Titre: Title:	User Activity Detection and Feature Selection Using WiFi CSI-Based Datasets
Auteur: Author:	Roya Alizadeh
Date:	2023
Type:	Mémoire ou thèse / Dissertation or Thesis
Référence: Citation:	Alizadeh, R. (2023). User Activity Detection and Feature Selection Using WiFi CSI-Based Datasets [Thèse de doctorat, Polytechnique Montréal]. PolyPublie. https://publications.polymtl.ca/57106/

URL de PolyPublie: PolyPublie URL:	https://publications.polymtl.ca/57106/
Directeurs de recherche: Advisors:	Yvon Savaria
Programme: Program:	Génie électrique

POLYTECHNIQUE MONTRÉAL
affiliée à l'Université de Montréal

User activity detection and feature selection using WiFi CSI-based datasets

ROYA ALIZADEH
Département de génie électrique

Thèse présentée en vue de l'obtention du diplôme de Philosophiae Doctor
Génie électrique

Août 2023

POLYTECHNIQUE MONTRÉAL
affiliée à l'Université de Montréal

Cette thèse intitulée :

User activity detection and feature selection using WiFi CSI-based datasets

présentée par **Roya ALIZADEH**
en vue de l'obtention du diplôme de Philosophiae Doctor
a été dûment acceptée par le jury d'examen constitué de :

Christian CARDINAL, président
Yvon SAVARIA, membre et directeur de recherche
François LEDUC-PRIMEAU, membre
Sébastien ROY, membre externe

DEDICATION

To my brother

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to my director of research, Prof. Yvon Savaria for his constant support, encouragements and supervision of my PhD research. I appreciate his patience and positive attitude which has helped me overcome difficult challenges during my studies. I thank him for providing me with excellent conditions to do my PhD studies at École Polytechnique of Montréal.

I owe a great debt of gratitude to Prof. Chahé Nerguizian for his valuable comments and questions for the completion of this thesis. I would like to thank him for having productive discussions during my research work. I would also like to thank the members of my thesis jury, Dr. François Leduc-Primeau, Dr. Christian Cardinal from École Polytechnique of Montréal, and Dr. Sébastien Roy from Sherbrooke University, for reviewing my work and providing constructive feedbacks.

I would like to thank all of my great friends in Montreal for accompanying me and supporting me during my studies. Last but not the least, my greatest and deepest gratitude goes to my family who have made many sacrifices in their lives for me.

RÉSUMÉ

Une ville intelligente a accès à beaucoup de données à travers divers systèmes interconnectés et des capteurs qui recueillent des informations sur ses résidents, son infrastructure et son environnement. Des dispositifs de type Internet des objets (IoT) et des capteurs sont déployés dans la plupart des villes pour collecter des données en temps réel sur différents aspects, tels que les mouvements des personnes, la qualité de l'air, la température, l'humidité, la circulation routière, la gestion des déchets et la consommation d'énergie. Ces données peuvent être utilisées par des systèmes basés sur l'intelligence artificielle (AI) pour améliorer les opérations de la ville, améliorer les services et prendre des décisions basées sur les données. L'utilisation d'algorithmes avancés peut aboutir à la création d'un environnement urbain efficace et durable pour les résidents.

Dans le cadre de la présente recherche, une méthode de reconnaissance des activités humaines dynamiques (HAR) est appliquée à plusieurs ensembles de données publiques basés sur le WiFi dont l'ensemble (WiAR) à la base de nos travaux. Une évaluation des performances est réalisée à l'aide d'un cadre de traitement des données basé sur l'apprentissage automatique. Après avoir détecté les activités humaines pertinentes, il est souhaitable de distinguer les personnes qui semblent effectuer une activité spécifique. Nous détectons automatiquement l'activité humaine en suivant l'évolution du spectrogramme des signaux RF WiFi et en comparant l'énergie liée au mouvement avec un seuil dynamique. Nous proposons une méthode utilisant la distribution d'énergie à la fois en fréquence et en temps en analysant le spectrogramme des signaux RF WiFi pour distinguer un changement d'énergie dû à un changement de vitesse de mouvement lors d'une activité humaine.

Ensuite, nous évaluons les performances de la méthode proposée lorsqu'elle utilise différentes caractéristiques détectées dans l'ensemble de données. La question est de savoir combien de caractéristiques utiliser et avec quelle dimensionalité pour obtenir de meilleures performances. Pour relever ce défi, nous combinons l'analyse en composantes principales (PCA) et l'algorithme de l'arbre de décision afin de déterminer les caractéristiques du signal qui permettent d'améliorer la précision du modèle et d'obtenir de bonnes performances. La méthode PCA effectue une projection selon les directions ayant la plus grande variabilité. Étant donné que le système varie dynamiquement dans un l'environnement, ses paramètres non déterministes peuvent être contrôlés par un algorithme heuristique ou un arbre de décision pour fournir une solution de prise de décision. L'application de la PCA avant

l'analyse par l'algorithme de l'arbre de décision permet une idenfication efficace des caractéristiques qui aident l'arbre de décision afin qu'il apprenne plus rapidement et produise une précision plus élevée avec le moins de caractéristiques possible. Nous mesurons la pertinence des caractéristiques en fonction : a) de la matrice de covariance de la PCA et b) de l'entropie dans la méthode de prédiction des activités humaines.

Il existe un compromis entre la latence, la précision et la sélection d'un ensemble de caractéristiques. Bien que le temps de calcul réduit soit un choix approprié dans certaines applications, il peut ne pas répondre à une exigence cible souhaitée telle qu'une grande précision ou l'estimation de la durée d'une activité humaine. Ainsi, cette recherche tente de choisir les caractéristiques les plus efficaces en fonction des métriques de performance, puis de recommander les caractéristiques les plus efficaces pour la tâche de détection explorée en fonction de la variabilité observée, du gain d'information et de la corrélation entre les caractéristiques. Les résultats montrent qu'en combinant la PCA et l'algorithme de l'arbre de décision, nous réduisons le nombre de dimension à seulement deux tout en obtenant d'excellentes performances.

ABSTRACT

A smart city is exposed to a lot of data through various interconnected systems and sensors that collect information about its residents, infrastructure, and environment. Internet of Things (IoT) devices and sensors are deployed throughout the city to collect real-time data on various aspects, such as people's movements, air quality, temperature, humidity, traffic flow, waste management, and energy consumption. This data can be used by artificial intelligence (AI) based systems to enhance city operations, improve services, and make data-driven decisions. Using advanced algorithms to process such data can result in an improved and sustainable urban environment for residents.

With that general goal in mind, a dynamic human activity recognition (HAR) method is proposed and applied to a public dataset for WiFi-based activity recognition (WiAR) in this research. A performance evaluation is carried out using a machine learning-based data processing framework. After detecting the related human activities, it is desirable to distinguish people who appear to be performing a specific activity. Our proposed method automatically detects human activity by tracking the evolution of the spectrogram of WiFi RF signals and comparing the movement-related energy with a dynamic threshold. We propose a method based on energy distribution in frequency and time that analyzes the spectrogram of WiFi RF signals to distinguish a change in energy due to a change in speed of motion during a human activity.

The thesis explores and characterizes the performance of the proposed method when it uses different signal features. An important research question is to determine how many signal features to use and with which number of dimensions to achieve good performance while keeping the computational effort at an acceptable level. To overcome this challenge, we apply the principal component analysis (PCA) method before the decision-tree algorithm to efficiently characterize features and to improve the model accuracy while achieving a high performance. PCA transforms data to retain the dimensions that have the highest variability. The environment to detect human activity changes dynamically. There is an element of randomness or uncertainty involved in determining the final outcome. Such non-deterministic behavior and parameters can be controlled by heuristic algorithms or by decision trees to provide a decision-making solution. Applying the PCA method before the decision-tree algorithm allows effective feature characterization and makes the decision-tree algorithm learn faster and achieve a higher accuracy with the least number of features. We measure

the relevance of features based on: a) the PCA covariance matrix, and b) the entropy in the human activity prediction method.

This work is motivated by the evidence of trade-offs between latency, accuracy, and selection of a set of features. Although less computational time is a proper choice in some applications, it may not comply with a desirable target requirement such as high precision or estimating the duration of human activity. Thus, this research attempts to characterize features based on performance metrics and then recommends the most effective features for the explored detection task according to the observed variability, information gain, and correlation between features. The results show that by applying the PCA algorithm before the decision-tree algorithm, we can reduce the number of considered feature dimensions to two while approaching the highest performance that was found to be possible.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

AI	Artificial Intelligence
ANN	Artificial Neural Network
AoA	Angle of Arrival
AP	Access Point
CA	Collision Avoidance
CDF	Cumulative Distribution Function
CFO	Carrier Frequency Offset
CFR	Channel Frequency Response
CIR	Channel Impulse Response
CNN	Convolutional Neural Network
COTS	Commercial off the Shelf
CPU	Central Processing Unit
CSI	Channel State Information
CSMA	Carrier Sense Multiple Access
DC	Direct Connect
DFS	Doppler frequency shift
DNN	Deep Neural Network
DOB–SCV	Distribution Optimally Balanced Stratified Cross-Validation
DS	Direct Sequence
FFT	Fast Fourier Transform
FLMCF	Floor Localization Based on Multi-Classifier Fusion
FMCW	Frequency-Modulated Continuous-Wave
GIS	Geographical Information System
GPS	Global Positioning System
HAR	Human Activity Recognition
HMM	Hidden Markov Model
ICT	Information and Communication Technology
IoT	Internet of Things
ISM	Industrial Scientific and Medical
ITN	Intelligent Transportation Network
ITS	Intelligent Transportation System
KNN	K-Nearest Neighbor
LAN	Local Area Network

LFMW	Linear Frequency Modulated Waveform
LOS	Line of Sight
LR	Logistic Regression
LSTM	Long Short Term Memory
MAC	Medium Access Control
MAPE-K	Monitoring Analysis Planning Execution-Knowledge Base
μD	Micro-Doppler
MIMO	Multiple-Input-Multiple-Output
ML	Machine Learning | Maximum Likelihood
MLP	Multi-Layer Perceptron Neural Network
MU-MIMO	Multi-user Multiple-Input-Multiple-Output
NB	Naive Bayes
NBSVM	Near-Bayesian Support Vector Machine
NIC	Network Interface Card
NLOS	Non-Line of Sight
OFDM	Orthogonal Frequency Division Multiplexing
OFDMA	Orthogonal Frequency Division Multiple Access
OS	Operating System
OSBE	Over-Sampling Balanced Ensemble
PCA	Principal Component Analysis
PHY	Physical Layer
PMF	Probability Mass Function
PNL	Preferred Network List
PSD	Power Spectral Density
RF	Radio Frequency
RFID	Radio Frequency Identification
RMS	Root Mean Square
RNN	Recurrent Neural Network
ROC	Receiver Operating Characteristic
ROCAUC	Receiver Operating Characteristic Area Under Curve
RSSI	Received Signal Strength Indicator
SAR	Synthetic Aperture Radars
SMOTE	Synthetic Minority Oversampling Technique
SSD	Subcarrier Shift Distribution
SSU	Static Sensing Unit
STFT	Short Time Fourier Transform

SVM	Support Vector Machine
SW-ARQ	Stop and Wait for Automatic Repeating reQuest
ToA	Time of Arrival
ToF	Time of Flight
USBE	Under-Sampling Balanced Ensemble
WiAR	WiFi-based Activity Recognition
WLAN	Wireless Local Area Network
WMA	Weighted Moving Average
WSVM	Wavelet Support Vector Machine

CHAPTER 1

INTRODUCTION

Smart cities around the world are implementing initiatives to promote greener and safer urban environments, with cleaner air and water, better mobility, and efficient public services. These initiatives are supported by technologies like the Internet of Things (IoT) and big data analytics, that form the base of the smart city model [1]. According to [2], for a city to be smart, it should use technology and data to deliver a better quality of life. It should create more livable, sustainable, and efficient urban environments that improve the well-being of residents while fostering economic growth and innovation.

1.1 Problem Description

Data analytics is the most important key to creating smart cities, improving urban mobility, and helping these cities better manage their infrastructure. It has an important role in all aspects of city operations and public services. Real-time data collected from sensors and other devices can help to reduce the costs of operation and improve convenience through better information services for users using advanced data analytics.

Human activity recognition (HAR) can be very useful in smart city platforms. HAR is the process of interpreting human motion using advanced data analytics. For instance, human motion can be detected by analyzing data recorded by sensors or WiFi devices. Another challenge with HAR is that activity recognition and detection should be automatic, possibly using artificial intelligence (AI) based algorithms.

1.2 Motivations

In smart cities, human activity recognition and people count (HARC) is an important area of AI research that could help urban planners make informed decisions about infrastructure development. Some authors have reported the capacity to estimate the number of people in a crowd from WiFi signals [9]. However, no dataset allowing crowd estimation was made

public. Several datasets allow human activity recognition from WiFi signals are available [10] [11] [12]. Consequently, the rest of this thesis focuses on HAR a simpler subproblem of HARC. Using advanced algorithms achieves higher accuracy in detecting and classifying human activities. This leads to more reliable results and reduces false positives and false negatives.

Technically, the complexity of human activity detection makes it a challenging domain in smart cities. It has been investigated by several researchers from computer vision [3], signal processing [13], and wireless networking [14]. The drawbacks of the computer-vision-based approach include 1) a requirement for a network of cameras to be installed in the area of interest, resulting in a high deployment cost, 2) an inability to work in the dark, 3) an inability to work behind walls, 4) a requirement for a Line-of-Sight (LoS) condition, and 5) significant privacy issues. Additionally, adapting the location of cameras on a site can be difficult, and complex environments may have numerous blind spots.

Some solutions use wearable devices for human activity recognition [4]. Wearable devices such as smartphones are small and lightweight, making them easy for individuals to wear and carry throughout the day. A limitation of this method is data availability. Indeed, only the participants who have a wearable device or suitable mobile application can be detected and

TABLE 1.1 A summary of approaches, technologies, advantages, and limitations of different techniques for human activity detection.

Approach	Technology	Advantages	Limitations
Vision-based	Camera [3]	High accuracy	High cost, complex computation, privacy issue
Wearable devices	Smartphone [4]	Low cost	The person must carry a device
Device Free	Sound-based method [5]	Low cost	Exposed to environmental noise and limited coverage area
	Infrared [6] [7]	Low cost	Very sensitive to noise, temperature variations, dust, and smoke
	WiFi [8]	Low cost, available everywhere, no privacy concerns	The performance of WiFi-based methods can be influenced by environmental factors like interference, signal attenuation, backscattering, and multipath reflections. It requires advanced analytical and machine-learning-based methods.

it requires a high training cost. Sound-based methods [5] are exposed to environmental noise and their coverage area is too small. Detecting human activity using Infrared IoT sensors [6] is possible but it is relatively unreliable as it is sensitive to electrical noise, temperature variation, dust, and smoke.

By contrast, radio frequency (RF) signals able to penetrate through objects, such as walls, combined with the ubiquity of wireless devices, such as WiFi routers, provide great potential for imaging, tracking [8], and occupancy estimation. This thesis exploits a WiFi-based public domain dataset [12] generated from an Intel 5300 wireless NIC card to apply advanced data analytics and machine learning techniques as a predominant technology to tackle the challenge of human activity detection. Table 1.1 summarizes the approach, technology, advantages, and limitations of the different techniques for human activity detection.

Despite these challenges and disadvantages, WiFi-based methods for activity detection remain a valuable and promising approach due to their widespread availability and cost-effectiveness. Applying deep learning to WiFi-based methods has received strong research interest.

1.3 Proposed Data Processing Framework for HAR

Unlike available surveys related to deep learning in human activity recognition [15], we propose a framework depicted in Figure 1.1 for a system that can automatically detect human activity. That framework is used as a test bed to process channel state information of WiFi signals that we took from the public domain WiAR dataset [12]. That dataset was generated using an Intel 5300 wireless NIC card and its associated CSI tool [16]. The wireless NIC card that was used to produce the test set contains 2 Tx, 3 Rx antennas and each antenna receives 30 subcarriers which creates 180-dimensional CSI vectors. The data was collected using an IEEE 802.11n monitor mode at 5 GHz. In a preprocessing step, 1) we create CSI signatures for each known human activity that can be used as inputs to the classification model [17], 2) since raw CSI data is not directly usable for feature extraction, the weighted moving average method is utilized to filter out noise from the CSI signals [18], 3) the PCA method is then applied for further denoising, and dimensionality reduction. This preprocessing is a vital phase that must be applied to enable successful training of machine learning models while preventing overfitting.

In reference to Figure 1.1, let us first emphasize that people perform the same activity at

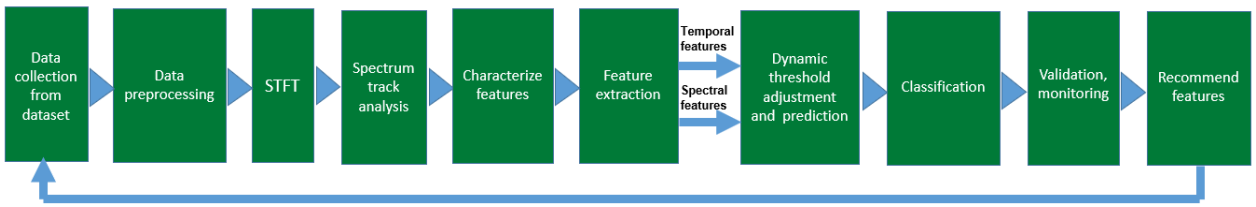


FIGURE 1.1 Proposed human activity recognition architecture.

different speeds and for varying durations. The framework first applies a Short-Time Fourier Transform (STFT) on the preprocessed data to enable analysing the CSI signals both in time and in frequency. Thus, the methods explored in this thesis, track the movement of persons in both time and frequency domains. The goal of characterizing feature components is to choose a proper subset of features that is sufficient to describe a target. In this part, we investigate whether partially orthogonal features allow gaining any substantial additional information or not. Orthogonal signals should provide additional information to the model. Features that are orthogonal to the others should contain information not contained in the others. Moreover, before training a model to extract some features, it is important to apply feature selection techniques and characterize them to avoid overfitting, long training time, and the use of noisy data as input.

The proposed framework applies PCA before the decision tree algorithm to automate the decision-making of features by transforming data in the directions that have the highest variability and information gain. Both temporal and spectral domain features are extracted. Among the various temporal features, dominant features such as chirp rate and frequency of movement are evaluated. Also, spectral features such as the delta band power that we proposed, and the spectral flux are considered. After deriving the values of features both in the time and the spectral domains, we estimate a dynamic threshold as a decision boundary of the classifier to predict the higher movement energy activities that have a higher speed. To estimate the optimal values of various dynamic thresholds, we predict the best threshold of the support vector machine (SVM) to classify the data. The proposed framework also allows applying a 5-fold cross-validation method and a confusion matrix is used to validate the model and monitor the data. It was also observed that there is a trade-off between latency, accuracy, and the set of selected features. This led us to explore the most effective selection of features based on observed variability, information gain, and correlation between features.

1.4 Summary of Contributions

The contributions of this thesis are summarized as follows :

1) **A data processing framework for HAR is proposed and implemented.** It is inspired by the MAPE-K loop method [19]. As HAR is a complex task due to the diversity of activities and the data variability, a method that can do it successfully represents a significant contribution. Previously published work does rather naive preprocessing such as applying a low-pass filter directly and inject the filtered data into ML-based processing methods, which is rather inefficient and much less flexible due to the burst and impulse noises in CSI streams. So, put simply, the proposed framework for HAR in Section 1.3 provides a structured context to combine preprocessing and ML processing methods to implement detection systems that are better organized and efficient.

