POLITECNICO DI TORINO

Master Degree
in Data Science and Engineering

Master Thesis

Radar-Based Human Activity Recognition In Indoor Environments

Supervisors
Dr. Luigi Borzì
Dr. Samaneh Zolfaghari
Prof. Masoud Daneshtalab
Co-Supervisor
Dr. Hajar Abedi
Dr. Mainak Chakraborty
Prof. George Shaker

Candidates
Ali Samimi Fard
Mohammadreza Mashhadigholamali

Academic Year 2025-2026

Contribution Statement

This thesis is the result of a collaborative effort by the two candidates, **Ali Samimi Fard** and **Mohammadreza Mashhadigholamali**. Both authors contributed equally to the conception and development of the research, including the design of experiments, implementation of analytical and computational methods, data preprocessing, model development and evaluation, and the interpretation of results. The preparation, writing, and revision of this document were carried out jointly, and both authors share full responsibility for the content presented in this thesis.

Abstract

Human Activity Recognition (HAR) plays an essential role in ambient assisted living, healthcare monitoring, and smart home environments. Existing approaches that rely on wearable sensors often face compliance limitations, while camera-based systems raise privacy concerns and suffer from performance degradation under poor lighting or occlusion conditions. To address these challenges, this thesis explores a contactless, privacy-preserving sensing framework based on Frequency-Modulated Continuous Wave (FMCW) radar.

This work investigates radar-based multi-class activity recognition and fall detection using a 60 GHz MIMO FMCW radar and multi-dimensional feature maps—Range-Doppler (RD), Range-Azimuth (RA), and Range-Elevation (RE). A complete signal processing pipeline is developed to generate structured spatiotemporal radar representations. Unlike conventional methods that treat radar maps as standalone images, this framework preserves their multi-dimensional structure and leverages it directly within machine learning (ML) and deep learning (DL) models.

Two datasets were collected in realistic indoor environments—a bedroom with a single radar and a living room with a dual-radar setup—capturing a broad range of daily activities and fall-related scenarios. Multiple ML and DL architectures were evaluated, including Support Vector Machines (SVM), Multi-Layer Perceptron (MLP), Convolutional Neural Networks (CNN), Long Short-Term Memory networks (LSTM), Convolutional LSTM (ConvLSTM), and hybrid CNN+LSTM and ResNet+LSTM models. Their performance and computational cost were assessed using cross-scene validation and leave-one-person-out schemes to examine generalization across environments and subjects.

Experimental results demonstrate that integrating RD, RA, and RE feature maps substantially improves recognition accuracy. In the bedroom dataset, the ConvLSTM achieved a macro F_1-score of up to 98.05% for 4-class activity recognition. For fall detection in the living room dataset, the proposed 3D CNN model obtained 94.33% accuracy and a macro F_1-score of 93.5% using combined RA+RE inputs. For 6-class activity recognition in the living room, the CNN+LSTM model achieved an accuracy of 79.11% and an F_1-score of 76.09%, further improved to

90.3% accuracy using a majority voting scheme.

Overall, the findings confirm that FMCW radar, combined with tailored spatiotemporal DL architectures, offers an effective, privacy-preserving, and generalizable solution for HAR and fall detection in indoor environments, supporting its applicability in real-world ambient assisted living systems.

Contents

List of Tables 6

List of Figures 9

1 Introduction 12
- 1.1 Motivation . . . 15
- 1.2 Research Goals . . . 15
- 1.3 Contributions . . . 16
- 1.4 Organization . . . 16

2 Background 18
- 2.1 Radar Basics . . . 18
  - 2.1.1 Doppler Effect . . . 19
  - 2.1.2 Millimeter Wave Radar . . . 19
  - 2.1.3 Pulsed-Radar Systems . . . 20
  - 2.1.4 Single Frequency Continuous Wave (SFCW) . . . 20
  - 2.1.5 Frequency Modulated Continuous Wave (FMCW) . . . 20
  - 2.1.6 Range Measurement . . . 21
  - 2.1.7 Velocity Measurement . . . 26
  - 2.1.8 Angle Detection . . . 27
  - 2.1.9 Maximum Angular Field of View and Angular Resolution . . . 30
- 2.2 Related Works . . . 31

3 Indoor Human Activity Recognition 38
- 3.1 Preprocessing . . . 38
  - 3.1.1 Blackman-Harris Window . . . 39
  - 3.1.2 Range-FFT Map . . . 40
  - 3.1.3 Moving Target Indicator (MTI) . . . 41
  - 3.1.4 Range-Doppler Map . . . 42
  - 3.1.5 Capon Algorithm . . . 42
  - 3.1.6 Principal Component Analysis (PCA) . . . 46

3.2 Bedroom Dataset . . . 47
3.2.1 Materials and Method . . . 48
3.2.2 Data Collection . . . 49
3.2.3 Activity Recognition . . . 51
3.2.4 Experimental Setup . . . 55
3.3 Living Room Dataset . . . 56
3.3.1 Materials and Method . . . 57
3.3.2 Data Collection . . . 57
3.3.3 Fall Detection . . . 59
3.3.4 Activity recognition . . . 63
3.4 Performance Metrics . . . 68

4 Experimental Evaluation Result 70
4.1 Bedroom Dataset . . . 70
4.1.1 Experimental Results . . . 70
4.1.2 Discussion . . . 88
4.1.3 Conclusion . . . 89
4.2 Living Room Dataset . . . 90
4.2.1 Fall detection . . . 90
4.2.2 Activity Recognition . . . 94

5 Summary and Future Outlook 108
5.1 Technical Contributions . . . 108
5.2 Open Issues and Future Directions . . . 109
5.3 Other Potential Application Areas . . . 111

List of Abbreviations 112

Bibliography 116

List of Tables

2.1 Summary of Research Papers. 31
3.1 Radar configuration and specification. 49
3.2 Activity sample counts under Cross-Scene-Validation approach. . . 55
3.3 Activity sample counts per subject in Leave-One-Person-Out Cross-Validation approach. (A_1) walking, (A_2) sitting on the bed, (A_3) sitting on the chair, (A_4) lying down on the bed, (A_5) lying down on the floor, (A_6) empty room, (A_7) transition. (S_1) subject one, (S_2) subject two and (S_3) subject three. 56
3.4 Radar configuration and specification. 57
3.5 Total sample count for each scenario. 63
3.6 Activity sample counts for 9-class classification. 68
3.7 Activity sample counts for 6-class classification. 68
4.1 Comparison of Accuracy and F_1-scores for different machine learning classifiers across varying activity sets. NB: Naive Bayes, DT: Decision Tree, RF: Random Forest. 71
4.2 Confusion matrix for the Support Vector Machines classifier with seven activities, showing percentages. 72
4.3 Confusion matrix for the Multi-Layer Perceptron classifier with seven activities, showing percentages. 72
4.4 Confusion matrix for the Naive Bayes classifier with seven activities, showing percentages. 73
4.5 Confusion matrix for the Decision Tree classifier with seven activities, showing percentages. 73
4.6 Confusion matrix for the Random Forest classifier with seven activities, showing percentages. 74
4.7 Confusion matrix for the K-Nearest Neighbor classifier with seven activities, showing percentages. 74
4.8 Confusion matrix of the Support Vector Machines in classifying six activities, showing percentages. 75
4.9 Confusion matrix of the Multi-Layer Perceptron classifier in classifying six activities, showing percentages. 75

4.10 Confusion matrix of the Naive Bayes classifier in classifying six activities, showing percentages. 76
4.11 Confusion matrix of the Decision Tree classifier in classifying six activities, showing percentages. 76
4.12 Confusion matrix of the Random Forest classifier in classifying six activities, showing percentages. 77
4.13 Confusion matrix of the K-Nearest Neighbor classifier in classifying six activities, showing percentages. 77
4.14 Confusion matrix of the Support Vector Machines in classifying five activities, showing percentages. 78
4.15 Confusion matrix of the Multi-Layer Perceptron in classifying five activities, showing percentages. 78
4.16 Confusion matrix of the Naive Bayes in classifying five activities, showing percentages. 79
4.17 Confusion matrix of the Decision Tree in classifying five activities, showing percentages. 79
4.18 Confusion matrix of the Random Forest in classifying five activities, showing percentages. 80
4.19 Confusion matrix of the K-Nearest Neighbor in classifying five activities, showing percentages. 80
4.20 Confusion matrix of the Support Vector Machines classifier in classifying four activities, showing percentages. 81
4.21 Confusion matrix of the Multi-Layer Perceptron in classifying four activities, showing percentages. 81
4.22 Confusion matrix of the Naive Bayes classifier in classifying four activities, showing percentages. 82
4.23 Confusion matrix of the Decision Tree in classifying four activities, showing percentages. 82
4.24 Confusion matrix of the Random Forest classifier in classifying four activities, showing percentages. 83
4.25 Confusion matrix of the K-Nearest Neighbor in classifying four activities, showing percentages. 83
4.26 Performance metrics for various deep learning models under Cross-Scene-Validation with Range-Doppler+Range-Azimuth+Range-Elevation inputs. 84
4.27 Comparison of ConvLSTM model performance for 7 activity classification under the Cross-Scene-Validation approach. 85
4.28 Comparison of ConvLSTM model performance for 6 activity classification under the Cross-Scene-Validation approach. 85
4.29 Comparison of ConvLSTM model performance for 5 activity classification under the Cross-Scene-Validation approach. 85

4.30 Comparison of ConvLSTM model performance for 4 activity classification under the Cross-Scene-Validation approach. 85
4.31 Comparison of ConvLSTM model performance presented as percentages using the Leave-One-Person-Out Cross-Validation approach. 87
4.32 Comparison of proposed 3D CNN model performance for various configurations. RE: Range-Elevation, RA: Range-Azimuth. 91
4.33 Comparison of proposed 3D CNN model complexity and computational resource for various configurations. RE: Range-Elevation, RA: Range-Azimuth. 92
4.34 Performance metrics for 3D CNN models with Range-Doppler+Range-Azimuth+Range-Elevation inputs for 9-class activity recognition. . 95
4.35 Computational cost for 3D CNN model with Range-Doppler+Range-Azimuth+Range-Elevation inputs for 9-class activity recognition. . 96
4.36 Classification performance of ResNet+LSTM model with Range-Doppler+Range-Azimuth+Range-Elevation inputs for 9-class activity recognition. 97
4.37 Classification performance of ResNet+LSTM model with Range-Doppler+Range-Azimuth+Range-Elevation inputs for 6-class activity recognition. 99
4.38 Computational cost of ResNet+LSTM model. 101
4.39 Classification performance of CNN+LSTM model with Range-Doppler+Range-Azimuth+Range-Elevation inputs for 6-class activity recognition. . 102
4.40 Accuracy of the model with different value of t. 103
4.41 Computational cost for CNN+LSTM model with Range-Doppler+Range-Azimuth+Range-Elevation inputs for 6-class activity recognition. . 104

List of Figures

2.1 Illustration of Doppler effect. 19
2.2 Various radars waveform illustration. 21
2.3 FMCW radar block diagram. 22
2.4 FMCW radar signal illustration in various domains. 22
2.5 Basic principle of a radar. 23
2.6 Transmitted and received signal in FMCW radar. 24
2.7 Multiple object detection using FMCW radar. 25
2.8 Doppler-FFT to separate two objects. 28
2.9 Phase difference between RX antenna. 29
2.10 Angle of arrival estimation in FMCW radar. 29
2.11 Maximum angular field of view. 30
3.1 Overview of the proposed preprocessing steps of FMCW radar-based HAR. 39
3.2 Three-term Blackman-Harris window and transform. 40
3.3 Range-Doppler map data processing. 42
3.4 Overview of the proposed framework for FMCW radar-based HAR. SVM: Support Vector Machines, MLP: Multi-Layer Perceptron, NB: Naive Bayes, DT: Decision Tree, RF: Random Forest, KNN: K-Nearest Neighbor. 48
3.5 BGT60TR13C radar device. 48
3.6 BGT60TR13C top view and antenna arrangement. 49
3.7 Bedroom layout and radar placement in the home-like data collection environment. 50
3.8 Examples of a participant performing the activities with corresponding feature maps. (A_1) walking, (A_2) sitting on the bed, (A_3) sitting on the chair, (A_4) lying down on the bed, (A_5) lying down on the floor, (A_6) empty room, (A_7) transition. 51
3.9 Architecture of the 3D CNN model implemented for activity classification using the bedroom dataset. 53
3.10 Architecture of the LSTM model implemented for activity classification using the bedroom dataset. 54

3.11 Architecture of the ConvLSTM model implemented for activity classification using the bedroom dataset. 54
3.12 Room layout indicating radars positions and activity locations. Activities: (B_1) lying on the floor, (B_2) lying on a sofa, (B_3) sitting on/getting up from a sofa, (B_4) picking an object from the floor, (B_5) picking an object from a table, (B_6) sitting on the floor, (B_7) sitting on a sofa, (B_8) standing, (B_9) walking. 58
3.13 The living room layout indicating radars positions and activity locations. 59
3.14 Overview of the proposed fall detection system. 60
3.15 Examples of a participant performing the activities with corresponding feature maps. 61
3.16 Architecture of the 3D CNN model implemented for fall detection using the living room dataset. 62
3.17 Overview of the proposed activity recognition system. 64
3.18 Examples of a participant performing the activities with corresponding feature maps. Activities: (B_1) lying on the floor, (B_2) lying on a sofa, (B_3) sitting on/getting up from a sofa, (B_4) picking an object from the floor, (B_5) picking an object from a table, (B_6) sitting on the floor, (B_7) sitting on a sofa, (B_8) standing, (B_9) walking. 64
3.19 Architecture of the ResNet-18 + LSTM model implemented for activity classification using the bedroom dataset. 66
3.20 Architecture of the CNN + LSTM model implemented for activity classification using the bedroom dataset. 67
4.1 Confusion matrices for the ConvLSTM model on various activity sets under Cross-Scene-Validation with Range-Doppler+Range-Azimuth+Range-Elevation inputs. 86
4.2 Training and validation loss curves for the ConvLSTM model under Cross-Scene-Validation with Range-Doppler+Range-Azimuth+Range-Elevation inputs. 87
4.3 Normalized confusion matrices for different proposed model configurations and the Range-Elevation+Range-Azimuth feature input. . 91
4.4 Training loss for different proposed model configurations and Range-Elevation+Range-Azimuth feature input. 92
4.5 Confusion matrix for the 3D CNN model for 9-class activity recognition. (B_1) lying on the floor, (B_2) lying on a sofa, (B_3) sitting on/getting up from a sofa, (B_4) picking an object from the floor, (B_5) picking an object from a table, (B_6) sitting on the floor, (B_7) sitting on a sofa, (B_8) standing, (B_9) walking. 96

4.6 Training and validation loss curves for the 3D CNN model with Range-Doppler+Range-Azimuth+Range-Elevation inputs for 9-class activity recognition. 97
4.7 Confusion matrix for the ResNet+LSTM model for 9-class activity recognition. (B_1) lying on the floor, (B_2) lying on a sofa, (B_3) sitting on/getting up from a sofa, (B_4) picking an object from the floor, (B_5) picking an object from a table, (B_6) sitting on the floor, (B_7) sitting on a sofa, (B_8) standing, (B_9) walking. 98
4.8 Training and validation loss curves for the ResNet+LSTM model with Range-Doppler+Range-Azimuth+Range-Elevation inputs for 9-class activity recognition. 99
4.9 Confusion matrix for the ResNet+LSTM model for 6-class activity recognition. (C_1) In-place movement, (C_2) lying on the floor, (C_3) lying on a sofa, (C_4) sitting on the floor, (C_5) sitting on a sofa, (C_6) walking. 100
4.10 Training and validation loss curves for the ResNet+LSTM model with Range-Doppler+Range-Azimuth+Range-Elevation inputs for 6-class activity recognition. 101
4.11 Confusion matrix for the CNN+LSTM model for 6-class activity recognition. (C_1) In-place movement, (C_2) lying on the floor, (C_3) lying on a sofa, (C_4) sitting on the floor, (C_5) sitting on a sofa, (C_6) walking. 102
4.12 Training and validation loss curves for the CNN+LSTM model with Range-Doppler+Range-Azimuth+Range-Elevation inputs for 6-class activity recognition. 103

Chapter 1

Introduction

Over the past decades, increasing the population of older adults has become a significant challenge for most developed and developing countries [1]. By the end of the year 2050, about 22% of the world population is expected to be over 65 [2]. This demographic shift leads to the emergence of diseases, such as neurodegenerative and non-communicable diseases, necessitating consistent patient monitoring [3]. Research suggests that timely and appropriate interventions could reduce the number of fatal incidents and hospitalizations of the elderly. However, considering existing resources and increasing demand for care, it is impossible to provide suitable care for people in need, staying at home or other care facilities [2]. Remote health and activity monitoring allows more people at home or nursing facilities to access constant surveillance. Such systems can be used to measure a broad range of physiological signs and recognize users' activity [4]. In past studies, these systems have been used to monitor a wide range of health issues, including Cardiovascular and Respiratory system-related diseases, mobility-related diseases, Neurological Disorders, and Diabetes [5].

Chronic diseases are a significant health concern for older adults, especially cardiovascular diseases and diabetes. These conditions can significantly impact their quality of life and healthcare needs [6]. However, these diseases can be prevented or even mitigated through healthy lifestyle choices, including a balanced diet, regular physical activity, and avoiding tobacco products [7, 8]. These conditions significantly limit mobility and physical activity in affected individuals, further impacting their health and well-being [9]. Accordingly, monitoring physical activity can be an effective way to identify potential issues and serves as a crucial indicator to determine whether individuals are engaging in sufficient exercise to prevent chronic diseases or exhibiting early symptoms of such conditions [10]. Also, previous studies highlight growing evidence that controllable factors such as physical activity, social interaction, and cognitive engagement can delay or prevent dementia. Since cognitive abilities are closely tied to independent living, monitoring activities that

boost cognitive function could be crucial for maintaining autonomy and delaying hospitalization [11]. In other words, monitoring an individual's physical activities is useful across various domains, including rehabilitation, sports science, and early detection of musculoskeletal or cognitive disorders, as well as in assessing fall risk and balance [12]. The field of Human Activity Recognition (HAR) has emerged as a highly active area of research over the last twenty years due to its significant applications across various sectors, including robotics, healthcare, remote monitoring, gaming, security surveillance, the educational sector, and Human-computer Interaction (HCI) [13, 14].

The activity recognition process involves identifying or detecting current activities based on data collected from various sensors [13]. Researchers divided activity recognition into three subcategories: 1) action-based, 2) motion/gesture-based and 3) interaction-based [14, 15]. Action refers to coordinating body movement to accomplish a specific task, while gesture refers to moving parts of the body in order to highlight speech. Interaction encompasses a range of activities typically involving two or more participants. For instance, shaking one's head to indicate disagreement is a gesture, walking is considered an action, and having a conversation between two or people and engaging in fighting is an interaction [14].

In previous studies, various sensors were employed to capture human movement. These sensors facilitate the precise and efficient recording of human activity. Among the array of sensors available, two primary categories are commonly used in movement analysis methods: wearable sensors and non-wearable sensors [16,17].

In the wearable category, individuals are required to carry sensors with them as they engage in various activities. In past research, a wide range of sensors, including accelerometer, gyroscope, barometer, and others, were utilized to measure changes in acceleration during human body movements [2, 18]. Although extensive research [19–21] has been conducted on using wearable sensors for activity recognition, there are some challenges in using such devices. For instance, some people might refuse to wear them or need to remember to use them or recharge their batteries. Wearable sensors can also be uncomfortable and tricky to wear all the time, especially during activities such as bathing or sleeping, which might be challenging for older adults. Also, slight differences in attachment sites can affect accelerometers' accuracy in evaluating gait parameters [22]. Although most wearable devices are affordable, the limitations mentioned above suggest the necessity to explore other options for studying human activity thoroughly.

In recent years, attention has shifted towards the non-wearable approach, which eliminates the need for individuals to carry or wear any sensors or devices. This strategy entails installing sensors within the environment where activities occur. These sensors then collect activity data, which can be analyzed for activity recognition. This group of devices are more user-friendly since they do not impose the burden of carrying them. Some examples of non-wearable sensors include

infrared-based technologies, Ambient light communication technologies, computer vision sensors, ultrasonic transmitters, and radars [6]. However, non-wearable sensors face their own set of challenges [15]. For example, Video cameras can detect and monitor the movements of elderly individuals, but concerns regarding privacy infringement may arise [17]. Additionally, Cameras for monitoring require a clear Line of Sight (LOS) to the subject, and their performance can be affected by environmental situations, such as light conditions [23] and clothing [24]. Other non-wearable sensors have their limitations. For instance, Infrared sensors interfere with sunlight, are susceptible to noise, and have limited working range [6].

It is also possible to use WiFi signals to detect and recognize human activities. WiFi sensing leverages existing wireless communication infrastructure, thereby enabling cost-effective and straightforward deployment, which results in eliminating the need for additional sensor installation and minimizing associated costs [25]. However, relying solely on WiFi for sensing presents challenges in daily life application, robustness, and privacy. WiFi signals are sensitive to environmental factors, leading to inaccuracies in sensing results, while privacy and security concerns arise from the potential capture of sensitive information. Additionally, WiFi is designed for wireless data transfer, not other applications. So, using this technology for other purposes can impact network quality significantly. Despite its advantages, WiFi sensing requires careful consideration of these challenges to ensure reliable and secure operation [26].

Radars, on the other hand, provide enhanced privacy protection compared to video-based methods, making them more comfortable for users. Also, radars can capture motion even in low-light conditions [17] . Radar systems are classified into two main types based on their waveform [27, 28]: Continuous Wave (CW) radar and impulse radar, each with distinct applications. Impulse radars are useful in vital signs detection, gesture recognition, and human tracking. Conversely, CW radar boasts a simpler architecture, facilitating easy system integration and lower power consumption, making it appealing for mobile and portable applications [27]. While CW radar has been investigated for human activity classification in several studies [29], its inability to measure absolute range and range resolution poses challenges [30, 31].

CW radars can be further categorized into two types: Frequency Modulated Continuous Wave (FMCW) radars and Single Frequency Continuous Wave (SFCW) radars, also known as Doppler radars. SFCW radars operate based on the Doppler phenomenon, where a frequency shift occurs when encountering an object. In contrast, FMCW radars transmit signals with varying frequencies and can also utilize the Doppler effect [28].

Previous initiatives in crafting radar-based systems for motion analysis have mainly concentrated on analyzing movements in a straight line to minimize the angle of motion's effect on micro-Doppler patterns [32–34]. However, straight-line

movements are rare in residential environments, with individuals often moving unpredictably. To bypass the issues related to the radar's angle relative to the moving subject, determining the individual's velocity by tracking the shift in their location over time (velocity = position/time) presents a viable solution. Employing a Multiple Input Multiple Output (MIMO) Frequency Modulated Continuous Wave (FMCW) radar system [35–37] offers the dual benefits of pinpointing subjects' locations while also gathering micro-Doppler data. This dual capability makes MIMO FMCW radar an ideal candidate for monitoring movement within homes and recognizing a wide range of activities [38, 39].

1.1 Motivation

The growing demand for continuous, reliable, and privacy-preserving monitoring of older adults—particularly in indoor settings such as care facilities—necessitates unobtrusive and scalable solutions. Although wearable sensors and vision-based systems have been extensively explored for HAR, they often encounter challenges related to user compliance, comfort, privacy concerns, and sensitivity to environmental factors.

Radar-based HAR, especially with MIMO FMCW technology, presents a compelling alternative. This research is driven by the potential of radar systems to deliver accurate, contactless, and privacy-respecting activity recognition in realistic indoor environments.

1.2 Research Goals

The overarching goal of this thesis is to investigate the feasibility, design considerations, and practical limitations of human activity recognition using FMCW radar in realistic indoor environments.

To achieve this, the work is guided by the following specific research goals:

- Design and implement a radar signal processing pipeline capable of converting raw FMCW radar data into structured and informative feature representations.
- Develop and collect two realistic radar-based activity datasets in indoor environments, enabling the study of challenging and underexplored activity classes.
- Evaluate and compare various ML and DL models—including classical classifiers, temporal models, and hybrid architectures—for both activity recognition and fall detection.

- Analyze the generalization ability of trained models across spatial variations and viewing angles to assess their robustness and transferability.
- Investigate model efficiency, computational complexity, and the feasibility of ensemble techniques in the context of real-time or embedded system deployment.

1.3 Contributions

The publications listed below were derived from the research conducted in this thesis. Some titles link directly to the main page of the corresponding paper.

- **Fall Detection in Ambient-Assisted Living Environments Using FMCW Radars and Deep Learning**
  Proceedings of the IEEE International Radar Conference (RADAR), 2025.
- **FMCW Radar-Based Human Activity Recognition: A Machine Learning Approach for Elderly Care**
  Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC), 2025.
- **Lightweight FMCW Radar Framework for Human Activity Recognition under Limited Data Conditions**
  Submitted to *Nature Scientific Reports*, under review.
- **Exploring Multi-View FMCW Radar Early-level Feature Fusion for Human Activity Recognition**
  Submitted to *22nd Workshop on Context and Activity Modeling and Recognition with AI - CoMoRe-AI 2026*, under review.

1.4 Organization

The remainder of this thesis is structured as follows:

- **Chapter 2**: Provides background on radar fundamentals and human activity recognition, including a literature review. It also includes a literature review covering FMCW radar systems and ML and DL approaches relevant to the field.
- **Chapter 3**: Describes the system methodology, including preprocessing, feature extraction, dataset collection, and ML/DL model design.

- **Chapter 4**: Presents the experimental evaluation results across the two datasets, comparing ML and DL models across multiple activity recognition tasks.
- **Chapter 5**: Summarizes the contributions, highlights remaining research challenges, and discusses future research directions.

Chapter 2

Background

This chapter provides an overview of the essential concepts and technologies that underpin radar-based HAR. We begin by introducing radar fundamentals, including the key principles and mechanisms that allow for the detection and tracking of motion. This is followed by a detailed examination of Frequency Modulated Continuous Wave (FMCW) radar systems, which form the basis for the radar setup employed in this research. Additionally, we review the existing body of work in radar-based HAR, highlighting significant advancements in the field and how they relate to the methods and objectives of this work.

2.1 Radar Basics

RADAR is the abbreviation of Radio Detection And Ranging [40]. Originally developed for military applications, radar technology has since been adopted across numerous non-military domains, including HAR [23], human gait analysis [38], vital sign monitoring [41], and the automotive industry [42], such as vehicle occupancy detection [43] and Door Open Warning Technology [44]. Radar is also widely used in Unmanned Aerial Vehicles (UAVs) applications [45].

The fundamental principle of radar is to transmit electromagnetic waves and receive their reflections from surrounding targets. By measuring the time delay between the emitted and received signals, the system can determine the range (distance) to the target. Additionally, if the target is moving, radar can estimate its velocity by exploiting the Doppler effect [46].

In the following sections, we introduce the technical concepts needed to understand radar operation and classification. We then categorize radars based on their operating frequency and signal type. This is followed by a detailed review of key FMCW radar parameters, such as chirp structure, intermediate signals, and bandwidth, along with explanations of how FMCW radar can measure distance,

velocity, and angle with high precision. We also introduce important definitions such as range resolution, velocity resolution, maximum detectable velocity, maximum angular field of view, and angular resolution [40].

2.1.1 Doppler Effect

The Doppler effect refers to the change in the frequency of a received signal compared with the originally transmitted signal, caused by the relative motion between the target and the radiation source. The direction of motion determines whether the frequency shift is positive or negative. When the target moves toward the radar, the perceived wavelength becomes shorter, resulting in a higher received frequency. Conversely, when the target moves away from the radar, the reflected signal has a longer wavelength and therefore a lower frequency [40]. Fig. 2.1 illustrates the Doppler effect.

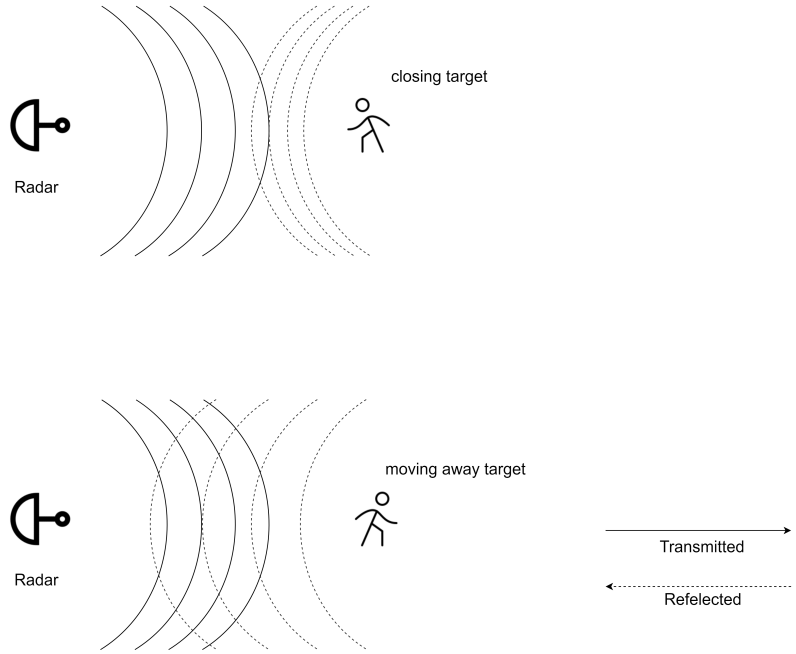


Figure 2.1: Illustration of Doppler effect.

2.1.2 Millimeter Wave Radar

Radars can be classified based on various criteria, one of which is the frequency band of the emitted signal. Millimeter Wave (mmWave) refers to electromagnetic waves with wavelengths ranging from 1 mm to 10 mm, corresponding to frequencies between 30 and 300 GHz [47]. One of the key advantages of mmWave technology

is its ability to support compact radar designs. Because shorter wavelengths allow for smaller antennas and electronic components, high-frequency mmWave systems can be integrated more easily into constrained spaces or embedded within smart furniture [48].

2.1.3 Pulsed-Radar Systems

Conventional pulsed radars emit a periodic sequence of pulsed waveforms. These radars can determine the range (distance) of an object by measuring the time delay between the transmitted and received signals [49]. Although pulsed radars offer long operating ranges and are capable of measuring both the target's range and velocity [50], they are generally unsuitable for short-range measurement applications [49]. Moreover, they require high peak power to achieve sufficient average transmitted power, which increases system complexity and limits their practicality for low-power or compact sensing scenarios [50].

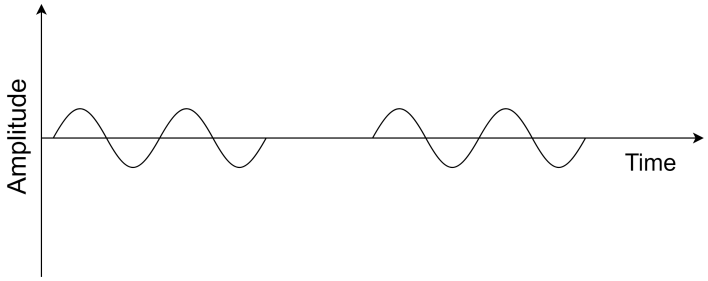
2.1.4 Single Frequency Continuous Wave (SFCW)

Unlike pulsed radars, this radar continuously emit electromagnetic signals with a fixed frequency. CW radar can measure the radial velocity of the object by measuring changes in the frequency of the received signal [28]. In order to calculate the range of the object, it is necessary to calculate the time delay between transmitting and receiving signals. As a result, unlike pulsed radar, SFCW radar cannot measure the object's distance [40].