2) **A workflow is defined to estimate the efficiency of features and detection parameters to improve the recognition performance.** This is a very challenging task explored in Chapter 4, Section 4.2. Activity recognition, as a classification task, shares a common challenge with other classification problems. For these methods to be effective, they must rely on feature extraction methods. Indeed, different activities may exhibit similar characteristics such as walking and running. Consequently, generating distinct features to uniquely represent these activities is a challenging endeavor. Several prior studies have employed machine learning techniques for human activity recognition [20]. They heavily depend on time-frequency transformation [21], and statistical approaches [22]. Authors in [11], [15] only discuss deep learning models such as CNNs and RNNs that can be used for activity recognition. There were no systematic approaches for feature extraction to effectively capture distinctive features for human activities. In this thesis, we expand the scope to the techniques that can be combined successfully with deep learning methods to tackle HAR which is a complex classification problem.

3) **A means to estimate the chirp rate is proposed to detect human activity.** Indeed, this feature proved quite effective when analyzing incoming channel state information (CSI) data from a public domain dataset. The chirp rate relates to the rate at which the signal frequency changes over time [23]. In the context of human activity detection, robust feature detection is essential for dealing with the challenges posed by stochastic environments because it can help to improve the accuracy and reliability of activity recognition systems. It was found that a chirp rate is somewhat invariant to the stochastic nature of the environment. The chirp rate was previously used in radar systems for detecting SAR images [24], [25], but it has never

been used in a context similar to or comparable to the applications explored in this thesis.

4) **We showed that applying PCA before the decision-tree algorithm allows effective characterization and automatic detection of features by transforming data along directions having the highest variability to obtain the largest information gain.** This solution, proposed for the first time in this thesis, addresses the problem of how many features should be used for effective human activity detection. Authors in [26] used some preprocessing methods for WiFi-based crowd estimation by extracting 13 spectral descriptors. However, estimating all those statistics is a computationally expensive process.

5) Finally, **a method is proposed to efficiently classify imbalanced datasets, and the impact of imbalance on feature selection and classification performance is also explored.** The training and evaluation of learning techniques require large annotated data samples which is a time-consuming task. Therefore, the scarcity of annotations poses a significant challenge for activity recognition, giving rise to another issue known as class imbalance. Handling class imbalance is discussed in machine-learning-based algorithms and is developed for the context of human activity detection in this thesis for the first time.

1.5 Publications Resulting from the Thesis

A HAR data processing framework was proposed and published as a first contribution in [17] IEEE 5G World Forum (5GWF) and Chapter 3. In that paper, 1) human activity was characterized based on PCA and spectrogram analysis, 2) CSI data was mapped to different kinds of human activities, 3) These activities were categorized by applying a time series classification algorithm based on the LSTM-RNN model, 4) A method was proposed to distinguish walking and running speeds with CSI changes, 5) HAR accuracy were estimated as well. It is of interest that the speed of movement was proposed as a means to predict people who appear to get into a vehicle, which was the intent of the paper, however, the database that was used in this work was not explicitly captured in the context of a smart transportation system.

A method to estimate features and an adaptive threshold for tracking signal activities in time and frequency from a spectrogram was proposed as a second contribution. The purpose of that estimation method was to try to automatically detect human activity assuming that it is somewhat analogous to people who appear to get into a vehicle as was proposed in [27] IEEE ICECS2022 and in Chapter 4. In [27], the impact of feature selection on the performance

of the classification was explored. This paper also raised the question of how many features should be used for effective human activity detection. Chapter 6 is an attempt to answer this question.

Contributions 3 and 4 listed above in Section 1.4 are the focus of Chapter 6, which also corresponds to a paper accepted recently to the IEEE Journal of Intelligent Transportation Systems. Finally, contribution 5 proposed in Chapter 5 was recently published in IEEE ICECS2023. Table 1.2 summarizes the list of publications in this thesis.

TABLE 1.2 List of publications.

R. Alizadeh, Y. Savaria, and C. Nerguizian, "Human activity recognition and people count for a smart public transportation system," in 2021 IEEE 4th 5G World Forum (5GWF), IEEE, 2021, pp. 182–187.
R. Alizadeh, Y. Savaria, and C. Nerguizian, "Automatic detection of people getting into a bus in a smart public transportation system," in 2022 29th IEEE International Conference on Electronics, Circuits and Systems (ICECS), IEEE, 2022, pp. 1–4.
R. Alizadeh, Y. Savaria and C. Nerguizian, "Characterization and Selection of WiFi Channel State Information Features for Human Activity Detection in a Smart Public Transportation System," in IEEE Open Journal of Intelligent Transportation Systems, doi : 10.1109/OJITS.2023.3336795.
R. Alizadeh, Y. Savaria, and C. Nerguizian, "Enabling Human Activity Recognition in Smart Public Transportation Systems in Presence of Dataset Imbalance," in 2023 30th IEEE International Conference on Electronics, Circuits and Systems (ICECS), IEEE, 2023, pp. 1–4.

1.6 Dissertation Outline

The thesis is organized as follows. In Chapter 2, we present a background overview of human activity recognition using WiFi signals. We describe the development of WiFi and the 802.11 standards and their extensions. We also overview some basic characteristics of wireless channels such as multi-path fading. We review wireless sensing using different types of signals and devices, such as radio-frequency identification (RFID), received signal strength indication (RSSI), mmWave, and CSI. We discuss the challenges of WiFi-based sensing and using deep learning for activity recognition. Methods for addressing imbalanced datasets and a concise overview of the chirp transform are also included.

In Chapter 3, we explore multi-path reflection models for human activity recognition.

We describe WiFi-based human activity recognition based on a proposed human activity recognition (HAR) processing framework. We explain how to pre-process data, reduce data dimensionality, and apply machine learning algorithms to detect different human activities. Performance comparison with the result of processing another dataset using the LSTM-RNN model and deep learning methods are taken into account in this chapter as well.

In Chapter 4, we tackle the challenge of automatically detecting human activity treated as a classification problem using higher movement energy. Suitable dynamic thresholds are identified to classify people according to higher movement energy, with a support vector machine (SVM). Results on human activity detection based on higher movement energy obtained with the WiAR dataset are also compared with those obtained using another dataset.

In Chapter 5, we propose a method to efficiently classify imbalanced datasets. We benchmark the impact of the imbalanced dataset and the feature selection in our machine learning classification algorithm in terms of performance. Performance comparison with another method and another dataset to handle imbalanced datasets are also performed in this chapter.

In Chapter 6, we extract the relevant features of multi-dimensional CSI data pre-processed with the STFT. This chapter notably proposes a method to extract chirp parameters from which HAR can be performed. The analysis proposed in this chapter first characterizes features by applying PCA before the decision-tree algorithm to automate a decision-making solution for feature selection. Moreover, based on the obtained results, we recommend the most effective features for the explored detection task according to the observed variability, information gain, and correlation between features. To validate the generality of the proposed methods, another comparison between results obtained with the WiAR dataset and those obtained with another dataset is performed in relation to the identification of the features important for HAR detection. Finally, Chapters 7 and 8, discuss and conclude this thesis, respectively.

CHAPTER 2

LITERATURE REVIEW ON METHODS TO RECOGNIZE HUMAN ACTIVITY FROM RADIO FREQUENCY SIGNALS AND RELATED SIGNAL PROCESSING TECHNIQUES

Human activity recognition plays an important role in people's everyday life and is a core technology that enables a wide range of applications such as vital signs monitoring [28], human-machine interactions [29], safety surveillance [30], and user identification [31]. Therefore, having a well-defined categorization of sensing technologies is essential to designing human activity recognition systems. Various sensing technologies utilize cameras [3], wearable sensors [4], and RFIDs [32] for human activity recognition. It should be noted that computer-vision and camera-based methods do not work in a dark environment and can be easily affected by line-of-sight obstacles and bad weather that may cause smoke and fog. Additionally, these methods introduce a privacy issue that restricts their wide use.

Methods based on wearable devices have limited sensing capability and require users' active participation that typically imposes wearing sensor devices or carrying specialized sensors. This is a serious limitation in security monitoring applications, such as intrusion detection at home. Researchers also explored RFID-based solutions for activity recognition. However, RFID devices are not common in our daily use and the readers are still expensive for a large-scale deployment. These aforementioned techniques all have key limitations that prevent practical and scalable human activity recognition that is non-intrusive while offering a low-cost solution.

WiFi-based human activity recognition [33], [34], [35], [36], [37] emerges as a promising solution that can overcome these problems. WiFi signals are almost ubiquitous in our daily activities, including our homes, workplaces, business areas, and even many outdoor streets. Radio propagation can reach tens of meters from a WiFi sender even with line-of-sight obstacles, allowing a wide range of activity sensing using only limited devices. With these advantages, it opens a promising option for pervasive and cost-effective activity recognition with very little overhead.

To measure the changes in the received WiFi signals, commercial network interface cards such as the Intel 5300 NIC [16] and Atheors 9580 NIC [38] are available to quantify the

physical layer properties over wireless channels such as the channel state information (CSI).

The received signal strength indicator (RSSI) is another measurable metric of wireless channels. However, it can only detect limited types of human activities due to the coarse-grained channel state information (single path loss value per packet). Additionally, the stability of the RSSI is not guaranteed even in a static indoor environment [39]. By contrast, CSI provides more fine-grained features including both amplitude and phase of multiple sub-carriers. Therefore, we can quantify the exact changes of reflected radio signals caused by human activities with the intent of correlating the CSI patterns with the activity features [12].

2.1 Human Activity Recognition Using WiFi Signals

Recent research has shown a rapid development of human activity recognition through WiFi signals given its advantages of low cost, ubiquitous deployment, and privacy protection. The key point is that different human movements and positions create different multipath distortions in WiFi signals and different patterns in the time and frequency domains of the CSIs. The original activities can be recognized by analyzing the underlying patterns of different WiFi signals. These signals are subject to complex distortions, but recent advances in machine learning bring new possibilities toward more intelligent and accurate recognition methods relating to signal distortions and features of interest.

In this chapter, we review the methods for activity recognition with wireless sensing and learning. We start by introducing the development of 802.11 WiFi standards, and the evolution trend toward wireless sensing. Moreover, we present some basic characteristics of wireless channels to better understand wireless-based activity sensing. We conduct a review of existing wireless sensing approaches using different types of signals such as RFID in Section 2.5, mmWave in Section 2.6, RSSI in Section 2.7, WiFi-based probe monitoring methods in Section 2.8 and CSI in Section 2.9. Then, we discuss the opportunities and challenges of WiFi-based sensing using deep learning methods for activity recognition. We present the CSI-based dataset used to analyze each human activity. We show how to overcome imbalanced sets of data in machine-learning-based analysis. Finally, since the duration and the dynamic speed of each activity are different for each person, we discuss an important feature, the chirp rate that corresponds to the duration and the dynamic speed range of each activity.

2.2 Evolution of WiFi

WiFi is classified under the IEEE standard 802.11 which refers to Wireless Local Area Networks (WLANs). The IEEE 802.11 standard specifies the Medium Access Control (MAC) and the Physical (PHY) layers for wireless connectivity in a local area within a radius of less than 100 meters. It addresses many important issues such as security, power management, and roaming. Therefore, many novel wireless computing applications have been created using WiFi technology [40]. In the next Section, we briefly review the evolution of 802.11 WiFi standards and discuss their evolution from wireless communication to wireless computing.

2.2.1 802.11 Standards and Its Extensions

The rapid development of wireless, broadband, and Internet technologies raises the use of the IEEE 802.11-based Wireless Local Area Network (WLAN) as a ubiquitous networking technology around the world. The 802.11 standard has been further developed by several extensions. Each extension has its specific characteristics, such as frequency band or coverage range. The main goal of each successive extension was to increase the data rate. Table 2.1 summarizes the most important extensions and their characteristics [41].

The 802.11 standard frequency band is 2.4 GHz, which is the unlicensed Industrial, Scientific, and Medical (ISM) short-range radio frequency band [42]. The 802.11 protocol is Carrier Sense Multiple Access (CSMA). The Collision Avoidance (CA) mechanism is used

Table 2.1 IEEE 802.11 standard [41]

IEEE standard	**802.11**	**802.11a**	**802.11b**	**802.11g**	**802.11n**	**802.11ac**	**802.11ax**
Alternative notation	**-**	**-**	**-**	**-**	**WiFi 4**	**WiFi 5**	**WiFi 6**
Publishing year	**1997**	**1999**	**1999**	**2003**	**2009**	**2013**	**2019**
Frequency band [GHz]	**2.4**	**5**	**2.4**	**2.4**	**2.4/5**	**5**	**2.4/5**
bandwidth [MHz]	**22**	**20**	**22**	**20**	**20,40**	**20,40, 80,160**	**20,40, 80,160**
Usable channels	**13**	**19**	**13**	**13**	**13,19**	**19**	**8**
Range [m]	**20–100**	**35–120**	**40–140**	**40–140**	**250**	**50**	**N/A**

in the 802.11 standard to avoid collision during carrier sensing.

IEEE 802.lla operates in the 5 GHz band and supports data rates in the range of 6 to 20 Mbps [43]. It uses Orthogonal Frequency Division Multiplexing (OFDM) instead of direct sequence (DS) spread spectrum and allows 12 non-overlapping channels in a local area. Because 802.lla operates in a higher frequency (5 GHz) band, it faces lesser radio interference than 802.11. Because of its higher data rate, this standard has great potential to support various multimedia applications in a LAN environment. IEEE 802.llb is an enhancement of the 802.11 standard [44]. It operates in the 2.4 GHz band and uses a direct DS spread spectrum. The bandwidth of an 802.11b channel is 22 MHz. IEEE 802.llg is an extension of 802.llb. It is designed to achieve data rates up to 54 Mbps [45]. However, to be compatible with 802.llb, it still uses the 2.4 GHz band, but OFDM is used instead of DS spread spectrum.

The 802.11 standard uses the Stop and Wait for Automatic Repeating reQuest (SW-ARQ) scheme to send data. In this protocol, the sender transmits a single packet and then waits for the acknowledgment. It creates a lot of overhead due to the time spent on sensing the channel before sending the packet and sending the acknowledgment after receiving each packet [46]. To overcome this challenge, IEEE 802.11n uses the BlockAck mechanism (sending single acknowledge instead of several acknowledges) to reduce the protocol overhead, improve the efficiency of the channel and enhance the throughput performance [47].

802.llac further increases the multi-station WLAN throughput to at least 1 Gbps and provides a single link throughput of at least 500 Mbps [48]. This is accomplished by enhancing the air interfaces in 802.11n with wider radio bandwidth (up to 160 MHz) in the 5 GHz band, Multiple-Input-Multiple-Output (MIMO) streams, multi-user MIMO (MU-MIMO), and high-density modulation. The standard 802.11ax (also known as WiFi 6) is a robust, highly efficient signal transmission for better operation at significantly lower signal strengths [49]. It uses orthogonal frequency-division multiple access (OFDMA), down-link MIMO and MU-MIMO, which further increases throughput. It offers real-world speeds that are roughly 30% faster than 802.llac, with theoretical maximum speeds up to around 10 Gbps. The transmission bandwidths for both IEEE 802.11ac and 802.11ax standards are 80 and 160 MHz to achieve a more precise time-based position determination with these WiFi signals [49]. In our work, we use the public domain 802.lln WiFi-based dataset [12] for activity recognition.

2.2.2 Wireless Sensing

With the rapid development of wireless communication over the past decade, considerable attention has been drawn on wireless communication. Because of the ubiquitous coverage and high speed of WiFi, the new generation of smart IoT devices and advanced computing technologies enable a promising direction from wireless communication to wireless sensing. Therefore, a broad range of computing applications, such as human-computer interactions on smart home appliances, elderly care, smart health care systems, safety surveillance, and smart city were created [50].

The movements of objects and humans impact the wireless signal propagation through reflection, diffraction, and scattering, which provide great opportunities to detect an object and human motion by analyzing the received wireless signals. The result of such an analysis can be leveraged for intelligent wireless sensing, e.g., activity recognition, target localization as well as tracking and target identification.

By contrast, vision-based sensing does not work in dark environments and requires a clear line-of-sight path. Wireless sensing, however, has a much lower dependency on the working environment. Additionally, wireless sensing can effectively protect users' privacy, which is quite important in security-sensitive application scenarios. Therefore, wireless sensing has attracted increasing attention in recent years. In the next section, we briefly introduce some characteristics of wireless channels that can affect wireless sensing.

2.3 Channel Characteristics of Wireless Systems

The spatial features of emitted wireless signals are at the basis of wireless-based target detection. In this section, we briefly discuss the characteristics of wireless channels.

2.3.1 Multipath Fading in Wireless Sensor Networks

Multipath fading has a key impact on wireless links. If the position of a receiver or a transmitter is varied by about half a wavelength, the level of attenuation of the signal changes dramatically. This phenomenon is considered to be a small-scale form of fading. Another important physical phenomenon is shadowing ; it is considered to be a large-scale fading. In this case, the position of the receiver and the transmitter are static. It corresponds

to the deviations of the RF signal from its mean due to the presence of large obstacles, which create shadow zones that cause deep fades [51]. It can occur due to reflection, refraction, scattering, and diffraction.

Multipath fading occurs when a signal reaches the receiver via multiple paths (some of which bounce off buildings, ground, and other objects). It causes variations in received signal power. The attenuated signals arrive at different times and phases. The multiple instances of the signals can cancel each other, causing the loss of signal or connection. The problem becomes more severe when higher data rates are explored [52].

Multipath fading is the main factor of signal degradation in an indoor environment. There are lots of static objects and some dynamic moving objects in indoor environments. Refraction, diffraction, and scattering effects caused by the ground and buildings and moving objects such as vehicles in outdoor environments are also important causes of signal degradation.

Several recent researchers have dived deep into the physical layer and leveraged the finer-grained power feature, channel response, to discriminate multipath characteristics. The transmission loss and multipath fading mostly are used to estimate static characteristics of wireless transmission. Such characteristic patterns can be leveraged for static sensing, such as target localization. The attenuation of radio signals during propagation is taken into account to perform indoor localization and detect targets through wireless sensing. The different locations of a target will cause quite diverse signal losses and fading effects [53].

For dynamic sensing of targets, researchers mainly focus on dynamic features derived from wireless channels. A moving object in an indoor environment affects the reflected signals, thus causing dynamic changes to the characteristics of wireless channels.

2.4 Wireless-based Activity Recognition

With the development of wireless technologies and sensing methodologies, the demand for high-performance applications such as human activity recognition, intrusion detection, gesture recognition, and user identification are increased. To implement such emerging applications, relevant features must be extracted from collected wireless signals. Various methods have been proposed based on specific features of different technologies, devices, and types of sensors. These methods can be divided into the following categories : 1) radio-frequency identification (RFID) based methods, 2) mmWave-based methods, 3)

received signal strength indicator (RSSI) based methods, 4) WiFi-based probe monitoring methods, and 5) channel state information (CSI) based methods. These various methods have enabled many novel recognition applications that could be used in smart home and office environments, safety protection, smart healthcare monitoring system, and smart city. In the next sections, we review the literature on wireless-based activity recognition approaches.

2.5 RFID-based Methods

Radio Frequency IDentification (RFID) is a promising technology to automatically identify and track tags by detecting the electromagnetic pulse from a nearby reader. Since the signal strength received by the reader is changed by the movement of humans in the vicinity, it can be used for human activity recognition. Table 2.2 summarizes the existing works focusing on RFID tag-based sensing.

TABLE 2.2 A summary of existing approaches for RFID-based human activity recognition

Authors	Applications	Summary
Yang et al. [54]	Elder Monitoring	Monitor daily behaviors of elder people in healthcare services
Damayanti et al. [32]	Exercise Monitoring	Detect the exercise details like the number of rotations and motion of the hand
Wang et al. [55]	Breathe Monitoring	Monitor breathing by attaching lightweight tags to users' clothes and track the periodic body movement caused by the inhaling and exhaling
Zhang et al. [56]	Gait Analysis	Construct a tag array attached to a wall to analyze people's gait characteristics

Yang et al. [54] proposed a dense convolutional network for RFID-based activity recognition to analyze the basic daily behaviors of elder people in healthcare services. Damayanti et al. [32] developed a device to detect exercise, to track the number of times some exercise has been repeated, and the motion of the hand to guide the person. RFID tags, a RFID reader, and a Hall effect sensor are used for the detection. Wang et al. [55] proposed a method to monitor users' breathing using commercial-off-the-shelf (COTS) RFID systems. In this system, passive lightweight RFID tags are attached to users' clothes. A commodity RFID reader reports the decoded backscatter radio waves. The periodic body movement due to inhaling and exhaling were detected by analyzing the low-level data

reported by commodity readers. Zhang et al. [56] designed a tag array attached to a wall to analyze people's gait characteristics. Although RFID-based methods achieve a relatively good recognition performance in some specific scenarios, RFID readers are expensive. Using them for large-scale deployment would lead to expensive systems. Additionally, RFID devices are susceptible to interference in certain environments, which might cause erroneous data to be analyzed and acted upon.