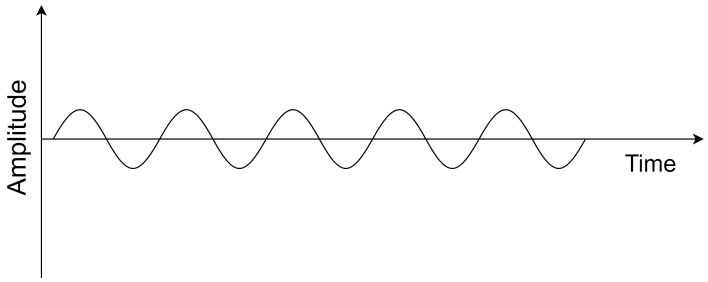
2.1.5 Frequency Modulated Continuous Wave (FMCW)

In FMCW radars, the transmitted signal increases linearly in frequency over a fixed duration. This is the key distinction between traditional pulsed radar systems and SFCW radars, which continuously transmit a constant-frequency waveform [41]. As a result, FMCW radars can simultaneously measure the range, velocity, and angle of a target [51]. Fig. 2.2 illustrates different radar waveform types. A typical FMCW radar consists of several functional blocks, as shown in Fig. 2.3.

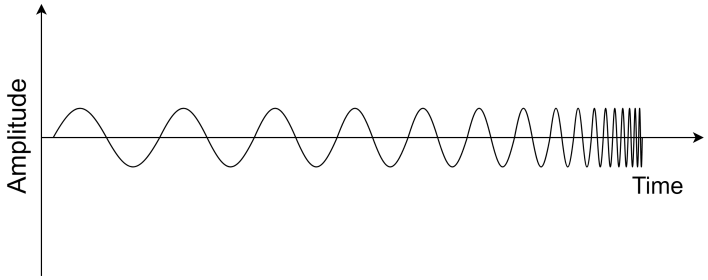
The ability to deploy multiple FMCW radars with only small spatial displacement is another advantage of this technology [52]. This feature is particularly beneficial in scenarios requiring full coverage of indoor spaces such as residential environments or long-term care facilities where multiple radar units must be installed throughout the area.



(a) Pulse radar waveform



(b) SFCW radar



(c) FMCW radar

Figure 2.2: Various radars waveform illustration.

2.1.6 Range Measurement

In pulsed radar systems, the range of the object is calculated by measuring the delay between the transmitted and received signal by the device. However, it is impossible to determine the target's velocity with this type of radar. In SFCW radars, the frequency shift between transmitted and received signals is measured by the device, and by using that, the speed can be calculated.

Despite SFCW Radars, FMCW radars can also measure the target's distance [43]. In FMCW radars, each transmitted signal is a sinusoidal wave that starts at a specific frequency (f_c), then sweeps up to a higher frequency (f_e) over a specific duration (T_c). This signal is called a 'chirp signal', and a specific number of chirp signals make one chirp frame [53, 54]. Fig. 2.4a illustrates a sample chirp signal amplitude over time and Fig. 2.4b shows the frequency versus time plot of a chirp

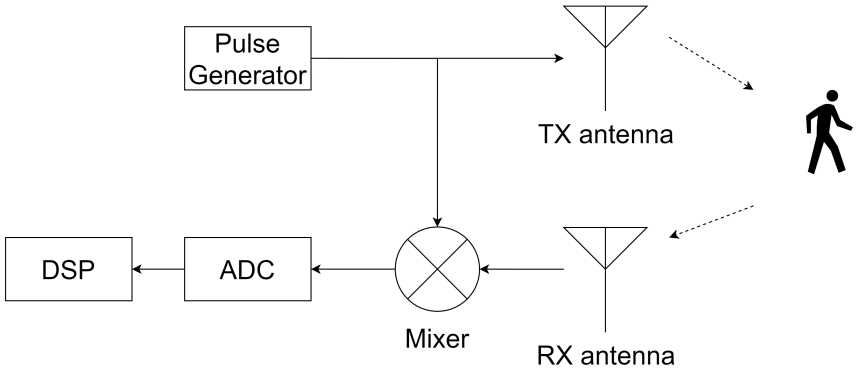
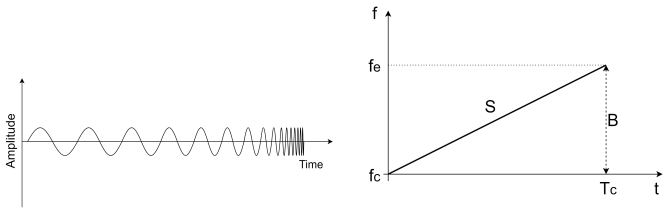


Figure 2.3: FMCW radar block diagram.

signal.



(a) Amplitude vs Time (b) Frequency vs Time

Figure 2.4: FMCW radar signal illustration in various domains.

We can calculate the slope of the sweep S, which is the rate of increase in the frequency of chirp and is a constant parameter. It can be computed using the following formula:

$$S = \frac{B}{T_c} \tag{2.1}$$

FMCW radar transmits a chirp signal through the transmitting antenna (TX) and captures signal reflected by objects with receiving antennas (RX). Fig. 2.5 displays the principle of a radar system.

these signals can be expressed with equations 2.2 and 2.3 [55]:

$$X_{TX} = \sin\left(\omega_{TX} + \phi_{TX}\right) \tag{2.2}$$

$$X_{RX} = \sin\left(\omega_{RX} + \phi_{RX}\right) \tag{2.3}$$

where X_{TX} and X_{RX} are transmitted and received signals, ω_{TX} and ω_{RX} are their initial frequencies, and ϕ_{TX} and ϕ_{RX} are their initial phases, respectively.

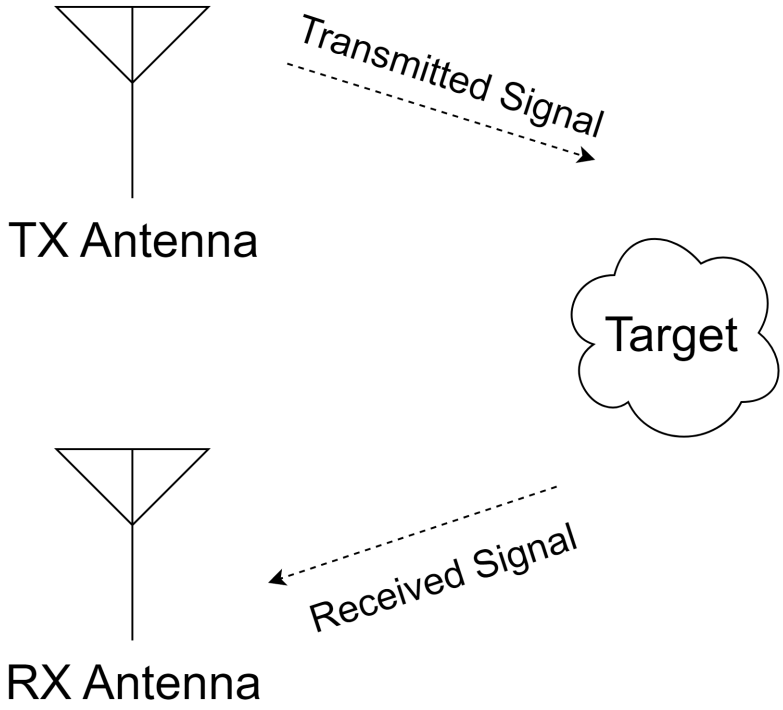


Figure 2.5: Basic principle of a radar.

FMCW radars utilize a 'mixer', a 3-port device with two inputs and one output, to combine the transmitted and received signals. The mixer creates a new signal with a new frequency, which is the Intermediate Frequency (IF) or Beat Signal. The maximum value of the intermediate frequency (f_{IFmax}) depends on the radar's hardware [50]. The beat frequency is equal to the difference in the frequency between two input signals, and the IF phase is equal to the difference between the two input signal phases. So, the IF is a sinusoidal wave, which can be expressed with the equation 2.4 [55]:

$$\begin{aligned} X_{IF} &= \sin\left((\omega_{TX} - \omega_{RX}) + (\phi_{TX} - \phi_{RX})\right) \\ &= A\sin\left(2\pi f_0 t + \phi_0\right) \end{aligned} \tag{2.4}$$

Also, the IF signal can be seen only in the interval where transmitted and received signals overlap (T). Fig. 2.6 illustrates transmitted, received, and IF signal and its frequency:

In FMCW radars, a portion of the transmitted chirp is mixed with the received signal by radar to produce a beat signal [56]. Also, the initial phase of IF signal ϕ_0 equals to $\phi_0 = 2\pi f_0 \tau$, where the initial frequency is $f_0 = S\tau = \frac{S2d}{C}$. In these

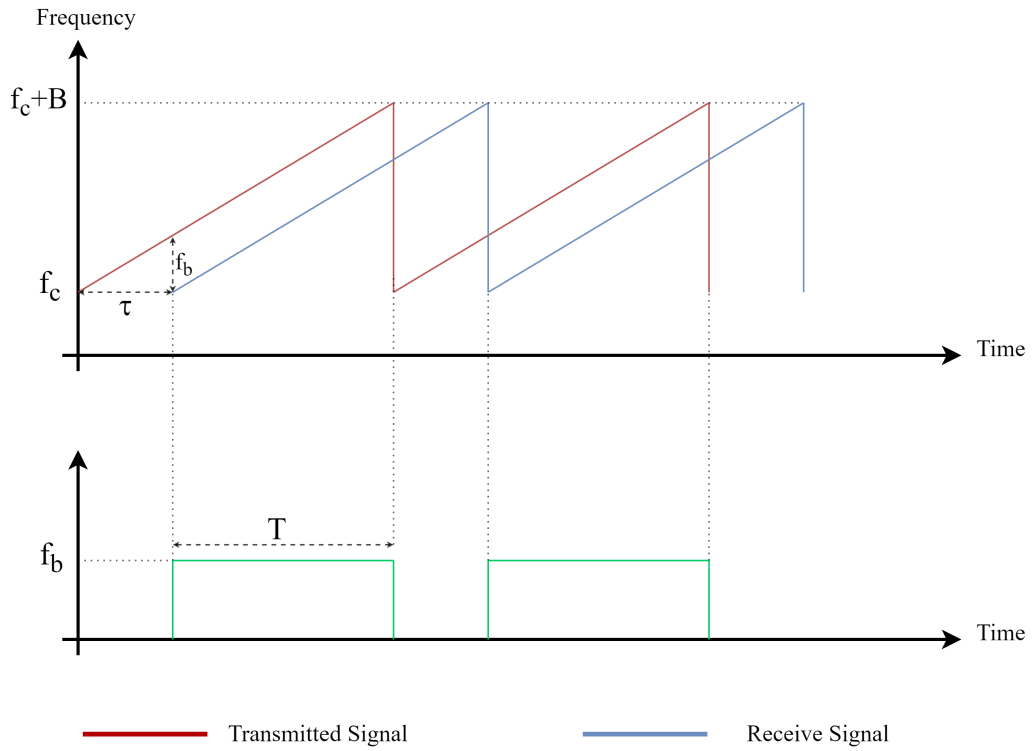


Figure 2.6: Transmitted and received signal in FMCW radar.

formulas, τ is the time delay between transmitting signal and receiving its reflection by radar, d is the distance between the object and the radar, and C refers to the propagation speed of the electromagnetic wave or speed of light, which is equal to $3 * 10^8$ and λ is the wavelength which can be computed using $\lambda = \frac{C}{f_0}$. So, we can rewrite the equation 2.4 as follows [50, 55]:

$$X_{IF} = A \sin\left(2\pi \frac{S2d}{C} t + \frac{4\pi d}{\lambda}\right) \tag{2.5}$$

So, once the frequency of the received signal by radar is known, we can calculate the distance d between the detected object and the radar using the following formula [50, 53]:

$$d = \frac{\tau C}{2} \tag{2.6}$$

The above explanation pertains to the case of a single detected object. The signal travel time varies if multiple objects are located at different distances. This results in different frequencies for the IF signals from each object, allowing us

to separate and calculate their respective distances easily [55]. This situation necessitates using the Fast Fourier Transformation (FFT) for separating different peaks, commonly known as range-FFT [55]. Fig. 2.7 illustrates multiple detected objects by radar, resulting in multiple strikes in the IF frequency spectrum.

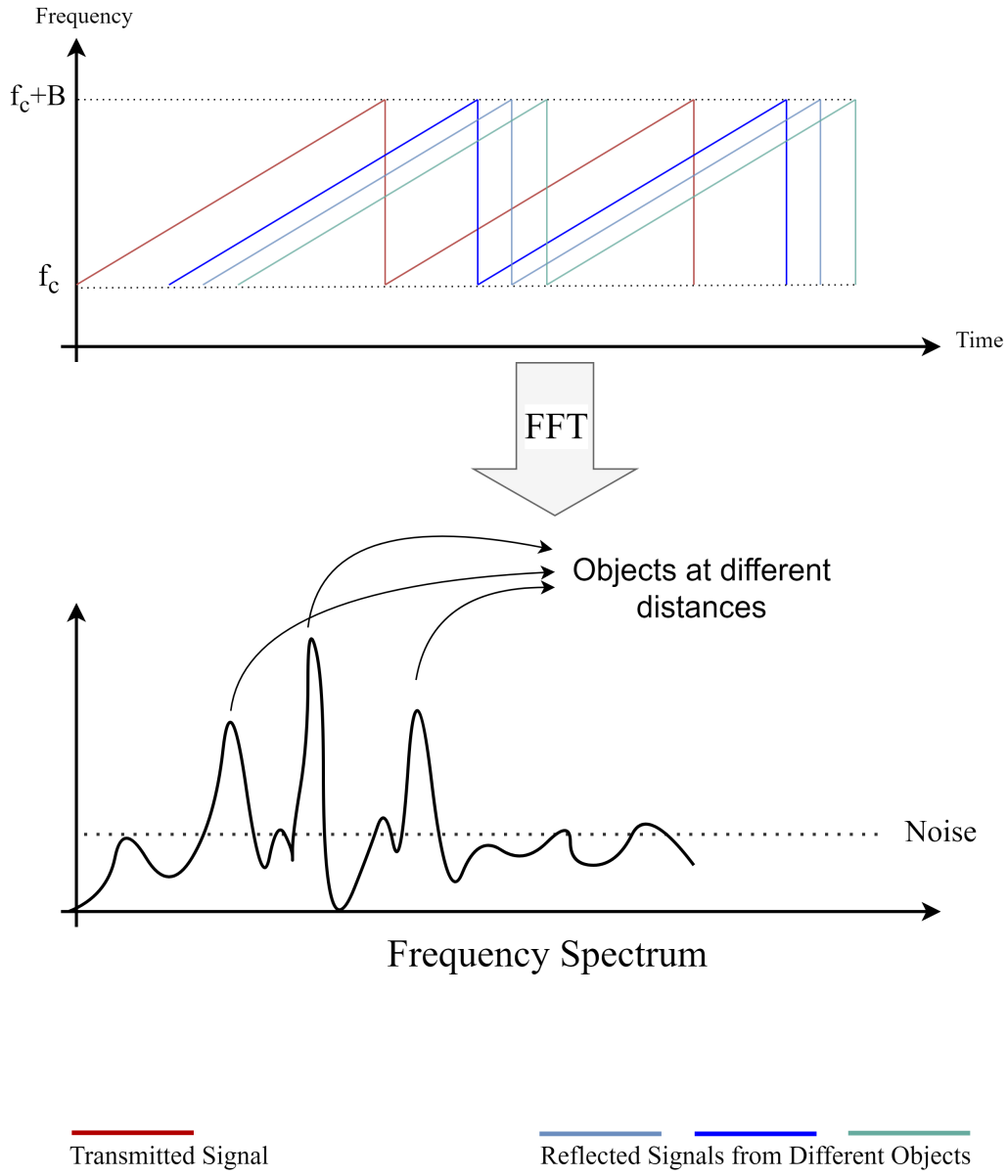


Figure 2.7: Multiple object detection using FMCW radar.

Range Resolution

Range resolution is the minimum detectable distance between two objects [40]. Based on the FFT theorem, the minimum distinguishable frequency component in an observation window T_c equals $1/T_c$ [50,55]. For FMCW radar with a bandwidth (B), range resolution R_{Res} can be determined using equation 2.7 [43, 50, 55]:

$$R_{Res} = \frac{C}{2B} \tag{2.7}$$

So, if we increase the bandwidth, we will have a higher range resolution [57]. In BGT60TR13C radar, ultra-wide bandwidth of 5.5 GHz results in a very good range resolution of approximately 3 cm [58].

2.1.7 Velocity Measurement

In order to classify the activity the subject is doing, it is necessary to calculate the velocity of the subject, as well as its position. In the velocity measurement process using a FMCW radar, the system must emit two chirps separated by a time interval $T_C = T + \tau$. Each chirp's reflection is subjected to an FFT to determine the target's distance (range-FFT). While each chirp's range-FFT will highlight peaks at identical locations, the phases of these peaks will differ. The change in phase between these two peaks is directly linked to the target's motion over the period T_C [55]. As a target moves a distance d in a time frame of T_C, the phase of the reflected signal from this object will change, which can be calculated from the equation 2.8:

$$\Delta\varphi = \frac{4\pi V T_C}{\lambda} \tag{2.8}$$

where V is the velocity, and λ is the wavelength. So, the velocity can then be extracted using the formula 2.9 [53, 55]:

$$V = \frac{\lambda \Delta\varphi}{4\pi T_C} \tag{2.9}$$

However, this velocity measurement method relies on the phase difference between received chirps by radar and has inherent ambiguities. Accurate measurement is possible only if $|\Delta\phi| < \pi$. Beyond this range, the phase difference becomes unclear, leading to potential errors in velocity calculation. By rearranging the formula above, we can find a condition for an unambiguous velocity measurement(equation 2.10) [43, 55]:

$$V < \frac{\lambda}{4T_C} \tag{2.10}$$

Furthermore, equation 2.11 informs us about the highest relative velocity (V_{max}) that can be effectively measured within this system. It indicates that higher velocities demand shorter intervals between chirp emissions [43].

$$V_{max} = \frac{\lambda}{4T_C} \tag{2.11}$$

This relationship underscores the trade-off between the measurement range and the temporal resolution in chirp-based radar velocity detection systems.

The two-chirp velocity measurement technique has limitations when dealing with scenarios involving multiple moving objects with varying velocities at the same distance from the radar. Since these objects share the same range, their reflected chirps exhibit identical IF. Consequently, the range-FFT produces a single peak representing the combined signal from all these objects at the same range. It is impossible to differentiate their velocities using a simple phase comparison technique [55]. To overcome this limitation and enable velocity measurement for multiple targets, the radar system employs a set of chirps transmitted in sequence, forming a chirp frame. This frame typically consists of N equally spaced chirps ($N > 2$). This approach allows the system to differentiate between targets with different velocities despite their identical range [50].

The processing involves using range-FFT on the reflected chirps, producing N peaks with different phases corresponding to each object's contribution. A second FFT, called Doppler-FFT, is then applied to the N phasors to distinguish the objects (Fig. 2.8), with ω_1 and ω_2 representing the phase differences between chirps [55].

Velocity Resolution

Velocity resolution is the minimum detectable change in the velocity of the object. Discrete Fourier Transform (DFT) indicate that two frequencies, ω_1 and ω_2, can be resolved if $\Delta\phi = \phi_2 - \phi_1 > \frac{2\pi}{N}$. N is the number of chirps in each frame, and $T_f = NT_c$. Given $\Delta\phi = \frac{4\pi V T_C}{\lambda}$, the velocity resolution (V_{res}) can be derived, showing it is inversely proportional to the time frame (T_f), as per equation 2.12 [50, 55].

$$V > V_{res} = \frac{\lambda}{2T_f} \tag{2.12}$$

2.1.8 Angle Detection

It is necessary to have the angle of the object, along with its range, to locate it in a 3D space. An FMCW radar can estimate the Angle of Arrival (AoA) of a reflected signal using at least two RX antennas separated by distance l. The angle of arrival estimation relies on the principle that a slight change in the distance of

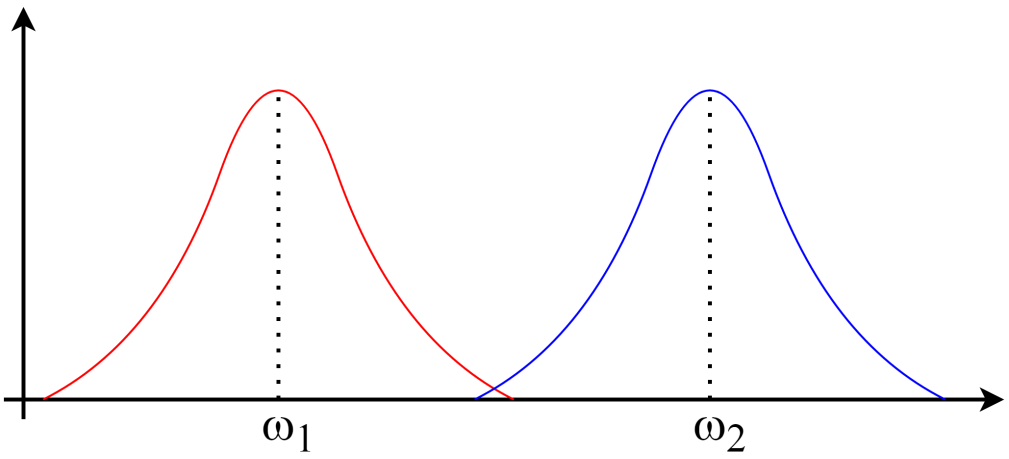


Figure 2.8: Doppler-FFT to separate two objects.

an object causes a phase shift in the range-FFT or Doppler-FFT peak. A chirp is transmitted from the TX antenna, and all antennae will receive its reflection. Since there is the distance between two RX antennas equal to l, the reflected signal must travel more distance equal to $\Delta d = l\sin(\theta)$ to reach the next antennas. This difference results in a phase change between received signals. This phase change is mathematically described in 2.13. Fig. 2.9 illustrates AoA and how distance between RX antennas results in phase difference.

$$\Delta\phi = \frac{2\pi\Delta d}{\lambda} \tag{2.13}$$

Finally, the AoA (θ) can be calculated from using equation 2.14:

$$\theta = \arcsin\frac{\Delta\phi\lambda}{2\pi l} \tag{2.14}$$

In this research, the distance between each two antennas is set as $\frac{\lambda}{2}$, so we can modify the equation 2.14 and rewrite it as follows (equation 2.15):

$$\theta = \arcsin\frac{\Delta\phi}{\pi} \tag{2.15}$$

Similarly, and considering antenna placement in this radar, AoA relative to the horizontal plane can be calculated by measuring the phase difference between RX1 and RX3, denoted as *azimuth*, and AoA relative to the vertical plane is called *elevation*, which can be estimated using the phase difference between RX1 and RX4 antennas.

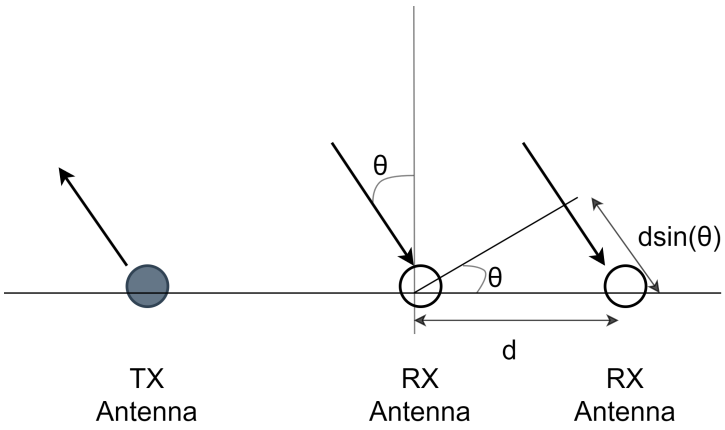


Figure 2.9: Phase difference between RX antenna.

The dependency of $\Delta\phi$ on $\sin(\theta)$ introduces nonlinearity, approximated by $\sin(\theta) \cong \theta$ for small θ. This approximation improves estimation accuracy when θ is small, as shown in Fig. 2.10.

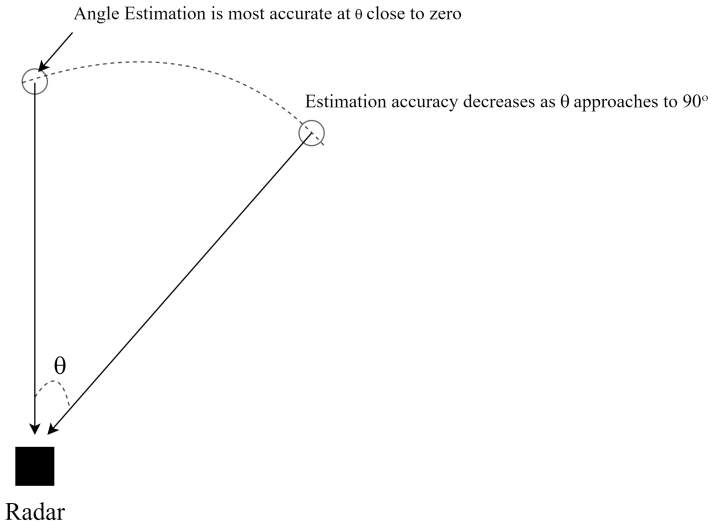


Figure 2.10: Angle of arrival estimation in FMCW radar.

In other words, radar is more sensitive to changes in the AoA when the object is placed precisely in front of the radar $\theta = 0$.

2.1.9 Maximum Angular Field of View and Angular Resolution

The radar's maximum angular field of view corresponds to the largest AoA it can estimate, as illustrated in Fig. 2.11.

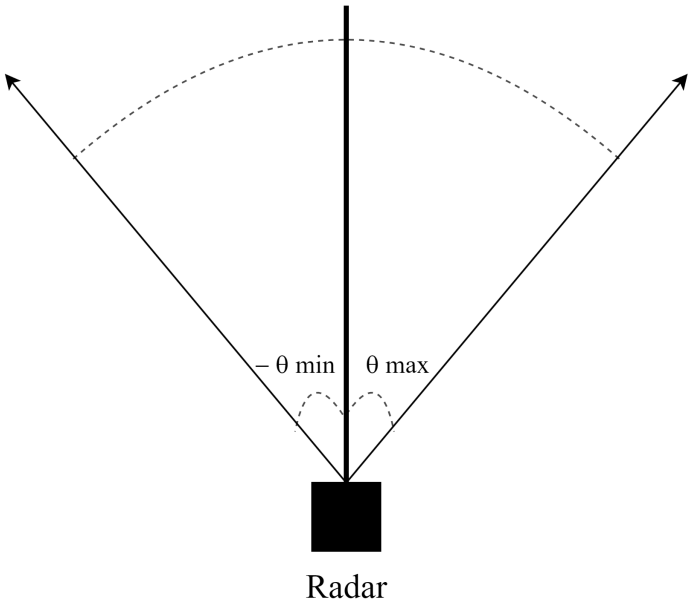


Figure 2.11: Maximum angular field of view.

The equation 2.16 provides the maximum field of view for two antennas spaced apart:

$$\theta_{max} = \sin^{-1}(\frac{\lambda}{2l}) \tag{2.16}$$

Usually, we choose $l = \frac{\lambda}{2}$ to get the largest possible angular field of view equal to $\pm 90°$ [50]. Also, the angle resolution θ_{Res} depends on the number of RX antenna, and it is derived from the following equation [59]:

$$\theta_{Res} = \frac{2}{N_{RX}} \tag{2.17}$$

where N_{RX} is the number of receiver antennas. However, an increasing number of RX antennas results in greater mathematical complexity and higher radar costs [60].

2.2 Related Works

Human activity recognition using radar technologies, particularly FMCW radar, has emerged as a robust solution for capturing human motion without needing wearable sensors. Recent advancements leverage sophisticated Machine Learning (ML) techniques, ranging from Deep Learning (DL) architectures to innovative feature extraction methods, to enhance the accuracy and efficiency of HAR systems. This section reviews pivotal studies that have contributed to the field.

Table 2.1: Summary of Research Papers.

Ref	Year	Radar Type	Sample Size	# Activities	Features	Model	Best Result
[61]	2020	University of Glasgow	-	-	Range-Doppler, Range-Time, Doppler-Time	1D CNN-LSTM, 2D CNN	93.4%
[62]	2020	TI IWR1642	15	6	Micro-Doppler Spectrograms, Range-Time Profiles	LSTM, Bi-LSTM	91%
[63]	2021	TI IWR6843ISK	8	6	Extract Range, Doppler, and Angle features with Neural Network	2D-CNN, 3D-CNN (+ LSTM)	!?
[17]	2022	University of Glasgow	-	-	Micro-Doppler features considered as 2D images	SVM	99.77%
[2]	2022	University of Glasgow	-	-	Time–Velocity images fed to VGG16	Improved VGG16 model	96.34%
[64]	2022	TI AWR 1642	19	3	Micro-Doppler images	SVM, KNN	73.7%
[65]	2022	TI WR1843	17	6	Point Cloud, Range-Doppler	3D CNN + LSTM	97.26%
[66]	2022	TI AWR1642 BOOST	10	4	Doppler, Range, and Angle signatures as 2D features, Range-Doppler-Time and Range-Angle-Time signatures as the 3D features	Transformer-sequential-decoder model	95.2%
[67]	2022	PulsON P410	14	9	Time-Velocity images	2D-CNN, GRUs	90.8%
[68]	2022	TI IWR6843ISK	4	7	Spatial–Temporal Point Clouds	CNN	97.6%
[69]	2022	University of Glasgow	-	-	Range-Doppler-Time	RD-CNN	96.5%
[70]	2022	TI 60 GHz FMCW	15	4	Time-Range-Doppler	CNN-LSTM	76%
[71]	2023	University of Glasgow	-	-	Doppler Time Map	KNN	96.40%
[72]	2023	AWR 1642 single-chip	13	5	Micro-Doppler characteristic	Residual-bi-LSTM-attention Hybrid Multi-Network	99.86%

Ref	Year	Radar Type	Sample Size	# Activities	Features	Model	Best Result
[73]	2023	University of Glasgow	-	-	Doppler-time map	ResNet50, VGG16, VGG19, MobileNetV2	93%
[74]	2023	TI IWR6843ISK	5	5	Range–Angle, Angle-Doppler, Range-Doppler-Angle	KNN, SVM, Random Forest, Bayes, Decision Tree	98.07%
[75]	2023	TI IWR6843AOI	5	5	Range-Velocity, Range-Elevation, Range-Azimuth	DyLite-RADHAR	97.5%
[76]	2023	Infineon BGT60TR13	14	1	Range-Doppler Image	CNN+LSTM	96%
[77]	2023	5.8 GHz FMCW	6	6	Micro-Doppler Images	SimpleNet, MobileNet-V2, ResNet, DINN, DIATRadHAR Net, SqueezeNet	95.4%
[78]	2023	5.8 GHz FMCW	6, 10	6, 5	Micro-Doppler Images	Lightweight Hybrid Vision Transformer	99.7%
[79]	2023	TI IWR 1443	10	4	Raw data, Dynamic Time Warping, Unsupervised-Encoded Level, Supervised-Encoded Level	LSTM, DTW	82%
[80]	2023	TI AWR1843	10	5	Micro-Doppler Time-Frequency map	Transformer network	85-97%
[81]	2023	University of Glasgow	-	-	Time-Range Map, Micro-Doppler Map	CNN, Transformer (parallel)	97.5%

Some researchers have focused on novel DL techniques to enhance the accuracy and efficiency of HAR systems. Chen et al. [66] tackled the challenge of recognizing continuous human motions with high similarity using a new Continuous Human Motion Recognition (CHMR) algorithm called CHMR-HS and a modified transformer-based learning model. By extracting and processing spectral-temporal features as both 2D and 3D signatures, including Doppler, range, and angle signatures (2D features) and Range-Doppler-time and Range-angle-time signatures (3D features), their model achieved a 95.2% accuracy in distinguishing similar continuous motions, based on data from 10 subjects performing 3 groups of similar activities. In this paper, authors installed a AWR1642-BOOST mmWave FMCW radar from Texas Instruments (TI) in the laboratory for data collection. Although they achieved high accuracy in classifying similar activities, in order to assess the proposed model better, it should be tested on more various activities.

Due to the power of transformer encoder-decoder networks for time-series data analysis, researchers have increasingly focused on for HAR in recent years. Sha Huan et al. [78] introduced a Lightweight Hybrid Vision Transformer (LH-ViT)

that combines efficient convolution operations with the self-attention mechanism, making it suitable for embedded applications. Using a micro-Doppler map as input, they achieved an impressive 99.7% accuracy on a self-established dataset containing reflected radar data of five daily activities recorded from ten subjects. This dataset was recorded using a FMCW radar operating at 5.8 GHz with 400 MHz bandwidth. Also, the performance of the model on a public dataset developed by Guangzhou University was 92.1%. This dataset was collected from six subjects performing six activities.

Another study [72] introduced a hybrid multi-network model that combines the strengths of Residual Neural Network (ResNet), Bi-directional Long Short-Term Memory (Bi-LSTM), and attention mechanisms. This model used micro-Doppler features to classify five activities performed by thirteen subjects. Authors collected data using TI AWR1642 single-chip FMCW radar in a laboratory environment, achieving an accuracy of 99.86%. Despite its high performance, the complexity and architecture of the model make it unsuitable for real-time implementation.

Jiang et al. [80] proposed a novel approach for HAR with transformer networks. Their study involved a dataset collected using TI AWR1843-BOOST from 10 volunteers performing five activities, including 1-Walking, 2-Standing still, 3-Standing up, 4-Sitting down and 5-Falling down. Initially, micro-Doppler time-frequency maps were extracted and mapped into a high-dimensional space. Then, the transformer multichannel attention mechanism was used to predict continuous actions. The proposed transformer-based network utilized a feature encoding layer and an Inception maximum pooling layer. The transformer network effectively recognized activities, achieving the highest accuracy of 97.23% for walking, and the lowest accuracy was 85.19% for squatting.

Gu et al. [81] present a lightweight DL model called RMPCT-Net, a multi-channel parallel Convolutional Neural Network (CNN) and transformer network designed for HAR applications using FMCW radar. The model architecture includes a dual-input system: one channel uses a CNN to extract spatial feature information from the Time-Range map, while another processes temporal features through transformer-based mechanisms from the micro-Doppler map. RMPCT-Net achieved a remarkable average accuracy of 97.5% on the University of Glasgow dataset.

Abdu et al. [17] used AlexNet, VGG16, and VGG19 pre-trained models to extract features from micro-Doppler signatures of the University of Glasgow dataset. To refine the extracted features, they used a simple channel attention network. The extracted features were fused using Canonical Correlation Analysis (CCA). Finally, the fused features were fed to an SVM classifier, achieving a accuracy of 99.77%.

Triani et al. [73] leveraged the VGG19 architecture with transfer learning for HAR. They compared the proposed method against traditional ML models, such as

SVM and k-NN, and the combination of CNN and ML methods. The Doppler-time map is extracted from a dataset provided by the University of Glasgow was fed to various pre-trained models such as ResNet50-TL, MobileNetV2-TL, VGG16-TL, and VGG19-TL. Then, SVM and k-NN algorithms were used as classifiers. The best accuracy was achieved by combining VGG19 as a feature extractor and SVM as a classifier, which was equal to 90%. The performance of the proposed model was compared against feeding the Doppler-time map directly to SVM and k-NN models and using solely pre-trained models for classification. Several researchers have made innovative contributions in the preprocessing phase to improve model performance. Nguyen et al. [77] presented a novel technique for enhancing performance of HAR by separating micro-Doppler signatures associated with limb movements from those of the torso using the ECM-Th-STFT separation algorithm. This algorithm is based on the Energy Concentration Measure (ECM) and a Th-STFT filter with a predefined threshold. They fed these novels extracted features to six existing Deep Convolutional Neural Network (DCNN) models and compared their performance. This method improved classification accuracy by up to 6%, reaching 95.4% with the SqueezeNet model compared to the original unseparated dataset.

Fei Xiang et al. [71] proposed a method that utilizes two-dimensional Principal Component Analysis (2D-PCA) followed by two-dimensional Linear Discriminant Analysis (2D-LDA) to process and classify human activity from FMCW radar data efficiently. 2D-PCA is used to reduce the dimensionality of the Doppler Time map, and the 2D-LDA algorithm is used to extract discriminant feature information. By employing a k-NN classifier, the method achieves a recognition rate of 96.40% while using the University of Glasgow dataset.

Senigagliesi et al. [64] explored the application of low-cost automotive radar combined with ML techniques in HAR. First, Range-Doppler maps are extracted. Then, authors evaluate four distinctive approaches for feature dimension reduction or further feature extraction, including 1- Using Principal Component Analysis (PCA) as feature selection followed by using SVM or k-NN as classifiers 2- Parameter extraction from Range-Doppler maps followed by the aforementioned ML algorithms as classifier. This step executed with and without applying a Butterworth filter after parameters extraction step. Finally, the last approach is feeding Range-Doppler maps to a VGG16 DL model. These methodologies were tested using a dataset collected using FMCW TI AWR 1642 radar comprising three different type of walking activities captured from 19 subjects who performed activities in hallway at Marche Polytechnic University. In their comparative analysis, the parameter extraction method, particularly when augmented with a Butterworth filter for signal smoothing, consistently outperformed the others, achieving a classification accuracy of up to 94.2% for activities, surpassing the more computationally intensive PCA and DL methods.

Feature fusion, or combining features from various domains, is another approach to increasing model performance and decreasing classifier complexity. Yu et al. [68] proposed a framework for fall detection and HAR. First, they calculated the point cloud of radar data. After applying the proposed denoising component, they introduced a new voxelization technique. They also used a novel data augmentation technique to address the commonly encountered issue of feature sparsity. Finally, they employed a dual-view CNN for classification, achieving 97.61% accuracy for fall detection and 98% for general activity classification. In this research, data were recorded from four subjects performing seven activities using TI IWR6843ISK-ODS evaluation board.

Huang et al. [65] recorded data from 17 subjects with IWR1843 TI FMCW radar and proposed a method combining point cloud data with Range-Doppler information to enhance accuracy. The authors used a 3D CNN + LSTM network to process the point cloud data and a 3D CNN for the Range-Doppler data. This fusion resulted in a 4D array that captures spatial and temporal features, improving model performance. Finally, they achieved a recognition accuracy of 97.26% for classifying seven activities.

Ding et al. [61] proposed a hybrid neural network model that leverages multidomain radar information for HAR. The model combines 1D CNN, LSTM, and 2D CNN to fuse Range-Doppler, time-Doppler, and time-range maps calculated from the University of Glasgow dataset. The authors fed Time-Range and Time-Doppler maps to two parallel 1D CNN-LSTM models. Additionally, the Range-Doppler map was fed to a 2D-CNN model. The output of these three models was fused and went through classification modules consisting of a fully connected layer. This multidomain fusion captures richer features, enhancing recognition accuracy. The best accuracy achieved in this research was 93.3%.

Sheng et al. [75] used triple-view signal maps (Range-Velocity, Range-Azimuth, and Range-Elevation) as input, integrating dynamic convolution and lightweight shuffle net structures into the SlowFast framework. The study collected data using TI IWR6843AOP evaluation board from 5 subjects in indoor and outdoor environments while performing five different activities. The methodology showed excellent performance, with 99.6% accuracy for outdoor and 97.5% for indoor activity recognition.

Kim and Seo [69] introduced a novel approach, combining Range–Time–Doppler (RTD) map with a Range-Distributed Convolutional Neural Network (RD-CNN). This approach extends traditional 2-D time-Doppler maps to 3-D by incorporating range information, enabling the extraction of more comprehensive features related to human activities. The RTD map employs Continuous Wavelet Transform (CWT) instead of Short-time Fourier Transform (STFT) to improve temporal and spectral resolution, capturing both bulk motions and micromotions. The proposed model was evaluated using the University of Glasgow radar dataset. This model

achieved a recognition accuracy of 96.49%, significantly outperforming traditional CNN models. Additionally, the model demonstrated robust performance even with unknown geometries, maintaining an average accuracy of 86.96% in different environments.

In another study on fall detection, Hsu et al. [76] introduced a light CNN model to identify Range-Doppler images with significant speed and distance variations, followed by a Bi-LSTM framework for fall detection. This method achieved an accuracy rate of nearly 96%. In this paper, authors collected data from 14 individuals, using Infineon BGT60TR13C radar in various locations, including an office, a meeting room, and 3 different bathrooms.

Innovative strides in non-contact human posture detection are showcased in the research by Liu et al. [74]. The proposed technique employs a combination of range, Doppler, and angle information to identify and classify sitting postures accurately. The authors proposed a novel method to correct angle measurement errors typically caused by range FFT bin shifts, enhancing the detection accuracy. The extracted features from TI IWR6843ISK radar raw data of 3 subjects sitting behind a desk were fed to various ML algorithms. Using an SVM model, they classified five distinct sitting postures with an average accuracy of 98.07%.

Most DL models require large datasets to provide reliable results. However, collecting data is challenging in some cases, particularly in the HAR domain. Therefore, developing models that perform well with small datasets is necessary. Yixin Zhao et al. [2] proposed an enhanced PCA combined with an improved VGG16 model to improve HAR accuracy, especially with small-scale datasets. They used a dataset from the University of Glasgow and extracted a Time-Velocity map from the raw data. They then applied an improved PCA algorithm for feature reduction and an enhanced VGG16 model for classification. This method achieved an accuracy of 96.34% and reduced training time by 12.8%, demonstrating the effectiveness of feature extraction and dimensionality reduction.

A group of researchers sought to change the data collection scenario to provide a more realistic dataset. Zhu et al. [67] presented a HAR system using distributed radar sensor networks combined with CNN-RNN architectures. They collected raw radar data from 14 participants performing nine types of activities. During the trial, participants could move along arbitrary trajectories within the specified area, and activities were performed with seamless interactive transitions, making it more reflective of daily human activity. The collected data were converted into spectrograms for input into the neural network. Time-Doppler maps were fed to a 2D-CNN model for spatial feature extraction. Subsequently, a Bidirectional Gated Recurrent (Bi-GRU) unit was used to capture the long- and short-term temporal dependencies in the extracted features. Finally, a Fully Connected Neural Network (FCNN) block made the final predictions. The best classification accuracy of approximately 90.8% was achieved in this research.

Mehta et al. [79] collected a dataset using TI IWR-1443 radar from ten subjects who performed four activities (sitting, sit-to-stand, walking, and falling) ten times each. They utilized Dynamic Time Warping (DTW) to classify activities from an unequal series of radar frames. Two strategies were employed: using raw radar array data and applying feature extraction with a Convolutional Variational Autoencoder (CVAE) before classification. A gradient-based frame equalization was applied to standardize the series length, followed by DTW for classification. For equalized data, LSTM networks were used for post feature extraction. Various feature extraction methods were explored, including supervised and unsupervised CVAE, CNN, and PCA. The LSTM approach was also adapted for handling non-equalized data, highlighting the flexibility and robustness of the method. For unequal series, the DTW algorithm achieved a testing accuracy of 80.81% using the Unsupervised Pixel-Level (UnSup-PLevel) method. The equalized series approach showed a slight improvement in performance due to the standardized frame length. The LSTM models, especially when paired with supervised CVAE for feature extraction, outperformed other techniques, achieving the highest accuracy of up to 85.74% for walking activity.

Chapter 3

Indoor Human Activity Recognition

Human Activity Recognition (HAR) is essential for monitoring daily activities in long-term care facilities, as it enables detection of abnormal patterns such as prolonged inactivity or unexpected transitions that may indicate health issues. This monitoring capability facilitates assessment of daily activity levels and triggers appropriate interventions when necessary. The system provides real-time, unobtrusive monitoring that supports timely caregiver responses, ultimately enhancing quality of care and helping elderly individuals maintain their independence.

In order to analyze the impact of radar resolution, number of installed radars, and environmental factors on the performance of the proposed approach for indoor HAR, we established two distinct datasets. The first dataset was gathered using a single radar installed in a bedroom at the Schlegel-UW Research Institute for Aging (RIA) in Canada, collecting data from three participants performing various activities. For the second dataset, two radars were installed in the RIA's living room, with data collected from eleven participants. While some activities overlap between datasets, others are unique to each collection environment. Moreover, the participants in the two datasets were entirely different, ensuring diversity in the collected data.

3.1 Preprocessing

In this section, we detail the preprocessing steps applied to the raw radar data. The data preprocessing steps are similar in both dataset. The only difference lies in how the data from the two radars were combined in the second dataset, which is described in the relevant section. Fig. 3.1 illustrates an overview of the proposed preprocessing steps for radar-based HAR. In the following sections we

provide detailed explanations of each module.

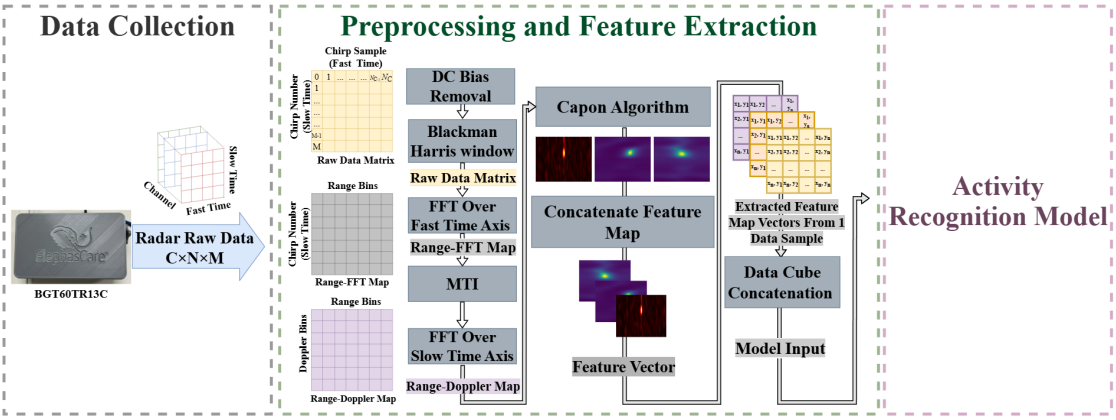


Figure 3.1: Overview of the proposed preprocessing steps of FMCW radar-based HAR.

In FMCW radars, a sequence of chirps is transmitted through the TX antenna, and each RX antenna receives the reflections of these signals. The data collected by each antenna are stored in a vector, with dimensions $C \times N \times M$, where C is the number of channels, N is the number of chirps per frame, and M is the number of samples per chirp. Each row, referred to as *fast time*, contains data from a single chirp or range bin, while each column, referred to as *slow time*, contains data from the same sample across different chirps [27]. In this study, we employ a radar with three receivers; hence, the output raw data has 3 channels.

3.1.1 Blackman-Harris Window

To preprocess the collected data, spectral leakage in the frequency domain was mitigated using the Blackman-Harris windowing function after removing the DC bias [82]. The Blackman-Harris window, commonly used in signal processing for spectral analysis and FFT, helps reduce spectral leakage by tapering the edges of the signal before applying the Fourier Transform. This minimizes discontinuities at the signal boundaries, which can otherwise introduce artificial frequency components. Fig. 3.2 illustrates the three-term Blackman-Harris Window and its transform [83]. The window function is defined as a sum of cosines (equation 3.1) [84]:

$$w(n) = a_0 - a_1 \cos\left(\frac{2\pi n}{N}\right) + a_2 \cos\left(\frac{4\pi n}{N}\right) - a_3 \cos\left(\frac{6\pi n}{N}\right) \tag{3.1}$$

where n is the index of the sample (0 to $N-1$) and N is the total number of points in the window. The coefficients a_0, a_1, a_2 and a_3 determine the window

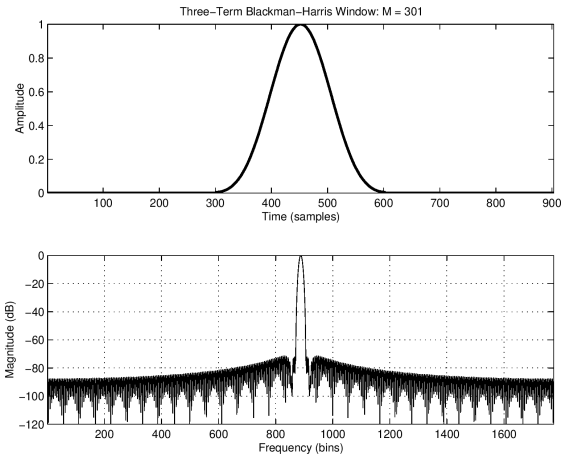


Figure 3.2: Three-term Blackman-Harris window and transform.

shape. For the 4-term Blackman-Harris window, the default coefficient values are set as follows [82]:

- $a_0 = 0.35875$
- $a_1 = 0.48829$
- $a_2 = 0.14128$
- $a_3 = 0.01168$

3.1.2 Range-FFT Map

The range of an object can be determined by applying the FFT along the fast-time axis, with the peak of the spectrum indicating the object's distance, known as the range-FFT [85]. In this context, let

$$x[n] = A\, e^{j\left(\omega_{\text{IF}}\, n\, +\, \phi_{\text{IF}}\right)}, \quad 0 \leq n < N_s \tag{3.2}$$

be the finite-duration, discrete-time radar beat signal, where $\omega_{\text{IF}} = \frac{2\pi f_{\text{IF}}}{F_s}$, f_{IF} is the beat frequency, F_s is the sampling rate, and $A\, e^{j\phi_{\text{IF}}}$ captures the amplitude and initial phase. Applying the discrete-time Fourier transform to $x[n]$ in equation 3.2 yields

$$X(\omega) = \mathcal{F}\{x[n]\} = \sum_{n=0}^{N_s-1} x[n] e^{-jn\omega} \tag{3.3}$$

$$= Ae^{j\phi_{\text{IF}}} P_{N_s}(\omega - \omega_{\text{IF}}) \tag{3.4}$$

where

$$P_N(\omega) = \sum_{n=0}^{N-1} e^{-jn\omega} = \frac{\sin(\omega N/2)}{\sin(\omega/2)} e^{-j\omega(N-1)/2} \tag{3.5}$$

where $\omega_{IF} = \frac{2\pi f_{IF}}{F_s}$ is the discrete angular frequency, and $P_N(\omega)$ is the window function with the length N, which influences the shape of the frequency response. The Discrete Fourier Transform (DFT) converts the time-domain signal into the frequency domain, revealing the frequencies present in the signal, which correspond to the ranges of the objects. The DFT of the sampled signal $X[n]$ is given by $X[k]$ and is referred as range-FFT (equation 3.6):

$$X[k] = \sum_{n=0}^{N_s-1} x[n] e^{-j\frac{2\pi kn}{N_s}} = A e^{j\phi_{\text{IF}}} P_{N_s}\left(\frac{2\pi k}{N_s} - \omega_{\text{IF}}\right) \tag{3.6}$$

for $0 \leq k < N_s$. Each frequency bin k of the DFT corresponds to a frequency $f_k = k\frac{F_s}{N_s}$. Since the beat frequency f_{IF} is proportional to the range ($f_{IF} = \frac{S2d}{C}$), the corresponding range d_k can be expressed as (equation 3.7):

$$d_k = f_k \frac{c}{2S} = k\frac{cF_s}{2SN_s} = k\frac{c}{2S\tilde{T_c}} = k\frac{c}{2\tilde{B}} \tag{3.7}$$

for $0 \leq k < N_s$, where $\tilde{T_c} = \frac{N_s}{F_s}$ is the length of the sampled IF signal in time and $\tilde{B} = S\tilde{T_c}$ is the sweep bandwidth of chirp signal. Thus, each index k in the DFT spectrum corresponds to a specific range d_k and is called range axis. The resulting DFT, when plotted against the range axis, shows peaks at these indices, which correspond to the range of detected objects.

3.1.3 Moving Target Indicator (MTI)

The remaining signal comprises two types of reflections. The first type is the echo from stationary objects in the environment, known as clutter, which includes reflections from walls, floors, beds, and chairs. The second type is the reflected signal from a person engaged in daily activities. To eliminate the undesired effect of clutter, a clutter removal algorithm must be applied. The MTI algorithm utilizes a linear filtering technique that effectively diminishes the signal strength of stationary objects while preserving the signal strength of moving targets. Among the various methods available for implementing an MTI filter in FMCW radar systems, the Finite Impulse Response (FIR) design stands out as a straightforward and efficient approach. At each time step, the peak absolute value across slow time for every range bin is denoted as $r_{i,max}$. The MTI filter output t_i is then computed

as a weighted average between this peak value and the previous MTI filter output t_{i-1}, using a weighting factor α (equation 3.8):

$$t_i = \alpha.r_{i,max} + (1 - \alpha).t_{i-1} \tag{3.8}$$

At the first time step t_0 , the MTI filter output t_i is initialized to zero. For each range bin, the MTI filter removes the influence of stationary objects by subtracting t_{i-1} from $r_{i,max}$, producing the filtered FFT value $r_{i,filt}$ (equation 3.9) [86]:

$$r_{i,filt} = abs(r_{i,max} + (1 - \alpha).t_{i-1}) \tag{3.9}$$

This method of subtracting an estimate of the background stationary clutter has been proven effective in eliminating stationary targets while having minimal impact on slow-moving objects. FIR MTI filters are advantageous due to their simple design and implementation, easily adjustable parameters, and linear phase characteristics [86, 87].

3.1.4 Range-Doppler Map

In the next step, we apply a second FFT on the vertical axis to obtain Doppler information for each channel. Then, the information from all channels is integrated to generate the Range-Doppler map. Fig. 3.3 illustrates range-FFT and Range-Doppler map.

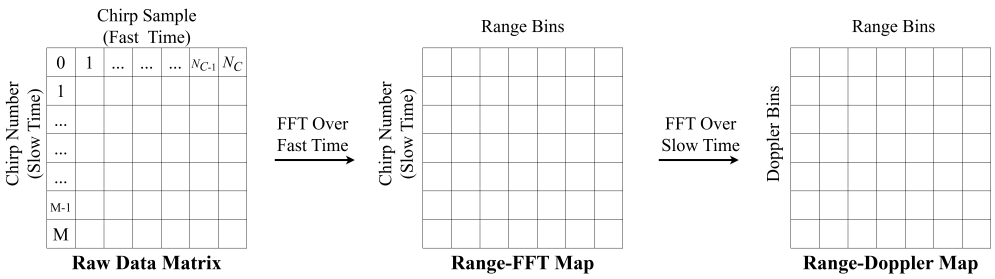


Figure 3.3: Range-Doppler map data processing.

3.1.5 Capon Algorithm

Multipath occurs when a signal travels through two or more distinct routes from the transmitter to the receiver. The specific number and characteristics of these multiple paths are influenced by the structure of the environment and the presence of various objects [19]. Due to the characteristics of electromagnetic signals, the amplitude of a multipath signal can sometimes become stronger than the amplitude

of the actual reflected signal. As a result, it is impossible to accurately approximate the accurate range profile of the object [19]. To determine the correct position of the subject and generate an accurate range-angle map, we apply the Capon algorithm to the Range-Doppler map.

The Capon algorithm is a filtering method that enables the estimation of the AoA despite noisy and distorted echo signals. The Capon beamformer, also called the Minimum Variance Distortionless Response (MVDR) beamformer, aims to minimize the output power of an array while ensuring a distortionless response at the desired angle. This approach is intended to improve signal detection in environments with noise and interference. By minimizing the output power in undesired directions, the algorithm reduces noise and interference, resulting in improvements in the Signal-to-Interference-plus-Noise Ratio (SINR). Additionally, this capability allows the Capon algorithm to perform as a spatial filter, focusing on a particular direction while suppressing signals from other directions.

The Capon algorithm achieves this goal through an optimization approach to minimize the beamformer's output power. Capon proposed a novel approach involving a constrained optimization problem. Mathematically, this is done by minimizing the variance of the beamformer's output. The objective is to determine the weight vector W so that the signal gain of the desired direction remains below a certain level (usually one). Also, it reduces the sidelobe effects but does not distort or affect signal coming from the desired direction. The solution entails determining the beamforming weights that minimize the output power while satisfying the constraint of maintaining a distortionless response. For simplicity, assuming the target is a single point, for the transmitted signal $s(t)$, the received signal at the l^{th} antenna element, reflected by the target, $x_l(t)$ can be modeled as follows:

$$x_l(t_f, t_s) = b_l \cdot e^{-j\left[2\pi f_b + \frac{2vt_s}{\lambda_{\max}} + \tau_l + \alpha_l + \Delta\psi_l(t_f, t_s)\right]} + e_l(t_f, t_s) \tag{3.10}$$

In equation 3.10, t_f is fast time index and t_s is slow time index. fast time index refers to time delay between each chirp and slow time index is chirp cycle time. b_1 is channel's mismatched magnitude, a_1 is channel's mismatched phase and f_b represents beat frequency. Also, v is radial velocity of target, λ_{max} is wavelength at start frequency, τ_1 is phase shift at l^{th} RX antenna, $\Delta\psi_l(t_f, t_s)$ is residual phase noise and $e_l(t_f, t_s)$ is additive noise. The frequency of the beat signal f_{IF} generated by an object located at a specific range in front of the radar is determined by:

$$f_b = S\frac{2 \cdot \mathrm{d}}{c} \tag{3.11}$$

In equation 3.11, the parameter S represents the rate of increase in chirp signal frequency, c denotes the speed of light and d is the distance of object from radar.

By rearranging the terms in equation 3.10 and combining the received signals

from all receiver channels into a column vector, it can be formulated as equation 3.12:

$$x_l(t_f, t_s) = \Gamma \cdot \mathbf{a}(\theta) \cdot y(v, f_b, t_f, t_s) + \mathbf{e}(t_f, t_s) \tag{3.12}$$

where

$$\Gamma = \begin{pmatrix} b_1 \cdot e^{-j\alpha_1} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & b_l \cdot e^{-j\alpha_l} \end{pmatrix}$$

$$a(\theta) = \begin{pmatrix} e^{-j\tau_1} \\ \vdots \\ e^{-j\tau_l} \end{pmatrix}$$

$$y(v, f_b, t_f, t_s) = e^{-j\left[2\pi f_b \cdot t_s + \frac{2vt_s}{\lambda_{\max}} + \Delta\psi_1(t_f, t_s)\right]}$$

where θ is AoA of the object in front of radar, Γ depends on the channel gain/phase mismatches, and a is the steering vector which depends on AoA. If there are multiple targets at different ranges, each with a distinct beat frequency f_{IF} , the received vector is the sum of all individual vectors received from each target. This can be mathematically expressed as expressed in equation 3.13:

$$x_l(t_f, t_s) = \Gamma \cdot A(\theta) \cdot Y(v, f_b, t_f, t_s) + e_l(t_f, t_s) \tag{3.13}$$

Here, A is an $L * K$ matrix where K represents the number of targets, with each column corresponding to the steering vector of a specific target. Matrix Y is a diagonal matrix with its diagonal entries populated by elements of $y(v, f_b, t_f, t_s)$. In equation 3.13, the vectors A, v, and f_b represent the unknown parameters associated with the targets. Among these, only the elements of A vary with receiver channel indices, indicating that matrix y does not affect the covariance of x. Specifically, the covariance matrix of x can be computed as equation 3.14, assuming the additive noise is uncorrelated with y:

$$R = \mathbb{E}\left\{\mathbf{x} \cdot \mathbf{x}^H\right\} = P_s \cdot \Gamma \cdot A(\theta) \cdot A^H(\theta) \cdot \Gamma^H + R_n \tag{3.14}$$

where P_s denotes the power of the signals, R_n represents the noise covariance matrix, assumed to be positive definite under the assumption of independent noise across receivers, The notation $E\{.\}$ denotes the expectation, and x^H denotes the Hermitian transpose of x, yielding the autocorrelation of the signal x.

Autocorrelation quantifies how similar a signal is to itself at different time points, with matrix R reflecting the self-similarity of signal x over time. the power output Φ can be formulated as equation 3.15:

$$\Phi = w^H R w \tag{3.15}$$

where Φ represents the output power, where w denotes a complex weight vector applied to the inputs of the array by the beamformer. The Capon beamforming problem is subsequently posed as a constrained optimization task. R is the covariance matrix of the input arrays and H is Hermitian transpose (conjugate transpose). The Capon beamforming problem is then formulated as a constrained optimization problem (equation 3.16):

$$\min_w w^H R w, \quad \text{s.t.} \quad w^H a(\theta_0) = 1. \tag{3.16}$$

This problem can be solved using standard Lagrange multiplier. The Lagrangian L is defined as follows (equation 3.17):

$$L = w^H R w - \lambda(w^H a - 1) \tag{3.17}$$

where λ is the Lagrange multiplier. Set differentiation L with respect to w^* gives optimal weight:

$$\frac{\partial L}{\partial w^*} = 2Rw - \lambda a = 0. \tag{3.18}$$

Solving for W results in equation 3.19:

$$W = \frac{R^{-1}a}{a^H R^{-1} a} \tag{3.19}$$

In equation 3.19, R^{-1} denotes the inverse of the covariance matrix R. The power spectrum of the beamformer output at direction θ represents the output power when the array is oriented towards that direction, expressed in equation 3.20:

$$\Phi(\theta) = w(\theta)^H R w(\theta). \tag{3.20}$$

Thus, the capon output power spectrum express as equation 3.21:

$$\Phi(\hat{\theta}) = \frac{1}{a^H(\hat{\theta}) \cdot R^{-1} \cdot a(\hat{\theta})} \tag{3.21}$$

where $\hat{\theta}$ is a test unknown AoA.

In summary, equation 3.21 is derived from optimal beamforming principles applied to the covariance matrix R. For a comprehensive understanding, readers are encouraged to consult textbooks or papers dedicated to array signal processing

that delve into the Capon beamforming algorithm, such as this paper [49], as it exceeds the scope of this study. The final 3D data structure, called the data cube, is formed by combining the RD, RA, and RE feature maps.

3.1.6 Principal Component Analysis (PCA)

In order to simplify the models and improve activity classification performance, we reduced the feature vector size before feeding it into the ML models. PCA is a widely used statistical technique that transforms a set of correlated variables into a set of uncorrelated variables, known as principal components [88]. The primary goal of PCA is to reduce the dimensionality of the data while retaining as much variance as possible. This process is crucial for handling high-dimensional datasets, which are often difficult to analyze and interpret directly.

Suppose we have a dataset $X = \{x_1, x_2, ..., x_n\}$ where each data point x_n is a vector in $\mathbb{R}^D$, and the dataset has been centered (i.e., the mean of the dataset has been subtracted from each data point). The goal of PCA is to project these data points onto a subspace of lower dimensionality M (where $M < D$) in such a way that the variance of the projected data is maximized. The practical steps for implementing PCA are as follows:

1. **Standardize the Data**
   Before applying PCA, it is crucial to standardize the data (equation 3.22). Standardization ensures that each feature contributes equally to the analysis, particularly when features have different units or variances. This is done by subtracting the mean and dividing by the standard deviation for each feature, resulting in all features having a mean of zero and a standard deviation of one.

$$x_{ij}^{scaled} = \frac{x_{ij} - \mu_j}{\sigma_j} \tag{3.22}$$

   where x_{ij} is the value of feature j for observation i, μ_j is the mean of feature j, and μ_j is the standard deviation of feature j.

2. **Compute the Covariance Matrix**
   Once the data is standardized, the next step is to compute the covariance matrix S (equation 3.23). This matrix captures the extent to which variables vary from their means relative to each other, summarizing the relationships between pairs of variables in the dataset X.

$$S = \frac{1}{N-1} X^T X \tag{3.23}$$

where N is the number of observations, and X is the matrix of data points with each row representing an observation and each column a feature.