2.6 MmWave-based Methods

In comparison with WiFi devices with a 2.4/5 GHz frequency band, mmWave-based devices generate a 60 GHz radio signal with a small wavelength. The associated shorter wavelength radio signals create stronger reflection from small objects. Table 2.3 summarizes existing works that explored using mmWave to track objects.

TABLE 2.3 A summary of existing approaches for mmWave-based human activity recognition

Authors/Approach	Applications	Summary
mmTrack [57]	Localization and Object Tracking	Passively localize and track multiple users simultaneously
M-gesture [58]	Gesture Recognition	Build a robust real-time gesture recognition method based on millimeter-wave radar
Meng et al. [59]	Multiple People Detection	Co-existing multiple people using millimeter-wave Sensing
mmEye [60]	Object Imaging	Super-resolution imaging system

For example, mmTrack [57] was implemented on a commodity 802.11ad device and evaluated indoor environments to passively localize and track multiple users simultaneously. This technique improves the spatial resolution by performing digital beamforming using the receiving antennas to detect multiple targets with a robust clustering technique. M-gesture [58] proposed a robust, high-precision in-air gesture sensing system to recognize different gestures for interactive computer graphics based on millimeter-wave radar. Meng et al. [59] proposed a method called mmGaitNet that used 60 GHz mmWave signals to detect multiple persons with about 90% accuracy. To further improve the sensing resolution and accuracy, mmEye [60] proposed a super-resolution imaging system to construct a 60 GHz

radar system using the MUSIC algorithm. They proposed a joint transmitter smoothing method, which jointly uses the transmit and receive arrays to boost the spatial resolution while not sacrificing the aperture of the antenna array. Since mmWave-based approaches highly rely on specialized radar devices that are relatively costly, it is not a suitable solution for large-scale deployment of daily activity sensing.

2.7 RSSI-based Methods

The received signal strength indicator (RSSI) indicates the path loss of wireless signals with respect to a certain distance using any commodity WiFi device [61]. Table 2.4 summarizes the existing works on sensing human activities using RSSI coming from WiFi-based devices.

TABLE 2.4 Summary of existing approaches for RSSI-based human activity recognition

Authors	Applications	Summary
Apidet et al. [62]	Indoor Target Localization	Focus on human movement effects on the accuracy using RSSI
Zixiang et al. [63]	Activity Recognition	Detect human activity based on a genetic algorithm using RSSI
Chuku et al. [64]	Localization	Outlier detection to remove errors in RSSI-based distance estimation using RSSI
Dib et al. [65]	Activity Recognition	Hierarchical threshold-based algorithm to classify RSSI data to detect human activities
Wu et al. [66]	Indoor 3D Localization	Multi-floor localization based on multi-classifier fusion
Mostofi et al. [67]	Crowd Count	Crowd count through walls Using WiFi
Depatla et al. [68]	Occupancy Measurement	Occupancy estimation using only WiFi power measurement

In [63], the authors extracted multiple features from collected RSSI data and utilized a genetic algorithm to recognize human daily activities. [69] proposed to use a single commercial-off-the-shelf transceiver pair to detect a user's breathing rate by measuring RSSI from 16 frequency channels. Authors in [64] proposed a solution to overcome shadowing effects caused by natural and man-made obstacles in order to increase accuracy. To further improve the sensing ability, Dib et al. [65] proposed a hierarchical

threshold-based algorithm to classify RSSI data into two classes. The first one encompasses static movements. The second one groups dynamic movements. In this paper, the features used in machine learning algorithms are unconventionally based on the statistical characterization parameters of the body channel in an indoor environment. These parameters include the average of a slow-fading channel component and a fast-fading component cumulative distribution function (CDF). Wu et al. [66] proposed a WiFi indoor 3D localization method based on multi-classifier fusion named Floor Localization based on Multi-Classifier Fusion (FLMCF) to estimate the height of the location.

Mostofi et al. [67] proposed a method to count the total number of people walking inside a building (or in general behind walls), using readily deployable WiFi transceivers that are installed outside the building, and only based on WiFi RSSI measurements. Key contributions of [67] are : 1) They showed that the effect of a single person on the WiFi link can be modeled using a process that they referred to as a "Renewal-type" random process. 2) They then showed that the inter-event times carry vital information on the total number of people, and are more robust to the attenuation caused by the walls. More specifically, they used theories from the Renewal process literature to model the effect of N people as a superposition of "Renewal-type" processes. They then derived the Probability Mass Function (PMF) of the inter-event times based on this model, and used it to estimate the number of people using a maximum likelihood (ML) estimator.

Authors in [68] were interested in counting the total number of people walking in an area based on only WiFi RSSI measurements between a pair of stationary transmitter/receiver antennas. They proposed a framework based on understanding two important ways that people leave their signature on the transmitted signal : blocking the line of sight (LOS) and scattering effects. By developing a simple motion model, they first characterized mathematically the impact of crowds on blocking the LOS. Then they characterized probabilistically the impact of the total number of people on the scattering effects and the resulting multipath fading component. By putting the two components together, they then developed a mathematical expression for the probability distribution of the received signal amplitude as a function of the total number of occupants using Kullback-Leibler divergence. To confirm the framework, they ran extensive indoor and outdoor experiments with up to nine people and showed that the proposed framework can estimate the total number of people with a good accuracy with only a pair of WiFi cards and the corresponding RSSI measurements.

Based on an outdoor experiment, [68] did both estimations with directional antennas and

with omnidirectional antennas. For the first case, directional antennas are used at both the transmitter and the receiver. More specifically, GD24-15 2.4 GHz parabolic grid antennas were used from Laird Technologies. This model has a 15 dBi gain with 21 degree horizontal and 17 degree vertical beamwidth and is suitable for IEEE 802.11 b/g applications. For the second case, omnidirectional antennas were used at both the transmitter and the receiver.

Although, RSSI has been widely deployed for human activity sensing in recent years, because of the coarse-grained channel state information, it detects limited types of human activities and its value diminishes rapidly with distance. Moreover, since the stability of the RSSI is not guaranteed even in a static indoor environment [39], it is unreliable in many applications.

2.8 WiFi-based Probe Monitoring Methods

Since nowadays each smartphone is generally associated with a specific human user, counting unique MAC addresses of smartphones can generate a fairly close estimate of the number of users in an indoor space and their movement patterns. Although indoor localization using wireless technology has been extensively studied, using probe packets for crowd monitoring has rarely been investigated. WiFi probe requests are special network frames emitted by wireless client devices to discover 802.11 Network Access Points (APs) in their vicinity. As the probe requests carry the unique MAC addresses of users (MAC randomization is so far used in a very limited scope), this enables tracking of a user in an indoor environment, if the location of their smartphone can be tracked.

In [70], authors proposed a non-invasive WiFi-based approach for indoor crowd analysis which works by passively monitoring the probe packets generated by smartphones. The authors achieved crowd monitoring in indoor environments through passive sniffing of WiFi probe requests generated by the client devices. Their aim was to localize multiple users (crowd) and to track their spatio-temporal pattern of movement over a longer term. The authors mentioned that localizing users for crowd analysis passively monitoring the WiFi probe requests appears to be a fairly novel problem worthy of deeper investigation.

By active localization, authors referred to the location fingerprinting technique which works by logging the RSSI values of multiple nearby wireless Access Points (APs) observed by the user's smartphone. Passive localization, on the other hand, deploys a set of listeners that passively listen to the WiFi probe requests generated by the smartphones around them.

By parsing the probe requests, they estimate the number of users and their locations.

In this paper, the authors proposed the MiamiMapper system which can capture the WiFi probes requests of users and can localize individual users as well as capture the flow of large crowds without requiring any Internet connection. The client smartphones are just required to keep their WiFi on in search of the neighboring APs which will emit probe packets. The proposed system captures those probe packets to uniquely identify the number of users and analyzes them for crowd modeling. The entire system is developed using COTS hardware and software.

Active localization is the approach by which pregenerated location fingerprints are used to localize a new user. Passive localization, on the other hand, uses some static Sensing Units (SSUs) to determine the location of a new user without the active participation of the user being located. So, while the active scheme captures AP beacons to localize a user device, the passive scheme uses the probe requests emitted by a user device and captured by a set of Locator(s).

2.8.1 MiamiMapper Test Setup

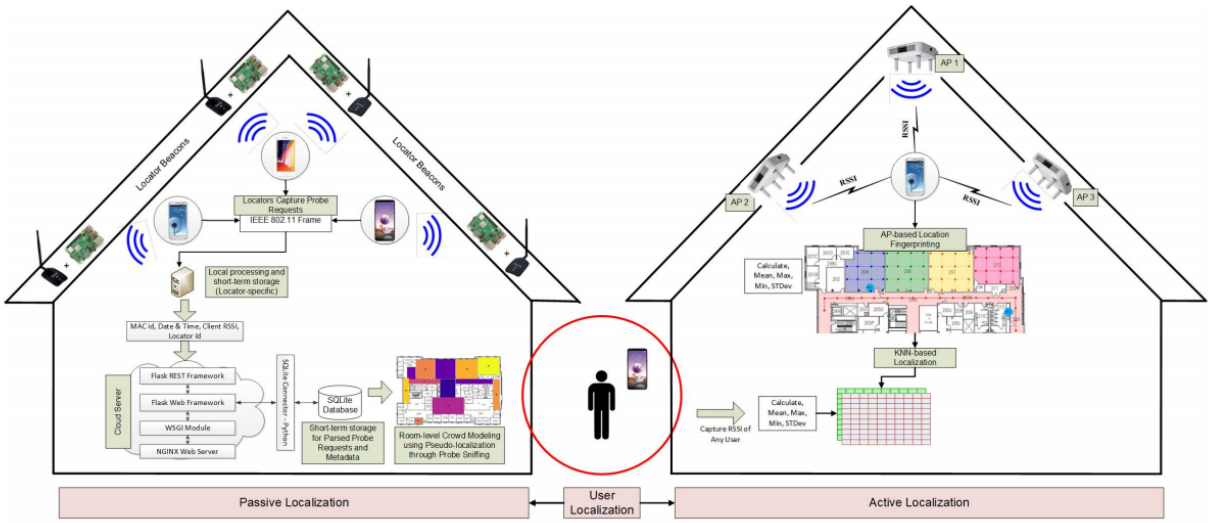


FIGURE 2.1 The MiamiMapper system architecture [70].

The overall system has three different stakeholders – Client, AP, and Locator. A Client is a person carrying a Wi-Fi-enabled smartphone whose location is to be tracked. APs are the wireless access points available in the surroundings of the Client. The Locator side is

comprised of 4 Static Sensing Units (SSU) and a Cloud Server. SSUs are deployed at strategic points. Clients moving through the range of an SSU will be identified uniquely through their MAC IDs. Below, the Client and Locator modules in more detail are described. Figure 2.1 illustrates the MiamiMapper system architecture as well.

Client Side : client devices are wireless devices like laptops and smartphones having Wi-Fi support.

Locator Side : Locator side is divided into two parts – four Static Sensing Units (SSUs) and the Cloud Server. The cloud server carries out the processing of probe requests captured and transmitted by the SSUs and uses the information for user localization.

2.8.2 Performance Metrics

Authors in [70] tested the MiamiMapper system using the following performance metrics :

— No. of Unique Devices (N_d) : It measures the number of unique MAC IDs seen through a day or week. This is the prime metric for crowd measurement.

— No. of Processed Packets : It measures the total number of probe packets being handled by the SSUs on an hourly basis.

— Location accuracy (L_{acc}) : It is the distance (measured in meters) between the actual location of the user to the predicted location. Location accuracy has been calculated for both active and passive localization schemes.

2.9 CSI-based Methods

The limited performance of RSSI measurements due to its noisy and unstable characteristics typically makes such methods inappropriate for human activity recognition. The main drawback of RSSI is that it fails to capture multipath effects. The wireless propagation channel is modeled as Channel Impulse Response (CIR) to fully characterize the individual paths [71]. Authors in [72] showed that compared to traditional signal strength (RSSI) measurements, CSI-based techniques achieve more stable and reliable amplitude and phase information over the wireless link using commodity off-the-shelf WiFi NICs card such as Intel 5300 Wireless Card. Each activity generates a specific pattern on the amplitude and phase of CSI signals. Therefore, CSI is used in a wide range of applications to recognize human activities. Table 2.5 summarizes the existing works to sense

human activities using CSI coming from WiFi-based devices.

TABLE 2.5 A summary of existing approaches for CSI-based human activity recognition

Authors/Approach	Applications	Summary
WiFine [73]	Gesture Recognition	Deployed a lightweight WiFi-based gesture recognition system on low-end edge devices
WiBFall [74]	Activity Recognition	Distinguish the falling activity in a bathroom.
WiEat [75]	Activity Recognition	Monitor users' eating activities
CARM [12]	Activity Recognition	Correlate the human movement velocity and the CSI dynamics to distinguish different activities.
FingerDraw [76]	Activity Recognition	Track finger drawings in the air leveraging WiFi signals
XModal-ID [77]	Person Identification Recognition	Recognize WiFi-video cross-modal gait-based person identification
CrossGR [78]	Gesture Recognition	Develop a deep neural network to perform gesture recognition using WiFi signals
WiVelo [79]	Walking Velocity Recognition	Leverage multipath effects to estimate fine-grained walking velocity using Wi-Fi signals
GaitID [80]	Identification Recognition	Develop a WiFi based human identification system
DeepCount [9]	Crowd Counting	Develop a WiFi-based crowd counting system using deep learning algorithms

WiFine [73] deployed a lightweight WiFi-based gesture recognition system on low-end edge devices. WiBFall [74] applied a CSI-based fall detection approach that constructed a radio propagation model and manually selected features to distinguish the falling activity in the bathroom. WiEat [75] proposed a device-free eating monitoring system to automatically monitor users' eating activities by identifying the fine-grained eating motions and detecting the duration of chewing and swallowing. CARM [12] proposed a novel CSI-speed model that correlates the human movement velocity and the CSI dynamics to detect specific human activities.

Fine-grained recognition applications were explored using CSI-based sensing technology as well. For example, FingerDraw [76] proposed a sub-wavelength level finger motion tracking system using WiFi signals from commodity devices, without attaching any sensor to the finger. To achieve more subtle activity recognition, XModal-ID [77] proposed a framework

to recognize cross-modal gait-based person identification. It uses the WiFi signal to detect the area where a person walks in and a video footage of a walking person in another area to determine whether it is the the same person in both cases or not. CrossGR [78] developed a deep neural network to perform gesture recognition by extracting gesture-related WiFi signal characteristics.

WiVelo [79] further developed accurate velocity estimation by spatial-temporal signal correlation features observed from different antennas. In this paper, two correlation features for direction and speed estimation were defined by extracting subcarrier shift distribution (SSD) from CSI signals. Moreover, the mesh model was designed by the antennas' locations to enable a fine-grained velocity estimation. GaitID [80] estimated various walking trajectories and speeds by extracting target-specific features that best characterize gait patterns. Then the normalization algorithms were applied to eliminate gait irrelevant perturbation in signals.

In order to solve the problem of human behavior perception in a multi-human environment, crowd counting using deep learning and WiFi signals were proposed in [9]. Since the use of WiFi to directly count the crowd is too complicated, Convolutional Neural Networks (CNNs) are used to automatically extract the relationship between the number of people and the channel, and Long Short Term Memory (LSTM) resolves the dependencies of the number of people and CSI.

To overcome the massive labeled data required by deep learning methods an online learning mechanism is added to determine whether or not someone is entering/leaving the room via an activity recognition model [9]. By massive training samples, the end-to-end learning approach can achieve an average of 86.4% prediction accuracy in an environment of up to 5 people. By applying a door switch mechanism into the activity recognition model, the accuracy of the deep learning model was improved up to 90% [9]. Authors in [9] suggested exploring how to improve the maximum distinguished number of people as future work.

Therefore, CSI-based methods are currently used in a wide range of applications and extensively applied in activity recognition systems. They can deliver accurate results without requiring excessive computational resources or complex hardware.

2.10 WiFi-based Dataset

In this section, we introduce the public WiFi-based dataset in terms of activity type, data format, data acquisition methods, and influential factors.

2.10.1 A Dataset for WiFi-Based Human-to-Human Interaction Recognition

The Dataset has been provided by [10] and includes twelve different interactions performed by 40 different pairs of subjects in an indoor environment. Each pair of subjects performed ten trials of each of the twelve interactions and the total number of trials recorded in this dataset for all the 40 pairs of subjects is 4800 trials (40 pairs of subjects $\times$ 12 interactions $\times$ 10 trials). The publicly available CSI tool is used to record the WiFi signals transmitted from a commercial off-the-shelf access point to a desktop computer that is equipped with an Intel 5300 network interface card. The recorded WiFi signals consist of the RSSI and the CSI values [10]. The MIMO streams are established between the access point and the Intel 5300 NIC card. Two transmit antennas at the access point and three receive antennas at the NIC card are considered [10].

This dataset can be exploited to explore WiFi-based human activity recognition methods such as those based on various machine learning algorithms that could be used to recognize different human-to-human interactions. All the interactions were performed in a line-of-sight manner with respect to the access point and the NIC at a frequency band of 5 GHz. Table 2.6 summarizes the specifications of this dataset. Figure 2.2 illustrates the description of the fields in the structure that contain a WiFi packet. Figure 2.3 demonstrates $IEEE$ 802.11n modulation and coding schemes [81].

TABLE 2.6 Specifications Table :

Subject :
Computer Science Applications Signal Processing Human-Computer Interaction Computer Vision and Pattern Recognition
Specific subject area :
Activity recognition, Human-to-human interaction recognition using Wi-Fi signals.
Type of data :
Raw dataset
How data were acquired :
The WiFi signals were captured in an indoor environment (furnished room) from 40 different pairs of subjects while performing twelve different human-to-human interactions. The CSI tool [1] was used to record the WiFi signals transmitted from a commercial off-the-shelf access point to a desktop computer that is equipped with an Intel 5300 Network Interface Card (NIC). The number of transmit antennas in the utilized access point are two antennas, while the number of receive antennas in the employed NIC is three antennas.
Data format :
Raw
Parameters for data collection :
The twelve interactions were thoroughly explained to the subjects before the beginning of data recording. All the interactions were performed in a line-of-sight manner with respect to the access point and the NIC. The access point was configured to operate at a frequency band of 2.4 GHz, wireless channel number 6, and channel bandwidth of 20 MHz.
Description of data collection :
The WiFi signals were recorded from 40 different pairs of subjects while performing twelve different human-to-human interactions. Each pair of subjects performed ten trials of each interaction. The recorded Wi-Fi signals consist of the Received Signal Strength Indicator (RSSI) values and the Channel State Information (CSI) values.
Data source location :
Institution : German Jordanian University, Department of Computer Engineering City/Town/Region : Amman, 11180 Country : Jordan Latitude and longitude (and GPS coordinates) for collected samples/data : $31.77673^{\circ}N$, $35.8025^{\circ}E$
Data accessibility :
Repository name : Mendeley Data Data identification number : 10.17632/3dhn4xnjxw.1 Direct URL to data : https ://data.mendeley.com/datasets/3dhn4xnjxw/draft ?a=90c726d4-5493-4efc-9ee6-973bcd922b31

Field	Description
timestamp_low	The arrival time of the Wi-Fi packet, which is represented by the lower 32 bits of NIC's clock [1]. This timestamp also represents the arrival time of the RSSI and CSI values comprised within the Wi-Fi packet.
Nrx	Nrx represents the number of antennas used at the receiver side (i.e., the NIC) and its value is set to 3.
Ntx	Ntx represents the number of antennas used at the transmitter side (i.e., the access point) and its value is set to 2.
noise	The measured noise over the channel.
agc	Represents the automatic gain control parameter of the NIC measured in dB. The value of this field along with the value in the noise field are necessary to convert the unit of the RSSI values from dB to dBm as described in the CSI tool.
RSSI_a	RSSI_a represents the RSSI value received at the first antenna of the NIC measured in dB.
RSSI_b	RSSI_b represents the RSSI value received at the second antenna of the NIC measured in dB.
RSSI_c	RSSI_c represents the RSSI value received at the third antenna of the NIC measured in dB.
CSI	The channel state information in the form of a complex three-dimensional matrix that has a dimension of Ntx × Nrx × Nsc. Nsc represents the number of subcarriers constructed using the Orthogonal Frequency-Division Multiplexing (OFDM) modulation scheme, which is applied to the utilized 20 MHz wide channel. The CSI tool specifies the value of the Nsc parameter to 30 subcarriers.
label	The recorded trial for any of the twelve human-to-human interactions consists of two types of intervals, namely the steady-state and the interaction intervals. During the steady state interval, the pair of subjects are standing against each other without performing any activity. On the other hand, during the interaction interval, the pair of subjects perform one of the twelve different human-to-human interactions. Thus, this field assigns a label to the Wi-Fi packet to specify whether the packet has arrived during the steady state interval or the interaction interval. In particular, the assigned label is a string of the form In, where n is an integer in the range of 1 to 13 that is assigned to each Wi-Fi packet as follows:

Label	Description
I1	A Wi-Fi packet is labeled as I1 if it arrived while the pair of subjects were performing the approaching interaction.
I2	A Wi-Fi packet is labeled as I2 if it arrived while the pair of subjects were performing the departing interaction.
I3	A Wi-Fi packet is labeled as I3 if it arrived while the pair of subjects were performing the handshaking interaction.
I4	A Wi-Fi packet is labeled as I4 if it arrived while the pair of subjects were performing the high five interaction.
I5	A Wi-Fi packet is labeled as I5 if it arrived while the pair of subjects were performing the hugging interaction.
I6	A Wi-Fi packet is labeled as I6 if it arrived while the pair of subjects were performing the kicking with the left leg interaction.
I7	A Wi-Fi packet is labeled as I7 if it arrived while the pair of subjects were performing the kicking with the right leg interaction.
I8	A Wi-Fi packet is labeled as I8 if it arrived while the pair of subjects were performing the pointing with the left hand interaction.
I9	A Wi-Fi packet is labeled as I9 if it arrived while the pair of subjects were performing the pointing with the right hand interaction.
I10	A Wi-Fi packet is labeled as I10 if it arrived while the pair of subjects were performing the punching with the left hand interaction.
I11	A Wi-Fi packet is labeled as I11 if it arrived while the pair of subjects were performing the punching with the right hand interaction.
I12	A Wi-Fi packet is labeled as I12 if it arrived while the pair of subjects were performing the pushing interaction.
I13	A Wi-Fi packet is labeled as I13 if it arrived while the pair of subjects were in the steady state interval.

FIGURE 2.2 Description of the fields in the structure that contain a WiFi packet [10].

MCS index	Spatial streams	Modulation type	Coding rate	Data rate (Mbit/s)			
				20 MHz channel		40 MHz channel	
				800 ns GI	400 ns GI	800 ns GI	400 ns GI
0	1	BPSK	1/2	6.50	7.20	13.50	15.00
1	1	QPSK	1/2	13.00	14.40	27.00	30.00
2	1	QPSK	3/4	19.50	21.70	40.50	45.00
3	1	16-QAM	1/2	26.00	28.90	54.00	60.00
4	1	16-QAM	3/4	39.00	43.30	81.00	90.00
5	1	64-QAM	2/3	52.00	57.80	108.00	120.00
6	1	64-QAM	3/4	58.50	65.00	121.50	135.00
7	1	64-QAM	5/6	65.00	72.20	135.00	150.00
8	2	BPSK	1/2	13.00	14.40	27.00	30.00
9	2	QPSK	1/2	26.00	28.90	54.00	60.00
10	2	QPSK	3/4	39.00	43.30	81.00	90.00
11	2	16-QAM	1/2	52.00	57.80	108.00	120.00
12	2	16-QAM	3/4	78.00	86.70	162.00	180.00
13	2	64-QAM	2/3	104.00	115.60	216.00	240.00
14	2	64-QAM	3/4	117.00	130.00	243.00	270.00
15	2	64-QAM	5/6	130.00	144.40	270.00	300.00
16	3	BPSK	1/2	19.50	21.70	40.50	45.00
17	3	QPSK	1/2	39.00	43.30	81.00	90.00
18	3	QPSK	3/4	58.50	65.00	121.50	135.00
19	3	16-QAM	1/2	78.00	86.70	162.00	180.00
20	3	16-QAM	3/4	117.00	130.70	243.00	270.00
21	3	64-QAM	2/3	156.00	173.30	324.00	360.00
22	3	64-QAM	3/4	175.50	195.00	364.50	405.00
23	3	64-QAM	5/6	195.00	216.70	405.00	450.00
...	4	...	...	...	...	...	...
31	4	64-QAM	5/6	260.00	288.90	540.00	600.00

FIGURE 2.3 IEEE 802.11n modulation and coding schemes [81].

2.10.2 WiAR : A Public Dataset for WiFi-Based Activity Recognition

The WiAR dataset can be used to explore means of distinguishing the impact of mobility factors in the presence of human activity on WiFi signals and then explore WiFi-based human behavior recognition in different indoor environments [11].

Experimental Setup To obtain this dataset, a commercial TP-Link wireless router was used as the transmitter operating in the IEEE 802.11n AP mode at 5 GHz. A Thinkpad 400 laptop running Ubuntu 10.04 was used as a receiver. This laptop is equipped with an off-the-shelf Intel 5300 card running a modified firmware. During the process of receiving WiFi signals, the receiver pings 30 pkts/s from the router and records the RSSI and CSI

from each packet. Three experimental environments were utilized including an empty room, a meeting room, and an office to collect data. Figure 2.4 illustrates different receiver types and the corresponding number of sub-carriers and frequency bands. This dataset is further described in Chapter 3, Section 3.5.

Types	Subcarriers	Frequency	Modify-firmware	Year
IWL5300	30(2)*	20HMz	Y	2011
IWL5300	30(4)	40HMz	Y	2011
USRP	N/A	N/A	Y	2013
Atheros	56	20HMz	N	2015
Atheros	114	40HMz	N	2015
Intel 7265†	–	–	–	2019

FIGURE 2.4 Different types of receivers and the corresponding number of sub-carriers using CSI tool [11].

Machine learning-based algorithms are required for the advanced processing of WiFi signals to achieve accurate recognition. Therefore, in the next section, we introduce some machine-learning-based methods to detect human activities.

2.11 Deep Learning and Classification Algorithms to Detect Human Activity

In the previous section, we have discussed different methods to collect data for human activity recognition. The next step is to explore the relationship between the signal features and the activities using proper deep learning and classification methods to detect accurate human activity.

Deep learning is a subset of machine learning based on artificial neural networks. It is capable of learning complex patterns and relationships present in the data set. It is inspired by the functioning of the human brain and consists of layers of neurons called interconnected nodes that process and transform data. The main characteristic of deep learning is the use of deep neural networks including one input layer, one or more hidden layer(s), and one output layer. Feed-forward neural networks and recurrent neural networks (RNN) are the popular types of deep learning architecture. Figure 2.5 illustrates the architecture of a feed-forward neural network [82].

RNNs are feed-forward neural networks that introduce a notion of time to the model. It includes a memory component, where information can flow in cycles through the network.

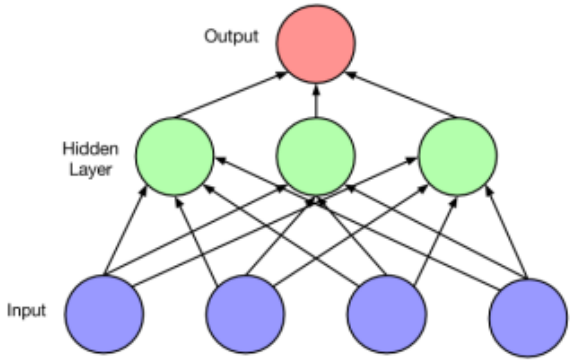


FIGURE 2.5 The architecture of a feed-forward neural network [82].

This allows the network to process sequences of data, such as time series observed in speech or natural languages. Recurrent neural networks are used for many supervised tasks like classification and recognition. Supervised machine learning is the machine learning technique that makes predictions or classifies data based on the labeled datasets [83].

Besides recurrent neural networks which can represent time dependencies, Markov chains can model transitions between states in an observed sequence as well. Hidden Markov models (HMM) have been studied since the 1960s [84]. For example, CARM [12] performed a hidden Markov model (HMM) to predict the sequential states of each human activity. However, this approach is limited because its states must be drawn from a sized discrete state space S. The dynamic programming algorithm that is used to perform efficient inference with hidden Markov models scale in time according to $O(|S|^2)$, where $|S|$ represents the number of states in the HMM [85]. Additionally, the size of the transition table to predict the probability of moving between two time-adjacent states is of the order of $|S|^2$. Therefore, it is infeasible to use HMM for a large set of possible hidden states [86]. Given the limitations of Markov models, it is reasonable to use recurrent neural networks capturing long-range time dependencies.

2.11.1 Classification Algorithms

Classification is a subset of a supervised machine learning algorithm. Supervised learning is a type of machine learning where the algorithm learns from labeled training data. In supervised learning, the algorithm is provided with input data (features) along with corresponding output labels (the correct category or class in which each input data point belongs). The algorithm's objective is to learn a mapping from the input features to the output labels so that it can make predictions on new, unlabeled data. Human activity

detection is an example of the classification. In this section, we describe the classification algorithms used in this thesis.

Long Short Term Memory (LSTM) : LSTM is a special kind of RNN which is capable of learning long-term dependencies. It was introduced by Hochreiter and Schmidhuber to overcome the vanishing gradient problem in 1997 [87]. In this neural network model, a memory block takes the place of each ordinary neuron in the hidden layer of a standard recurrent neural network [82].

LSTMs help preserve the error that can be back-propagated through time and layers. By maintaining a more constant error, they allow recurrent nets to continue to learn over 1000 time steps. Similarly to computer memory, LSTM enables the information to be stored in, written to, or read from a cell. The cell makes decisions about what to store, and when to allow reads, writes, and erases via gates that open and close.

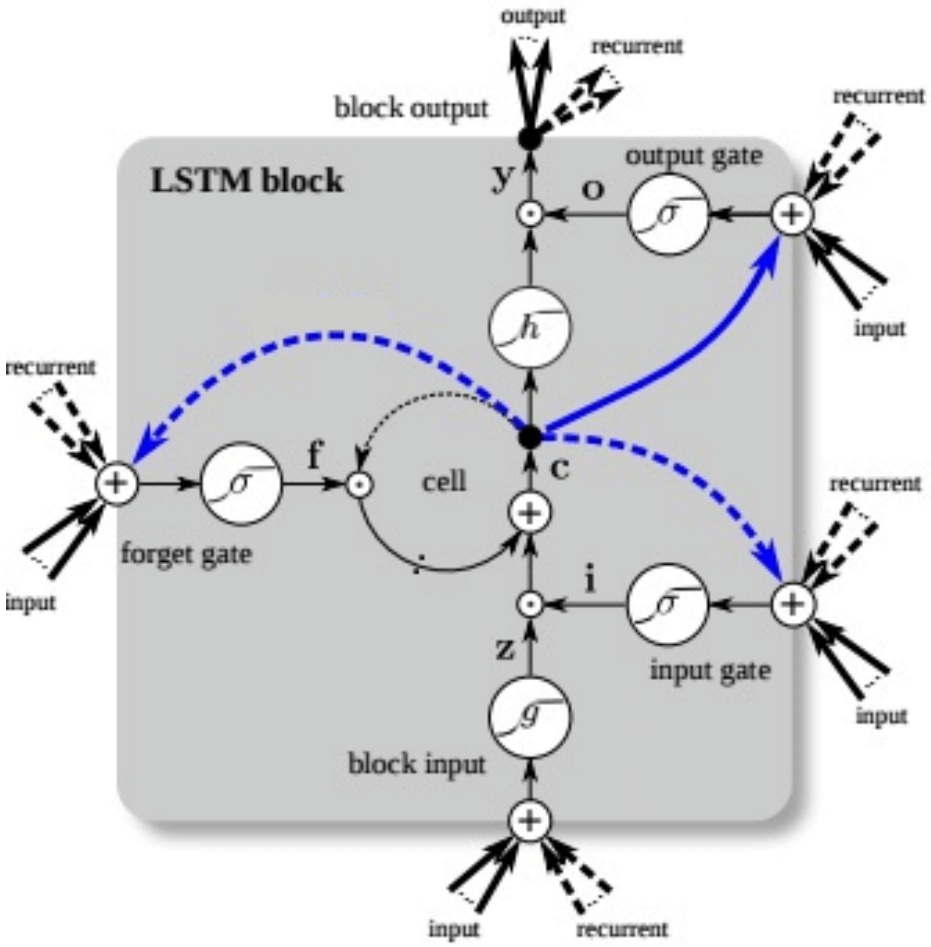


FIGURE 2.6 LSTM block diagram [82].

The LSTM block shown in Figure 2.6 has an input gate, a forget gate, and an output gate which regulate the flow of information into and out of the cell. The forget gate decides how much of the old memory to preserve, the input gate determines what new information to store in this cell, and the output gate gives the output of each cell the same dynamic range as an ordinary tanh hidden unit. Figure 2.6 illustrates different LSTM gates, block input, and block output as the following :

Block input : This unit is labeled z^t as presented in Equation (2.1). $\mathbf{x^t}$ is the input vector at time t, $\mathbf{W_z}$ is the weight connecting $\mathbf{x^t}$ to the three gates and block input, $\mathbf{R_z}$ is the recurrent weight connecting $\mathbf{y^{t-1}}$ to the three gates and block input, b_z is the bias vector and g is the tanh activation function.

Block input is a node that takes activation in the the standard way from the input layer x^t at the current time step as well as from the hidden layer at the previous time step y^{t-1}. The summed weighted input is run through a tanh activation function g.

$$z^t = g(\mathbf{W_z}x^t + \mathbf{R_z}y^{t-1} + b_z) \tag{2.1}$$

Input gate : This unit labeled i^t as presented in Equation (2.2). $\mathbf{W_i}$ is the weight connecting $\mathbf{x^t}$ to the three gates and block input, $\mathbf{R_i}$ is the recurrent weight connecting $\mathbf{y^{t-1}}$ to the three gates and block input, b_i is the bias vector, σ represents the logistic sigmoid function. A gate is a sigmoidal unit that takes activation from the current data point x^t as well as from the hidden layer at the previous time step. A node is called a gate because its value is used to multiply the value of another node. It is a gate in the sense that if its value is zero, then the flow from the other node is cut off. If the value of the gate is one, the flow passes through.

$$i^t = \sigma(\mathbf{W_i}x^t + \mathbf{R_i}y^{t-1} + b_i) \tag{2.2}$$

Forget gate : This unit is labeled as f^t presented in Equation (2.3). $\mathbf{W_f}$, is the weight connecting $\mathbf{x^t}$ to the three gates and block input, $\mathbf{R_f}$ is recurrent weight connecting $\mathbf{y^{t-1}}$ to the three gates and block input, b_f is the bias vector. It provides a method by which the network can learn to flush the contents of the internal state. This is especially useful in continuously running networks.

$$f^t = \sigma(\mathbf{W_f}x^t + \mathbf{R_f}y^{t-1} + b_f) \tag{2.3}$$

Cell state : This unit is labeled c^t as presented in Equation (2.4). $\odot$ is a point-wise multiplication, $\mathbf{i^t}$ is the output of input gate at time t, $\mathbf{z^t}$ is the output of block input at

time t, $\mathbf{f^t}$ is the output of forget gate at time t, $\mathbf{c^{t-1}}$ is the cell state at time $t-1$.

$$c^t = \mathbf{i^t} \odot \mathbf{z^t} + \mathbf{f^t} \odot c^{t-1} \tag{2.4}$$

Output gate : This unit is labeled as o^t presented in Equation (2.5). $\mathbf{W_o}$ is the weight connecting $\mathbf{x^t}$ to the three gates and block input, $\mathbf{R_o}$ is recurrent weight connecting $\mathbf{y^{t-1}}$ to the three gates and block input, b_o is the bias vectors. σ represents the logistic sigmoid function.

$$o^t = \sigma(\mathbf{W_o}x^t + \mathbf{R_o}y^{t-1} + b_o) \tag{2.5}$$

Block output : This unit is labeled as y^t presented in Equation (2.6). $\odot$ is a pointwise multiplication, $\mathbf{o^t}$ is the output of the output gate at time t. The cell state at time t is run through a *tanh* activation function to produce $\mathbf{h(c^t)}$ and then it is multiplied by the value of the output gate to provide y^t.

$$y^t = \mathbf{o^t} \odot \mathbf{h(c^t)} \tag{2.6}$$

Keras categorical-entropy model is applied at the last predicted layer to perform classification.

Support Vector Machine (SVM) : The SVM is a robust classifier because of its versatility and usefulness in grappling with both linear and non-linear challenges. The computational interpretation of the decision-making mechanism with an SVM is described by Equation (2.7) [88].

$$c(f) = sign(\sum_{j=1}^{n} c_j L_j k(I, I_j) + d) \tag{2.7}$$

where I_j is the j^{th} attribute vector, c_j is the category mark, n shows the feature vector dimension. While L_j represents the Lagrange multipliers and d is the bias term. The performance of the SVM classifier depends on the right kernel selection. The linear and polynomial kernels of the SVM are presented as Equations (2.8) and (2.9), respectively.

$$k(I, I_j) = I^T I_j \tag{2.8}$$

$$k(I, I_j) = (I^T I_j + 1)^p \tag{2.9}$$

where p is the degree of the polynomial.

2.12 Techniques to Handle Imbalanced Datasets

A dataset is imbalanced when the number of data available for different classes is different. Since the minority class in an imbalanced dataset carries a risk of not being properly detected, proper metrics and classification algorithms should be considered to predict and detect the desired features. The classifier performance of the applications with imbalanced datasets cannot be analyzed by the classic accuracy metric, which is a percentage of testing samples detected correctly by the algorithm. Indeed, there is a high risk of false negative and false positive predictions. Algorithms related to imbalanced data handling can be categorized into three groups : data-level, algorithmic-level, and ensemble methods. **Data-level approach** focuses on re-sizing the training datasets to balance datasets such as under-sampling and over-sampling. The under-sampling method reduces the number of samples in the majority class. On the other hand, the over-sampling method creates new samples in the minority class. However, the disadvantage of the under-sampling method is that when the samples of the minority class are too rare, we lose tremendous amounts of data from the majority class [89]. Additionally, over-sampling may make the decision regions of the learner smaller and more specific and cause overfitting problems.

Change-direction methods have been proposed to address the problem of balancing and re-sizing a dataset [90]. This class of techniques augments the dataset by creating synthetic data samples based on the original data by performing the best-known over-sampling algorithm called the Synthetic Minority Over-sampling Technique (SMOTE) [91]. The advantage of SMOTE is that it makes the decision regions larger and less specific. However, some classification performance degradation remains associated with data distribution as well [90]. Improved minority over-sampling methods were proposed, such as borderline-SMOTE1 and borderline-SMOTE2 [92]. Those methods only over-sample the minority data near the borderline. Data filtering was proposed in [93] to classify the majority of samples into safe regions, noisy regions, and borderline regions.

Methods in **algorithmic-level** apply algorithms to overcome the imbalanced data problem. The Adaboost algorithm [94] increases the weights of misclassified samples and decreases the weights of the ones classified properly. The authors of [95] proposed a nearest neighbor classification model and performed a convex optimization technique to learn its weight parameters to overcome the imbalanced data problem. In [96] a wavelet support vector machine (WSVM) was considered to perform a filter feature selection technique to remove redundant and irrelevant data. Near-Bayesian Support Vector Machine (NBSVM)

was proposed for imbalanced classification problems combining decision boundary shift and unequal regularization costs by [97].