3. **Compute Eigenvectors and Eigenvalues**
   Perform eigendecomposition on the covariance matrix S to obtain the eigenvalues and eigenvectors (equation 3.24):

$$Sv_i = \lambda_i v_i \tag{3.24}$$

   where λ_i is the eigenvalue and v_i is the corresponding eigenvector.

4. **Sort Eigenvalues and Select Principal Components**
   Sort the eigenvalues in descending order and arrange the eigenvectors accordingly. The eigenvectors corresponding to the largest eigenvalues are the directions of maximum variance.

5. **Select the Number of Principal Components**
   Choose the top M eigenvectors based on the cumulative variance explained by the eigenvalues. Typically, you choose M such that the cumulative variance is around 90-95%.

6. **Project the Data onto the New Subspace**
   Multiply the original data matrix X by the matrix B formed by the selected M eigenvectors.

$$Z = XB \tag{3.25}$$

   in equation 3.25, Z is the matrix of transformed data in the new subspace.

We evaluated the performance of various numbers of principal components in PCA and selected the value that yielded the optimal performance. PCA helps to reduce noise and handle multicollinearity by transforming correlated features into uncorrelated principal components. This simplifies the model, speeds up training, and reduces the chance of overfitting. It also improves computational efficiency and model performance.

3.2 Bedroom Dataset

In the first stage of this work, we propose a framework that leverages RD, RA, and RE feature maps extracted from FMCW radars for indoor activity recognition. This approach addresses the growing need for innovative Ambient-Assisted Living (AAL) solutions that utilize HAR to support independent living for older adults.

Such tools enable remote health monitoring, facilitate rehabilitation, enhance overall well-being, and allow for early detection of potential health issues. The radar data is processed using the methodology detailed in Section 3.1 to generate RD, RA, and RE maps, which serve as input features for various ML and DL models. Fig. 3.4 illustrates the prototype system architecture for HAR implementation.

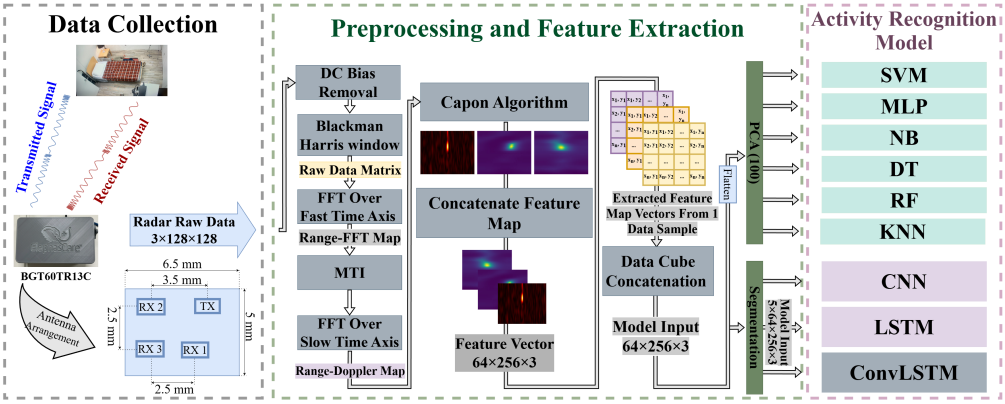


Figure 3.4: Overview of the proposed framework for FMCW radar-based HAR. SVM: Support Vector Machines, MLP: Multi-Layer Perceptron, NB: Naive Bayes, DT: Decision Tree, RF: Random Forest, KNN: K-Nearest Neighbor.

3.2.1 Materials and Method

In the first step, we utilized a single mmWave FMCW radar system, the BGT60TR13C developed by Infineon Technologies AG. This device transmits a sawtooth waveform at frequencies between 58–63.5 GHz using one transmitter and three receivers, with an adjustable chirp duration. It employs a 12-bit Analog-to-Digital Converter (ADC), providing a 74 dB dynamic range and up to 4 Mega Samples Per Second (MSps). The radar configuration and key parameters are summarized in Table 3.1. Due to its L-shaped antenna array, the device can estimate the horizontal angle (azimuth) using the RX1 and RX3 antennas, the vertical angle (elevation) using the RX2 and RX3 antennas, and the target's velocity [58]. Fig. 3.5 shows the BGT60TR13C radar, its shield, and their placement on the baseboard.

Figure 3.5: BGT60TR13C radar device.

Table 3.1: Radar configuration and specification.

Parameters	Value
Radar Model	BGT60TR13C
Start Frequency	61 GHz
End Frequency	62 GHz
Transmit Output Power	1-8 dBm
ADC Sampling Rate	2 Msps (Max)
Frame Rate	10
Chirps Per Frame	128
Number of Tx Antennas	1
Number of Rx Antennas	3
Range Resolution	15 cm
Max Unambiguous Range	4.8 m

Moreover, this radar has a relatively low output power (maximum 8 dBm), posing no negative health effects on humans [89]. Several studies have demonstrated its performance and efficiency [90–94]. Additionally, the Federal Communications Commission (FCC) has designated the 60 GHz band as a priority frequency range for healthcare-related applications [95]. Fig. 3.6 illustrates the antenna arrangement of the BGT60TR13C radar.

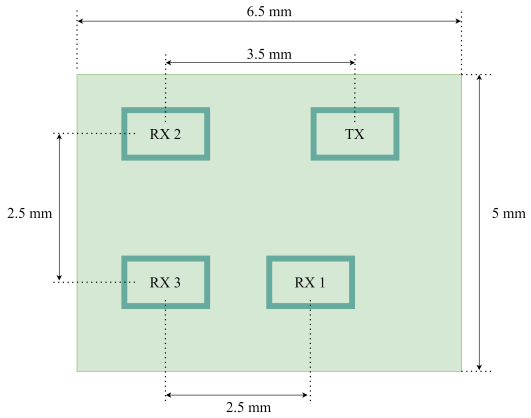


Figure 3.6: BGT60TR13C top view and antenna arrangement.

3.2.2 Data Collection

For data collection, the radar sensor was installed in a bedroom environment and positioned to ensure optimal coverage, minimizing side-lobe interference and maximizing detection performance. It was mounted at a height of 210 cm and angled

downward by 30 degrees. Three healthy subjects participated in the study, completing a total of 16 trials. Two subjects performed six trials each, while the third performed four.

Each trial began with a one-minute recording of an empty room. Participants then carried out a set of predefined activities: walking for 2 minutes, sitting on a bed for 2 minutes, lying on the bed for 5 minutes, and lying on the floor for 5 minutes. In some sessions, participants also sat on a chair for 2 minutes. Data were collected continuously, capturing both steady-state activities and transitions.

All transition movements were grouped into a single "Transition" class to simplify classification and maintain data balance. The dataset consists of seven activity classes: Empty Room, Walking, Sitting on the Bed, Sitting on a Chair, Lying on the Bed, Lying on the Floor, and Transition.

The data collection protocol was approved by the University of Waterloo's Research Ethics Committee, and all procedures complied with relevant safety and ethical guidelines. The radar system used in this study is FCC-certified, ensuring adherence to applicable regulatory standards. Fig. 3.7 illustrates the bedroom layout and radar installation, and Fig. 3.8 shows an example of a participant performing each activity along with the corresponding feature maps.

Figure 3.7: Bedroom layout and radar placement in the home-like data collection environment.

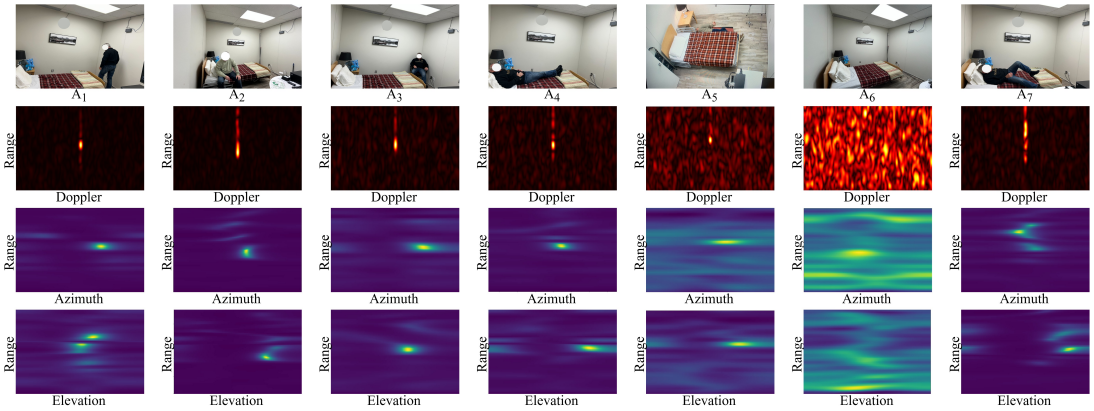


Figure 3.8: Examples of a participant performing the activities with corresponding feature maps. (A_1) walking, (A_2) sitting on the bed, (A_3) sitting on the chair, (A_4) lying down on the bed, (A_5) lying down on the floor, (A_6) empty room, (A_7) transition.

3.2.3 Activity Recognition

A ML or DL model classifies each input data vector into one of the predefined activity categories. In this work, we implemented several classical ML classifiers as well as multiple DL architectures, and then evaluated and compared their performance in recognizing human activities.

After extracting RD, RA, and RE feature maps from the raw radar data, as described in Section 3.1, we applied PCA before feeding the data into ML models. This step reduces dimensionality, accelerates training, and helps mitigate overfitting.

In contrast, for DL models such as CNN, LSTM, and ConvLSTM, no feature-reduction technique was applied. Instead, the data was segmented into sets of five consecutive data cubes to balance temporal granularity with computational efficiency. If all five cubes shared the same activity label, they were merged into a single segment. If the labels within a segment differed (for example, three matching labels and two differing), the matching cubes were discarded, segmentation was paused, and resumed starting from the next sequence of cubes corresponding to the subsequent activity. This procedure ensures label consistency across segments and prevents merging conflicting activity annotations.

Model hyperparameters were determined using a systematic grid search strategy, ensuring an effective balance between computational efficiency and predictive performance. Any parameters not explicitly mentioned were left at their default values.

Support Vector Machine (SVM):

SVM is a powerful classification algorithm that separates data using an optimal hyperplane and can model both linear and non-linear decision boundaries through kernel functions [96, 97].

- `probability=True`
- `C=10`
- `kernel='rbf'`

Multi-Layer Perceptron (MLP):

MLP is a neural network architecture composed of an input layer, one or more hidden layers, and an output layer. It processes data through feedforward connections and is trained using backpropagation to minimize prediction error via gradient descent [98, 99].

- `hidden_layer_sizes=(128, 64)`
- `max_iter=300`

Naïve Bayes (NB):

NB is a probabilistic classifier based on Bayes' theorem, assuming feature independence given the class label. Despite this assumption, it often performs well across various tasks due to its simplicity and efficiency [100].

Decision Tree (DT):

DT segments data based on feature values, leading to classification or prediction at the leaf nodes. They are non-parametric and can adapt to complex datasets without a predefined model structure [101].

Random Forest (RF):

RF is an ensemble learning method that constructs multiple decision trees during training and outputs the mode of their predictions for classification tasks or the mean for regression [102].

- `n_estimators=100`

k-Nearest Neighbor (kNN):

k-NN is an instance-based learning algorithm that classifies data points based on the majority class of their nearest neighbors [103].

- `n_neighbors=9`

Convolutional Neural Network (CNN):

The implemented CNN model consists of four 3D convolutional blocks with 8, 16, 32, and 64 filters, respectively. Each block uses a $(3 \times 3 \times 3)$ kernel and the Exponential Linear Unit (ELU) activation function, followed by a $1 \times 2 \times 2$ max pooling layer. The network includes two fully connected layers: one with 128 neurons and another with 7, 6, 5, or 4 neurons (corresponding to the number of activities). Dropout is applied after each layer with rates increasing from 0.2 to 0.5, excluding the output layer. The model is trained using the Adam optimizer with a cross-entropy loss function, a learning rate of 1×10^{-3}, and a maximum of 100 epochs with early stopping (patience = 10, min_delta = 0). The architecture is shown in Fig. 3.9.

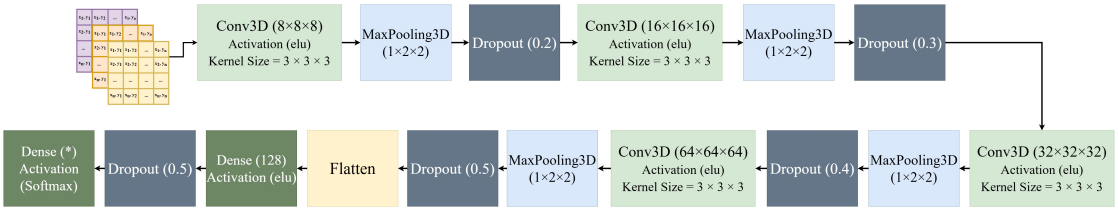


Figure 3.9: Architecture of the 3D CNN model implemented for activity classification using the bedroom dataset.

Long Short-term Memory (LSTM)

The LSTM architecture consists of two Bi-LSTM layers with 256 units and a dropout rate of 0.5. It includes two fully connected layers: one with 128 neurons and another with 7, 6, 5, or 4 neurons (corresponding to the number of activities). The model is trained using the Adam optimizer with a decay factor of 0.9 and an initial learning rate of 1×10^{-3}. This learning rate is reduced to 10% of its initial value at the 200th epoch. Training proceeds for up to 400 epochs with early stopping (patience = 40, min_delta = 0). The architecture is shown in Fig. 3.10.

Convolutional Long Short-Term Memory (ConvLSTM)

The ConvLSTM model includes a single ConvLSTM block with 32 filters, a (3×3) kernel, and ReLU activation. This block is followed by batch normalization, 3D

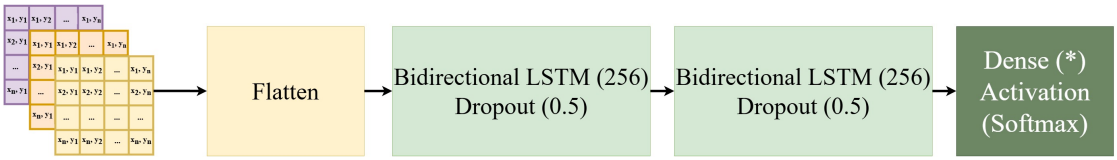


Figure 3.10: Architecture of the LSTM model implemented for activity classification using the bedroom dataset.

max pooling ($1 \times 2 \times 2$), and dropout (0.3). The output is then passed to a fully connected layer with 64 neurons and ReLU activation, followed by dropout (0.5). The final fully connected layer contains 7, 6, 5, or 4 neurons, corresponding to the number of classes, with softmax activation. The model is trained using Stochastic Gradient Descent (SGD) with momentum 0.9 and weight decay 1×10^{-4}, optimizing a categorical cross-entropy loss. Training uses a learning rate of 1×10^{-4}, a batch size of 64, and a maximum of 100 epochs with early stopping (patience = 10, min_delta = 0). Fig. 3.11 shows the architecture.

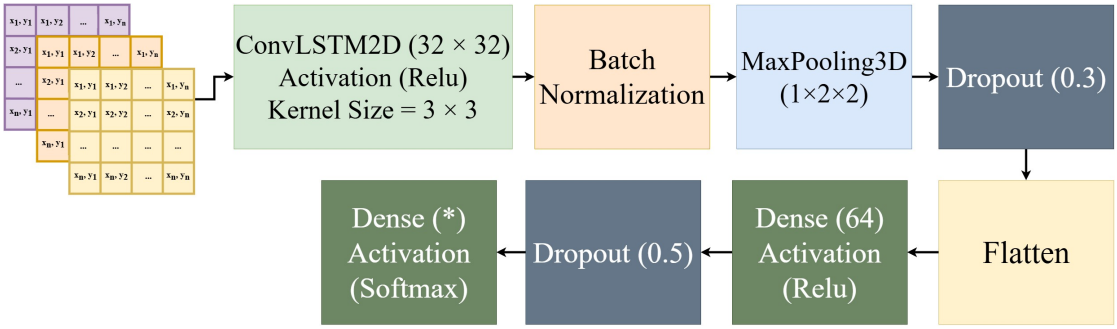


Figure 3.11: Architecture of the ConvLSTM model implemented for activity classification using the bedroom dataset.

The ConvLSTM [104] architecture, which combines CNN [105,106] and LSTM [107] networks, captures spatial relationships and temporal dependencies essential for analyzing complex radar signals. Its effectiveness has been investigated in fall detection [108], gesture recognition [109], and activity recognition [110]. The LSTM component uses memory cells with gating mechanisms to regulate information flow, enabling selective retention or discarding of state data. Conventional ML models often struggle with the similarities between RD, RA, and RE maps, making it hard to extract subtle patterns. DL models like CNN excel at uncovering these features, and ConvLSTM leverages this ability to process intricate radar data effectively [108, 109].

Traditional LSTM excel at modeling temporal sequences but struggle to capture spatial structures in radar data. Meanwhile, CNN extract spatial features but are not designed for sequence modeling. A straightforward combination of CNN and

LSTM models may fail to effectively capture both spatial relationships within frames and temporal dependencies across consecutive frames [111]. ConvLSTM addresses these limitations by integrating convolutional layers into LSTM gates, allowing simultaneous modeling of spatial and temporal patterns. This makes ConvLSTM particularly suited for joint spatial-temporal analysis, such as activity recognition using FMCW radar signals [111] and human action recognition in videos [112].

3.2.4 Experimental Setup

To validate our approach, we employed two distinct strategies. In the first strategy, Cross-Scene Validation (CSV), we used 80% of the data from 14 of the 16 distinct scenes (recording sessions) for training and reserved the remaining 20% for validation. The two remaining scenes were held out for testing. In the second strategy, Leave-One-Person-Out Cross-Validation (LOPO-CV), we repeated the evaluation three times—each time leaving one subject out—and computed the accuracy and F_1-score. In each fold, 80% of the data from the two available subjects was used for training, while the remaining 20% was used for validation. The unseen subject's data served exclusively as the test set. For example, in the first iteration, the first subject's data was held out for testing, while the second and third subjects' data were split for training and validation. In subsequent iterations, the roles were rotated accordingly. Tables 3.2 and 3.3 summarize the dataset distribution for each validation strategy.

Table 3.2: Activity sample counts under Cross-Scene-Validation approach.

Activity	**Training**	**Validation**	**Testing**	**Total**
Walking (A_1)	6,495	1,625	1,250	9,370
Sitting on the Bed (A_2)	5,130	1,285	705	7,120
Sitting on the Chair (A_3)	1,255	310	1,205	2,770
Lying Down on the Bed (A_4)	12,870	3,220	1,540	17,630
Lying Down on the Floor (A_5)	9,810	2,455	3,040	15,305
Empty Room (A_6)	3,405	850	640	4,895
Transition (A_7)	4,305	1,075	790	6,170
Total	**43,270**	**10,820**	**9,170**	**63,260**

Table 3.3: Activity sample counts per subject in Leave-One-Person-Out Cross-Validation approach. (A_1) walking, (A_2) sitting on the bed, (A_3) sitting on the chair, (A_4) lying down on the bed, (A_5) lying down on the floor, (A_6) empty room, (A_7) transition. (S_1) subject one, (S_2) subject two and (S_3) subject three.

Activity	S_1	S_2	S_3
A_1	3020	3800	2583
A_2	1800	2661	2684
A_3	1205	1568	-
A_4	5327	6142	6181
A_5	5993	6252	3082
A_6	1514	2030	1382
A_7	1749	2170	2298

To evaluate the models, we applied a normalization technique in which the mean and standard deviation were computed for each feature channel using the combined training and validation data. These parameters were then used to normalize both the training–validation subset and the test set, ensuring consistent scaling and improving the learning process. All computations were carried out on a high-performance laptop equipped with an AMD Ryzen *7 6800H* processor, *32* GB of RAM, and an NVIDIA GeForce RTX *3060* GPU running Windows *11*. Model development was performed using Python *3.8.19*, Scikit-learn *1.3.2*, and TensorFlow *2.10*.

For a comprehensive evaluation, multiple metrics were reported for each validation approach. For ML models, accuracy along with weighted and macro-averaged F_1-scores is presented. For DL models, accuracy, precision, recall, and the F_1-score are reported.

3.3 Living Room Dataset

In the next stage of this study, two wall-mounted FMCW radars were deployed in a realistic living room environment to monitor human activities. The radar signals were processed using the methodology described in Section 3.1 to generate RD, RA, and RE maps. The RA and RE feature maps were then used as input to a 3D CNN, forming the basis of a novel fall detection system designed to address the growing need for scalable solutions in elderly care.

Although Doppler signatures are promising for fall detection, the Doppler signature of certain fall types can closely resemble other movements, such as sitting, making differentiation challenging. Additionally, the Doppler signature of the same activity may vary significantly when observed from different angles. Consequently, relying solely on Doppler information can lead to a high rate of false

alarms, reducing model reliability in fall detection tasks [113, 114].

Beyond fall detection, the RD, RA, and RE representations were used to classify a broader set of activities performed within the living room environment. We implemented and evaluated several DL architectures, including a 3D CNN, a modified ResNet18 combined with an Long Short Term Memory (LSTM) network, and a hybrid CNN+LSTM model. The proposed system achieved high accuracy while maintaining computational efficiency, making it well-suited for deployment in AAL environments.

3.3.1 Materials and Method

In this phase, we utilized two mmWave FMCW radar systems identical to those used in Section 3.2. To investigate the impact of deploying multiple radars, optimize radar specifications, and maintain a balanced computational complexity for the implemented DL models, we adjusted their configurations. Specifically, two radars were installed in the living room, each positioned to capture data from a different perspective. Table 3.4 provides a comprehensive summary of the radar parameters employed in this phase.

Table 3.4: Radar configuration and specification.

Parameters	**Value**
Radar Model	BGT60TR13C
Start Frequency	61 GHz
End Frequency	61.5 GHz
Transmit Output Power	5 dBm
ADC Sampling Rate	2 Msps
Frame Rate	10
Chirps Per Frame	128
Number of Tx Antennas	1
Number of Rx Antennas	3
Range Resolution	30 cm
Max Unambiguous Range	4.8 m

3.3.2 Data Collection

To collect the data, two radar devices were installed at a height of 210 cm above the ground—one positioned above a window and the other above a television. Both devices were tilted at an angle of 30° to maximize room coverage. Eleven healthy subjects participated in the study, each performing nine specific activities

within the living room. All participants performed every activity at each of five predefined locations, as illustrated in Fig. 3.13.

The activities included standing, sitting on/getting up from a sofa, walking, picking an object from a table, picking an object from the floor, sitting on a sofa, lying on a sofa, sitting on the floor, and lying on the floor. Fig. 3.12 shows examples of a participant performing each of the specified activities.

Data collection procedures were approved by the Schlegel-UW RIA and the University of Waterloo. All activities adhered to the relevant safety protocols and ethical guidelines. Sessions were segmented by both activity type and location to ensure controlled, consistent, and reproducible data acquisition conditions.

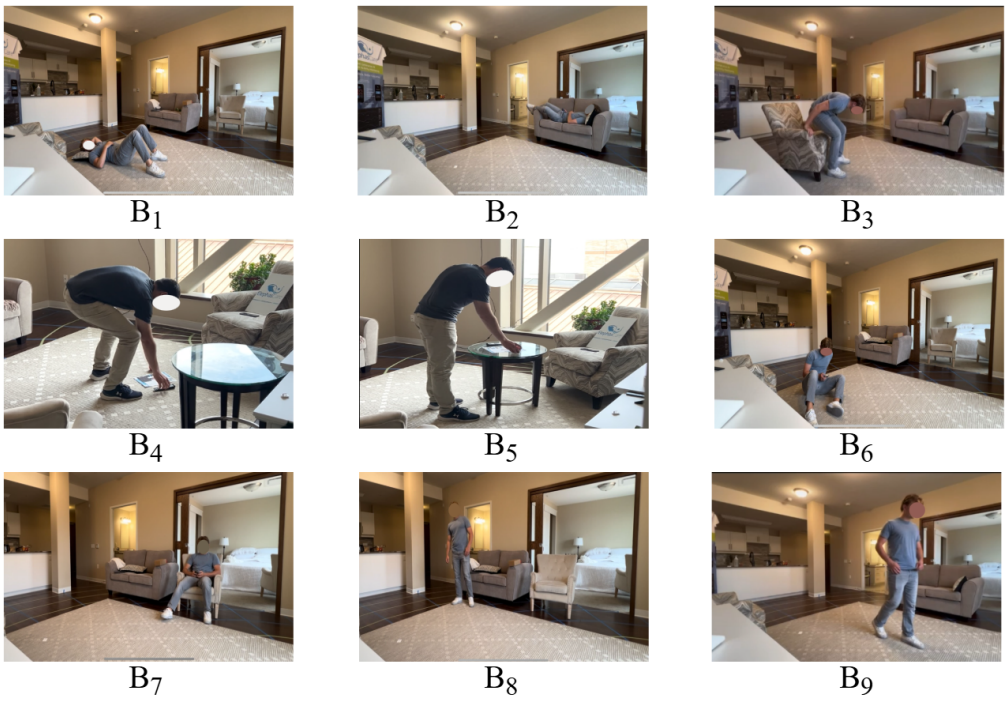


Figure 3.12: Room layout indicating radars positions and activity locations. Activities: (B_1) lying on the floor, (B_2) lying on a sofa, (B_3) sitting on/getting up from a sofa, (B_4) picking an object from the floor, (B_5) picking an object from a table, (B_6) sitting on the floor, (B_7) sitting on a sofa, (B_8) standing, (B_9) walking.

Figure 3.13: The living room layout indicating radars positions and activity locations.

3.3.3 Fall Detection

As an initial step in analyzing the living room dataset, this study investigates the binary classification of fall versus non-fall scenarios. For the non-fall class, two activities are considered: sitting on a sofa and lying on a sofa. The fall class is represented by lying on the floor. These activities were chosen due to the relatively clear distinction between static and dynamic postures, compared to the more challenging task of differentiating visually similar static activities.

Due to ethical and practical constraints, real fall data was not collected, as inducing falls in human participants poses significant safety risks [115]. Instead, lying on the floor is employed as a surrogate for fall events. This choice is supported by the fact that lying on the floor is uncommon in residential care units and is often associated with adverse incidents, including falls. Thus, detecting an individual lying on the floor is a practical and reliable indicator of a fall [116]. Fig. 3.14 illustrates the architecture of the proposed system prototype.

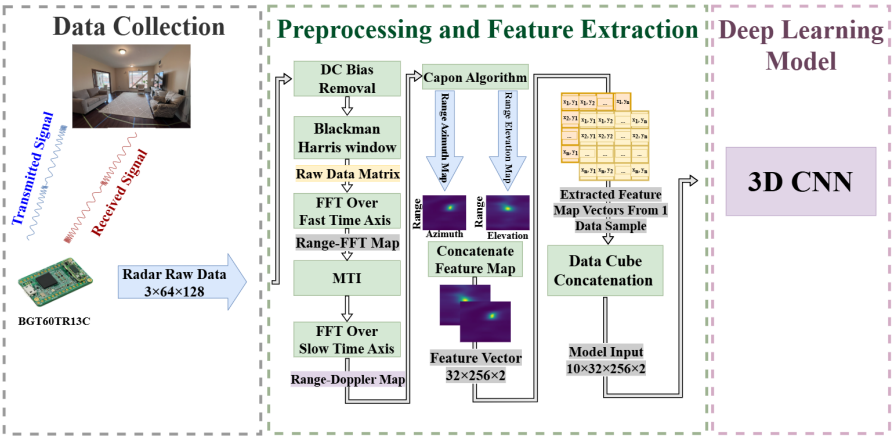


Figure 3.14: Overview of the proposed fall detection system.

Existing research supports this approach, demonstrating that the inability to rise after a fall is a crucial marker of fall-related incidents, particularly among older adults. For instance, one study indicated that approximately 80% of individuals aged 90 or older are unable to get up after a fall, with many remaining on the floor for extended periods due to a lack of assistance or cognitive impairment [117]. Additionally, most existing datasets for fall detection are collected from young, healthy participants simulating falls. For safety, these simulations are typically conducted on mattresses. However, in real-life falls, people often instinctively try to break their fall with their hands—a reaction that is not accurately reproduced in controlled trial settings. As a result, these datasets may not fully represent real-world fall incidents, thereby limiting the generalizability of the developed models [118]. By utilizing lying on the floor as an approximation of falls, this methodology ensures participant safety while maintaining the relevance of the dataset for practical applications, such as AAL environments where timely intervention is crucial. The following sections explore the implemented DL model in detail.

Deep Learning Model Architecture

After preprocessing the raw data, as described in Section 3.1, we construct RA and RE feature maps by combining angular information from the Capon beamformer with range information obtained through the range Fast Fourier Transform (range-FFT). Fig. 3.15 shows examples of a participant performing each scenario, along with the corresponding feature maps.

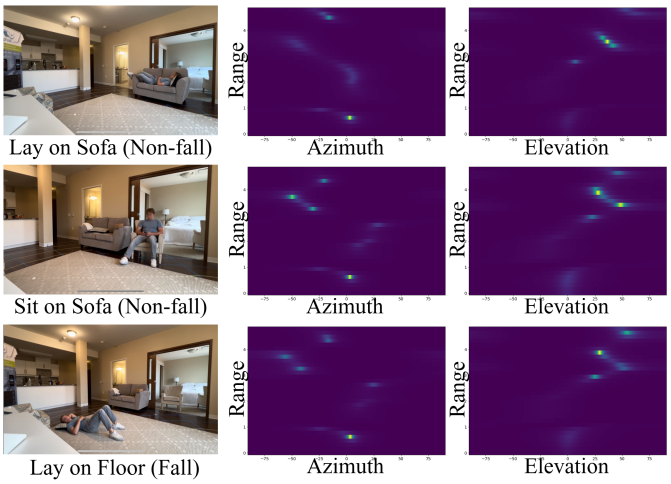


Figure 3.15: Examples of a participant performing the activities with corresponding feature maps.

Following the computation of RA and RE feature vectors, these vectors are concatenated to form a 3D data structure referred to as a *data cube*. Every ten consecutive data cubes are then merged, yielding a compact feature representation that incorporates range, azimuth, elevation, and temporal information.

Given this 3D structure, a 3D CNN is employed to capture the spatial and temporal dependencies present in the data cubes. The model architecture is illustrated in Fig. 3.16.

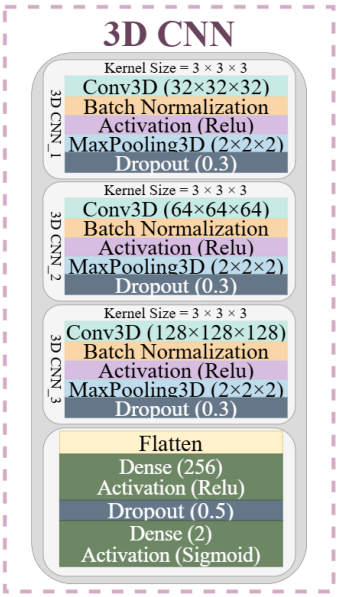


Figure 3.16: Architecture of the 3D CNN model implemented for fall detection using the living room dataset.