Ensemble methods combine multiple base classifiers. A data-level algorithm is applied for each base classifier as a pre-processing method. The SMOTEBOOST proposed by [98] is the most popular ensemble method. However, the authors of [99] considered combining three methods 1) re-sampling, 2) ensemble building, and 3) applying a fusion rule to solve the imbalanced problem of data classification.

Data skewed characteristics are considered another problem of imbalanced datasets in validation methods, performance metrics, and data shift processes. Indeed, imbalanced datasets may cause different distributions between training and test partitions. As a result, it leads to inaccurate conclusions when learning data from the training set. This phenomenon is called dataset shift or more specifically covariate shift [100]. The authors of [101] proposed an algorithm to optimally balance the SCV (DOB–SCV) to overcome dataset shifts. In [102], the authors considered three algorithms : 1) Adaboost, 2) Under-Sampling Balanced Ensemble (USBE) and 3) Over-Sampling Balanced Ensemble (OSBE) to solve multi-class problems with a single-label method. In the present study, we propose an ensemble method to evaluate the performance of a human activity recognition (HAR) model.

2.13 A Brief Review of the Chirp Transform

Chirp signals have been developed in many digital signal-processing applications and were initially employed in radar technologies [103]. The frequency of chirp signals either increases (up-chirp) or decreases (down-chirp) with time [104]. They are frequently encountered in various signal processing applications, such as sonar [105], radar, and biomedical devices [106]. Some of these applications rely on chirp signal transmission as in the case of sonar [105], while others model the received signal after Doppler spread as chirp signals, e.g., in synthetic aperture radars (SARs) [107], and heart sound signals [108]. In the literature, various techniques have been developed for the estimation of chirp parameters, including the airborne SAR [24] and the Doppler frequency rate [25].

For automotive radar applications, it is usually called linear frequency modulated waveform (LFMW). The frequency modulation in the Doppler domain is called the micro-Doppler (μD) effect [109]. Gesture recognition is performed using a 77 GHz

Frequency-Modulated Continuous-Wave (FMCW) radar system based on the μD signatures in [109]. These authors applied a simple k-nearest neighbor (KNN) classifier to evaluate the importance of the five micro-Doppler-based handcrafted features for gesture recognition. The overall classification accuracy was 84%.

The chirp-sequence frequency modulation in Figure 2.7 obtains the range r and radial velocity v_r of objects. It uses short-time linear frequency ramps with chirp duration T_{chirp} and bandwidth f_{sweep} at 77 GHz carrier frequency. There are N sampling points in each chirp, and one chirp-sequence cycle consists of K chirp ramps.

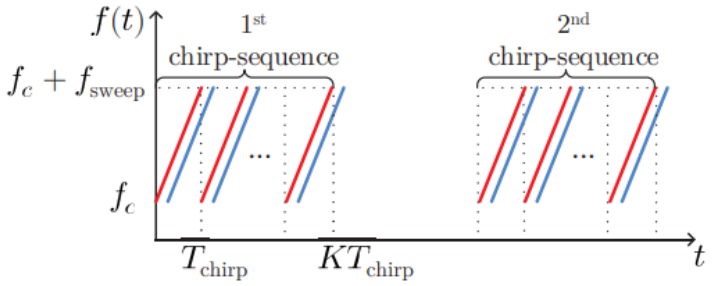


FIGURE 2.7 Chirp-sequence frequency modulation scheme. The red lines denote the transmitted chirp signals, and the blue ones are the received signals [109].

Chirps were found to be robust against noise when trying to detect features of radar signals, particularly when the signal shifts in frequency over time [110]. It was thus conjectured that reflected RF signals in the presence of humans moving are analogous to reflected radar signals in the presence of a moving target. To our knowledge, nobody has ever reported the detection of human activity using chirp parameters.

2.14 Summary

In this chapter, we presented WiFi 802.11 standards for wireless communication. Among different 802.11 standards, we conclude that IEEE 802.lln WiFi-based method enables more accurate and reliable activity sensing. We presented multipath fading in wireless channels and discussed why CSI-based measurements of WiFi signals can create a rich spectrum and robust information. We explained how to detect different human activities analyzing attenuated CSI signals using different methods. In this study, we used the public domain WiAR dataset based on the 802.11n WiFi standard to do the analysis. We conclude that deep learning

methods have brought new possibilities to many IoT applications. In particular, it has enabled intelligent methods to detect different human activities. Using the deep learning method, we can directly extract useful features from raw CSI data using the dataset and classify them by applying classification algorithms. We explained different methods to overcome the imbalanced dataset problem. The reported review of the literature leads to the research hypothesis that extracting chirp parameters from the incoming CSI data can feed a robust machine learning-based model predicting human activity characteristics. This hypothesis will be validated in the rest of the present thesis.

CHAPTER 3

HUMAN ACTIVITY RECOGNITION

In this chapter, we describe a WiFi-based human activity recognition method based on a proposed processing framework. We use the publicly available WiAR dataset [12]. We explain how to pre-process data, reduce the dimension of data, and apply machine learning algorithms to detect different human activities.

3.1 Introduction

Human activity recognition has brought lots of attention in many applications including smart cities. We introduced different methods of activity recognition in the literature review (Chapter 2). The raw CSI data is not sufficient to detect human activity due to the low signal-to-noise ratio associated with these signals. In this chapter, we explore how to estimate each human activity with advanced deep learning methods using CSI data extracted from a public domain WiAR dataset described in Section 3.5. We also explore the so-called long short-term memory (LSTM), an advanced deep learning method [87] for this application. Systems based on the LSTM structure are specifically designed to handle long-term dependencies in sequential data. They comprise mechanisms to capture and remember information over extended sequences. Since human activity is a sequential phenomenon, we present CSI data with the LSTM model, which provides richer features from the time and frequency domains for activity recognition. We propose a human activity recognition (HAR) framework to predict what a person is doing based on movement traces. The reported experimental results show that the accuracy achieved by HAR is 94% when the Adam optimizer method is used.

The rest of this chapter is organized as follows. Section 3.2 presents the Channel State Information (CSI) that can be extracted from a WiFi signal. In Section 3.3, we present a multi-path reflection model that can be used to support human activity recognition. In Section 3.4, we propose a HAR data processing framework. Section 3.5 describes the dataset and the sources of uncertainty that make it hard to process. Data preprocessing steps are explained in Section 3.6. The means to detect human activity are characterized in Section 3.7. The proposed preprocessing method combines PCA and the short-time Fourier transform

(STFT). The actual classification method based on a time series processed with the LSTM-RNN model to detect people is discussed in Section 3.8. A method to detect walking and running speeds is explored in Section 3.9. The performance obtained by the proposed method exploiting the LSTM-RNN model is reported and compared to that obtained when processing the WiAR dataset. The methods proposed by other authors are also discussed in Sections 3.10, 3.11, and 3.12, respectively. Finally, we summarize this chapter in Section 3.13.

3.2 Channel State Information (CSI)

IEEE 802.11n (WiFi-4) devices comprise multiple transmit and receive antennas to multiply the capacity of a radio link using multiple-input and multiple-output (MIMO) to exploit multipath propagation [111]. Let $X(f,t)$ and $Y(f,t)$ be the frequency-time representations of transmitted and received signals, respectively. The signal is generally composed of sub-carriers with carrier frequency f at time t between a given Tx-Rx pair. The channel frequency response (CFR) can be expressed by the $H(f,t)$ matrix that is multiplied by the $X(f,t)$ vector to give the received vector signal $Y(f,t)$ that can be expressed as Equation (3.1).

$$Y(f,t) = H(f,t)X(f,t) + \mathcal{N} \tag{3.1}$$

where $\mathcal{N}$ refers to an additive white Gaussian noise vector in which each term has zero mean and variance σ, $\mathcal{N}(0,\sigma)$ [112]. Since the Intel 5300 WiFi NIC card reports CSI values as a CFR for 30 OFDM subcarriers per Tx-Rx link at each specific time, the indices of these subcarriers correspond to the grouping (Bandwidth = 20 MHz, subcarrier grouping Ng = 2 and to the number of subcarriers Ns = 30) as mentioned in Figure 3.1. The right-most column in Figure 3.1 indicates the subcarrier indices k which depend on the channel BW and Ng. Each CSI stream contains 30 complex matrices with dimensions $N_{Tx} \times N_{Rx}$, where N_{Tx} and N_{Rx} represent the number of transmit and receive antennas, respectively. The CSI values for each subcarrier of a given Tx-Rx link is called a CSI time series and the total number of dimensions in the CSI data is given by $N_{Tx} \times N_{Rx} \times 30$.

As Figure 3.2 illustrates, given two transmitting (Tx) antennas and three receiving

BW	Grouping *Ng*	*Ns*	Carriers for which matrices are sent
20 MHz	1	56	All data and pilot carriers: –28, –27,…–2, –1, 1, 2,…27, 28
	2	30	–28,–26,–24,–22,–20,–18,–16,–14,–12,–10,–8,–6,–4,–2,–1, 1,3,5,7,9,11,13,15,17,19,21,23,25,27,28
	4	16	–28,–24,–20,–16,–12,–8,–4,–1,1,5,9,13,17,21,25,28
40 MHz	1	114	All data and pilot carriers: –58, –57, …, –3, –2, 2, 3,…, 57, 58
	2	58	–58,–56,–54,–52,–50,–48,–46,–44,–42,–40,–38,–36,–34,–32,–30, –28,–26,–24,–22,–20,–18,–16,–14,–12,–10,–8,–6,–4,–2, 2,4,6,8,10,12,14,16,18,20,22,24,26,28, 30,32,34,36,38,40,42,44,46,48,50,52,54,56,58
	4	30	–58,–54,–50,–46,–42,–38,–34,–30,–26,–22,–18,–14,–10,–6, –2, 2,6,10,14,18,22,26,30,34,38,42,46,50,54,58

FIGURE 3.1 Number of subcarriers and carrier grouping (IEEE 802.11n Standards) [113].

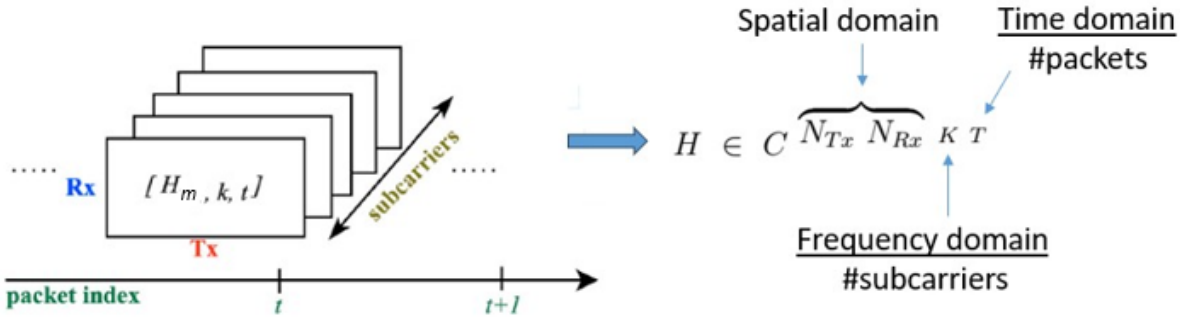


FIGURE 3.2 CSI data is a time series of CSI matrices of MIMO-OFDM channels [114].

antennas (Rx), we can express the matrix H as :

$$H(t) = \begin{pmatrix} H_{1,1}(t) & H_{1,2}(t) & \dots & H_{1,k}(t) & \dots & H_{1,30}(t) \\ H_{2,1}(t) & H_{2,2}(t) & \dots & H_{2,k}(t) & \dots & H_{2,30}(t) \\ H_{3,1}(t) & H_{3,2}(t) & \dots & H_{3,k}(t) & \dots & H_{3,30}(t) \\ H_{4,1}(t) & H_{4,2}(t) & \dots & H_{4,k}(t) & \dots & H_{4,30}(t) \\ H_{5,1}(t) & H_{5,2}(t) & \dots & H_{5,k}(t) & \dots & H_{5,30}(t) \\ H_{6,1}(t) & H_{6,2}(t) & \dots & H_{6,k}(t) & \dots & H_{6,30}(t) \end{pmatrix} \tag{3.2}$$

where $m = N_{Tx}N_{Rx}$, $H_{m,k}(t)$ represents the CSI values in the m^{th} Tx-Rx pair at time t for the k^{th} subcarrier. For each Tx-Rx pair, we obtain 30 CSI values from 30 OFDM subcarriers used by 802.11n [115]. Thus, we get $2 \times 3 \times 30 = 180$ CSI values for each received 802.11n frame when the sender has 2 antennas and the receiver has 3 antennas. The sequence of CSI values

for each subcarrier of a given Tx-Rx pair is called a CSI stream, denoted as Equation (3.3), which corresponds to the rows of the of matrix $H(t)$ expressed in Equation (3.2).

$$\vec{\mathbf{H}}\mathbf{(t)} = [H_{1,1}(t), H_{1,2}(t), ..., H_{1,30}(t), \cdots H_{m,k}(t), \cdots, H_{6,1}(t), H_{6,2}(t), ..., H_{6,30}(t)]^T \quad (3.3)$$

where $\vec{\mathbf{H}}\mathbf{(t)}$ represents the CSI vector obtained at time t and T stands for the transposition operation. The sampling rate is 1000 packets/s to ensure fine-grained information about human activity. For the duration time t, the CSI is given by $[H(1), H(2), ..., H(t)]$. The Intel 5300 NIC reports CSI for 30 groups of subcarriers, spread evenly among all subcarriers of a 20/40 MHz channel [16]. Based on Figure 3.3, the variation of CSI across all the subcarriers in the same Tx-Rx link is similar. However, according to [16], variations of the same subcarrier in various links are different.

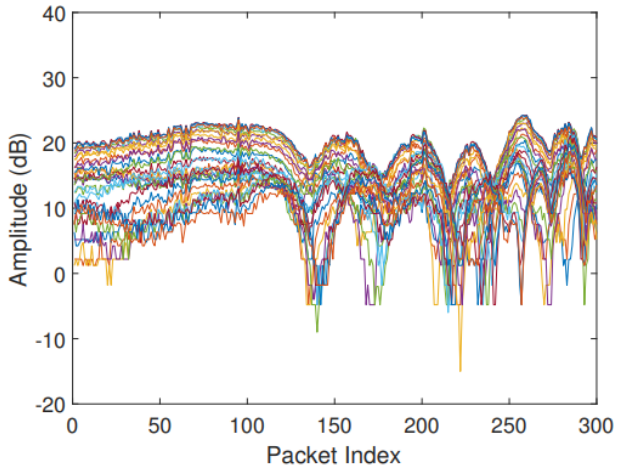


FIGURE 3.3 CSI amplitude variation across 30 subcarriers for the same Tx-Rx link [16].

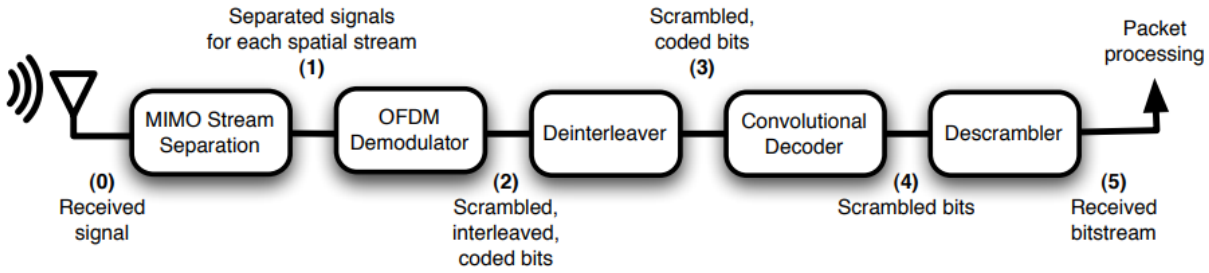


FIGURE 3.4 The 802.11n MIMO-OFDM decoding process [114].

The CSI data is reported by the NIC in a format specified by the standard [115] in the form of packets. Figure 3.4 illustrates the 802.11n MIMO-OFDM decoding process [114].

The MIMO receiver separates the RF signal (0) for each spatial stream (1). Demodulation converts the separated signals into bits (2). Bits from the multiple streams are deinterleaved and combined (3) followed by convolutional decoding (4) to correct errors. Finally, scrambling that randomizes bit patterns is removed and the packet is processed (5) [114]. We address the challenges associated with combining CSI data from 30 OFDM subcarriers in Section 3.6.

3.3 Understanding Multi-path Reflection Usable for Human Activity Recognition

According to [114], the CSI can be used to characterize the CFR of a telecommunication link. The CSI contains information on time delay, amplitude attenuation and phase shift of an incoming signal propagated from a sender to a receiver [30]. Figure 3.5 illustrates the

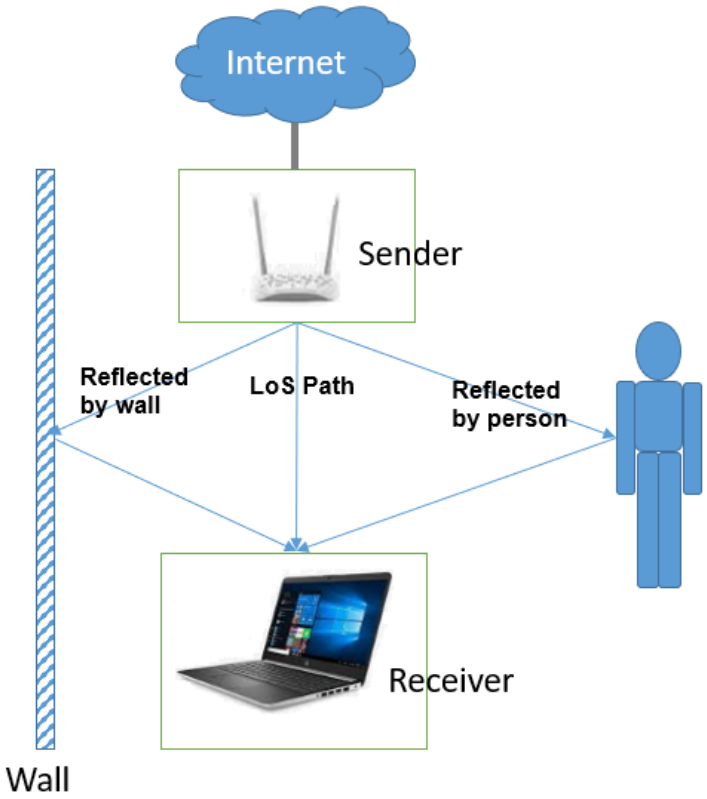


FIGURE 3.5 HAR system architecture.

main elements of a HAR system. In that illustrative example, the access point and laptop contain 2 and 3 antennas, respectively. As this figure shows, the received signal from the WiFi transceiver consists of both static and dynamic paths. Therefore, the CFR $H(f,t)$ depends on both static reflected paths such as walls and dynamic reflected paths modulated by human

activities. $H(f,t)$ is represented by [12] :

$$H(f,t) = e^{-j2\pi \triangle ft}(H_s(f) + \sum_{k=1}^{P_d} a_k(f,t)e^{-j2\pi f\tau_k(t)}), \tag{3.4}$$

where H_s is the aggregation of the static CFR, $a_k(f,t)$ and $\tau_k(t)$ are the complex channel attenuation and the time of flight for path k, respectively, and P_d is the number of dynamic paths. The carrier frequency offset (CFO) term $e^{-j2\pi \triangle ft}$ also comprise a phase shift caused by the carrier frequency difference $\triangle f$ between the sender and the receiver [116]. Wireless signals travel at the speed of light c and the delay of the k^{th} path represented by τ_k.

$$\tau_k = d_k(t)/c \tag{3.5}$$

where $d_k(t)$ is the signal propagation distance. Thus, a dynamic CFR H_d is given by :

$$H_d(f,t) = \sum_{k=1}^{P_d} a_k(f,t)e^{-j2\pi f d_k(t)/c} = \sum_{k\in P_d} a_k(f,t)e^{-j2\pi d_k(t)/\lambda} \tag{3.6}$$

$$d_k(t) = \int_{-\infty}^{t} \upsilon_k(u)du \tag{3.7}$$

where f and λ represent the carrier frequency and the wavelength, $\lambda = c/f$, and υ is the rate of path length change corresponding to the movement of a human. Equations (3.6) and (3.7) relate the variation of CSI to the motion speed υ [12]. For example, the running activity has the highest rate of path length change in higher frequencies, while the walking activity has a lower rate of path length change reflected in lower frequencies. However, the standing activity has no path length change. As a result, we can detect different activities by correlating CSI power and motion speed.

The CFR power is obtained by $|H(f,t)|^2$. Based on Parseval's theorem the total energy of a signal in the time domain equals the spectral power in the frequency domain [117]. We estimate the energy/power in the frequency domain over a time interval by transforming the Short-Time Fourier transform (STFT) of the human activity CSI signal. Thus the frequency range obtained from the STFT transform reflects the frequency of movement.

3.4 Human Activity Recognition (HAR) Data Processing Framework

Figure 3.6 illustrates the proposed HAR data processing framework. It contains modules to monitor and analyze data, perform planned response actions, and execute actions by applying knowledge representation of the system inspired by the Monitoring-Analysis-Planning-Execution-Knowledge-Base (MAPE-K) loop administration [19]. We use the public dataset for WiFi-based activity recognition (WiAR) proposed in [12] as a data set for evaluating the performance of HAR systems. In the HAR data processing framework that we propose, a data pre-processing module denoises the raw collected data using principal component analysis (PCA). Moreover, the start and the end of an activity are detected from PCA results.

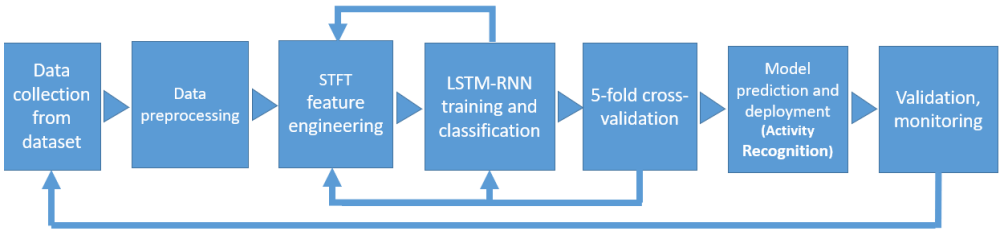


FIGURE 3.6 Proposed HAR data processing framework.

The proposed system utilizes STFT to extract features. Within the WiAR dataset, CSI values are received from multiple paths and their values change as humans move. To address this challenge, STFT is employed to identify distinct frequency components associated with various motion speeds. STFT facilitates the analysis of both fast and slow movements in human activities.

Since different people perform the same activity at different speeds, LSTM-RNN model is applied to the training and classification component of the HAR data processing framework to build a robust CSI-based human activity model. In the validation component, training samples (80%) are separated from test samples (20%). Moreover, a 5-fold cross-validation is adopted to prevent overfitting. A prediction component maximizes the prediction accuracy based on features. In the monitoring component, variations of the CSI and frequency/energy of movement are monitored and the report related to the number of people is provided.

3.5 Dataset

A WiFi-based device-free human activity recognition dataset is available from [12]. In this dataset, only the amplitude of CSIs was considered. [12] provided a public dataset using a desktop computer equipped with Intel 5300 WiFi card and $IEEE$ 802.11n access point. The CSI tool developed by Halperin et al. [16] obtains CSI values transmitted by the access point. All the experiments for this dataset were performed in the 5 GHz frequency band with 20 MHz bandwidth channels in three different indoor environments. In the $IEEE$ 802.11n setting, 20 MHz channels are divided into 312.5 kHz bands called subcarriers, each of which sends independent data simultaneously. Convolutional coding is applied across the bits for error correction and bits are interleaved to spread them in frequency. Each subcarrier in a packet is modulated equally [114].

In this dataset, $N_{Tx} = 2$ and $N_{Rx} = 3$. The CSI is a collection of $N_{Tx} \times N_{Rx}$ matrices $H(s)$ in which each describes the RF path between all pairs of N_{Tx} transmit and N_{Rx} receive antennas. The Network Interface Card (NIC) conveys CSI in accordance with the format outlined in reference [115]. A 802.11n NIC firmware probes a receiver to gather CSI and to learn CSI from a received packet [114]. Thus, the Multiple Input Multiple Output (MIMO) system subject to OFDM modulation, contains 30 subcarriers in each channel, can capture 6×30 subcarriers.

Table 3.1 Collected dataset characteristics [12].

Activities	Run, walk, fall, stand, pick up, sit
Total number of activities	3400 different activities are performed by 25 volunteers individually
WiFi devices	Desktop computer equipped with Intel 5300 WiFi card and IEEE 802.11n access point
Bandwidth	20 MHz
Frequency	5 GHz
Number of antennas	$2Tx \times 3Rx$
Number of rooms used	3

Table 3.2 HAR actual number of activities.

Fall	Walk	Run	Stand
443	1465	1209	304

In the dataset, 3400 different activity samples are performed by 25 volunteers of varying ages and genders. Moreover, different directions of movement with different speeds in different environments are considered in [12] to obtain a robust activity training model and to prevent overfitting.

Table 3.1 summarizes collected dataset characteristics [12]. As this table shows, the activities performed are : running, walking, falling, standing, picking up and sitting. In total, 3 different rooms were used. Each CSI packet is labeled with the activity of the person. In this chapter, among different activities, we consider four activities falling, walking, running, and standing. Table 3.2 represents the actual number of different activities in this dataset.

3.5.1 Sources of Uncertainty

In a real system, when collecting data from WiFi devices, various sources of uncertainty come into play. These include factors like humidity, echoes, environmental features and parameters, multi-path reflections, reflections from stationary and moving objects, the presence of other individuals, and interactions between two or more people.

All of these factors introduce variability and stochastic (random) characteristics into the dataset in a non-deterministic environment. In this context, non-deterministic means that the outcomes or measurements cannot be precisely predicted or determined due to the complex interplay of these uncertain factors.

Indeed, this non-determinism can present significant challenges when the goal is to effectively analyze and model the data, as outlined in Chapter 6. In environments where data can exhibit stochastic behavior due to various uncertain factors, it becomes crucial to employ robust statistical and machine-learning techniques to extract meaningful patterns and insights from the data. This motivated us to explore in Chapter 6 specific methods and approaches to perform robust data analysis and modeling in the presence of non-deterministic factors.

3.6 Data Preprocessing

The raw CSI stream of data in the WiAR dataset cannot be used for processing directly. This is because each CSI stream contains lots of high frequency noise which is not related to

human activity.

The source of CSI noise comes from the internal state transitions of the transceiver such as rate adaptation and transmission power change. Moreover, the raw collected data contains many redundant CSI values from multiple antenna pairs and different channels which cause overhead to the learning modules and decrease the performance of activity recognition. To address these issues, we do 1) Direct Current (DC) component removal and 2) smoothing [9]. To remove the DC component, we suppress strong signals from the LoS path. In this way, we subtract the constant offsets from CSI streams which is calculated through a long-term averaging (4 seconds) over the subcarrier. In fact, the energy in the DC frequency corresponds to the paths from the transmitter to the receiver and it is not involved in human activity. Then, we use a moving average filter to smooth CSI streams and to remove the noise affecting the amplitude. We use the Weighted Moving Average (WMA) technique as expressed in Equation (3.8) to process the amplitude values of each subcarrier and to remove noise [9].

$$A'_t = \frac{1}{m + ... + 1}[mA_t + (m-1)A_{t-1} + ... + A_{t-m+1}], \tag{3.8}$$

where $[A_t, A_{t-1}, ..., A_{t-m+1}]$ denotes the amplitude value sequence of k^{th} subcarrier at time t ; A'_t is the weighted averaged value ; m is the weighted relationship between the current value and the historical values. We set m to the length of the packet for each subcarrier.

Additionally, we use Principal Component Analysis (PCA) for **1) noise removal**. High-bandwidth burst and impulse noises in the CSI data make it challenging to obtain a smooth CSI stream when applying a low-pass filter. However, when using the PCA algorithm for denoising, we can achieve better results [118]. Moreover, it reduces the complexity by reducing the dimension. PCA allows concentrating on the dimensions of the signal which carry the most important information.

The important information in the data is captured in its principal components [12]. However, CSI also includes a great deal of noise. It was notably found in [12] that the h_1 principal component observed in the presence of human activities is mostly noise. As each PCA component contains all CSI signal information, excluding the first principal component does not result in the loss of important information concerning HAR in incoming CSI data. Note also that there is a trade-off between computational complexity and classification performance when selecting the number of PCA components for feature extraction. However, it was conjectured that discarding the ones with lower eigenvalues can be done without sacrificing performance, which is what we did.

2) Dimensional reduction We address the challenges associated with combining CSI data from 30 OFDM subcarriers using PCA. Dimensional reduction is the process of reducing the dimension of large data sets, by transforming a large set of variables into a smaller one that still contains most of the information and minimizes the projection error. The first principal component contains the largest possible variance in the data set. The second principal component is uncorrelated and perpendicular to the first principal component and contains the next highest variance and so on.

Algorithm 1 PCA Algorithm

1: Normalize input data
2: Calculate the covariance matrix of input data
3: Calculate eigenvectors and eigenvalues
4: Sort the eigenvectors according to eigenvalues in decreasing order
5: Choose first k eigenvectors and that will be the new k dimensions

As a result, the PCA algorithm is a powerful tool for discovering the time-varying correlations between CSI streams [12]. Algorithm 1 summarizes PCA analysis. SVD is utilized to perform PCA. In this research, we obtain matrix H from CSI streams. Eigenvectors of correlation matrix $H^T \times H$ with dimension of $N \times N$ are calculated, where N is the number of CSI streams, $N = 180$. Following [12], we reconstruct movement signals by computing $h_i = H \times q_i$, where q_i and h_i are the i^{th} eigenvector and the i^{th} principal components, respectively. In fact, the eigenvectors of the correlation matrix are the directions of the axes where there is the most variance and we call them principal components. Eigenvalues are simply the coefficients attached to eigenvectors, which give the amount of variance carried in each principal component. By ranking eigenvectors in order of highest eigenvalues to lowest, we get the principal components in order of significance.

In this analysis, the first six principal components contain the highest eigenvalues in descending order. Figure 3.7 shows the PCA amplitude [dB] versus packet number for six PCA components of human activity labeled walking. The first principal component is ignored due to the noise [12]. The second principal component of PCA is considered for human activity detection due to having the highest eigenvalue after the first principal component. We discard the remaining PCA components because they have much lower amplitudes, often close to zero.

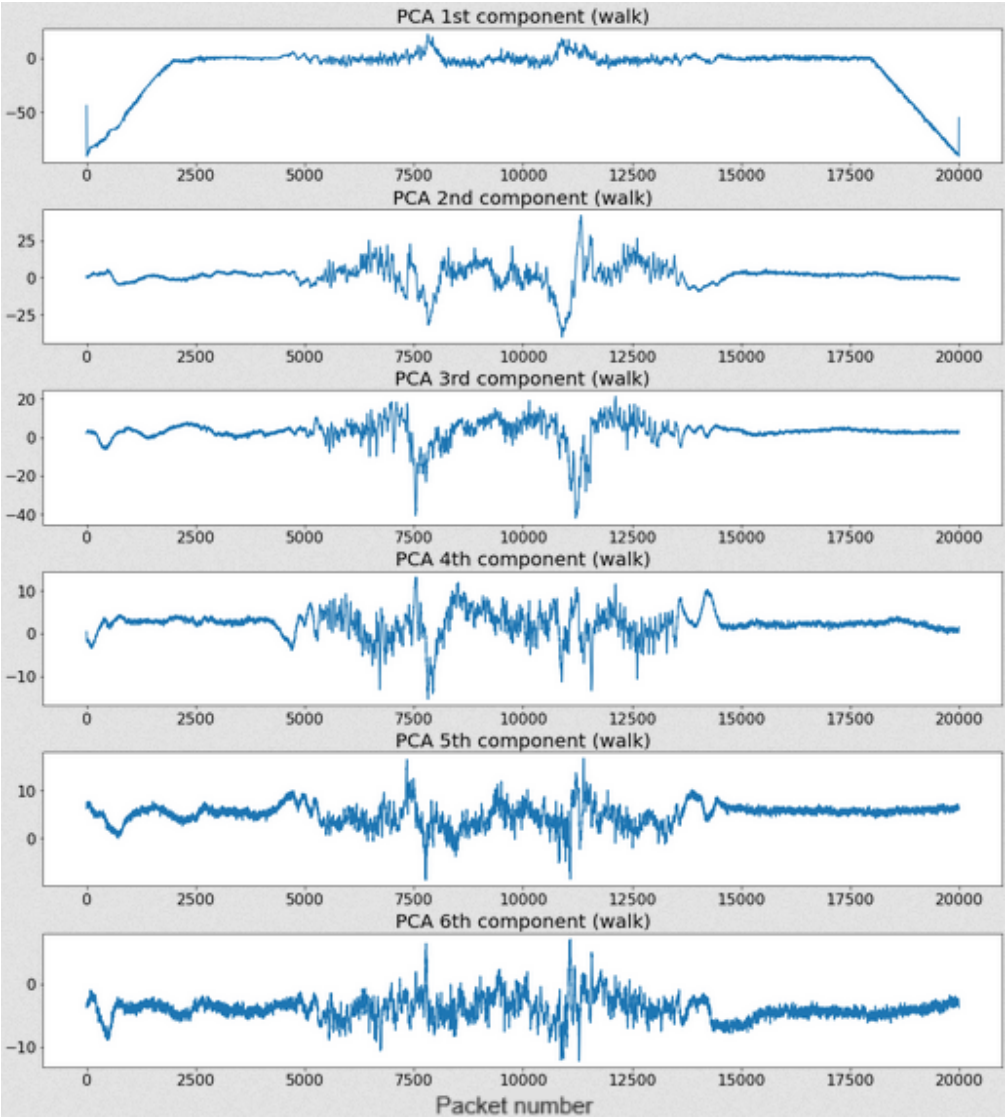


FIGURE 3.7 PCA amplitude [dB] versus packet number for six PCA components of human activity labeled walking.

3.7 Characterizing Human Activity

In this section, we characterize human activities based on PCA and spectrogram analysis. Before applying PCA to CSI streams, we do 1) Direct Current (DC) component removal 2) smoothing [9], and then 3) we use the PCA technique to get smooth CSI streams by removing burst and impulse noises. These steps are described in Section 3.6.

Figure 3.8 reports the value of the second PCA component for four known activities : falling, standing, walking, and running as a function of the packet number. The packet number is a time index that corresponds to the order in which the packets were received. Since the

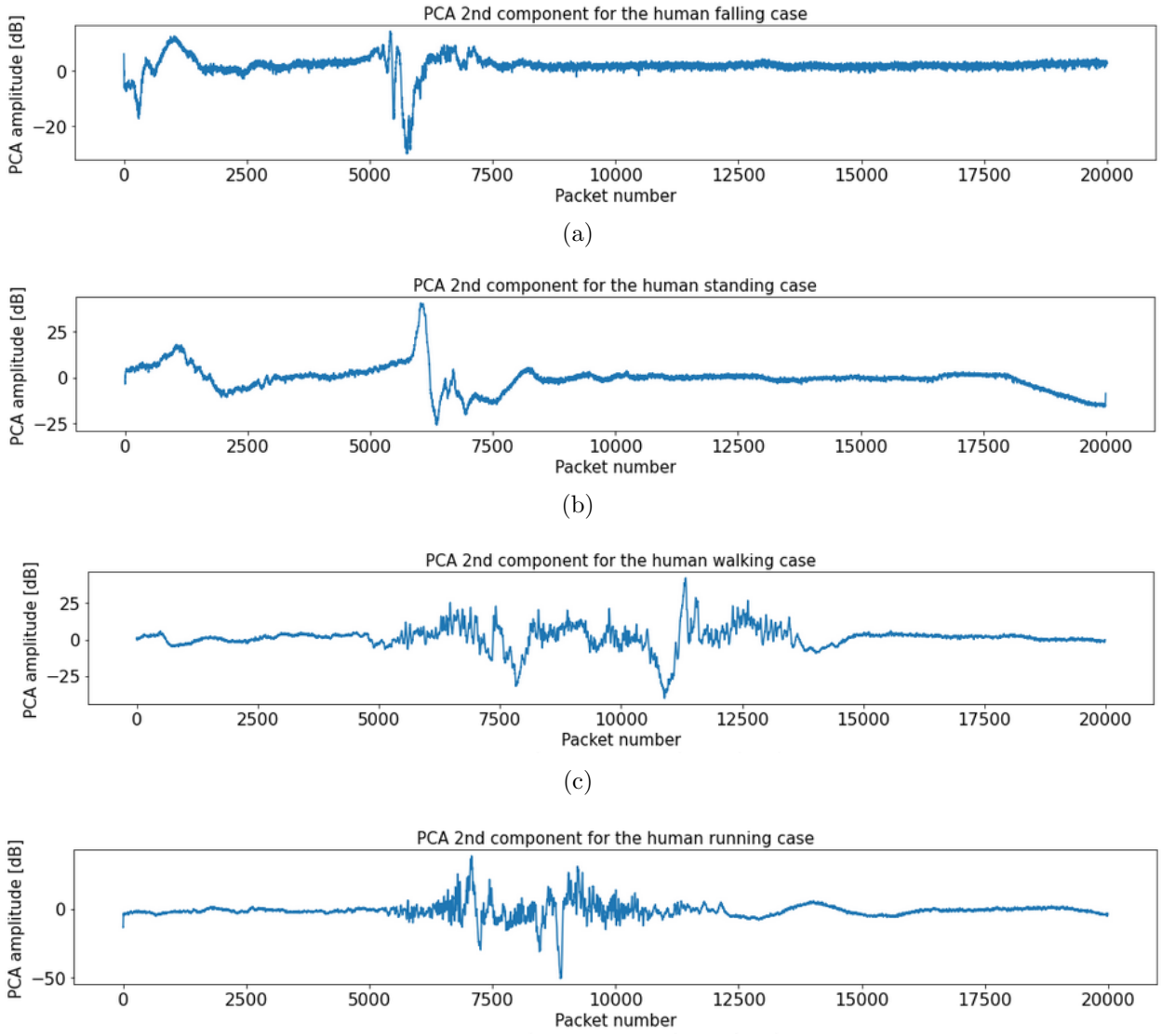


FIGURE 3.8 Value of the 2nd PCA component for four known human activities labeled a) falling b) standing c) walking and d) running.

first PCA component is mostly noise from the standpoint of HAR, we neglect it and consider the second one for feature extraction [12]. As Figure 3.8 illustrates, applying PCA on the CSI stream results in low-noise CSI data and the wave features can be extracted much more clearly. We distinguish the start and the end of activities by tracking the variation of signals (variance) for each activity. Based on Figure 3.8, the duration of activities for falling, standing, walking, and running are between packet numbers of $5000 - 6000$, $2000 - 6500$, $5000 - 14000$, and $5500 - 14000$, respectively. For the falling activity, there is a transient walking activity between packet numbers of $0 - 2000$. According to the reflection model represented in Equation (3.4), the CSI amplitude has a continuous fluctuation and a large variance during an activity. At the same time, it is steady or contains very low variance in the absence of activity. In the WiAR dataset, each activity occurs between no activity state. As we see in Figure 3.8, the speed of walking and running varies over time and they are not constant. Additionally, the walking and the running activities have similar behaviors. However, the CSI variation occurs in a short time for the running activity which indicates that we need spectrogram analysis. The duration of falling activity is short. For standing activity, we observe some attenuation of CSI signals in the duration of standing time. There is a short time where the detected activity switches from a walking state to a no activity state for standing activity, which happens between packet numbers of $6500 - 8000$.