The 3D CNN is particularly suited for this task, as it captures both spatial and temporal dimensions across three axes, whereas traditional 2D CNN are limited to spatial information only. This capability is crucial for our application, where accurately classifying activities like sitting and lying requires capturing temporal dynamics and spatial patterns within RA and RE maps. Moreover, due to the inherent similarities between radar-based RA and RE maps, it is challenging to distinguish subtle patterns and extract meaningful features solely through conventional methods. DL models, such as CNN, help reveal hidden information within these feature maps, allowing the model to generalize more effectively and perform robustly in real-world applications. This robustness is essential, as it allows the model to adapt to small environmental changes that may otherwise alter the feature maps, ensuring consistent performance across different settings. Our approach aligns with prior studies that have demonstrated the effectiveness of 3D CNN in applications like continuous hand gesture recognition [119], fall detection [120] and human action recognition [121].

For training, a learning rate of *0.0001*, a batch size of *8*, and a maximum of *200* epochs were used. Early stopping was applied with a patience of *20* epochs and minimum improvement set to *0* to prevent overfitting. The SGD optimizer with momentum *0.9* and weight decay *1e-4* was employed alongside the *Binary Cross-Entropy* loss function, which is appropriate for binary classification tasks. These hyperparameters were selected through a grid search to identify the optimal configuration.

Data Normalization Prior to training, data normalization was applied to ensure consistency across the dataset and to facilitate model convergence. The mean and standard deviation for each channel were computed from the training–validation set and used to normalize both the training–validation and test sets.

Data Augmentation Several data augmentation techniques were applied during training to enhance the model's robustness and generalization. These augmentations include Gaussian noise with a mean of *0* and a standard deviation of *0.1*, added with a probability of *0.5*. Additionally, a time-warping (scaling) augmentation was applied, in which data samples were scaled by a factor randomly chosen between *0.9* and *1.1*, with a probability of *0.5*.

Experimental Setup

The experiments were conducted on a system equipped with an AMD Ryzen *7 6800H* processor, *32* GB of RAM, and an NVIDIA GeForce RTX *3060* GPU, running Windows *11*. Python version *3.8.19* and TensorFlow version *2.10* were used to implement and execute the DL model. The model was trained and validated on data collected from 10 subjects, with 80% of the data used for training and the remaining 20% for validation. An evaluation is conducted on data from an additional, unseen subject to assess the model's generalization capability. The total number of samples used for the fall and non-fall scenarios is presented in Table 3.5.

Table 3.5: Total sample count for each scenario.

Scenario	Train	Validation	Test
Fall	96,870	24,220	12,500
Non-fall	196,920	49,240	25,000
Total	**293,790**	**73,460**	**37,500**

3.3.4 Activity recognition

In this stage, activities performed by the subjects were classified. Initially, nine different activities were classified using the 3D CNN model. Subsequently, the same set of activities was classified using the ResNet+LSTM model. To reduce training and testing times and to improve model performance, similar activities—such as picking an object from a table and from the floor, standing, and Sitting on/getting up from a sofa—were merged into a single class called *In-place movement*. As a

result, the classification problem was reduced from nine classes to six. The six-class classification was conducted using a modified ResNet+LSTM model (Section 3.3.4), where adjustments were made to the final layer. Finally, the six-class classification was also performed using a CNN+LSTM model. Fig. 3.17 illustrates the architecture of the proposed system prototype.

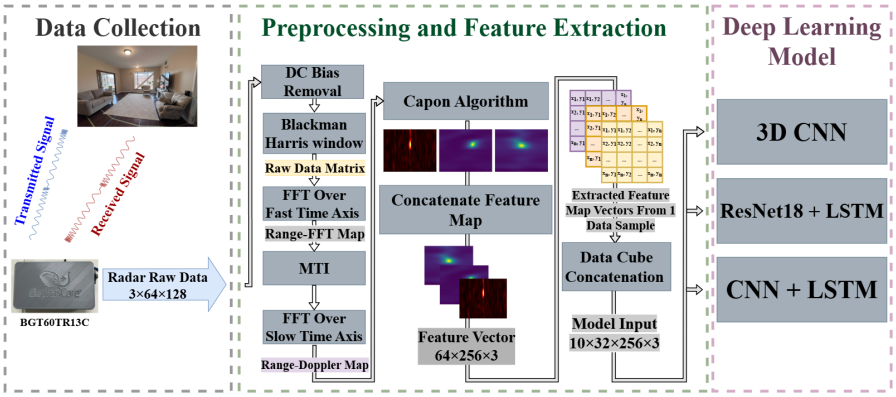


Figure 3.17: Overview of the proposed activity recognition system.

Deep Learning Models

Following the preprocessing of raw data, as detailed in Section 3.1, the RD, RA, and RE feature maps were employed for the activity recognition experiments. This contrasts with the fall detection experiment in Section 3.3.3, where only the RA and RE feature maps were used. Fig. 3.18 illustrates examples of a participant performing each specified activity, alongside the corresponding feature maps.

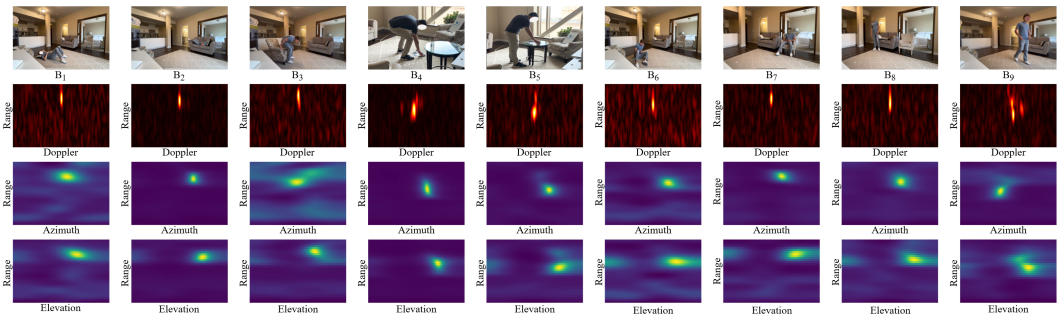


Figure 3.18: Examples of a participant performing the activities with corresponding feature maps. Activities: (B_1) lying on the floor, (B_2) lying on a sofa, (B_3) sitting on/getting up from a sofa, (B_4) picking an object from the floor, (B_5) picking an object from a table, (B_6) sitting on the floor, (B_7) sitting on a sofa, (B_8) standing, (B_9) walking.

3D CNN In the first step, nine activities were classified using the proposed 3D CNN model introduced in Section 3.3.3. For training, a learning rate of *0.001*, a batch size of *32*, and a maximum of *100* epochs were used as hyperparameters. An early stopping criterion with a patience of *10* epochs and a minimum improvement (*min_delta*) of *0* was also applied.

Additionally, the number of neurons in the final dense layer was set to 9 to match the number of activity classes in the task. All other hyperparameters remained the same as those described for the model in Section 3.3.3.

Modified ResNet-18 + LSTM In this study, a hybrid DL model was developed, combining CNN for spatial feature extraction and Bi-LSTM networks for sequential modeling. This design enables effective learning from spatio-temporal data, such as sequences of video frames or structured time-series images. The model is optimized for multiclass classification with nine target classes.

The CNN component is based on a ResNet-18 architecture [122], modified to enhance computational efficiency. To lighten the ResNet-like CNN while maintaining strong representational power, *SeparableConv2D* layers were employed instead of standard *Conv2D* layers. Separable convolutions factorize a standard convolution into a depthwise convolution, which applies a single filter to each input channel, followed by a pointwise convolution that linearly combines the output channels [123, 124]. This decomposition significantly reduces both the number of parameters and the computational cost compared to standard convolutions [123].

The CNN component processes spatial information, while the Bi-LSTM captures temporal dependencies by analyzing the sequence of CNN-extracted features across time steps. Regularization techniques, including *dropout*, *batch normalization*, and *L2 weight regularization*, were integrated to prevent overfitting and to promote generalization. Each convolutional operation is followed by batch normalization to stabilize and accelerate the training process [125], and a $ReLU$ activation function introduces non-linearity. Additionally, *MaxPooling* is applied early in the network to downsample feature maps and reduce computational complexity while preserving essential spatial features.

To apply the CNN independently at each time step, a *TimeDistributed* wrapper is employed, ensuring that the same CNN weights are shared across all 10 frames, yielding a sequence of CNN-processed feature vectors. After extracting spatial features from each frame, a Bi-LSTM network with 256 hidden units processes the sequence both forward and backward in time, capturing long-range dependencies and contextual information from both past and future frames [126]. A dropout rate of 0.5 is applied within the Bi-LSTM to regularize learning and prevent overfitting.

The output of the Bi-LSTM is fed into a sequence of dense (fully connected) layers. The first dense layer consists of 256 neurons with $ReLU$ activation, L2 regularization, batch normalization, and a dropout rate of 0.6. The second dense

layer consists of 128 neurons, also employing *ReLU* activation, L2 regularization, batch normalization, and a dropout rate of 0.6. These layers progressively map the encoded sequence features into a lower-dimensional space while enhancing the robustness and generalization of the model. Finally, a dense output layer with nine neurons and *softmax* activation is used for classification into nine distinct classes.

For training, a learning rate of *0.001*, a batch size of *32*, and a maximum of *100* epochs were set as hyperparameters. An early stopping criterion with a patience of *10* epochs and minimum improvement = *0* was applied. The SGD optimizer with a momentum of *0.9* and a weight decay of *1e-4* was selected to optimize the learning process. The model architecture is illustrated in Fig. 3.19.

Figure 3.19: Architecture of the ResNet-18 + LSTM model implemented for activity classification using the bedroom dataset.

CNN+LSTM The third model developed in this section is a hybrid DL architecture that integrates CNN for spatial feature extraction and Bi-LSTM networks for temporal sequence modeling. Additionally, an attention mechanism was incorporated to enhance sequence learning by focusing on relevant temporal features. The model was designed for multiclass classification with six target classes.

The CNN component processes individual data frames independently using a *TimeDistributed* wrapper, ensuring shared convolutional weights across all time steps. Each input sequence consists of 10 frames. To efficiently extract spatial features, similar to the second proposed model, the CNN backbone employs multiple stacked *SeparableConv2D* layers instead of standard convolutions. Each separable convolutional layer is followed by batch normalization [125], a *ReLU* activation function, and *MaxPooling* to progressively downsample feature maps.

Specifically, the network applies three blocks of separable convolutions with increasing filter sizes (32, 64, and 128 filters) and consistent kernel sizes of 5×5. After feature extraction, the output feature maps are flattened across spatial dimensions for each time step.

The extracted spatial features are then passed to a two-stage Bi-LSTM module. The first Bi-LSTM layer, configured with 128 hidden units, returns the full sequence of outputs, enabling subsequent temporal attention modeling. An *Attention* layer is then applied to dynamically weight the temporal features by learning the importance of different time steps. Following the attention mechanism, a

second Bi-LSTM layer with 128 hidden units aggregates the sequence into a fixed-size context vector that captures bidirectional temporal dependencies. Dropout regularization with a rate of 0.5 is applied to both BiLSTM layers to mitigate overfitting.

Finally, the output of the sequence modeling module is passed to a fully connected dense layer with six neurons and a *softmax* activation function, performing classification into the six target activity classes.

For training, a learning rate of *0.01*, a batch size of *32*, and a maximum of *100* epochs were used as hyperparameters. An early stopping criterion with a patience of *10* epochs and minimum improvement = *0* was applied to terminate training if no improvement was observed in validation loss. Optimization was performed using the SGD optimizer with a momentum of *0.9* and a weight decay of *1e-4* to promote efficient and stable convergence. The model architecture is illustrated in Fig. 3.20.

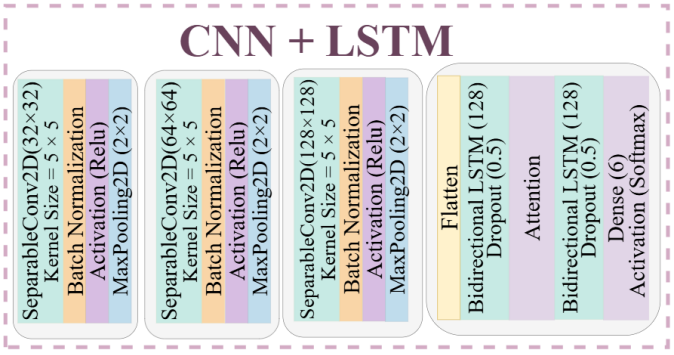


Figure 3.20: Architecture of the CNN + LSTM model implemented for activity classification using the bedroom dataset.

Experimental Setup

The proposed models were trained and validated on data collected from 10 subjects, with 80% of the data used for training and the remaining 20% for validation. The evaluation of the models was conducted using data from the remaining, unseen subject to assess and ensure the models' generalization capability. All procedures were executed on the same PC described in Section 3.3.3. The total number of samples for each activity is presented in Tables 3.6 and 3.7.

Table 3.6: Activity sample counts for 9-class classification.

Activity Label	**Training**	**Validation**	**Testing**	**Total**
Lying on the floor (B_1)	97,240	24,310	12,040	133,590
Lying on a sofa (B_2)	96,340	24,090	12,500	132,930
Sitting on/getting up from a sofa (B_3)	94,990	23,750	12,500	131,240
Picking an object from the floor (B_4)	54,920	13,730	6,500	75,150
Picking an object from a table (B_5)	55,280	13,830	6,500	75,610
Sitting on the floor (B_6)	96,010	24,010	12,500	132,520
Sitting on a sofa (B_7)	100,580	25,150	12,500	138,230
Standing (B_8)	99,370	24,850	12,500	136,720
Walking (B_9)	95,370	23,850	12,500	131,720
Total	790,100	197,570	100,040	1,087,710

Table 3.7: Activity sample counts for 6-class classification.

Activity Label	**Training**	**Validation**	**Testing**	**Total**
In-Place movement (C_1)	304,590	76,150	37,980	418,720
Lying on the floor (C_2)	97,240	24,310	12,040	133,590
Lying on a sofa (C_3)	96,340	24,090	12,500	132,930
Sitting on the floor (C_4)	96,010	24,010	12,500	132,520
Sitting on a sofa (C_5)	100,580	25,150	12,500	138,230
Walking (C_6)	95,370	23,850	12,500	131,720
Total	790,130	197,560	100,020	1,087,710

3.4 Performance Metrics

The performance of the proposed models are assessed using several key metrics: accuracy, precision, recall, F_1-score, and confusion matrix. These metrics provide a comprehensive evaluation of the models' classification performance and reveal patterns of classification errors.

Accuracy quantifies the overall correctness of the predictions by accounting for both True Positive (TP) and True Negative (TN) relative to the total number of predictions made. It is calculated as the ratio of the sum of TP and TN to the total number of cases, which also includes False Positive (FP) and False Negative (FN), as defined in Equation 3.26:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{3.26}$$

Precision quantifies the accuracy of positive predictions made by the model. It is calculated as the ratio of TP to the sum of TP and FP, emphasizing the quality of positive predictions. A high precision value indicates that the model produces few False Positive (FP), while a perfect precision of 100% is achieved when no FP are present. The formula for precision is defined in Equation 3.27:

$$Precision = \frac{TP}{TP + FP} \tag{3.27}$$

Recall represents the model's ability to correctly identify actual positive cases. It is calculated as the ratio of TP to the sum of TP and FN, emphasizing the model's sensitivity. A high recall value indicates that the model effectively minimizes FN, ensuring that most positive instances are detected. The formula for recall is defined in Equation 3.28:

$$Recall = \frac{TP}{TP + FN} \tag{3.28}$$

The F_1-score provides a balanced assessment of the model's performance by combining precision and recall into a single value. This metric is particularly useful when the class distribution is imbalanced, as it offers a more informative evaluation than accuracy alone. The F_1-score is defined as the harmonic mean of precision and recall, calculated using Equation 3.29:

$$F_1 - score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{3.29}$$

The weighted F_1-score accounts for the number of instances in each class, while the macro-averaged F_1-score treats all classes equally, making it particularly useful for assessing performance on imbalanced datasets.

The confusion matrix is a widely used tool in classification tasks, providing a detailed summary of model performance by displaying the counts of TP, FP, FN, and TN. It offers insights into how well each class is correctly predicted and highlights the types of misclassifications that occur, making it a valuable resource for performance analysis.

Chapter 4

Experimental Evaluation Result

This chapter presents the experimental evaluation of the proposed radar-based HAR system. The performance of various ML and DL models is assessed using the two collected datasets. The evaluation includes multi-class activity recognition and binary fall detection, explored under different feature representations and model architectures. Results are reported using standard performance metrics and analyzed to highlight system strengths, limitations, and generalization capabilities across diverse indoor scenarios.

4.1 Bedroom Dataset

This section presents the experimental results obtained using the Bedroom Dataset, which was collected in a realistic indoor environment and includes several daily activities performed by multiple subjects. A range of ML and DL models is evaluated for multi-class activity classification. The experiments aim to assess model performance, examine feature representation effectiveness, and investigate how reducing the number of target activities affects classification accuracy in this controlled setting.

4.1.1 Experimental Results

We evaluated the performance of several ML classifiers for recognizing different human activities using radar-derived features. The classifiers were trained and tested on reduced feature vectors extracted from RD, RA, and RE maps under the CSV protocol. To optimize dimensionality reduction, we tested multiple principal component settings in PCA and selected 100 components, which provided

the highest overall accuracy. Detailed performance metrics for each classifier are presented in Table 4.1.

Next, DL models including CNN, LSTM, and ConvLSTM were trained and evaluated using combined RD, RA, and RE (RD+RA+RE) feature maps across various activity sets under CSV. We further analyzed ConvLSTM performance for individual feature maps and pairwise combinations. To evaluate generalization to unseen subjects, we also tested ConvLSTM under the LOPO-CV protocol using the combined RD, RA, and RE features.

Table 4.1: Comparison of Accuracy and F_1-scores for different machine learning classifiers across varying activity sets. NB: Naive Bayes, DT: Decision Tree, RF: Random Forest.

#Activities	**Metrics (%)**	**ML-based Activity Classification**					
		SVM	**MLP**	**NB**	**DT**	**RF**	**KNN**
7	**Weighted F_1-score**	70.82	68.95	30.14	50.31	54.65	51.69
	Macro F_1-score	70.41	65.85	33.41	49.57	57.15	53.41
	Accuracy	**70.97**	68.71	31.89	51.31	56.34	54.02
6	**Weighted F_1-score**	76.28	75.40	32.13	54.41	61.16	57.85
	Macro F_1-score	77.75	75.58	36.03	55.51	63.89	61.35
	Accuracy	**76.34**	75.40	34.09	55.71	62.59	59.49
5	**Weighted F_1-score**	76.86	78.20	36.42	59.68	63.01	58.29
	Macro F_1-score	79.86	80.99	42.58	61.20	66.13	63.85
	Accuracy	76.63	**78.25**	42.89	60.21	63.72	59.34
4	**Weighted F_1-score**	88.39	89.99	68.29	76.15	74.27	72.63
	Macro F_1-score	89.08	90.43	64.78	75.73	75.63	74.16
	Accuracy	88.80	**90.28**	66.78	77.36	77.15	76.17

We evaluated system performance across different activity sets, initially classifying seven activities and subsequently removing less significant labels incrementally to focus on primary classes. The "Transition" class was removed first, followed by the "Empty Room" class, and finally the "Lay Down on the Floor" class. This step-by-step approach allowed us to refine the model's performance and better assess its effectiveness for core activities.

For seven activity classes, SVM achieved the highest accuracy (70.97%) and macro F_1-score (70.41%), followed by MLP with strong performance in macro F_1-score (65.85%) and accuracy (68.71%) still lagged behind SVM in accuracy.

Table 4.2: Confusion matrix for the Support Vector Machines classifier with seven activities, showing percentages.

Actual Label \ Predicted Label	Empty room	Enter the room and walk	Lay down on the bed	Lay down on the floor	Sit on the bed	Transitions	Sit on the chair
Empty room	91.74	1.25	5.45	1.56	0.0	0.0	0.0
Enter the room and walk	0.0	97.60	0.56	1.12	0.48	0.24	0.0
Lay down on the bed	0.45	0.26	89.75	1.75	0.19	7.59	0.00
Lay down on the floor	8.90	3.09	30.12	56.64	0.13	0.33	0.79
Sit on the bed	0.0	3.67	0.99	0.0	76.59	18.76	0.0
Transitions	0.0	9.65	4.51	2.38	18.3	55.39	9.77
Sit on the chair	27.05	0.25	11.54	9.38	0.08	0.17	51.54

Table 4.3: Confusion matrix for the Multi-Layer Perceptron classifier with seven activities, showing percentages.

Actual Label \ Predicted Label	Empty room	Enter the room and walk	Lay down on the bed	Lay down on the floor	Sit on the bed	Transitions	Sit on the chair
Empty room	73.83	1.25	13.24	10.44	0.31	0.16	0.78
Enter the room and walk	0.00	96.64	0.80	1.28	0.32	0.96	0.00
Lay down on the bed	1.23	0.39	84.82	2.47	2.14	8.89	0.06
Lay down on the floor	10.68	2.76	20.70	60.41	1.74	0.85	2.86
Sit on the bed	0.14	2.40	0.14	0.00	55.99	41.18	0.14
Transitions	0.00	6.77	4.64	2.63	19.67	57.02	9.27
Sit on the chair	24.15	0.83	8.63	11.54	1.16	1.08	52.61

Table 4.4: Confusion matrix for the Naive Bayes classifier with seven activities, showing percentages.

Actual Label \ Predicted Label	Empty room	Enter the room and walk	Lay down on the bed	Lay down on the floor	Sit on the bed	Transitions	Sit on the chair
Empty room	94.70	0.00	3.89	1.25	0.16	0.00	0.00
Enter the room and walk	0.00	30.14	1.12	11.19	37.01	20.30	0.24
Lay down on the bed	26.09	0.06	49.51	5.91	14.21	4.22	0.00
Lay down on the floor	53.84	0.33	28.02	7.62	6.90	1.54	1.74
Sit on the bed	0.00	9.45	1.27	2.82	54.44	32.02	0.00
Transitions	0.00	5.14	2.63	7.64	42.11	33.96	8.52
Sit on the chair	70.12	0.00	4.07	0.50	0.91	0.00	24.40

Table 4.5: Confusion matrix for the Decision Tree classifier with seven activities, showing percentages.

Actual Label \ Predicted Label	Empty room	Enter the room and walk	Lay down on the bed	Lay down on the floor	Sit on the bed	Transitions	Sit on the chair
Empty room	49.69	0.78	29.44	15.11	2.34	0.16	2.49
Enter the room and walk	0.08	83.77	1.84	6.08	3.92	3.28	1.04
Lay down on the bed	1.88	2.27	71.45	8.24	5.71	9.86	0.58
Lay down on the floor	10.55	3.68	34.26	39.09	4.93	2.53	4.96
Sit on the bed	0.00	16.93	3.39	2.12	61.07	16.36	0.14
Transitions	0.25	12.66	7.27	6.27	23.68	43.98	5.89
Sit on the chair	21.41	1.33	25.98	21.99	4.15	2.41	22.74

Table 4.6: Confusion matrix for the Random Forest classifier with seven activities, showing percentages.

Actual Label \ Predicted Label	Empty room	Enter the room and walk	Lay down on the bed	Lay down on the floor	Sit on the bed	Transitions	Sit on the chair
Empty room	62.62	0.47	33.80	3.12	0.00	0.00	0.00
Enter the room and walk	0.00	93.69	0.72	3.12	1.92	0.56	0.00
Lay down on the bed	0.78	0.71	85.27	2.21	1.36	9.67	0.00
Lay down on the floor	7.49	3.78	52.14	36.04	0.13	0.36	0.07
Sit on the bed	0.00	10.72	0.71	0.14	79.55	8.89	0.00
Transitions	0.00	18.92	3.01	7.39	15.91	49.75	5.01
Sit on the chair	22.66	0.75	39.83	17.51	0.00	0.00	19.25

Table 4.7: Confusion matrix for the K-Nearest Neighbor classifier with seven activities, showing percentages.

Actual Label \ Predicted Label	Empty room	Enter the room and walk	Lay down on the bed	Lay down on the floor	Sit on the bed	Transitions	Sit on the chair
Empty room	79.13	0.47	17.60	2.80	0.00	0.00	0.00
Enter the room and walk	0.08	89.29	1.28	6.71	1.84	0.64	0.16
Lay down on the bed	4.48	0.52	88.90	2.21	1.04	2.86	0.00
Lay down on the floor	22.04	1.31	45.83	30.19	0.10	0.20	0.33
Sit on the bed	0.00	3.53	4.23	0.71	85.61	5.64	0.28
Transitions	0.00	8.27	11.15	7.39	31.95	29.07	12.16
Sit on the chair	43.32	0.08	31.37	7.39	0.00	0.25	17.59

Reducing the number of classes to six improved all models' performance. This indicates that reducing the number of activities simplifies the classification task, benefiting most models. However, SVM still led in accuracy (76.34%), with MLP showing significant gains in both weighted (75.40%) and macro F_1-scores (75.58%). Additionally, based on the SVM confusion matrices in Table. 4.2 and Table 4.8, the average correct classification rate is over 74% while considering all seven activity classes and exceeds 82% on average for the six activity classes, respectively. This suggests that SVM is most effective for recognizing a diverse set of activities within this more complex scenario.

Table 4.8: Confusion matrix of the Support Vector Machines in classifying six activities, showing percentages.

Actual Label \ Predicted Label	Empty room	Enter the room and walk	Lay down on the bed	Lay down on the floor	Sit on the bed	Sit on the chair
Empty room	91.59	1.40	4.98	2.02	0.00	0.00
Enter the room and walk	0.08	98.40	0.32	0.88	0.32	0.00
Lay down on the bed	0.52	0.52	96.11	1.69	0.17	0.00
Lay down on the floor	8.54	2.83	28.42	58.90	0.13	1.18
Sit on the bed	0.00	6.35	0.99	0.00	92.52	0.14
Sit on the chair	27.55	0.17	8.55	9.13	0.00	54.61

Table 4.9: Confusion matrix of the Multi-Layer Perceptron classifier in classifying six activities, showing percentages.

Actual Label \ Predicted Label	Empty room	Enter the room and walk	Lay down on the bed	Lay down on the floor	Sit on the bed	Sit on the chair
Empty room	67.13	1.09	17.45	11.21	0.62	2.49
Enter the room and walk	0.08	97.60	0.32	1.44	0.40	0.16
Lay down on the bed	0.71	0.45	93.90	2.79	2.01	0.13
Lay down on the floor	8.48	3.15	23.55	61.07	1.25	2.50
Sit on the bed	0.00	3.10	1.27	0.00	95.35	0.28
Sit on the chair	22.32	0.58	7.63	10.21	1.66	57.59

Table 4.10: Confusion matrix of the Naive Bayes classifier in classifying six activities, showing percentages.

Actual Label \ Predicted Label	Empty room	Enter the room and walk	Lay down on the bed	Lay down on the floor	Sit on the bed	Sit on the chair
Empty room	97.51	0.00	1.25	1.25	0.00	0.00
Enter the room and walk	0.00	41.97	1.76	14.79	41.49	0.00
Lay down on the bed	25.89	3.83	50.29	3.57	16.35	0.06
Lay down on the floor	56.18	1.48	25.13	8.11	5.98	3.12
Sit on the bed	0.00	42.45	1.13	2.26	54.16	0.00
Sit on the chair	72.20	0.00	1.33	0.66	0.58	25.23

Table 4.11: Confusion matrix of the Decision Tree classifier in classifying six activities, showing percentages.

Actual Label \ Predicted Label	Empty room	Enter the room and walk	Lay down on the bed	Lay down on the floor	Sit on the bed	Sit on the chair
Empty room	54.05	0.78	21.65	18.69	0.93	3.89
Enter the room and walk	0.16	86.73	2.40	4.40	5.04	1.28
Lay down on the bed	1.49	3.11	77.16	9.80	7.85	0.58
Lay down on the floor	11.30	4.57	35.02	38.44	6.37	4.30
Sit on the bed	0.28	15.23	5.08	2.40	76.16	0.85
Sit on the chair	21.41	2.07	22.74	22.32	2.82	28.63

Table 4.12: Confusion matrix of the Random Forest classifier in classifying six activities, showing percentages.

Actual Label \ Predicted Label	Empty room	Enter the room and walk	Lay down on the bed	Lay down on the floor	Sit on the bed	Sit on the chair
Empty room	74.30	0.47	23.05	2.18	0.00	0.00
Enter the room and walk	0.00	94.56	0.72	3.04	1.68	0.00
Lay down on the bed	0.39	2.92	90.79	2.53	3.37	0.00
Lay down on the floor	10.22	4.07	43.86	41.43	0.10	0.33
Sit on the bed	0.00	9.45	0.14	0.14	90.27	0.00
Sit on the chair	30.54	0.50	32.70	11.95	0.00	24.32

Table 4.13: Confusion matrix of the K-Nearest Neighbor classifier in classifying six activities, showing percentages.

Actual Label \ Predicted Label	Empty room	Enter the room and walk	Lay down on the bed	Lay down on the floor	Sit on the bed	Sit on the chair
Empty room	85.20	1.09	10.90	2.80	0.00	0.00
Enter the room and walk	0.08	90.01	1.84	5.60	2.08	0.40
Lay down on the bed	3.70	0.65	92.41	2.08	1.17	0.00
Lay down on the floor	22.67	1.22	42.61	33.18	0.13	0.20
Sit on the bed	0.00	3.95	3.81	0.71	90.83	0.71
Sit on the chair	49.21	0.17	23.32	7.14	0.08	20.08

However, with seven activity classes, there is notable confusion between similar activities, such as "Lying on the bed" and "Lying on the floor". Additionally, natural movements during activities such as "Lying on the bed" or "Sitting on the bed" can lead to misclassification as the "Transition" class. Furthermore, brief periods without specific movements may cause confusion between the "Empty room" and "Lying on the floor" or "Sitting on the chair". When the "Transition" class is removed, reducing the classification to six activity classes, the classification performance becomes more balanced across the remaining activities. However, challenges persist in distinguishing between similar activities, such as "Lying on

the bed" and "Lying on the floor".

With five activity classes, MLP excelled, achieving the highest macro F_1-score (80.99%) and accuracy (78.25%), indicating its effectiveness in fewer activity scenarios. SVM also performed well, particularly in macro F_1-score (79.86%). Table 4.15 which illustrates MLP confusion matrix, highlights the classifier's robustness in distinguishing between the reduced set of activities with the average correct classification rate is over 82% for all five activity classes, particularly in recognizing "Entering the room and walking", "Lay down on the bed", and "Sit on the bed."

Table 4.14: Confusion matrix of the Support Vector Machines in classifying five activities, showing percentages.

		Predicted Label				
		Enter the room and walk	Lay down on the bed	Lay down on the floor	Sit on the bed	Sit on the chair
Actual Label	Enter the room and walk	98.24	0.56	0.80	0.40	0.00
	Lay down on the bed	0.91	95.85	1.95	1.30	0.00
	Lay down on the floor	3.02	33.48	61.73	0.13	1.64
	Sit on the bed	7.05	0.99	0.00	91.82	0.14
	Sit on the chair	0.25	25.06	16.35	0.00	58.34

Table 4.15: Confusion matrix of the Multi-Layer Perceptron in classifying five activities, showing percentages.

		Predicted Label				
		Enter the room and walk	Lay down on the bed	Lay down on the floor	Sit on the bed	Sit on the chair
Actual Label	Enter the room and walk	97.04	0.56	1.60	0.72	0.08
	Lay down on the bed	0.52	94.68	2.79	1.82	0.19
	Lay down on the floor	2.86	28.91	63.40	1.77	3.06
	Sit on the bed	3.10	0.42	0.00	96.33	0.14
	Sit on the chair	0.66	15.60	18.09	1.00	64.65

Table 4.16: Confusion matrix of the Naive Bayes in classifying five activities, showing percentages.