The disadvantage of feature extraction using principal components is that it only considers the time and amplitude of the signal, while the frequency domain characteristics are important keys in activity recognition. For example, running and walking activities have similar waveforms. However, in the frequency domain, most of the CSI energy of the running subject appears at higher frequencies. After denoising the signal and creating the matrix of feature vectors, we use a short-time Fourier transform (STFT) to extract features in both the time and frequency domain using the spectrogram tool available in Python as explained in Section 3.7.1.

3.7.1 STFT and Spectrogram

Since different people typically perform the same activity at different speeds and different time durations, STFT is performed to enable the analysis of CSI signals in both time and frequency. The STFT of $x(t)$ is given by [119] :

$$X(\tau, \omega) = \int_{-\infty}^{\infty} x(t) w(t - \tau) e^{-j\omega t} dt \tag{3.9}$$

where $w(\tau)$ is the window function, commonly a Hann window or Gaussian window centered around zero [120] and τ specifies the center of the moving window [119]. In practice, the procedure for computing the STFT is to divide a longer time signal into shorter segments of equal length and then compute the Fourier transform separately on each shorter segment [120]. To visualize the results of the STFT and illustrate how the frequency components of a signal change over time, the spectrogram is utilized. In fact, the spectrogram projects the signal in terms of three dimensions : time, frequency, and power.

The Scipy.signal.spectrogram package is used to compute the spectrogram. The "NFFT" parameter determines the number of data points used in each block for the FFT. The "noverlap" parameter controls the overlap between segments. The parameters like NFFT and noverlap can be adjusted to further control the spectrogram's resolution and appearance based on the required specification. The window parameter is used to specify the windowing function applied to each segment of the signal before computing the STFT [121]. In this research, the default window function of the Hanning window is employed. The Hanning window is a bell-shaped window that reduces spectral leakage and provides good frequency resolution [121]. The "Fs" parameter represents the sampling rate, which is the number of points sampled per second. It is chosen to be 1000 $samples/second$ for HAR. As the frequency of human activity is up to 400 Hz, the rate of 1000 $samples/second$ satisfies the Nyquist criterion $f_s \geqslant 2f_c$, where f_s is the sampling frequency and f_c is the frequency of a specific human activity. Finally, the absolute values of the spectrogram are rescaled using $10log_{10}(|X_k|)$ in dB to have a better color code. Table 3.3 illustrates the parameters required to plot the spectrogram.

In the STFT, the signal is divided into overlapping or non-overlapping segments, commonly called frames. The frame size is a crucial parameter in the STFT. The description specifies that the frame size should be a power of two. This choice is common in signal processing due to computational efficiency and properties related to the Fast Fourier Transform (FFT), which are often used for computing the Fourier transform. Figure 3.9 illustrates time-frequency analysis for human activities labeled running with different frame sizes. The frame size is set using the NFFT parameter. It is set to 32, 64, 128, and 1024 for Figures 3.9(a), 3.9(b), 3.9(c) and 3.9(d), respectively. According to Figure 3.9, there is a trade-off between frequency and time resolution in the STFT method. By selecting a larger frame size (NFFT), we achieve higher resolution in the frequency domain and lower resolution in the time domain. The vertical axis of a spectrogram represents the discrete frequency steps, the horizontal axis represents time, and each pixel represents the amount of energy [dB] in each frequency component at a particular time frame.

TABLE 3.3 Spectrogram parameters.

matplotlib.pyplot.spectrogram arguments	
x	Sequence containing the data
NFFT	Number of data points used in each block for the FFT
Fs	Sampling frequency
Fc	Center frequency of x
noverlap	Number of points of overlap between blocks
cmap	Color map
scale	Scaling of values in the spec $\{default, linear, dB\}$
window	The windowed version of the segment, the default value is window-hanning()

As Figures 3.9(a) and 3.9(b) illustrate, by selecting a smaller frame size like $NFFT = 32$ and $NFFT = 64$, the frequency resolution decreases. Thus, when there are fewer frequency bins, the Fourier transform is calculated over shorter time segments. This results in an improved time resolution.

By contrast, according to Figures 3.9(d), when a higher frame size $NFFT = 1024$ is selected, the time resolution decreases. Thus, there are more frequency bins when the Fourier transforms are calculated over a longer time. Consequently, the frequency resolution improves. As Figures 3.9(c) shows, selecting the frame size to be $NFFT = 128$ results in the best trade-off between time and frequency resolution for the running activity.

Figure 3.10 illustrates the spectrogram for four human activities : falling, standing, walking, and running using the WiAR dataset. The color bar on the right of the spectrogram indicates the intensity [in dB] shown by varying the color or brightness. Intensity is defined as a power per unit area. This figure shows how the energy of each frequency component evolves with time. The high-energy components are colored in red.

Different activities could have similar duration but different frequencies and vice versa. For example, based on Figure 3.10, running and falling produce similar frequencies. However,

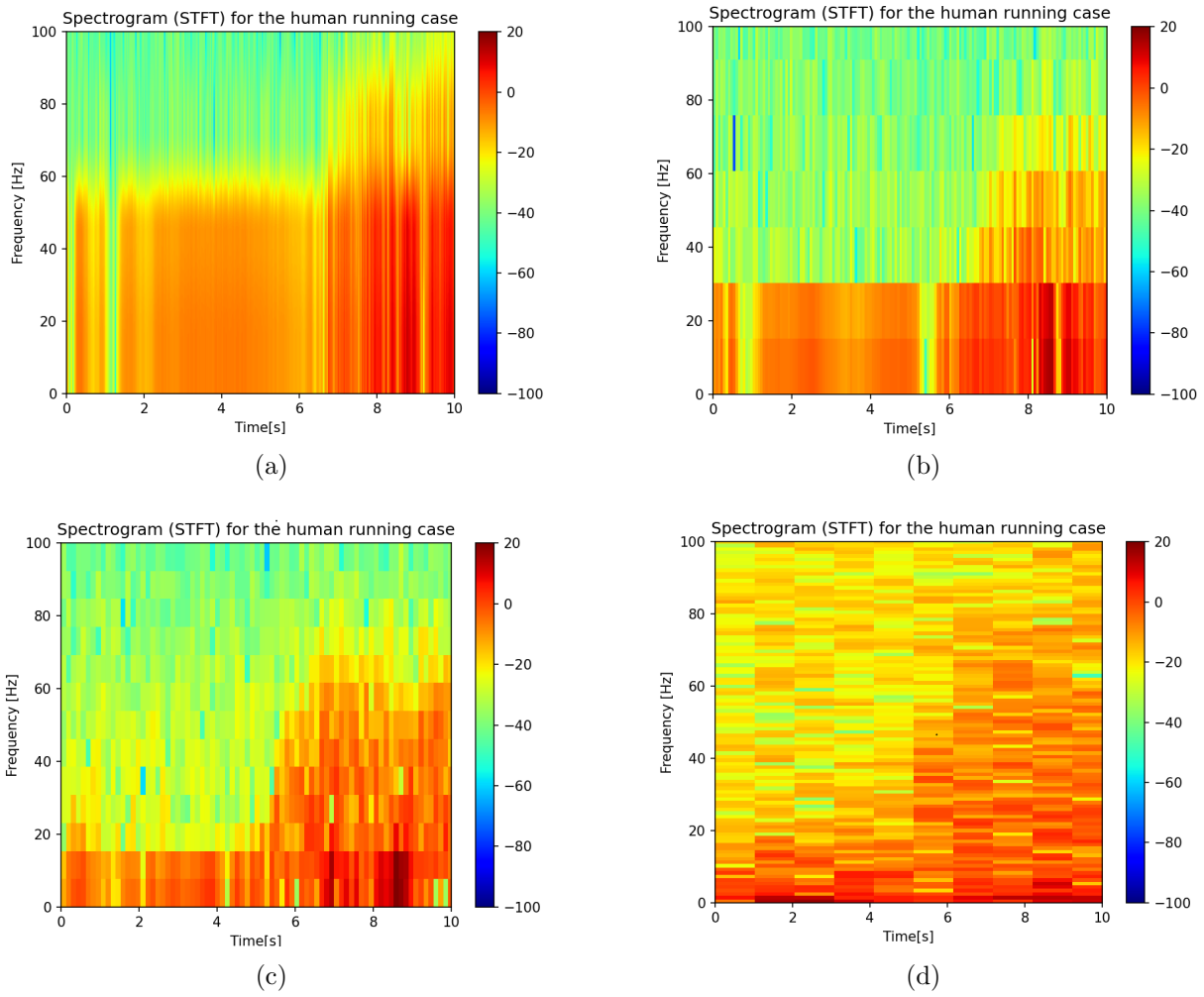
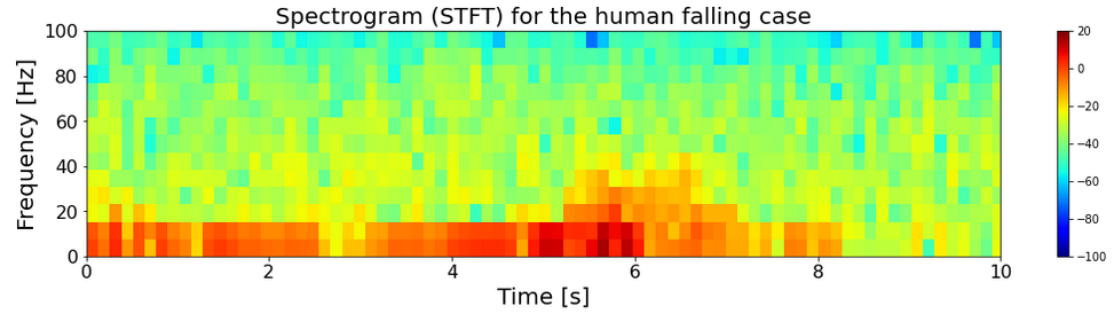
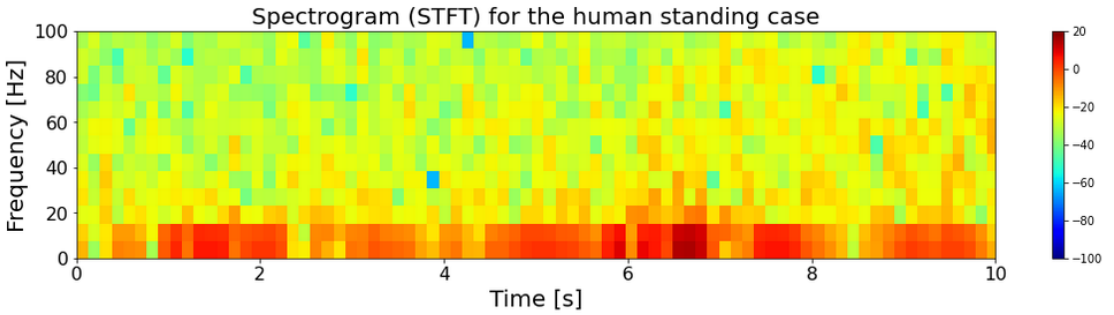


FIGURE 3.9 Time-frequency analysis for human activities labeled as running with frame sizes equal to : a) 32, b) 64, c)128, and d) 1024.

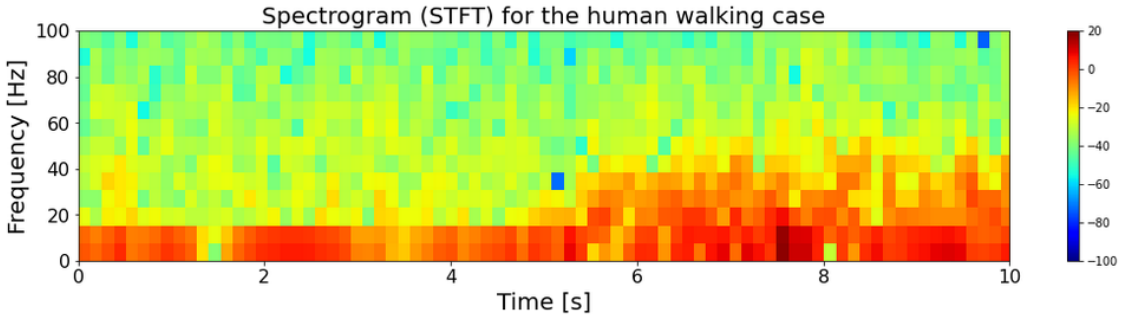
the duration of the falling activity is shorter than that of the running activity. According to Figure 3.10, the falling activity causes a sudden decrease of energy of less than 0.5 second. For standing, walking, and running activities, the high energy band is concentrated around $0-15\ Hz$, $19-40\ Hz$, and $20-65\ Hz$ frequency bands, respectively. In fact, in the running activity, most CSI energy is concentrated at higher frequencies. It is observed that for the falling activity, first the person moves slowly and most of the CSI energy is concentrated at low frequencies. Then there is a rapid transition to higher frequency components. After having some movements such as rolling, there is a fast transition to zero energy and a silent state. For the standing activity, we observe a steady state condition. The results also show the attenuation of CSI signals for slow walking after standing for a short time.



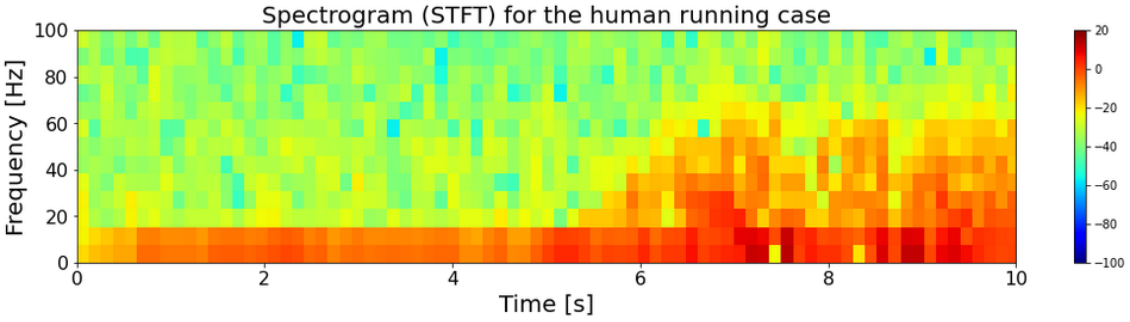
(a)



(b)



(c)



(d)

FIGURE 3.10 Time-frequency analysis for four known human activities labeled a) falling b) standing c) walking and d) running.

3.8 Time Series Classification Based on LSTM-RNN Model

Based on the literature, the hidden Markov model (HMM) [12] and k-nearest neighbors (KNN) [122] are used for WiFi-based activity recognition. However, we need both time and frequency dimensions to distinguish different kinds of activities. Human activities often exhibit temporal dependencies, where the current activity is influenced by previous activities. The KNN model [122] ignores the time dimension. It treats each feature independently and does not take into account the temporal order of data points. By contrast, the HMM model is based on a finite-state machine and relies on 200 ms time interval. Additionally, it requires a state transition matrix estimation, which is a bottleneck when the size of the data grows. Moreover, different activities require different time durations. For example, falling lasts less than one second, while walking takes more time. Also, different time durations are observed for the same activity. In the KNN model, a fixed length of data is required, and in the HMM model it is not sufficiently flexible in this aspect.

To address those challenges, a Long Short-Term Memory (LSTM) Recurrent Neural Network (RNN) model is used for human activity recognition time series classification. The LSTM network model is a type of artificial recurrent neural network (RNN) able to learn and remember time series of input data, even with long-term dependencies. It is also of interest that LSTM does not require initial state estimation. It contains feedback connections, known as recurrent connections to address the vanishing gradient problem and enable the modeling of long-term dependencies in sequential data found in activity recognition. This model learns to extract features from sequences of observations and maps them to different activity types. The memory of the LSTM can detect similar activities such as sitting down and falling. The number of hidden layers in the LSTM-RNN model is chosen to be 200. The number of hidden layers in the LSTM-RNN model is roughly the same as the size of the output of the LSTM-RNN, $N_{Tx} \times N_{Rx} \times 30$ OFDM subcarriers [123]. We select the batch size equal to 200 as well. To validate and optimize the model, the cross entropy loss function and the efficient Adam version of the stochastic gradient descent with a learning rate of $1e^{-4}$ are applied to compute the loss and to optimize the *dynamic LSTM-RNN* network, respectively.

From the total 3400 available samples, 80% were used for training and 20% for testing. The five-fold cross-validation method is applied on top of the optimization method to validate the model. The five-fold cross-validation method divides the dataset into five subsets of equal

size. The four subsets are used for training the classifier and one subset is for the testing classifier. This process is iterated five times and each time one subset is randomly chosen for testing. In fact, with this method, all samples of the dataset are utilized for both training and testing. The other advantage of using the five-fold cross-validation method is that since LSTM-RNN is a stochastic process, we repeat the evaluation of the model multiple times. Then, we calculate the mean and standard deviation of the performance.

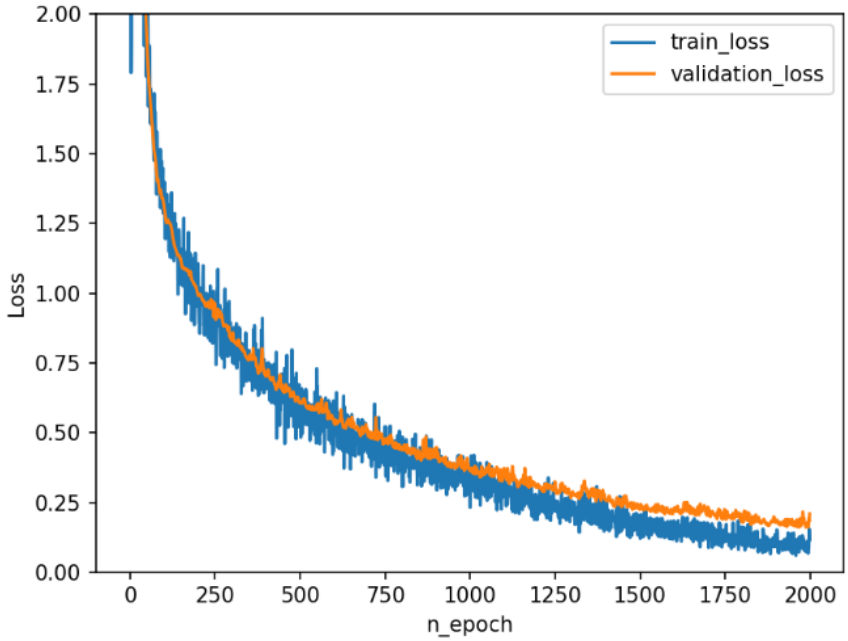
Figure 3.11 illustrates the accuracy and loss function of HAR using the WiAR dataset for each epoch. In this figure, the results are depicted for both the training and validation datasets. Based on the experimental results, the mean and standard deviation of accuracy are 94% and ±1.4%, respectively.

Using the WiAR dataset, the HAR confusion matrix is reported in Table 3.4 for the four considered activities labeled falling, walking, running, and standing. The predicted columns are the RNN-LSTM classifier output. The confusion matrix is a technique to summarize the performance of a classification algorithm. It shows how each class is predicted and reports the rate of the samples that are correctly or incorrectly classified. Based on Table 3.4, 12 samples were misclassified for falling activity. While for walking, running, and standing activities 18, 12, and 19 samples were classified wrongly, respectively. Additionally based on this matrix, 3, 23, 28, and 7 samples are misclassified as falling, walking, running, and standing, respectively.