Actual Label \ Predicted Label	Enter the room and walk	Lay down on the bed	Lay down on the floor	Sit on the bed	Sit on the chair
Enter the room and walk	42.29	1.92	14.55	41.25	0.00
Lay down on the bed	4.28	69.11	4.09	15.83	6.68
Lay down on the floor	1.51	47.27	7.92	6.14	37.16
Sit on the bed	42.03	0.99	2.82	54.16	0.00
Sit on the chair	0.00	7.05	0.58	0.66	91.70

Table 4.17: Confusion matrix of the Decision Tree in classifying five activities, showing percentages.

Actual Label \ Predicted Label	Enter the room and walk	Lay down on the bed	Lay down on the floor	Sit on the bed	Sit on the chair
Enter the room and walk	85.93	2.48	5.76	4.32	1.52
Lay down on the bed	2.92	76.57	10.06	9.34	1.10
Lay down on the floor	4.17	37.88	46.78	6.04	5.12
Sit on the bed	13.82	4.51	2.68	78.42	0.56
Sit on the chair	1.99	32.37	25.89	3.90	35.85

Table 4.18: Confusion matrix of the Random Forest in classifying five activities, showing percentages.

Actual Label \ Predicted Label	Enter the room and walk	Lay down on the bed	Lay down on the floor	Sit on the bed	Sit on the chair
Enter the room and walk	94.24	0.88	3.12	1.76	0.00
Lay down on the bed	2.66	90.46	2.86	4.02	0.00
Lay down on the floor	3.88	48.62	47.31	0.13	0.07
Sit on the bed	10.72	0.28	0.14	88.86	0.00
Sit on the chair	0.75	59.25	15.44	0.00	24.56

Table 4.19: Confusion matrix of the K-Nearest Neighbor in classifying five activities, showing percentages.

Actual Label \ Predicted Label	Enter the room and walk	Lay down on the bed	Lay down on the floor	Sit on the bed	Sit on the chair
Enter the room and walk	90.33	1.60	5.76	2.08	0.24
Lay down on the bed	0.58	95.78	2.08	1.49	0.06
Lay down on the floor	1.28	62.45	35.74	0.16	0.36
Sit on the bed	3.95	4.23	0.56	90.97	0.28
Sit on the chair	0.25	64.81	13.20	0.17	21.58

For four classes, MLP continued to lead with the highest macro F_1-score (90.43%) and accuracy (90.28%). Its confusion matrix for four activities, depicted in Table 4.21, exceeds an 89% average correct classification rate, which highlights the strong performance of the MLP in simplified classification tasks.

Table 4.20: Confusion matrix of the Support Vector Machines classifier in classifying four activities, showing percentages.

Actual Label \ Predicted Label	Enter the room and walk	Lay down on the bed	Sit on the bed	Sit on the chair
Enter the room and walk	98.88	0.64	0.48	0.00
Lay down on the bed	0.84	97.73	1.43	0.00
Sit on the bed	5.64	0.99	93.23	0.14
Sit on the chair	0.58	35.10	0.00	64.32

Table 4.21: Confusion matrix of the Multi-Layer Perceptron in classifying four activities, showing percentages.

Actual Label \ Predicted Label	Enter the room and walk	Lay down on the bed	Sit on the bed	Sit on the chair
Enter the room and walk	98.64	0.64	0.56	0.16
Lay down on the bed	0.32	97.21	2.27	0.19
Sit on the bed	2.54	1.13	96.19	0.14
Sit on the chair	0.83	28.13	1.74	69.29

Table 4.22: Confusion matrix of the Naive Bayes classifier in classifying four activities, showing percentages.

		Predicted Label			
		Enter the room and walk	Lay down on the bed	Sit on the bed	Sit on the chair
Actual Label	Enter the room and walk	41.57	4.72	53.64	0.08
	Lay down on the bed	4.61	70.67	17.52	7.20
	Sit on the bed	40.34	1.97	57.69	0.00
	Sit on the chair	0.00	5.39	1.24	93.36

Table 4.23: Confusion matrix of the Decision Tree in classifying four activities, showing percentages.

		Predicted Label			
		Enter the room and walk	Lay down on the bed	Sit on the bed	Sit on the chair
Actual Label	Enter the room and walk	92.73	2.48	3.84	0.96
	Lay down on the bed	2.27	88.84	8.18	0.71
	Sit on the bed	12.13	4.94	81.95	0.99
	Sit on the chair	1.58	46.14	8.22	44.07

Table 4.24: Confusion matrix of the Random Forest classifier in classifying four activities, showing percentages.

		Predicted Label			
		Enter the room and walk	Lay down on the bed	Sit on the bed	Sit on the chair
Actual Label	Enter the room and walk	97.12	0.88	2.00	0.00
	Lay down on the bed	4.02	92.41	3.57	0.00
	Sit on the bed	9.59	0.28	90.13	0.00
	Sit on the chair	1.16	69.54	0.00	29.29

Table 4.25: Confusion matrix of the K-Nearest Neighbor in classifying four activities, showing percentages.

		Predicted Label			
		Enter the room and walk	Lay down on the bed	Sit on the bed	Sit on the chair
Actual Label	Enter the room and walk	93.69	2.96	2.64	0.72
	Lay down on the bed	0.91	97.21	1.88	0.00
	Sit on the bed	3.67	5.50	90.13	0.71
	Sit on the chair	0.50	76.51	0.08	22.90

CSV The performance comparison of different models evaluated over multiple activity categories with combination of RD, RA, and RE (RD+RA+RE) feature maps as models' input is presented in Table 4.26. This table illustrates a consistent performance pattern across all activity sets: DL models (CNN, LSTM, and ConvLSTM) significantly outperform traditional ML approaches (SVM and MLP in Table 4.1). In particular, ConvLSTM consistently achieves the highest performance across all activity sets, with peak accuracies of 90.51% for 7-activity

classification and 97.87% for 4-activity classification. This trend indicates that ConvLSTM's ability to capture both spatial and temporal features provides a substantial advantage for HAR tasks using radar data.

Table 4.26: Performance metrics for various deep learning models under Cross-Scene-Validation with Range-Doppler+Range-Azimuth+Range-Elevation inputs.

#Activities	**Model**	**Metrics (%)**			
		Accuracy	**Precision**	**Recall**	**F_1-score**
7	CNN	89.48	86.66	**88.81**	87.17
	LSTM	88.50	85.09	87.35	85.64
	ConvLSTM	**90.51**	**86.75**	88.51	**87.31**
6	CNN	92.60	91.59	93.67	91.83
	LSTM	94.03	92.87	95.19	93.55
	ConvLSTM	**95.29**	**94.08**	**96.02**	**94.73**
5	CNN	92.57	94.09	94.40	93.79
	LSTM	93.99	95.08	95.08	94.79
	ConvLSTM	**96.06**	**96.86**	**96.15**	**96.39**
4	CNN	96.28	97.12	96.17	96.49
	LSTM	96.91	97.61	96.79	97.11
	ConvLSTM	**97.87**	**98.39**	**97.80**	**98.05**

The classification results of the ConvLSTM model with different feature inputs over various numbers of activities are summarized in Tables 4.27–4.30. Additionally, Fig. 4.1 shows the confusion matrices for different tasks, while Fig. 4.2 illustrates the training and validation loss curves of the ConvLSTM model across different activity sets using the RD+RA+RE feature maps.

As shown in Tables 4.27–4.30, classification performance improves as the number of classes decreases. Specifically, the F_1-score increases from 87.31% for seven classes to 98.05% for four classes. The RE map consistently achieves the best results across all tasks among single-feature inputs. For pairwise combinations, except for four classes, RD+RE maps outperform other combinations. Integrating all three features yields the highest performance, slightly surpassing RD+RE maps. This highlights the significance of combining motion (Doppler) and spatial dimensions (azimuth and elevation) to comprehensively represent activities, thereby enhancing classification accuracy.

The training process, shown in Fig. 4.2, demonstrates a steady decrease in both training and validation loss, reaching suitably low levels. This reflects effective

model learning, the absence of overfitting, and strong generalization capability.

Table 4.27: Comparison of ConvLSTM model performance for 7 activity classification under the Cross-Scene-Validation approach.

Metrics (%)	Model Input						
	RD	**RA**	**RE**	**RD+RA**	**RD+RE**	**RA+RE**	**RD+RA+RE**
Accuracy	76.94	71.76	89.53	86.26	90.40	87.51	90.51
Precision	80.12	74.32	86.02	84.96	87.05	84.47	86.75
Recall	79.98	77.18	87.33	87.16	88.95	86.42	88.51
F_1**-score**	77.66	72.95	86.12	85.51	87.70	84.59	87.31

Table 4.28: Comparison of ConvLSTM model performance for 6 activity classification under the Cross-Scene-Validation approach.

Metrics (%)	Model Input						
	RD	**RA**	**RE**	**RD+RA**	**RD+RE**	**RA+RE**	**RD+RA+RE**
Accuracy	82.70	78.22	94.09	92.60	95.58	91.23	95.29
Precision	85.53	83.60	92.79	92.69	94.43	90.66	94.08
Recall	87.80	86.96	95.14	94.74	96.43	93.67	96.02
F_1**-score**	85.00	82.41	93.50	93.28	95.20	91.35	94.73

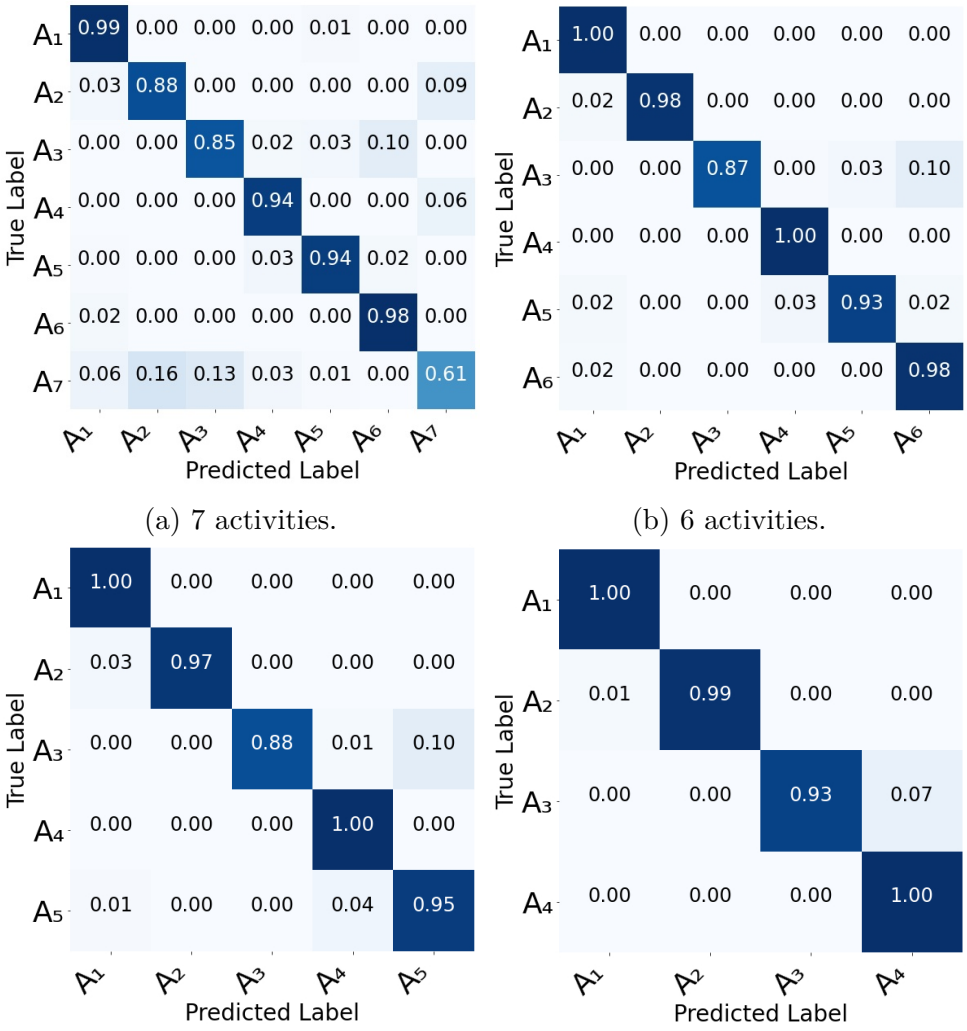
Table 4.29: Comparison of ConvLSTM model performance for 5 activity classification under the Cross-Scene-Validation approach.

Metrics (%)	Model Input						
	RD	**RA**	**RE**	**RD+RA**	**RD+RE**	**RA+RE**	**RD+RA+RE**
Accuracy	78.42	79.59	95.93	88.37	95.48	93.28	96.06
Precision	83.79	86.31	96.53	91.26	95.98	94.30	96.86
Recall	82.43	87.10	96.11	92.28	95.73	95.11	96.15
F_1**-score**	80.79	84.26	96.24	90.71	95.67	94.43	96.39

Table 4.30: Comparison of ConvLSTM model performance for 4 activity classification under the Cross-Scene-Validation approach.

Metrics (%)	Model Input						
	RD	**RA**	**RE**	**RD+RA**	**RD+RE**	**RA+RE**	**RD+RA+RE**
Accuracy	86.49	95.43	96.81	96.49	97.45	97.66	97.87
Precision	87.28	95.94	97.43	97.33	97.92	98.22	98.39
Recall	86.89	95.53	96.67	96.30	97.37	97.49	97.80
F_1**-score**	85.76	95.56	96.96	96.68	97.58	97.80	98.05

LOPO-CV We report the LOPO-CV results for the ConvLSTM model, which demonstrated the strongest performance under the CSV approach. The combined



(a) 7 activities.

(b) 6 activities.

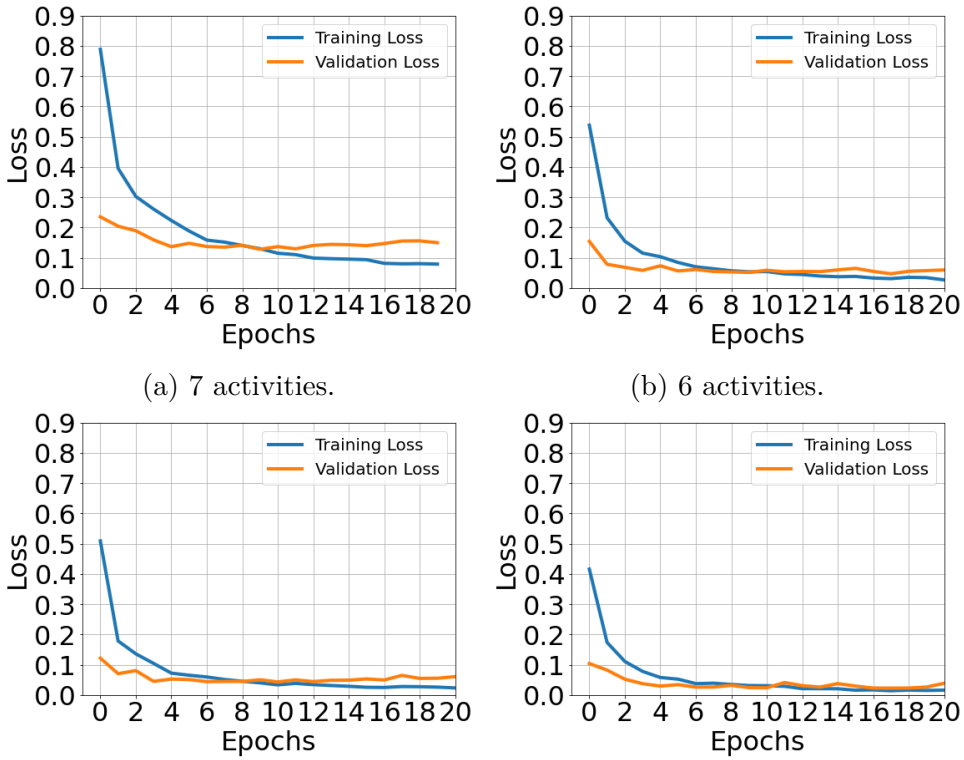
(c) 5 activities.

(d) 4 activities.

Figure 4.1: Confusion matrices for the ConvLSTM model on various activity sets under Cross-Scene-Validation with Range-Doppler+Range-Azimuth+Range-Elevation inputs.

RD+RA+RE feature maps were used, as they consistently provided the most reliable results. Table 4.31 summarizes the ConvLSTM performance for different activity sets. Both accuracy and F_1-score improve as the number of activities decreases, rising from 89.56% (87.15%) for seven activities to 93.00% (91.12%) for four activities. These findings demonstrate the model's robustness and generalization capability in subject-independent evaluation, as well as the effectiveness of integrating RD, RA, and RE feature maps.

Compared with the CSV approach, the LOPO-CV strategy evaluates performance using a larger number of test samples, providing a more realistic measure



(a) 7 activities.

(b) 6 activities.

(c) 5 activities.

(d) 4 activities.

Figure 4.2: Training and validation loss curves for the ConvLSTM model under Cross-Scene-Validation with Range-Doppler+Range-Azimuth+Range-Elevation inputs.

of model generalizability across individuals. The performance of the ConvLSTM using single or pairwise feature-map inputs was not analyzed under this validation method.

Table 4.31: Comparison of ConvLSTM model performance presented as percentages using the Leave-One-Person-Out Cross-Validation approach.

Test Subject	**7 Activity**		**6 Activity**		**5 Activity**		**4 Activity**	
	Accuracy	**F_1-score**	**Accuracy**	**F_1-score**	**Accuracy**	**F_1-score**	**Accuracy**	**F_1-score**
Subject 1	89.08	85.49	92.19	90.79	92.35	91.15	92.55	91.03
Subject 2	93.14	91.23	96.90	96.25	97.96	97.89	97.95	97.49
Subject 3	86.45	84.72	94.35	93.88	94.51	93.41	88.52	84.84
Average	89.56	87.15	94.48	93.64	94.94	94.15	93.00	91.12

Conventional ML models struggle to distinguish patterns in RD, RA, and RE maps, making subtle feature extraction challenging. In contrast, DL models such

as CNN effectively capture discriminative features, while ConvLSTM further enhances radar data processing by modeling both spatial and temporal dependencies [108, 109].

4.1.2 Discussion

This study validates the effectiveness of FMCW radar for HAR through a unified framework that integrates RD, RA, and RE feature maps across multiple ML and DL models. Performance was evaluated using both CSV and LOPO-CV validation strategies to assess generalization under varying conditions.

Among the ML models, SVM achieved the best performance for 6- and 7-activity classification, while MLP performed better for 4- and 5-activity tasks. This suggests that SVM is more robust in complex feature spaces with overlapping classes, whereas MLP benefits from simplified classification scenarios. However, the MLP requires careful regularization and sufficient computational resources to avoid overfitting. DT and RF provided consistent results but did not surpass SVM or MLP.

Analysis of confusion matrices reveals recurring misclassifications between "sitting on a chair" and "lying down," likely due to similarities in static radar signatures and minimal movement. The limited representation of certain classes—particularly sitting—may also have hindered the model's ability to learn distinctive features. These findings highlight the importance of dataset balancing and broader sample diversity for improving classifier performance.

The use of DL models further enhances classification accuracy. ConvLSTM delivered the strongest results, achieving accuracies of 90.51% under CSV and 89.56% under LOPO-CV, with corresponding F_1-scores above 87%. By integrating spatial and temporal modeling, ConvLSTM excels in distinguishing activities with subtle motion differences, such as "sitting on a bed" versus "sitting on a chair." These observations align with prior studies demonstrating the benefits of spatiotemporal learning for HAR [108, 109, 111, 112].

Among the radar representations, RE consistently outperformed RD and RA, underscoring the importance of vertical spatial information. Combining RD with RE typically improved accuracy, highlighting their complementary role. The inclusion of RA provided smaller gains, suggesting potential redundancy in some scenarios.

We demonstrated the advantages of using combined RD, RA, and RE maps as 3D data structures that effectively capture both motion and spatial patterns. Experimental results showed that SVM performs best on complex classifications involving 7 and 6 activities, while MLP excels in simpler 4- and 5-activity tasks, highlighting the need to match classifiers with task complexity.

4.1.3 Conclusion

In the first dataset, we installed one FMCW mmWave in the bedroom environment, and extracted 3 RD, RA, and RE feature maps as data vector feed them to various ML and DL models. Our approach focuses on system architecture optimization, including radar setup configuration, preprocessing pipeline development, and the generation of multi-dimensional feature maps for classifying intricate human activities. Unlike conventional image-based radar data processing, our framework processes raw radar data into structured feature maps that capture both temporal and spatial characteristics of human activities, balancing computational efficiency with recognition accuracy.

We demonstrated the feasibility and advantages of integrating RD, RA, and RE feature maps as 3D data vectors to effectively capture motion and spatial details. The experimental validation in realistic environments showed that among conventional ML models, SVM effectively handled complex, overlapping features in 7- and 6-activity classifications, while MLP excelled with less ambiguous activities in 4- and 5-activity scenarios. This highlights the importance of classifier selection based on data characteristics.

Furthermore, by leveraging DL models, particularly the ConvLSTM architecture, we successfully extracted spatiotemporal patterns from radar feature maps, thereby surpassing traditional approaches in performance. The ConvLSTM model effectively captured both motion and spatial dependencies, demonstrating robust performance across diverse scenarios and when tested on unseen data through CSV and LOPO-CV. Our results emphasize the benefits of using multiple feature maps for improved recognition accuracy, with RE proving the most effective, highlighting the crucial role of vertical spatial information in precise activity recognition.

The proposed solution demonstrated strong performance even when trained on a limited dataset, making it suitable for real-world applications in healthcare monitoring and smart environments that require non-intrusive activity monitoring. This advantage is particularly valuable when comprehensive data collection may be impractical due to domain-specific constraints.

Despite promising results, several limitations remain, including challenges in distinguishing highly similar activities and managing periods of inactivity. Future research should address these limitations by expanding dataset diversity, incorporating a wider range of activities, and evaluating the computational cost of models for large-scale implementation, real-time applications, and deployment on edge devices. Additionally, further exploration of lightweight DL architectures could improve accuracy and adaptability, enhancing the framework's suitability for long-term care and AAL environments.

4.2 Living Room Dataset

This section presents the experimental results obtained using the Living Room Dataset. This dataset, collected in a more spacious and complex environment, involves a larger group of subjects performing a diverse set of activities. Data were captured using two radars installed at different positions to provide multi-perspective coverage. The evaluation includes both binary fall detection and multi-class activity classification tasks. Several DL models are tested to assess their performance, computational complexity, robustness to spatial variations, and scalability in more realistic, less constrained settings.

4.2.1 Fall detection

This subsection presents the experimental evaluation of fall detection using the Living Room Dataset. Falls are critical events in healthcare monitoring. The dataset includes multiple instances of fall and non-fall activities recorded from two radar perspectives. One DL model is evaluated to assess its ability to accurately distinguish falls from similar daily movements using individual feature maps (RE, RA) as well as their combination (RE+RA). The results provide insight into model effectiveness, generalization across viewpoints, and the applicability of these feature maps for the fall detection task.

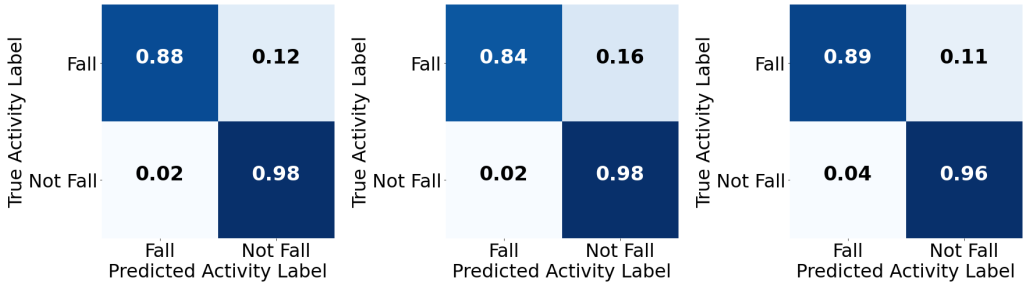
Experimental Results

The experimental methodology was designed to comprehensively evaluate both the performance and computational efficiency of the proposed fall detection system across multiple configurations and input modalities. The investigation focused on two key aspects: (1) the effect of varying the number of 3D CNN blocks and (2) a comparative analysis of three distinct input configurations—RE, RA, and the combined RE+RA feature maps. This approach enables a systematic assessment of how architectural complexity and feature representation influence detection accuracy and model robustness.

Table 4.32 provides a comprehensive comparison of the model's performance across different configurations. The RE+RA input consistently exhibited superior performance, achieving the highest accuracy across various 3D CNN block arrangements. The model reached a peak accuracy of 94.33% with one 3D CNN block. Although adding additional 3D CNN blocks resulted in slight variations in performance, the RE+RA input maintained the highest accuracy levels, significantly outperforming models with either RE or RA inputs alone. Fig. 4.3a, 4.3b, and 4.3c illustrate the normalized confusion matrices for the RE+RA input configuration with one, two, and three 3D CNN blocks, respectively.

Table 4.32: Comparison of proposed 3D CNN model performance for various configurations. RE: Range-Elevation, RA: Range-Azimuth.

# of Proposed 3D CNN Blocks	Metrics (%)	Model Input		
		RE	RA	RE+RA
1	**Accuracy**	89.04	81.38	94.33
	Precision	88.98	81.03	94.36
	Recall	89.04	81.38	94.33
	F_1**-score**	87.59	78.42	93.5
2	**Accuracy**	87.9	82.74	93.45
	Precision	87.78	82.45	93.61
	Recall	87.9	82.74	93.45
	F_1**-score**	86.19	79.69	92.38
3	**Accuracy**	87.39	83.01	93.61
	Precision	87.26	82.89	93.58
	Recall	87.39	83.01	93.61
	F_1**-score**	85.43	79.59	92.75

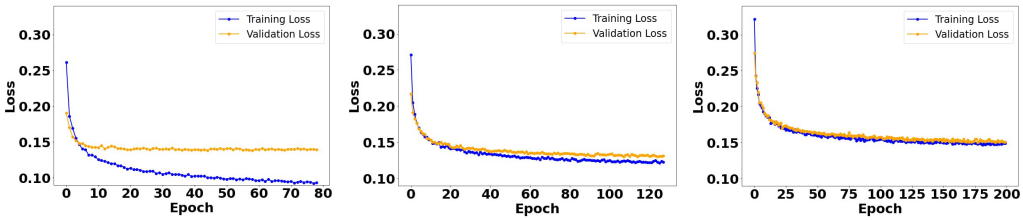


(a) One 3D CNN block. (b) Two 3D CNN blocks. (c) Three 3D CNN blocks.

Figure 4.3: Normalized confusion matrices for different proposed model configurations and the Range-Elevation+Range-Azimuth feature input.

The training progression, depicted by the loss curves for different 3D CNN configurations in Fig. 4.4, indicates that increasing the number of blocks improves convergence behavior. In particular, models with multiple 3D CNN blocks exhibited greater training stability and more consistent reduction in loss across epochs.

Computational Efficiency A comprehensive analysis of computational requirements is presented in Table 4.33. The relationship between the number of 3D CNN blocks and overall computational demands reveals two notable trends: (1) increasing the number of 3D CNN blocks results in higher floating-point operations per



(a) One 3D CNN block. (b) Two 3D CNN blocks. (c) Three 3D CNN blocks.

Figure 4.4: Training loss for different proposed model configurations and Range-Elevation+Range-Azimuth feature input.

second (FLOPs) and greater memory consumption, and (2) the total parameter count decreases with additional blocks due to architectural modifications, such as the incorporation of pooling layers. Across all configurations, the RE+RA input required the largest memory allocation and the longest processing time. In this study, FLOPs are reported in GigaFLOPS (GFLOPS) and MegaFLOPS (MFLOPS), model size is expressed in megabytes (MB), the number of parameters is reported in millions (M), and inference time is measured in milliseconds (ms).

Table 4.33: Comparison of proposed 3D CNN model complexity and computational resource for various configurations. RE: Range-Elevation, RA: Range-Azimuth.

# of Proposed 3D CNN Blocks	**Metrics**	**Features**		
		RE	**RA**	**RE+RA**
1	**#Parameter (M)**	83.8	83.8	83.8
	Inference Time (ms)	7.82	7.78	10.36
	#Flops (MFLOPS)	312	311	453.5
	Model Size (MB)	640	640	640
	Memory Usage (MB)	424.11	425.92	561.78
2	**#Parameter (M)**	16.8	16.8	16.8
	Inference Time (ms)	9.84	8.04	9.50
	#Flops (GFLOPS)	1.3	1.3	1.4
	Model Size (MB)	128	128	128
	Memory Usage (MB)	432.77	436.41	573.14
3	**#Parameter (M)**	4.4	4.4	4.4
	Inference Time (ms)	10.09	9.17	9.28
	#Flops (GFLOPS)	1.7	1.7	1.8
	Model Size (MB)	34.2	34.2	34.2
	Memory Usage (MB)	433.25	433.98	570.39

Our experimental results lead to several key findings:

- The integration of elevation and azimuth data is crucial for achieving robust fall detection performance. While increasing the number of 3D CNN blocks does not substantially enhance accuracy, it improves training convergence and reduces the total parameter count.
- System designers must carefully balance the trade-offs between model complexity and computational efficiency, considering specific deployment constraints and accuracy requirements.

These insights provide valuable guidance for optimizing fall detection systems across various operational scenarios and resource constraints.

Discussion

In this step, we evaluated the effectiveness of FMCW radar for fall detection using a framework that incorporates RA and RE feature maps individually and in combination (RA+RE) within a 3D CNN model. System performance was assessed using the LOPO-CV validation strategy.

The experimental results demonstrate that combining RA and RE feature maps substantially improves fall detection accuracy. The RE+RA configuration achieved a peak accuracy of 94.33% using a single 3D CNN block, indicating that the integration of complementary spatial information enables more effective differentiation between fall and non-fall scenarios.

Our analysis further highlights a trade-off between model complexity and computational efficiency. Although increasing the number of 3D CNN blocks enhances training stability and reduces the parameter count, it does not consistently improve classification accuracy. This suggests that beyond a certain depth, additional blocks contribute more to architectural optimization than to meaningful gains in detection performance.

Despite these encouraging results, several limitations remain. Data collected under controlled conditions may not capture the full variability of real-world environments, and using "lying on the floor" as a proxy for falls may not fully reflect the dynamics of actual fall events. Future work should include evaluation in more diverse and realistic scenarios, as well as exploration of alternative architectures and augmentation techniques to further improve robustness and efficiency.

Conclusion

This step introduces a system optimized for AAL environments, combining FMCW radar technology with DL to achieve reliable and cost-effective fall detection.

Utilizing two wall-mounted mmWave radars, the system captures human motion across multiple positions in the room and transforms the data into RA and RE feature maps for input to a 3D CNN model. The proposed model achieved high performance, with an accuracy of 94.33% and a macro F_1-score of 93.5%, demonstrating a strong balance between reliability and generalizability. These findings highlight the advantages of integrating range, azimuth, and elevation information and confirm the effectiveness of 3D CNN architectures in capturing both spatial and temporal characteristics for enhanced fall detection.

Future research may extend this work by examining additional environmental configurations and optimizing real-time processing for deployment across various AAL settings. With robust detection capability under diverse lighting conditions and a strong emphasis on user privacy, the proposed system offers a scalable solution for elderly care, with significant potential to enhance safety and overall quality of life.

4.2.2 Activity Recognition

This subsection presents the experimental evaluation of multi-class activity recognition using the Living Room Dataset. The dataset includes various daily activities performed by 11 subjects in a realistic environment, with radar data collected from two different viewpoints. Multiple DL models were trained and tested using combinations of RD, RA, and RE feature maps. The experiments were conducted under two classification scenarios—6-class and 9-class—to investigate the trade-off between task complexity and model performance.

Experimental Results

In this study, we evaluated the performance of several DL models in recognizing human activities using RD, RA, and RE feature representations. Detailed evaluation metrics for each model are provided in their respective sections. Both classification performance and computational cost were thoroughly assessed.

First, a 3D CNN was trained and tested for nine-class activity recognition. Next, a ResNet combined with LSTM (ResNet+LSTM) was evaluated for both nine-class and six-class tasks. Finally, we designed and implemented a hybrid CNN+LSTM model and assessed its performance for the six-class activity recognition task.

To enhance model robustness and performance, we implemented a Majority Voting approach for the last DL model. In this method, instead of immediately displaying the model's classification output, the predicted activity labels are stored in an empty vector while the model continues processing subsequent samples. This process is repeated for t samples. At the end of this sequence, the final classified

activity is determined by identifying the majority label among the t predictions. Although this approach may introduce a slight delay in displaying the output, it effectively reduces false activity detections by considering a broader temporal context, ultimately improving overall system performance.

3D CNN Table 4.34 presents the overall performance of the 3D CNN model trained on the combined RD, RA, and RE features for nine-class activity recognition. The model achieved an accuracy of 60.98%, with a precision of 57.14%, recall of 56.40%, and an F_1-score of 55.53%. These results indicate moderate performance, suggesting that although the model learns meaningful spatiotemporal features, its generalization capability remains limited for this more complex classification scenario.

Table 4.34: Performance metrics for 3D CNN models with Range-Doppler+Range-Azimuth+Range-Elevation inputs for 9-class activity recognition.

Model	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	F_1**-score (%)**
3D CNN	60.98	57.14	56.40	55.53

Fig. 4.5 presents the confusion matrix, highlighting class-wise performance. The model performs well on activities such as lying on the floor and walking. However, it struggles with actions like picking up an object from the floor or table, and sitting on or getting up from a sofa, which are often misclassified as standing. This confusion is likely due to overlapping spatiotemporal radar signatures.

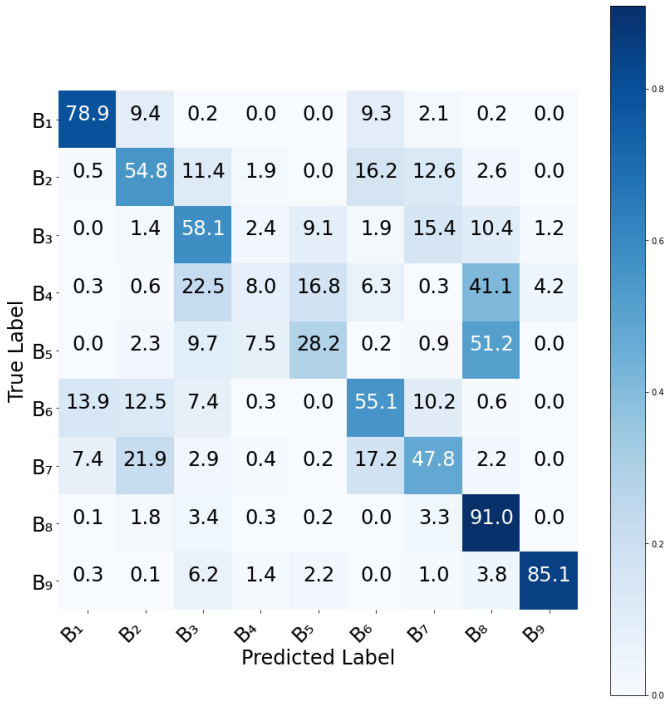


Figure 4.5: Confusion matrix for the 3D CNN model for 9-class activity recognition. (B_1) lying on the floor, (B_2) lying on a sofa, (B_3) sitting on/getting up from a sofa, (B_4) picking an object from the floor, (B_5) picking an object from a table, (B_6) sitting on the floor, (B_7) sitting on a sofa, (B_8) standing, (B_9) walking.

The training and validation loss curves in Fig. 4.6 show steady convergence over 100 epochs, indicating stable training behavior.

In addition to classification performance, we evaluated the model's computational complexity. As shown in Table 4.35, the 3D CNN requires approximately 64.70 GFLOPs and consists of 4.5 M parameters. These values reflect a high computational cost, which may limit the feasibility of deploying this model on resource-constrained edge devices.

Table 4.35: Computational cost for 3D CNN model with Range-Doppler+Range-Azimuth+Range-Elevation inputs for 9-class activity recognition.

Model	**#Flops (GFLOPS)**	**#Parameter (M)**
3D CNN	64.70	4.5

Overall, the 3D CNN demonstrates limited effectiveness in recognizing all nine activities, particularly those involving similar or overlapping motion patterns.

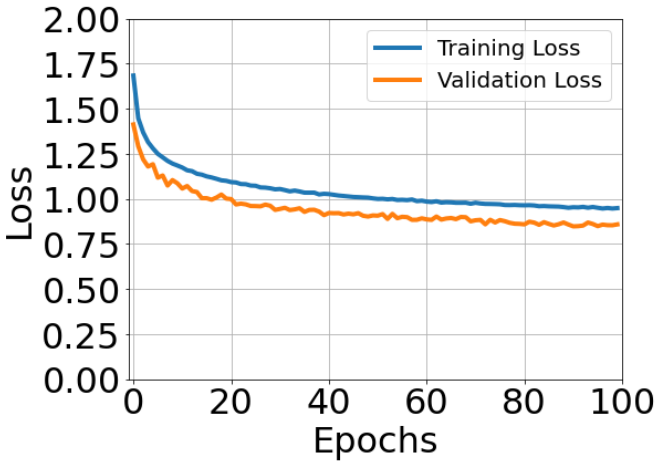


Figure 4.6: Training and validation loss curves for the 3D CNN model with Range-Doppler+Range-Azimuth+Range-Elevation inputs for 9-class activity recognition. These findings underscore the need to explore alternative DL architectures to improve classification accuracy and robustness.

ResNet+LSTM Table 4.36 presents the classification performance and computational cost of the ResNet+LSTM model for nine-class activity recognition using RD, RA, and RE features. The model achieved an overall accuracy of 63.54%, with a precision of 61.56%, recall of 59.28%, and F_1-score of 58.89%. These results indicate a modest improvement over the 3D CNN model.

Table 4.36: Classification performance of ResNet+LSTM model with Range-Doppler+Range-Azimuth+Range-Elevation inputs for 9-class activity recognition.

Model	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F_1-score (%)**
ResNet+LSTM	63.54	61.56	59.28	58.89

Fig. 4.7 presents the confusion matrix for the ResNet+LSTM model in the nine-class activity recognition task. Similar to the 3D CNN model (Fig. 4.5), it performs well on distinguishable activities such as walking and lying on the floor. However, it shows difficulty in recognizing overlapping or similar activities, including picking up an object from the table or floor, sitting or getting up from a sofa, and standing still.

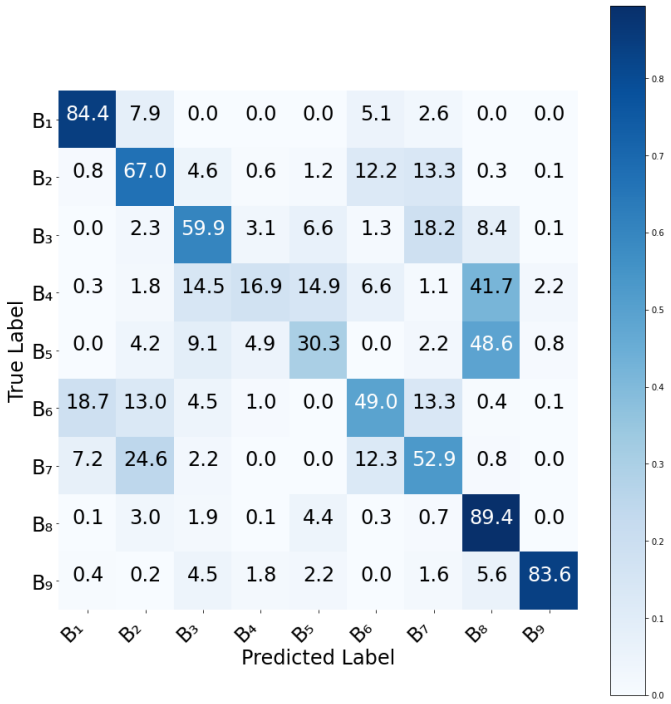


Figure 4.7: Confusion matrix for the ResNet+LSTM model for 9-class activity recognition. (B_1) lying on the floor, (B_2) lying on a sofa, (B_3) sitting on/getting up from a sofa, (B_4) picking an object from the floor, (B_5) picking an object from a table, (B_6) sitting on the floor, (B_7) sitting on a sofa, (B_8) standing, (B_9) walking.

The training and validation loss curves for the ResNet+LSTM model are shown in Fig. 4.8. The validation loss closely follows the training loss throughout the training process, with no significant divergence between the two. This behavior indicates that the model generalizes well to unseen data and does not exhibit signs of overfitting. The convergence of both curves further suggests stable training and effective learning of relevant spatiotemporal features.

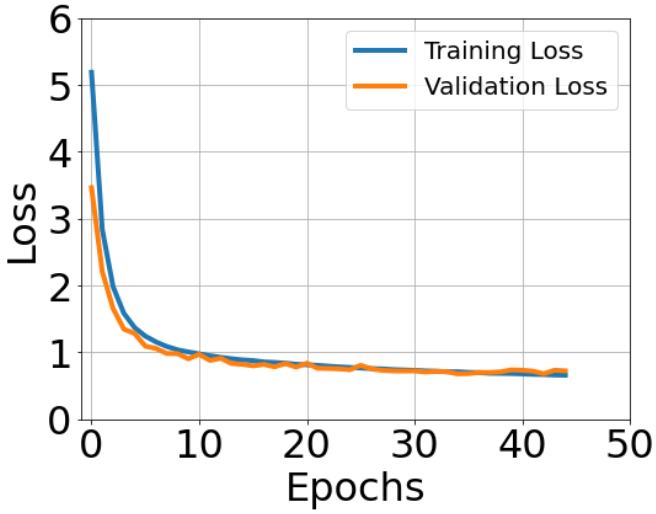


Figure 4.8: Training and validation loss curves for the ResNet+LSTM model with Range-Doppler+Range-Azimuth+Range-Elevation inputs for 9-class activity recognition.

As mentioned earlier, to reduce training time and enhance classification performance, similar activity classes were merged into a broader category, *In-place movement*. This consolidation led to a significant increase in the accuracy of the ResNet+LSTM model. Table 4.37 presents the model's performance on this six-class problem using RD, RA, and RE features. Compared to the nine-class setting, the model shows consistent improvements across all metrics. It achieved high accuracy with a strong balance between precision and recall, indicating that class combination improved generalization and reduced inter-class confusion.

Table 4.37: Classification performance of ResNet+LSTM model with Range-Doppler+Range-Azimuth+Range-Elevation inputs for 6-class activity recognition.

Model	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F_1-score (%)**
ResNet+LSTM	75.28	72.10	70.17	70.82

Fig. 4.9 presents the confusion matrix for the ResNet+LSTM model on the six-class task. The model shows strong performance in identifying activities such as *In-place movement* (90.3%), *lying on the floor* (73.8%), and *walking* (82.2%). Although some misclassification remains, the overall distribution indicates enhanced class separability compared to the more granular nine-class setting.

Figure 4.9: Confusion matrix for the ResNet+LSTM model for 6-class activity recognition. (C_1) In-place movement, (C_2) lying on the floor, (C_3) lying on a sofa, (C_4) sitting on the floor, (C_5) sitting on a sofa, (C_6) walking.

The training and validation loss curves shown in Fig. 4.10 indicate rapid convergence and stable learning behavior, confirming that the model avoids overfitting and maintains strong generalization performance throughout training.

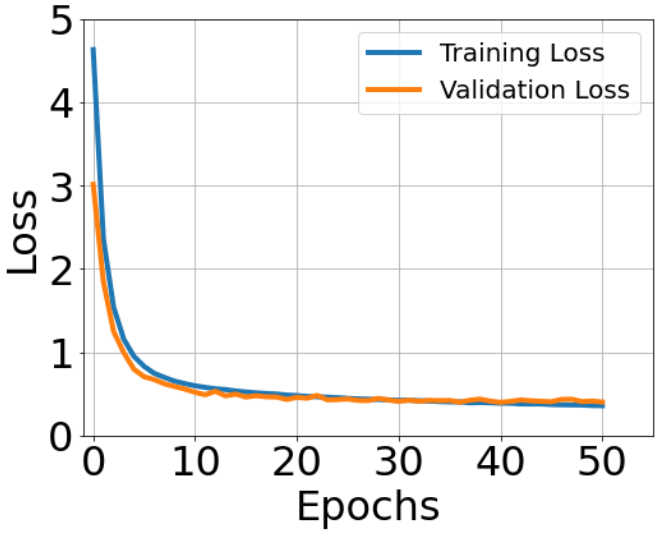


Figure 4.10: Training and validation loss curves for the ResNet+LSTM model with Range-Doppler+Range-Azimuth+Range-Elevation inputs for 6-class activity recognition.

In terms of computational efficiency, the ResNet+LSTM model is considerably lighter than the 3D CNN. As shown in Table 4.38, it requires only 23.45 GFLOPs and approximately 3.19 M parameters. This reduced complexity makes it a promising candidate for real-time or edge deployment, where computational constraints are significant.

Table 4.38: Computational cost of ResNet+LSTM model.

Model	**FLOPs (GFLOPs)**	**Parameters (M)**
ResNet+LSTM	23.45	3.19

Overall, while the ResNet+LSTM model offers performance comparable to the 3D CNN, its substantially lower computational cost and stable generalization performance make it more suitable for practical implementation in real-world human activity recognition systems.

CNN+LSTM Table 4.39 presents the classification performance of the CNN+LSTM model for six-class activity recognition using RD, RA, and RE features. The model achieved an overall accuracy of 79.11%, with a precision of 76.27%, recall of 76.23%, and F_1-score of 76.09%. These results indicate that the CNN+LSTM architecture provides strong and consistent performance across the different activity classes.

Table 4.39: Classification performance of CNN+LSTM model with Range-Doppler+Range-Azimuth+Range-Elevation inputs for 6-class activity recognition.

Model	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	F_1**-score (%)**
CNN+LSTM	79.11	76.27	76.23	76.09

Fig. 4.11 shows the confusion matrix of the ResNet+LSTM model, highlighting its ability to accurately distinguish between various activities. High diagonal values indicate strong performance in recognizing certain activities, such as "In-Place Movement" and "Walking." The model shows moderate confusion between "sitting on a sofa" and "sitting on the floor," indicating areas for further optimization.

Figure 4.11: Confusion matrix for the CNN+LSTM model for 6-class activity recognition. (C_1) In-place movement, (C_2) lying on the floor, (C_3) lying on a sofa, (C_4) sitting on the floor, (C_5) sitting on a sofa, (C_6) walking.

In the next step, as explained in Section 4.2.2, the Majority Voting technique was applied with various values for the parameter t. Table 4.40 presents the model's accuracy for each tested value. Given the model's sampling frequency, where each segment consists of 10 samples (equivalent to one second of data), a setting of $t = 5$ means that the model classifies and labels five consecutive data

segments without immediately displaying the result. Instead, the predicted labels are stored in a vector. After processing these five segments (totaling five seconds), the final activity label is determined based on the most frequently occurring label within this vector. This approach effectively enhances classification stability by minimizing the impact of occasional misclassifications.

Table 4.40: Accuracy of the model with different value of t.

t	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Accuracy (%)	79.11	79.29	82.30	83.57	84.83	85.52	85.63	86.62	87.21	87.49	88.00	89.06	88.80	89.22	89.49	89.74	89.80	89.01	**90.30**	89.80

The training and validation loss curves for the CNN+LSTM model are illustrated in Fig. 4.12. The model demonstrates stable training with a steady decline in training loss and a consistent validation loss. The absence of significant divergence between the two curves suggests effective generalization and low risk of overfitting.

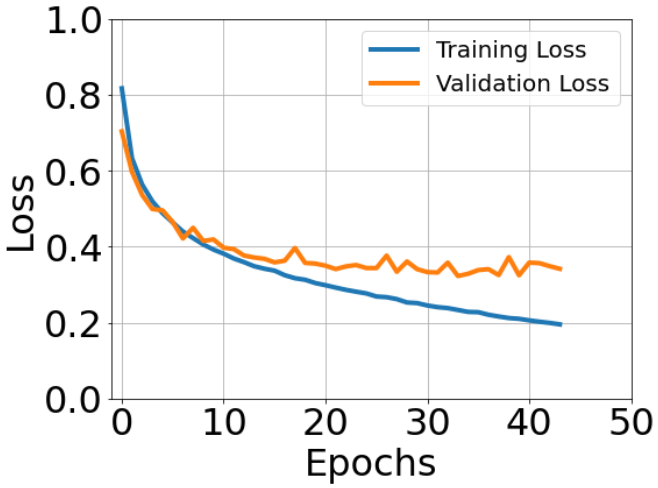


Figure 4.12: Training and validation loss curves for the CNN+LSTM model with Range-Doppler+Range-Azimuth+Range-Elevation inputs for 6-class activity recognition.

In terms of computational cost, the CNN+LSTM model contains approximately 17.31 M parameters and requires 2.3 GFLOPs, as reported in Table 4.41. This combination of relatively low computational complexity and competitive performance makes the model well suited for real-time or resource-constrained deployment environments.

Table 4.41: Computational cost for CNN+LSTM model with Range-Doppler+Range-Azimuth+Range-Elevation inputs for 6-class activity recognition.

Model	**#Flops (GFLOPS)**	**#Parameter (M)**
CNN+LSTM	2.3	17.31

Overall, the CNN+LSTM model delivers robust classification performance while maintaining computational efficiency, positioning it as a practical and effective solution for real-world human activity recognition systems.

Discussion

The activity recognition experiments conducted on the living room dataset provide valuable insights into the effectiveness and robustness of radar-based HAR systems under realistic residential conditions. The results show that deep learning models—particularly the hybrid ResNet18+LSTM and CNN+LSTM architectures—achieved the highest classification accuracies in both the 9-class and 6-class configurations. This demonstrates the importance of jointly exploiting spatial and temporal information for accurate activity recognition.

Compared to the bedroom dataset, the living room environment posed greater challenges due to increased environmental complexity, diverse furniture layout, broader motion trajectories, and a wider range of subject orientations. Despite these complexities, several deep models maintained strong performance, highlighting their robustness to scene variability. The inclusion of range-Doppler, range-azimuth, and range-elevation inputs further enhanced feature richness, improving the recognition of visually and kinematically similar activities, such as "sitting on sofa" versus "sitting on floor."

The 3D CNN model, which performed well in the more controlled bedroom scenario, exhibited a significant drop in accuracy in the living room setting. This can be attributed to multiple factors. First, the bedroom dataset involved a single radar in a confined and structured space, allowing the model to learn consistent spatiotemporal patterns. In contrast, the living room dataset incorporated data from two radars, introducing additional complexity in spatial alignment and requiring models to generalize across multiple viewpoints. Since 3D CNNs are highly sensitive to spatial consistency and localized patterns, any discrepancies in radar fusion or recording conditions likely degraded their performance.

Furthermore, 3D CNNs are inherently limited in modeling long-range temporal dependencies. Although they capture short-term spatiotemporal cues effectively, they are less capable of distinguishing activities with overlapping temporal signatures. For example, activities such as "picking an object from the floor" and "picking an object from a table" contain several motion segments resembling the

"standing" posture. LSTM-based architectures address this limitation by explicitly modeling temporal evolution across frames, enabling more reliable discrimination of sequential patterns.

The ResNet18+LSTM model achieved strong performance in both the 9-class and 6-class tasks, confirming the effectiveness of combining deep spatial feature extraction with temporal modeling. The residual connections in ResNet18 enhance learning of fine-grained radar features, while the LSTM layer captures longer-term dynamics. However, the performance improvement offered by this deeper architecture did not justify its substantially higher computational cost.

To improve robustness and mitigate confusion among activities with similar motion patterns, we consolidated the original 9 classes into 6 broader categories. This class merging led to a noticeable improvement in classification accuracy across all models. This indicates that some activity distinctions were too subtle for radar-based features alone to resolve reliably, especially in complex environments like the living room.

Interestingly, the CNN+LSTM model not only demonstrated substantially lower computational cost but also outperformed the ResNet18+LSTM model in 6-class tasks. This finding highlights that deeper and more complex spatial networks do not always guarantee better performance in radar-based HAR tasks. Instead, the CNN+LSTM model strikes an optimal balance between accuracy and efficiency, making it more suitable for deployment in ambient assisted living systems.

To further enhance performance, we implemented a majority voting ensemble strategy, where outputs were fused at the decision level. This approach led to a consistent improvement in classification accuracy compared to using a single model. Specifically, when using a temporal window of $t = 19$ seconds—meaning the model's predictions over the last 19 seconds were considered and the final label was chosen by majority vote—the accuracy reached up to 90%. This demonstrates that aggregating temporal predictions helps to smooth out transient misclassifications and reinforces consistent activity trends.

However, this improvement came at the cost of increased computational complexity and decision latency. The ensemble approach required running model concurrently, and the majority voting introduced a delay proportional to the window length. While this may be acceptable in high-resource or offline settings, it introduces challenges for real-time or embedded deployments, where rapid inference and low power consumption are critical.

Overall, this comparative analysis underscores that model complexity and ensemble strategies must be balanced with computational demands and deployment constraints. In scenarios where efficiency is paramount, the CNN+LSTM model alone offers a robust and scalable solution. However, when higher accuracy is essential and system resources are available, decision-level fusion via majority voting serves as an effective method to further enhance recognition performance.

Conclusion

The activity recognition experiments conducted using the Living Room Dataset highlight the feasibility and effectiveness of radar-based sensing in realistic domestic environments. By leveraging multiple feature representations—including RD, RA, and RE—we evaluated several model architectures to explore the trade-offs between classification accuracy, computational complexity, and generalization capability.

Among the models tested, the ResNet+LSTM architecture achieved satisfactory performance, particularly in the 6-class classification task. This demonstrates that combining spatial feature extraction through deep residual networks with temporal modeling via LSTM units forms a solid framework for radar-based activity recognition. However, its high computational demand limits its feasibility for real-time or large-scale deployments. In contrast, the hybrid CNN+LSTM model not only outperformed ResNet+LSTM in accuracy but also offered a better balance between performance and efficiency, effectively capturing both spatial and temporal features while maintaining lower computational complexity.

The results also revealed that reducing the number of activity classes—from nine to six—significantly improved classification performance. This suggests that some activities produce similar radar signatures, making them inherently more difficult to distinguish. Nonetheless, the proposed models demonstrated promising results even in the more complex 9-class scenario, highlighting the potential of deep learning to extract meaningful patterns from radar data.

Importantly, the models accurately classified activities performed at different locations within the room, suggesting that they learned activity-specific features rather than relying on positional cues. This contributes to better generalizability across different indoor layouts and deployment contexts.

Moreover, the implementation of ensemble techniques, such as a majority voting strategy, yielded improved overall performance. However, such approaches are not ideal for real-time or large-scale applications due to their added computational overhead.

Despite these promising outcomes, several research challenges remain. The high resource requirements of complex models, limited subject diversity, and constrained environmental variation in the current dataset are key areas for improvement. Future work should focus on developing lightweight architectures, expanding the dataset to include more subjects and diverse environments, and implementing domain adaptation techniques for cross-environment deployment. In addition, real-time processing on edge devices should be explored to enable practical and scalable integration into everyday settings.

Overall, these experiments confirm the viability of using MIMO FMCW radar combined with deep learning techniques for reliable, privacy-preserving human

activity recognition in indoor settings. The findings support radar as a strong candidate for ambient intelligence applications, particularly in elderly care, home monitoring, and smart living environments.

Chapter 5

Summary and Future Outlook

5.1 Technical Contributions

This thesis presents a comprehensive study of HAR using FMCW radar and various ML and DL models, with contributions on dataset creation, signal processing, model development, and empirical evaluation.

An end-to-end radar signal processing pipeline was designed and implemented to transform raw FMCW radar signals into structured feature maps, including RD, RA, and RE representations. The pipeline integrates essential preprocessing techniques such as static clutter removal, FFT-based transformations, and the application of the Capon algorithm.

Two original datasets were collected using FMCW radar devices in fully furnished and realistic indoor environments. The first dataset includes recordings from three subjects in a bedroom using a single radar, while the second involves eleven subjects in a living room monitored by two radars placed at different viewpoints. Both datasets focus on underexplored and challenging daily activities with overlapping motion patterns, offering valuable resources to the radar-based activity recognition community.

The evaluation of multi-view activity classification demonstrates that the trained models achieve robust classification performance across varying spatial positions within the room. This suggests that the models capture activity-specific motion features rather than relying on static positional cues, thereby indicating strong potential for generalization to unseen environments.

A broad range of classification models was implemented and assessed, including classical ML algorithms such as SVM, MLP, k-NN, DT, and RF, as well as DL models including CNN, LSTM, ConvLSTM, CNN+LSTM, and ResNet+LSTM.

These models were evaluated on both multi-class activity recognition and binary fall detection tasks, providing insights into trade-offs in accuracy, computational cost, and deployability.

Finally, ensemble learning methods, particularly majority voting strategies, were employed to aggregate predictions from multiple inputs. These techniques led to measurable improvements in classification performance, while also highlighting trade-offs concerning computational complexity in real-time applications.

5.2 Open Issues and Future Directions

While this thesis demonstrates the viability of radar-based HAR in realistic indoor environments, several critical challenges remain to be addressed to enable robust, scalable, and ethically sound deployment. This section outlines the primary open issues and potential directions for future research.

Dataset Limitations and Standardization Challenges The absence of large-scale, standardized radar datasets significantly limits the ability to fairly benchmark and compare HAR methodologies. Existing studies often rely on small, custom datasets with limited subject diversity, environmental variability, and inconsistent evaluation protocols. Moreover, the field lacks foundation models, large-scale pretraining approaches, and studies addressing domain shift.

Future research should prioritize the development of publicly available datasets that encompass a broad range of demographics, environments, and activity types. These datasets should include multi-radar setups, diverse indoor layouts, and longitudinal recordings to support robust generalization. Establishing standardized benchmarking protocols and requirement-driven dataset specifications would further improve comparability and accelerate progress in the field.

Real-World Deployment and Robustness Challenges Although laboratory experiments yield promising results, deploying HAR systems in real-world settings introduces new challenges. Environmental factors such as changes in furniture layout, the presence of pets, multiple occupants, and even external conditions like weather can degrade performance. Clutter noise remains a critical issue, with existing rejection algorithms often failing under complex conditions.

To ensure robust operation, future systems should incorporate adaptive learning mechanisms capable of handling environmental dynamics. Additionally, improving the interpretability of model decisions is essential for building user trust, particularly in healthcare, assisted living, and other high-stakes applications.

Multi-Person Activity Recognition Most current HAR systems are designed for single-subject scenarios, limiting their applicability in shared environments. Real-world settings often involve multiple individuals performing concurrent activities, which introduces challenges such as signal overlap, occlusion, and multipath interference.

Advancing toward multi-person activity recognition will require improved signal separation techniques, MIMO radar array configurations, and attention-based tracking mechanisms capable of distinguishing individuals and their activities in crowded indoor environments.

Privacy and Ethical Considerations As radar-based HAR systems move toward mainstream deployment, privacy and ethical concerns become increasingly significant. Although radar offers better privacy than camera-based alternatives, issues related to data misuse, consent, and fairness persist.

Future work should explore privacy-preserving strategies such as on-device inference, federated learning, and encrypted model sharing. In parallel, the development of ethical guidelines and regulatory standards will be essential to ensure the responsible and equitable use of these technologies.

Integration with Edge Computing and Real-Time Processing Deep learning models used in HAR are often computationally demanding, posing challenges for deployment on resource-constrained edge devices. Real-time inference is particularly important for time-sensitive applications such as fall detection, home automation, and elderly care.

Future research should focus on lightweight neural architectures, model compression techniques (e.g., pruning, quantization), and hardware-aware optimization. Efficient fusion strategies and adaptive processing pipelines will also be necessary to balance accuracy, latency, and power consumption in embedded systems.

Advanced Signal Processing and Feature Engineering Traditional radar feature maps such as RD, RA, and RE offer valuable information but may not fully capture the rich structure inherent in radar signals.

Future directions should include exploring advanced signal processing techniques such as tensor decomposition, manifold learning, and physics-informed features. These methods can enhance representation capacity, robustness, and interpretability, particularly in complex and cluttered environments.

Multimodal Sensor Fusion Combining radar with complementary sensing modalities—such as Inertial Measurement Units (IMUs), infrared sensors, microphones, and vision-based systems—can significantly improve recognition performance and contextual awareness. Radar fusion from multiple perspectives also

helps reduce occlusion effects and improve spatial coverage.

Future efforts should focus on deep sensor fusion frameworks capable of jointly modeling heterogeneous data streams, dynamically weighting their contributions based on context, and improving resilience to sensor failures or environmental noise.

5.3 Other Potential Application Areas

Beyond activity recognition, radar sensing technology offers promising opportunities across a broad spectrum of domains, owing to its non-intrusive, privacy-preserving, and robust nature. In the healthcare sector, radar systems can non-invasively monitor vital signs such as respiration and heart rate. This capability makes them suitable for sensitive medical applications, including neonatal care, continuous monitoring of the elderly, and chronic disease management. In rehabilitation and physical therapy, radar enables objective assessments of patient recovery by facilitating gait analysis, monitoring adherence to prescribed exercises, and evaluating progress in both sports and clinical contexts.

In smart building environments, radar contributes to energy-efficient infrastructure through occupancy detection and intelligent control of lighting systems, enhancing overall building automation. Within the automotive and transportation industries, radar plays a key role in ensuring passenger safety by enabling in-vehicle monitoring systems capable of detecting driver fatigue, initiating emergency responses, and ensuring compliance with safety regulations.

The technology also shows strong potential in the domain of security and surveillance. Unlike vision-based systems, radar can detect motion and monitor behavior in a privacy-preserving manner, making it ideal for use in sensitive environments. In industrial settings, such as construction sites, chemical plants, and mining operations, radar-based monitoring can improve workplace safety by tracking worker movements and ensuring adherence to safety protocols.

Further applications extend into education and human-computer interaction, where radar facilitates gesture-based interfaces and engagement tracking. These features are particularly beneficial in sterile environments, such as laboratories or surgical rooms, and in accessibility-critical scenarios. In the field of sports and fitness, radar systems provide biomechanical analysis and real-time performance feedback, aiding in injury prevention and enhancing athletic training.

Lastly, radar sensing supports the development of assistive technologies for individuals with disabilities. Capabilities such as gesture control and fall detection empower users by enhancing autonomy and improving their quality of life. Overall, the versatility and unobtrusiveness of radar technology make it a compelling solution for a wide range of applications beyond traditional activity recognition.

List of Abbreviations

2D-LDA two-dimensional Linear Discriminant Analysis. 34

2D-PCA two-dimensional Principal Component Analysis. 34

AAL Ambient-Assisted Living. 47, 57, 60, 89, 93, 94

ADC Analog-to-Digital Converter. 48

AoA Angle of Arrival. 27–30, 43–45

Bi-GRU Bidirectional Gated Recurrent. 36

Bi-LSTM Bi-directional Long Short-Term Memory. 33, 36, 53, 65–67

CCA Canonical Correlation Analysis. 33

CHMR Continuous Human Motion Recognition. 32

CNN Convolutional Neural Network. 8, 33–37, 51, 53, 54, 56, 57, 61–66, 71, 83, 88, 90, 91, 93–97, 101, 103–106, 108

ConvLSTM Convolutional Long Short-Term Memory. 51, 53–55

CSV Cross-Scene Validation. 55, 70, 71, 85, 86, 88, 89

CVAE Convolutional Variational Autoencoder. 37

CW Continuous Wave. 14, 20

CWT Continuous Wavelet Transform. 35

DCNN Deep Convolutional Neural Network. 34

DFT Discrete Fourier Transform. 27, 41

DL Deep Learning. 15–17, 31–34, 36, 48, 51, 54, 56, 57, 60, 62, 63, 65, 66, 70, 71, 83, 87–90, 93, 94, 97, 108

DT Decision Trees. 52, 88, 108

DTW Dynamic Time Warping. 37

ECM Energy Concentration Measure. 34

ELU Exponential Linear Unit. 53

FCC Federal Communications Commission. 49, 50

FCNN Fully Connected Neural Network. 36

FFT Fast Fourier Transformation. 25–28, 36, 39, 40, 42

FIR Finite Impulse Response. 41, 42

FMCW Frequency Modulated Continuous Wave. 14–16, 18, 20–23, 26, 27, 31–35, 39, 41, 47, 48, 56, 57, 88, 89, 93, 108

FN False Negative. 68, 69

FP False Positive. 68, 69

GHz Gigahertz. 19, 26, 33, 48, 49

HAR Human Activity Recognition. 13, 15, 18, 31–36, 38, 47, 48, 70, 84, 88, 104, 108–110

HCI Human-computer Interaction. 13

IF Intermediate Frequency. 23–25, 27

IMUs Inertial Measurement Units. 110

k-NN K-Nearest Neighbor. 34, 53, 108

LH-ViT Lightweight Hybrid Vision Transformer. 32

LOPO-CV Leave-One-Person-Out Cross-Validation. 55, 71, 85, 86, 88, 89, 93

LOS Line of Sight. 14

LSTM Long Short Term Memory. 35, 37, 51, 53–55, 57, 63, 64, 71, 83, 94, 97–99, 101–106, 108

MIMO Multiple Input Multiple Output. 15, 110

ML Machine Learning. 15–17, 31, 33, 34, 36, 46, 48, 51, 54, 56, 70, 83, 87–89, 108

MLP Multi-Layer Perceptron. 52, 71, 74, 78, 80, 83, 88, 89, 108

mmWave Millimeter Wave. 19, 20, 32, 48, 57, 89, 94

MSps Mega Samples Per Second. 48

MTI Moving Target Indicator. 41, 42

MVDR Minimum Variance Distortionless Response. 43

NB Naïve Bayes. 52

PCA Principal Component Analysis. 34, 36, 37, 46, 47, 51, 70

RA Range-Azimuth. 46–48, 51, 54, 56, 57, 60–62, 64, 70, 71, 83, 84, 86–90, 92–95, 97, 99, 101, 106, 108, 110

RADAR Radio Detection And Ranging. 18

RD Range-Doppler. 46–48, 51, 54, 56, 57, 64, 70, 71, 83, 84, 86–89, 94, 95, 97, 99, 101, 106, 108, 110

RD-CNN Range-Distributed Convolutional Neural Network. 35

RE Range-Elevation. 46–48, 51, 54, 56, 57, 60–62, 64, 70, 71, 83, 84, 86–90, 92–95, 97, 99, 101, 106, 108, 110

ResNet Residual Neural Network. 33

RF Random Forest. 52, 88, 108

RIA Research Institute for Aging. 38, 58

RNN Recurrent Neural Network. 36

RTD Range–Time–Doppler. 35

SFCW Single Frequency Continuous Wave. 14, 20, 21

SGD Stochastic Gradient Descent. 54, 62, 66, 67

SINR Signal-to-Interference-plus-Noise Ratio. 43

STFT Short-time Fourier Transform. 35

SVM Support Vector Machines. 33, 34, 36, 52, 71, 74, 78, 83, 88, 89, 108

TI Texas Instruments. 32–37

TN True Negative. 68, 69

TP True Positive. 68, 69

UAVs Unmanned Aerial Vehicles. 18

UnSup-PLevel Unsupervised Pixel-Level. 37

Bibliography

[1] P. Siva, A. Wong, P. Hewston, G. Ioannidis, D. J. Adachi, D. A. Rabinovich, A. Lee, and A. Papaioannou, "Step length measurement in the wild using fmcw radar," *Journal of Ambient Intelligence and Humanized Computing*, 1 2024.

[2] Y. Zhao, H. Zhou, S. Lu, Y. Liu, X. An, and Q. Liu, "Human activity recognition based on non-contact radar data and improved pca method," *Applied Sciences (Switzerland)*, vol. 12, 7 2022.

[3] A. Ferrari, D. Micucci, M. Mobilio, and P. Napoletano, "Deep learning and model personalization in sensor-based human activity recognition," *Journal of Reliable Intelligent Environments*, vol. 9, pp. 27–39, 3 2023.

[4] S. Mirjalali, S. Peng, Z. Fang, C.-H. Wang, and S. Wu, "Wearable sensors for remote health monitoring: Potential applications for early diagnosis of covid-19," *Advanced materials technologies*, vol. 7, no. 1, p. 2100545, 2022.

[5] L. P. Malasinghe, N. Ramzan, and K. Dahal, "Remote patient monitoring: a comprehensive study," *Journal of Ambient Intelligence and Humanized Computing*, vol. 10, pp. 57–76, 2019.

[6] S. Zolfaghari, S. Suravee, D. Riboni, and K. Yordanova, "Sensor-based locomotion data mining for supporting the diagnosis of neurodegenerative disorders: A survey," *ACM Computing Surveys*, vol. 56, 8 2023.

[7] W. C. Willett, J. P. Koplan, R. Nugent, C. Dusenbury, P. Puska, and T. A. Gaziano, "Prevention of chronic disease by means of diet and lifestyle changes," *Disease Control Priorities in Developing Countries. 2nd edition*, 2006.

[8] Y. Li, J. Schoufour, D. D. Wang, K. Dhana, A. Pan, X. Liu, M. Song, G. Liu, H. J. Shin, Q. Sun, *et al.*, "Healthy lifestyle and life expectancy free of cancer, cardiovascular disease, and type 2 diabetes: prospective cohort study," *bmj*, vol. 368, 2020.

[9] B. O'Flynn, J. Torres, J. Connolly, J. Condell, K. Curran, and P. Gardiner, "Novel smart sensor glove for arthritis rehabiliation," in *2013 IEEE International Conference on Body Sensor Networks*, pp. 1–6, IEEE, 2013.

[10] M. Walsh, J. Barton, B. O'Flynn, C. O'Mathuna, A. Hickey, and J. Kellett,

"On the relationship between cummulative movement, clinical scores and clinical outcomes," in *SENSORS, 2012 IEEE*, pp. 1–4, IEEE, 2012.

[11] E. Teixeira, H. Fonseca, F. Diniz-Sousa, L. Veras, G. Boppre, J. Oliveira, D. Pinto, A. J. Alves, A. Barbosa, R. Mendes, *et al.*, "Wearable devices for physical activity and healthcare monitoring in elderly people: A critical review," *Geriatrics*, vol. 6, no. 2, p. 38, 2021.

[12] S. Majumder, T. Mondal, and M. J. Deen, "Wearable sensors for remote health monitoring," *Sensors*, vol. 17, no. 1, 2017.

[13] J. Yang, J. Lee, and J. Choi, "Activity recognition based on rfid object usage for smart mobile devices," *JOURNAL OF COM-PUTER SCIENCE AND TECHNOLOGY*, vol. 26, pp. 239–246, 2011.

[14] G. Saleem, U. I. Bajwa, and R. H. Raza, "Toward human activity recognition: a survey," *Neural Computing and Applications*, vol. 35, pp. 4145–4182, 2 2023.

[15] Z. Hussain, M. Sheng, and W. E. Zhang, "Different approaches for human activity recognition: A survey," *arxiv preprint*, 6 2019.

[16] S. Ahmed, Y. Seo, and S. H. Cho, "Gait asymmetry evaluation using fmcw radar in daily life environments," in *Bioinformatics and Biomedical Engineering* (I. Rojas, O. Valenzuela, F. Rojas Ruiz, L. J. Herrera, and F. Ortuño, eds.), (Cham), pp. 116–127, Springer Nature Switzerland, 2023.

[17] F. J. Abdu, Y. Zhang, and Z. Deng, "Activity classification based on feature fusion of fmcw radar human motion micro-doppler signatures," *IEEE Sensors Journal*, vol. 22, pp. 8648–8662, 5 2022.

[18] Y. Chen and Y. Xue, "A deep learning approach to human activity recognition based on single accelerometer," in *2015 IEEE International Conference on Systems, Man, and Cybernetics*, pp. 1488–1492, Institute of Electrical and Electronics Engineers Inc., 1 2016.

[19] H. Abedi, J. Boger, P. P. Morita, A. Wong, and G. Shaker, "Hallway gait monitoring using novel radar signal processing and unsupervised learning," *IEEE Sensors Journal*, vol. 22, pp. 15133–15145, 8 2022.

[20] L. Chen, J. Hoey, C. D. Nugent, D. J. Cook, and Z. Yu, "Sensor-based activity recognition," 2012.

[21] J. Wang, Y. Chen, S. Hao, X. Peng, and L. Hu, "Deep learning for sensor-based activity recognition: A survey," *Pattern Recognition Letters*, 7 2017.

[22] M. Li, P. Li, S. Tian, K. Tang, and X. Chen, "Estimation of temporal gait parameters using a human body electrostatic sensing-based method," *Sensors (Switzerland)*, vol. 18, 6 2018.

[23] I. Ullmann, R. G. Guendel, N. C. Kruse, F. Fioranelli, and A. Yarovoy, "A survey on radar-based continuous human activity recognition," *IEEE Journal of Microwaves*, vol. 3, pp. 938–950, 7 2023.

[24] A. K. Seifert, M. Grimmer, and A. M. Zoubir, "Doppler radar for the extraction of biomechanical parameters in gait analysis," *IEEE Journal of Biomedical and Health Informatics*, vol. 25, pp. 547–558, 2 2021.

[25] N. Damodaran, E. Haruni, M. Kokhkharova, and J. Schäfer, "Device free human activity and fall recognition using wifi channel state information (csi)," *CCF Transactions on Pervasive Computing and Interaction*, vol. 2, pp. 1–17, 2020.

[26] Y. Ma, G. Zhou, and S. Wang, "Wifi sensing with channel state information: A survey," *ACM Comput. Surv.*, vol. 52, jun 2019.

[27] Z. Peng and C. Li, "Portable microwave radar systems for short-range localization and life tracking: A review," 3 2019.

[28] S. Ahmed, K. D. Kallu, S. Ahmed, and S. H. Cho, "Hand gestures recognition using radar sensors for human-computer-interaction: A review," 2 2021.

[29] K. Papadopoulos and M. Jelali, "A comparative study on recent progress of machine learning-based human activity recognition with radar," *Applied Sciences*, vol. 13, p. 12728, 11 2023.

[30] J.-M. Muñoz-Ferreras, Z. Peng, R. Gómez-García, and C. Li, "Review on advanced short-range multimode continuous-wave radar architectures for healthcare applications," *IEEE Journal of Electromagnetics, RF and Microwaves in Medicine and Biology*, vol. 1, no. 1, pp. 14–25, 2017.

[31] B. Fu, N. Damer, F. Kirchbuchner, and A. Kuijper, "Sensing technology for human activity recognition: A comprehensive survey," *IEEE Access*, vol. 8, pp. 83791–83820, 2020.

[32] A. K. Seifert, M. G. Amin, and A. M. Zoubir, "Toward unobtrusive in-home gait analysis based on radar micro-doppler signatures," *IEEE transactions on bio-medical engineering*, vol. 66, pp. 2629–2640, 9 2019.

[33] K. Saho, K. Uemura, K. Sugano, and M. Matsumoto, "Using micro-doppler radar to measure gait features associated with cognitive functions in elderly adults," *IEEE Access*, vol. 7, pp. 24122–24131, 2019.

[34] P. Addabbo, M. L. Bernardi, F. Biondi, M. Cimitile, C. Clemente, and D. Orlando, "Temporal convolutional neural networks for radar micro-doppler based gait recognition†," *Sensors (Switzerland)*, vol. 21, pp. 1–15, 1 2021.

[35] H. Abedi, "Use of millimeter wave fmcw radar to capture gait parameters," *American Journal of Biomedical Science & Research*, vol. 6, pp. 122–123, 11 2019.

[36] P. Addabbo, M. L. Bernardi, F. Biondi, M. Cimitile, C. Clemente, and D. Orlando, "Gait recognition using fmcw radar and temporal convolutional deep neural networks," in *2020 IEEE 7th International Workshop on Metrology for AeroSpace (MetroAeroSpace)*, pp. 171–175, 2020.

[37] A. Boroomand, G. Shaker, P. P. Morita, A. Wong, and J. Boger, "Autonomous gait speed estimation using 24ghz fmcw radar technology," in *2018*

IEEE EMBS International Conference on Biomedical & Health Informatics (BHI), pp. 66–69, 2018.

[38] C. Y. Hsu, Y. Liu, Z. Kabelac, R. Hristov, D. Katabi, and C. Liu, "Extracting gait velocity and stride length from surrounding radio signals," in *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, vol. 2017-May, pp. 2116–2126, Association for Computing Machinery, 5 2017.

[39] H. Abedi, A. Ansariyan, P. P. Morita, A. Wong, J. Boger, and G. Shaker, "Ai-powered noncontact in-home gait monitoring and activity recognition system based on mm-wave fmcw radar and cloud computing," *IEEE Internet of Things Journal*, vol. 10, pp. 9465–9481, 6 2023.

[40] B. R. Mahafza, *Radar systems analysis and design using MATLAB*. CRC Press, 2013.

[41] S. Ahmed and S. H. Cho, "Machine learning for healthcare radars: Recent progresses in human vital sign measurement and activity recognition," *IEEE Communications Surveys and Tutorials*, vol. 26, pp. 461–495, 2024.

[42] H. H. Meinel, "Evolving automotive radar - from the very beginnings into the future," in *8th European Conference on Antennas and Propagation, EuCAP 2014*, pp. 3107–3114, Institute of Electrical and Electronics Engineers Inc., 2014.

[43] H. Abedi, C. Magnier, V. Mazumdar, and G. Shaker, "Improving passenger safety in cars using novel radar signal processing," *Engineering Reports*, vol. 3, 12 2021.

[44] G.-R. Liu, M.-Z. Zhou, L.-L. Wang, and H. Wang, "A radar-based door open warning technology for vehicle active safety; a radar-based door open warning technology for vehicle active safety," *2016 International Conference on Information System and Artificial Intelligence (ISAI)*, 2016.

[45] P. Huegler, F. Roos, M. Schartel, M. Geiger, and C. Waldschmidt, "Radar taking off," *IEEE Microwave Magazine*, vol. 19, pp. 43–53, 11 2018.

[46] M. I. Skolnik, "Introduction to radar," *Radar handbook*, vol. 2, p. 21, 1962.

[47] D. M. Sheen, D. L. McMakin, and T. E. Hall, "Chapter 9 - detection of explosives by millimeter-wave imaging," in *Counterterrorist Detection Techniques of Explosives* (J. Yinon, ed.), pp. 237–277, Amsterdam: Elsevier Science B.V., 2007.

[48] A. Swindlehurst, E. Ayanoglu, P. Heydari, and F. Capolino, "Millimeter-wave massive mimo: The next wireless revolution?," *IEEE Communications Magazine*, vol. 52, pp. 56–62, 9 2014.

[49] H. Abedi, S. Luo, V. Mazumdar, M. M. Riad, and G. Shaker, "Ai-powered in-vehicle passenger monitoring using low-cost mm-wave radar," *IEEE Access*, vol. 10, pp. 18998–19012, 2021.

[50] S. Hu, S. Cao, N. Toosizadeh, J. Barton, M. G. Hector, and M. J. Fain,

"Radar-based fall detection: A survey," *IEEE Robotics & Automation Magazine*, 2024.

[51] J. Hasch, E. Topak, R. Schnabel, T. Zwick, R. Weigel, and C. Waldschmidt, "Millimeter-wave technology for automotive radar sensors in the 77 ghz frequency band," *IEEE Transactions on Microwave Theory and Techniques*, vol. 60, no. 3, pp. 845–860, 2012.

[52] M. Alizadeh, G. Shaker, J. C. M. D. Almeida, P. P. Morita, and S. Safavi-Naeini, "Remote monitoring of human vital signs using mm-wave fmcw radar," *IEEE Access*, vol. 7, pp. 54958–54968, 2019.

[53] G. Li, Z. Zhang, H. Yang, J. Pan, D. Chen, and J. Zhang, "Capturing human pose using mmwave radar," in *2020 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, pp. 1–6, 2020.

[54] W. Chen, H. Yang, X. Bi, R. Zheng, F. Zhang, P. Bao, Z. Chang, X. Ma, and D. Zhang, "Environment-aware multi-person tracking in indoor environments with mmwave radars," *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, vol. 7, sep 2023.

[55] C. Lovescu and S. Rao, "The fundamentals of millimeter wave radar sensors," *Texas Instruments, Julio*, 2020.

[56] G. M. Brooker *et al.*, "Understanding millimetre wave fmcw radars," in *1st international Conference on Sensing Technology*, vol. 1, 2005.

[57] X. Li, Y. He, and X. Jing, "A survey of deep learning-based human activity recognition in radar," 5 2019.

[58] "Bgt60tr13c - xensiv™ 60ghz radar sensor for advanced sensing." https://www.infineon.com/cms/en/product/sensor/radar-sensors/radar-sensors-for-iot/60ghz-radar/bgt60tr13c/, 2023.

[59] X. Li, X. Wang, Q. Yang, and S. Fu, "Signal processing for tdm mimo fmcw millimeter-wave radar sensors," *IEEE Access*, vol. 9, pp. 167959–167971, 2021.

[60] M. Alizadeh, H. Abedi, and G. Shaker, "Low-cost low-power in-vehicle occupant detection with mm-wave fmcw radar," in *2019 IEEE SENSORS*, pp. 1–4, 2019.

[61] W. Ding, X. Guo, and G. Wang, "Radar-based human activity recognition using hybrid neural network model with multidomain fusion," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 57, no. 5, pp. 2889–2898, 2021.

[62] A. Shrestha, H. Li, J. Le Kernec, and F. Fioranelli, "Continuous human activity classification from fmcw radar with bi-lstm networks," *IEEE Sensors Journal*, vol. 20, no. 22, pp. 13607–13619, 2020.

[63] P. Zhao, C. X. Lu, B. Wang, N. Trigoni, and A. Markham, "Cubelearn: End-to-end learning for human motion recognition from raw mmwave radar

signals," *IEEE Internet of Things Journal*, vol. 10, no. 12, pp. 10236–10249, 2023.

[64] L. Senigagliesi, G. Ciattaglia, D. Disha, and E. Gambi, "Classification of human activities based on automotive radar spectral images using machine learning techniques: a case study," in *2022 IEEE radar conference (RadarConf22)*, pp. 1–6, IEEE, 2022.

[65] Y. Huang, W. Li, Z. Dou, W. Zou, A. Zhang, and Z. Li, "Activity recognition based on millimeter-wave radar by fusing point cloud and range–doppler information," *Signals*, vol. 3, no. 2, pp. 266–283, 2022.

[66] Y.-S. Chen, K.-H. Cheng, Y.-A. Xu, and T.-Y. Juang, "Multi-feature transformer-based learning for continuous human motion recognition with high similarity using mmwave fmcw radar," *Sensors*, vol. 22, no. 21, p. 8409, 2022.

[67] S. Zhu, R. G. Guendel, A. Yarovoy, and F. Fioranelli, "Continuous human activity recognition with distributed radar sensor networks and cnn–rnn architectures," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–15, 2022.

[68] C. Yu, Z. Xu, K. Yan, Y.-R. Chien, S.-H. Fang, and H.-C. Wu, "Noninvasive human activity recognition using millimeter-wave radar," *IEEE Systems Journal*, vol. 16, no. 2, pp. 3036–3047, 2022.

[69] W.-Y. Kim and D.-H. Seo, "Radar-based human activity recognition combining range–time–doppler maps and range-distributed-convolutional neural networks," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–11, 2022.

[70] A. Gorji, A. Bourdoux, S. Pollin, H. Sahli, *et al.*, "Multi-view cnn-lstm architecture for radar-based human activity recognition," *Ieee Access*, vol. 10, pp. 24509–24519, 2022.

[71] F. Xiang, X. Nie, C. Cui, W. Nie, and X. Dong, "Radar-based human activity recognition using two-dimensional feature extraction," in *2023 3rd International Conference on Consumer Electronics and Computer Engineering (ICCECE)*, pp. 267–271, IEEE, 2023.

[72] C. Li, X. Wang, J. Shi, H. Wang, and L. Wan, "Residual neural network driven human activity recognition by exploiting fmcw radar," *IEEE Access*, 2023.

[73] L. R. Triani, N. Ahmadi, and T. Adiono, "Human activity recognition based on fmcw radar using cnn and transfer learning," in *2023 Asia Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pp. 248–253, IEEE, 2023.

[74] G. Liu, X. Li, C. Xu, L. Ma, and H. Li, "Fmcw radar-based human sitting posture detection," *IEEE Access*, 2023.

[75] B. Sheng, Y. Bao, F. Xiao, and L. Gui, "Dyliteradhar: Dynamic lightweight

slowfast network for human activity recognition using mmwave radar," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5, IEEE, 2023.

[76] W.-L. Hsu, J.-X. Liu, C.-C. Yang, and J.-S. Leu, "A fall detection system based on fmcw radar range-doppler image and bi-lstm deep learning," *IEEE Sensors Journal*, 2023.

[77] N. Nguyen, T. Nguyen, M. Pham, and Q. Tran, "Improving human activity classification based on micro-doppler signatures separation of fmcw radar," in *2023 12th International Conference on Control, Automation and Information Sciences (ICCAIS)*, pp. 454–459, IEEE, 2023.

[78] S. Huan, Z. Wang, X. Wang, L. Wu, X. Yang, H. Huang, and G. E. Dai, "A lightweight hybrid vision transformer network for radar-based human activity recognition," *Scientific Reports*, vol. 13, no. 1, p. 17996, 2023.

[79] R. Mehta, S. Sharifzadeh, V. Palade, B. Tan, A. Daneshkhah, and Y. Karayaneva, "Deep learning techniques for radar-based continuous human activity recognition," *Machine Learning and Knowledge Extraction*, vol. 5, no. 4, pp. 1493–1518, 2023.

[80] L. Jiang, M. Wu, L. Che, X. Xu, Y. Mu, and Y. Wu, "Continuous human motion recognition based on fmcw radar and transformer," *Journal of Sensors*, vol. 2023, no. 1, p. 2951812, 2023.

[81] M. Gu, Z. Chen, K. Chen, and H. Pan, "Rmpct-net: a multi-channel parallel cnn and transformer network model applied to har using fmcw radar," *Signal, Image and Video Processing*, vol. 18, no. 3, pp. 2219–2229, 2024.

[82] X. WANG, K. LEI, X. Yang, M. Li, and X. Wang, "Harmonic analysis based on blackman-harris self-multiplication window," in *2020 5th Asia Conference on Power and Electrical Engineering (ACPEE)*, pp. 2165–2169, 2020.

[83] J. O. Smith, *Spectral Audio Signal Processing*. http://ccrma.stanford.edu/-˜jos/sasp/, accessed 2024. online book, 2011 edition.

[84] F. Harris, "On the use of windows for harmonic analysis with the discrete fourier transform," *Proceedings of the IEEE*, vol. 66, no. 1, pp. 51–83, 1978.

[85] X. Zeng, H. S. L. Báruson, and A. Sundvall, "Walking step monitoring with a millimeter-wave radar in real-life environment for disease and fall prevention for the elderly," *Sensors*, vol. 22, no. 24, p. 9901, 2022.

[86] C. Will, P. Vaishnav, A. Chakraborty, and A. Santra, "Human target detection, tracking, and classification using 24-ghz fmcw radar," *IEEE Sensors Journal*, vol. 19, no. 17, pp. 7283–7299, 2019.

[87] M. Ash, M. Ritchie, and K. Chetty, "On the application of digital moving target indication techniques to short-range fmcw radar data," *IEEE Sensors Journal*, vol. 18, no. 10, pp. 4167–4175, 2018.

[88] A. Maćkiewicz and W. Ratajczak, "Principal components analysis (pca)," *Computers & Geosciences*, vol. 19, no. 3, pp. 303–342, 1993.

[89] "Health effects of mmwave radiation." https://www.infineon.com/dgdl/Infineon-Health%20Effects%20of%20mmWave%20Radiation-PI-v01_01-EN.pdf?fileId=5546d46266a498f50166f1ada0520444, 2018.

[90] T. Stadelmayer, M. Stadelmayer, A. Santra, R. Weigel, and F. Lurz, "Human activity classification using mm-wave fmcw radar by improved representation learning," in *Proceedings of the 4th ACM Workshop on Millimeter-Wave Networks and Sensing Systems, mmNets 2020*, Association for Computing Machinery, Inc, 9 2020.

[91] C. Y. Aydogdu, S. Hazra, A. Santra, and R. Weigel, "Multi-modal cross learning for improved people counting using short-range fmcw radar," in *2020 IEEE International Radar Conference (RADAR)*, pp. 250–255, IEEE, 2020.

[92] J. Weib, R. Perez, and E. Biebl, "Improved people counting algorithm for indoor environments using 60 ghz fmcw radar," in *Proceedings of the IEEE National Radar Conference*, vol. 2020-September, Institute of Electrical and Electronics Engineers Inc., 9 2020.

[93] R. Hernangómez, T. Visentin, L. Servadei, H. Khodabakhshandeh, and S. Stańczak, "Improving radar human activity classification using synthetic data with image transformation," *Sensors*, vol. 22, no. 4, p. 1519, 2022.

[94] H. Khodabakhshandeh, T. Visentin, R. Hemangomez, and M. Putz, "Domain adaptation across configurations of fmcw radar for deep learning based human activity classification," in *Proceedings of the International Radar Symposium (IRS)*, vol. 2021-June, IEEE Computer Society, 6 2021.

[95] X. Zeng, F. Wang, B. Wang, C. Wu, K. J. R. Liu, and O. C. Au, "In-vehicle sensing for smart cars," *IEEE Open Journal of Vehicular Technology*, vol. 3, pp. 221–242, 2022.

[96] M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt, and B. Scholkopf, "Support vector machines," *IEEE Intelligent Systems and their applications*, vol. 13, no. 4, pp. 18–28, 1998.

[97] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, pp. 273–297, 1995.

[98] M. Riedmiller and A. Lernen, "Multi layer perceptron," *Machine Learning Lab Special Lecture, University of Freiburg*, vol. 24, 2014.

[99] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *nature*, vol. 323, no. 6088, pp. 533–536, 1986.

[100] G. I. Webb, E. Keogh, and R. Miikkulainen, "Naïve bayes.," *Encyclopedia of machine learning*, vol. 15, no. 1, pp. 713–714, 2010.

[101] Y.-Y. Song and L. Ying, "Decision tree methods: applications for classification and prediction," *Shanghai archives of psychiatry*, vol. 27, no. 2, p. 130, 2015.

[102] L. Breiman, "Random forests," *Machine learning*, vol. 45, pp. 5–32, 2001.

[103] L. E. Peterson, "K-nearest neighbor," *Scholarpedia*, vol. 4, no. 2, p. 1883, 2009.

[104] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-c. Woo, "Convolutional lstm network: A machine learning approach for precipitation nowcasting," *Advances in neural information processing systems*, vol. 28, 2015.

[105] G. Bhavanasi, L. Werthen-Brabants, T. Dhaene, and I. Couckuyt, "Patient activity recognition using radar sensors and machine learning," *Neural Computing and Applications*, vol. 34, no. 18, pp. 16033–16048, 2022.

[106] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural Computation*, vol. 1, no. 4, pp. 541–551, 1989.

[107] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[108] L. Ma, M. Liu, N. Wang, L. Wang, Y. Yang, and H. Wang, "Room-level fall detection based on ultra-wideband (uwb) monostatic radar and convolutional long short-term memory (lstm)," *Sensors*, vol. 20, no. 4, 2020.

[109] F. Kong, J. Deng, and Z. Fan, "Gesture recognition system based on ultrasonic fmcw and convlstm model," *Measurement*, vol. 190, p. 110743, 2022.

[110] W. Ye, J. Cheng, F. Yang, and Y. Xu, "Two-stream convolutional network for improving activity recognition using convolutional long short-term memory networks," *IEEE Access*, vol. 7, pp. 67772–67780, 2019.

[111] C. Ding, L. Zhang, H. Chen, H. Hong, X. Zhu, and C. Li, "Human motion recognition with spatial-temporal-convlstm network using dynamic range-doppler frames based on portable fmcw radar," *IEEE Transactions on Microwave Theory and Techniques*, vol. 70, no. 11, pp. 5029–5038, 2022.

[112] M. Majd and R. Safabakhsh, "A motion-aware convlstm network for action recognition," *Applied Intelligence*, vol. 49, pp. 2515–2521, 2019.

[113] B. Erol, M. Amin, Z. Zhou, and J. Zhang, "Range information for reducing fall false alarms in assisted living," in *2016 IEEE Radar Conference (RadarConf)*, pp. 1–6, 2016.

[114] M. Shen, K.-L. Tsui, M. A. Nussbaum, S. Kim, and F. Lure, "An indoor fall monitoring system: Robust, multistatic radar sensing and explainable, feature-resonated deep neural network," *IEEE Journal of Biomedical and Health Informatics*, vol. 27, no. 4, pp. 1891–1902, 2023.

[115] S. S. Khan and J. Hoey, "Review of fall detection techniques: A data availability perspective," *Medical Engineering & Physics*, vol. 39, pp. 12–22, 2017.

[116] S.-G. Miaou, P.-H. Sung, and C.-Y. Huang, "A customized human fall detection system using omni-camera images and personal information," in *1st*

Transdisciplinary Conference on Distributed Diagnosis and Home Healthcare, 2006. D2H2., pp. 39–42, 2006.

[117] J. Fleming and C. Brayne, "Inability to get up after falling, subsequent time on floor, and summoning help: prospective cohort study in people over 90," *Bmj*, vol. 337, 2008.

[118] N. Pannurat, S. Thiemjarus, and E. Nantajeewarawat, "Automatic fall monitoring: A review," *Sensors*, vol. 14, no. 7, pp. 12900–12936, 2014.

[119] Z. Zhang, Z. Tian, and M. Zhou, "Latern: Dynamic continuous hand gesture recognition using fmcw radar sensor," *IEEE Sensors Journal*, vol. 18, no. 8, pp. 3278–3289, 2018.

[120] W. Li, D. Zhang, Y. Li, Z. Wu, J. Chen, D. Zhang, Y. Hu, Q. Sun, and Y. Chen, "Real-time fall detection using mmwave radar," in *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 16–20, 2022.

[121] W. Li, W. Wang, D. Zhang, and Gegentuya, "Human motion recognition method using millimeter-wave radar based on 3dtsnet," in *2023 5th International Conference on Electronics and Communication, Network and Computer Technology (ECNCT)*, pp. 18–22, 2023.

[122] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[123] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *CoRR*, vol. abs/1704.04861, 2017.

[124] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[125] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proceedings of the 32nd International Conference on Machine Learning* (F. Bach and D. Blei, eds.), vol. 37 of *Proceedings of Machine Learning Research*, (Lille, France), pp. 448–456, PMLR, 07–09 Jul 2015.

[126] A. Graves, S. Fernández, and J. Schmidhuber, "Bidirectional lstm networks for improved phoneme classification and recognition," in *Artificial Neural Networks: Formal Models and Their Applications – ICANN 2005* (W. Duch, J. Kacprzyk, E. Oja, and S. Zadrożny, eds.), (Berlin, Heidelberg), pp. 799–804, Springer Berlin Heidelberg, 2005.