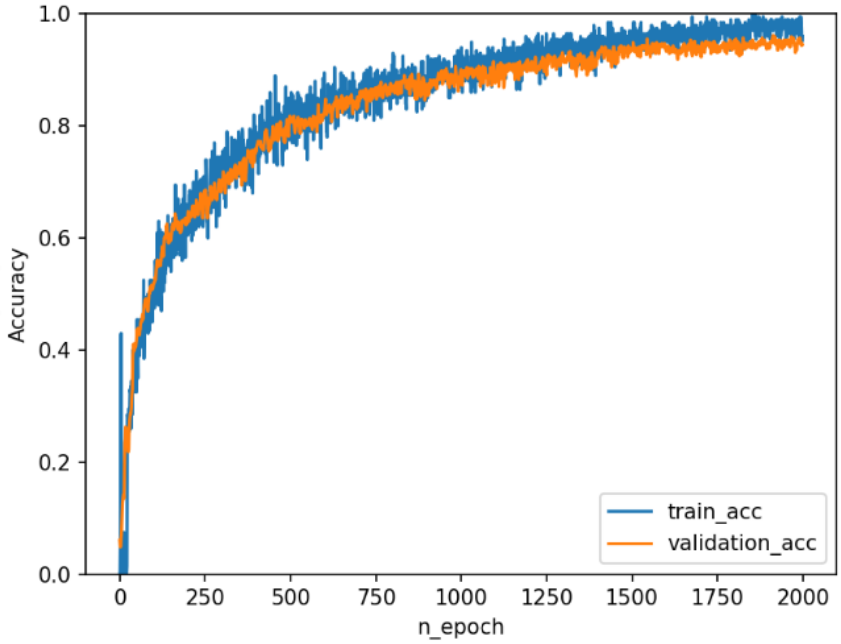
Table 3.4 HAR Confusion Matrix.

		Predicted			
		Fall	Walk	Run	Stand
Actual	Fall	428	3	8	1
	Walk	1	1442	13	4
	Run	1	9	1188	2
	Stand	1	11	7	281

The dataset from [12] allows detecting specific human activities when a single person is active. Figure 3.12 reports the number of people for different activities. The blue and red colors represent the actual and predicted activity, respectively. The RNN-LSTM classifier predicts human activities. The number of occurrences of each activity is counted from the LSTM-RNN prediction output as a predicted activity, while the actual count is the number of existing samples before training and classification. Based on this analysis, the respective error rates for the falling, running, standing, and walking activities are 3.3%, 1.7%, 7.5%, and 1.5%, respectively.



(a)



(b)

FIGURE 3.11 a) HAR training and validation loss b) training and validation accuracy using Adam optimizer.

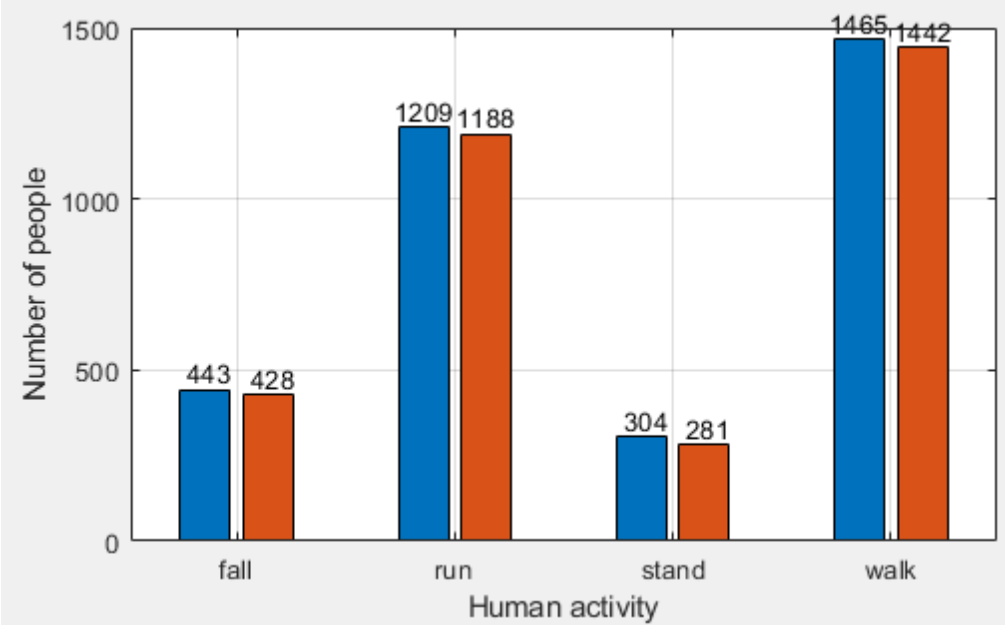


FIGURE 3.12 Number of people versus different activities. The blue and red colors represent the actual and predicted activity, respectively.

3.9 Method to Verify Walking and Running Speed with CSI Changes

Considering the round-trip path-length change, the CSI frequency f is determined by the Doppler frequency formula Equation (3.10) [124].

$$f = 2\nu cos(\theta)/\lambda \tag{3.10}$$

$$\nu = \frac{1}{2} f \lambda sec(\theta) \tag{3.11}$$

where ν is the speed of human activity, λ is the WiFi signal wavelength, and θ is the angle defined by the direction of the moving object and the transmitter line of sight to the moving object. Without defining the angle, it is difficult to determine whether a person is moving towards or away from the link.

Based on Equation (3.10), the maximum and minimum frequency of Doppler shift f is $\pm 2\nu/\lambda$. From this equation, we can calculate the CSI speed of human activity. We exclude $\nu = 0$ for walking and running activities. The wavelength λ for the frequency of 5 GHz WiFi signal is between 5.15 and 5.79 cm. We consider $\lambda = 5.15$ cm and read frequency f from the spectrogram of Figure 3.10 described in Section 3.7.1. Using Equation (3.11), we can verify the speed of running and walking activities.

Based on Figure 3.10, for the walking activity in the 19 to 40 Hz frequency range, we obtain the speed of the walking activity to be between 0.49 m/s and 1.03 m/s. While for the running activity analyzed in the 20 Hz and 65 Hz frequency range, we obtain the speed of running activity to be between 0.51 m/s and 1.67 m/s. A constant low frequency is related to the standing activity. Since human walking speed is no more than 5 m/s, the maximum and minimum of Doppler frequency shift (DFS) for $\lambda = 5.15$ cm is about ± 200 Hz. Moreover, $2\pi f_D \Delta_t$ should be in the range of 2π. Thus, $\Delta t_{max} = \frac{1}{2f_{Dmax}} = \frac{1}{400}$ which corresponds to the minimum packet rate of 400 Hz. To estimate the HAR direction of movement, we need to measure θ by calculating the angle of arrival (AoA) and time of flight (ToF) which is not in the scope of this chapter.

3.10 Human Activity Recognition ; Comparison with Andrii's Dataset [125]

Andrii et al. [125] also provided a CSI-based dataset for human activity recognition. Table 3.5 summarizes the collected dataset characteristics [125]. As this table shows, the activities performed are : walking, sitting, laying down, standing, getting up and getting down. In total, 3 different rooms were used. Each CSI packet is labeled with an activity of the person.

Table 3.5 Collected dataset characteristics [125].

Activities	walk, sit, stand, lay down, stand, get up/down
WiFi router	TP-Link TL-WDR4300
Bandwidth	20 MHz
Frequency	5 GHz
Number of antennas	$2Rx \times 2Tx$
Number of rooms used	3

Table 3.6 summarizes the collected number of samples for each activity. Figure 3.13 illustrates the received CSI amplitude versus packet numbers for the walking activity. Based on this figure, the time interval when the walking activity occurs is specified as a pair of packet numbers such as $4300 - 7200$.

Table 3.7 summarizes the confusion matrix for detecting the six different considered human activities with our method. In that case, 80% of the data is considered as a training set and 20% as a test set. The average accuracy for the training set and the test set is 91.3% and 51.4%, respectively. The main reason for the low accuracy of the test set is that

TABLE 3.6 The collected number of samples for each activity.

Activity	Stand	Sit	Get-down	Get-up	Walk	Lay-down
Number of samples	57	134	125	49	714	163

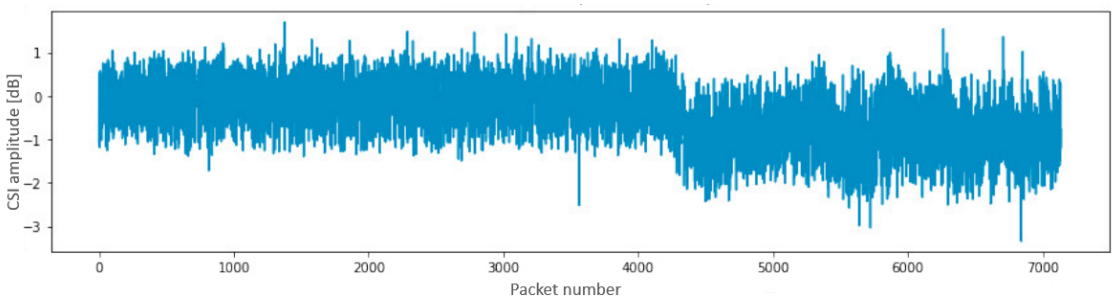


FIGURE 3.13 Received CSI amplitude versus packet numbers for the walking activity.

TABLE 3.7 LSTM-based model confusion matrix for human activity detection for the dataset from [125].

		Predicted					
		Stand	Sit	Get-down	Walk	Get-up	Lay-down
Actual	Stand	0.43	0.02	0.00	0.29	0.23	0.03
	Sit	0.01	0.50	0.23	0.00	0.00	0.26
	Get-down	0.03	0.03	0.51	0.00	0.00	0.43
	Walk	0.11	0.02	0.00	0.78	0.09	0.00
	Get-up	0.11	0.02	0.04	0.43	0.40	0.00
	Lay-down	0.04	0.21	0.21	0.00	0.04	0.50

this dataset is highly imbalanced and there is insufficient amount of data. The model favors the majority class and performs poorly on the minority class.

TABLE 3.8 Balanced LSTM-based model confusion matrix for human activity detection for the dataset from [125].

		Predicted					
		Stand	Sit	Get-down	Walk	Get-up	Lay-down
Actual	Stand	0.86	0.00	0.00	0.01	0.13	0.00
	Sit	0.00	0.96	0.02	0.00	0.00	0.02
	Get-down	0.00	0.04	0.86	0.00	0.00	0.11
	Walk	0.04	0.00	0.00	0.96	0.00	0.00
	Get-up	0.03	0.00	0.00	0.01	0.96	0.00
	Lay-down	0.00	0.07	0.04	0.00	0.00	0.88

Table 3.8 reports the balanced LSTM-based model confusion matrix for human activity detection from the dataset reported in [125]. The random oversampling technique is applied to handle the imbalanced nature of the dataset and to make it better balanced. It improved the average accuracy from 51.4% to 90.4%. However, the disadvantage of the over-sampling method is that it may make the decision regions of the learner smaller and more specific, with the risk of overfitting. In this thesis, we address the class imbalance in Chapter 5.

3.11 Performance Comparison of Human Activity Recognition Using Deep Learning Methods

In this section, a comparative analysis of the performance with different deep-learning-based algorithms is discussed for human activity detection using the WiAR dataset. Long Short Term Memory (LSTM) networks, a class of recurrent neural networks (RNN) are well-suited for time-series data analysis [126]. They can capture long-term dependencies in the data, while remembering and utilizing the context over time, thus making them suitable for tasks involving sequences and temporal patterns. The specifics and parameters of the LSTM classification model are described in Section 3.8. KNN models consider local similarity but struggle with CSI's high dimensionality and their sequential nature. By contrast, LSTM models demonstrate superior performance in sequential HAR utilizing CSI data. LSTM effectively learns and remembers long-term dependencies in the CSI data, leading to higher accuracy in activity recognition.

In KNN, the model is trained considering k nearest neighbors of the input data in the training set. The selection of the parameter k as the number of neighbors is a very important issue that can affect the decision made by the KNN classifier. A predefined rule for the selection of the value of k does not exist. In this study, the number of nearest neighbors k is determined experimentally as Figure 3.14. This figure illustrates the accuracy versus the number of neighbors for the KNN model. According to this figure, the KNN model achieves the maximum accuracy 81% considering $k = 8$.

The Multilayer Perceptron (MLP) model is commonly used for classification tasks [127]. However, when applied to sequential activities with CSI, it exhibits limitations in capturing temporal dependencies. Table 3.9 summarizes the performance comparison for different deep-learning-based algorithms of LSTM, KNN and MLP models. The models are trained using the training partition (80% of the data) and evaluated using the test set (20% of the data) of the WiAR dataset. The MLP model is trained with consideration for two hidden layers.

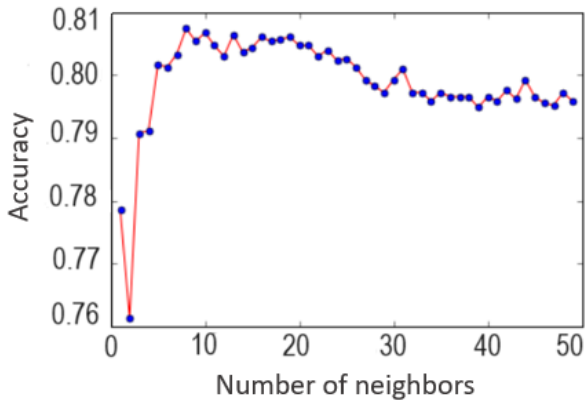


FIGURE 3.14 Accuracy versus number of neighbors for KNN model.

Based on the results in this table, the LSTM model performs better than KNN and MLP models to analyze time-series data.

TABLE 3.9 Performance comparison with different machine learning algorithms.

Algorithm	Test set accuracy %
LSTM	94
KNN	81
MLP	73

3.12 Performance Comparison of Human Activity Recognition Obtained by Other Authors

Wei Wang et al. [12] employed the HMM model to detect human activity using the CSI-based WiAR dataset. For a new environment and a new person that the system has never been trained on, the HMM model achieved average recognition accuracy of more than 80%. The authors in [128] mentioned that the accuracy of the activity recognition methods using the HMM model cannot exceed a certain level due to the uncertainty of human behavior.

The drawback of the HMM technique is that it is unable to capture complex movements with a sequence of micro-activities. HMMs assume that the future state of the system depends only on its current state and is independent of its previous states. This assumption may not hold for complex human activities, where the current action may depend on a history of

prior actions and environmental conditions. Additionally, in traditional HMMs, the transition probabilities between states are fixed and predefined. Complex human activities often involve non-linear, dynamic transitions that are challenging to model with fixed-state transitions. Linlin Guo et al. [11] used the WiAR dataset to detect human activities. They achieved an average accuracy of 83% by applying Naive Bayes (NB) machine learning algorithm and 90% using the CNN deep learning algorithm in different indoor environments. The NB algorithm is based on Bayes' theorem and is called "naive" because it makes a simplifying assumption that all features used for classification are conditionally independent, even though this may not be entirely accurate in practice. Thus, our method using the LSTM algorithm achieved the highest average accuracy of 94% ±1.4% for human activity detection.

3.13 Summary

This chapter proposed a HAR method to detect human activities. A dataset based on $IEEE$ 802.11n CSI measurements is used. This data contains fine-grained movement information. Since wireless signals can traverse walls and do not require a line-of-sight condition, they can be leveraged to enable HAR. A data processing framework inspired by the MAPE-K loop was proposed to analyze dynamic changes.

Although PCA is a powerful method to reduce dimension and remove noise, it contains time domain information. Spectrogram analysis was considered to obtain time-frequency domain features and to calculate the frequency of movements. Doppler frequency shifts were then analyzed to validate the speed of movement. In comparison with different deep learning models, the time series LSTM-RNN model is considered a better choice as this classifier, achieving a 94% ±1.4% accuracy of detection.

CHAPTER 4

DETECTING HUMAN ACTIVITY WITH HIGHER MOVEMENT ENERGY

In this chapter, a framework and a method are proposed to detect human activity with higher movement energy based on maximum energy changes in Sections 4.1 and 4.2, respectively. Additionally, the impact of feature selection is investigated on machine learning classification performance in Sections 4.3 and 4.4. Comparison with another dataset to detect human activity with higher movement energy is taken into account in Section 4.5. Finally, a summary and conclusions drawn from that part of our thesis work are presented in Section 4.6.

4.1 A Framework to Automatically Detect Human Activity with Higher Movement Energy

In this section, we propose the architecture of a system that can detect human activity automatically with higher movement energy as Figure 4.1. The dataset and data preprocessing were described in Chapter 3, Sections 3.5, and 3.6.

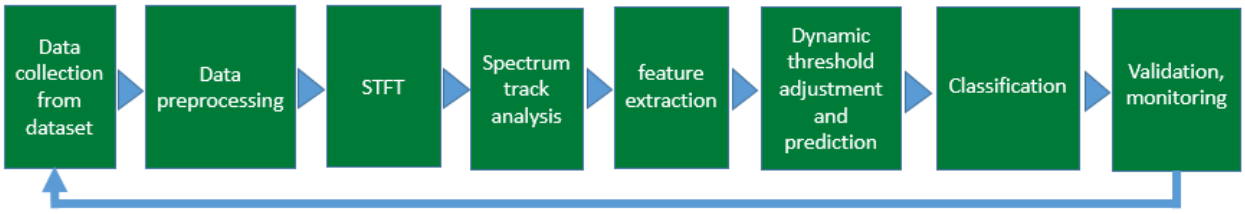


FIGURE 4.1 A data processing framework to automatically detect human activity with higher movement energy.

Figure 4.2 illustrates the spectrogram that reflects a human standing activity computed using the WiAR dataset introduced in Chapter 3, Section 3.7.1. That specific signal represents a standing activity in which the subject is walking at the end. We track the movement of a person in both the time and frequency domains in the ”spectrum track analysis” component.

Then, the PSD and an average of the band power are estimated in the feature extraction

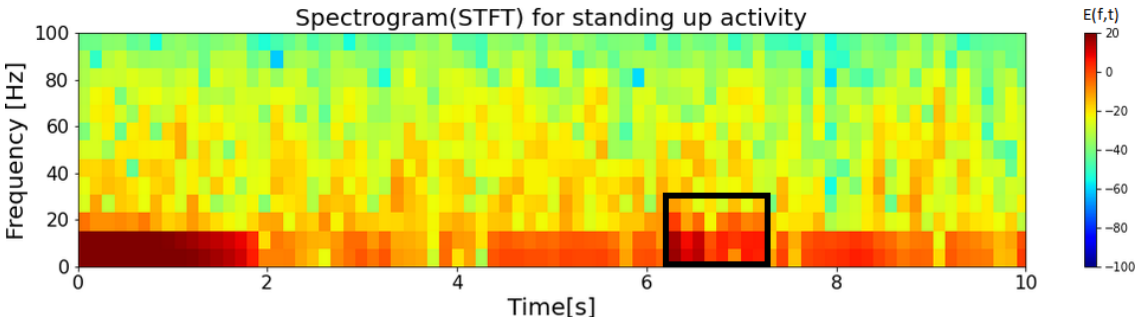


FIGURE 4.2 Time-frequency analysis for human standing activity [17].

component. Moreover, we consider the frequency of movement as another dimension of the signal features that can be used to improve the performance. The adaptive threshold is obtained by the decision boundary of the SVM classifier. Performance evaluation is performed in the validation module. In the monitoring component, spectrum features are tracked to detect people with higher movement energy.

4.2 Method to Detect Human Activity with Higher Movement Energy

Figure 4.3 illustrates the method proposed to detect people with higher movement energy. Based on the diagram, the first step is to average the spectrum in each rectangular sliding window in order to obtain the PSD. The PSD is the accumulated distribution of the spectrogram over time. The normalized PSD is calculated by dividing the PSD at a specific frequency by its sum over the whole spectrum, $\hat{P} = P/sum(P)$, for each sliding window.

Deciding what to choose as a proper feature is one of the challenges with the proposed method. Figure 4.4 illustrates the PSD of the time-domain signal for a standing activity corresponding to the spectrogram of Figure 4.2. As Figure 4.2 reflects, there is a short time during which the subject is walking in the period of rectangular shape. Figure 4.4 shows that the maximum area under the curve of the PSD corresponds to the area of the rectangular shape highlighted in Figure 4.2. The area under the PSD curve is the average power of that signal expressed by Equation (4.1) [129].

$$P = \int_{-\infty}^{\infty} S(f)df \tag{4.1}$$

where $S(f)$ is the PSD $|X(\tau, w)|^2$ obtained from Equation (3.9), with $\omega = 2\pi f$, where f is

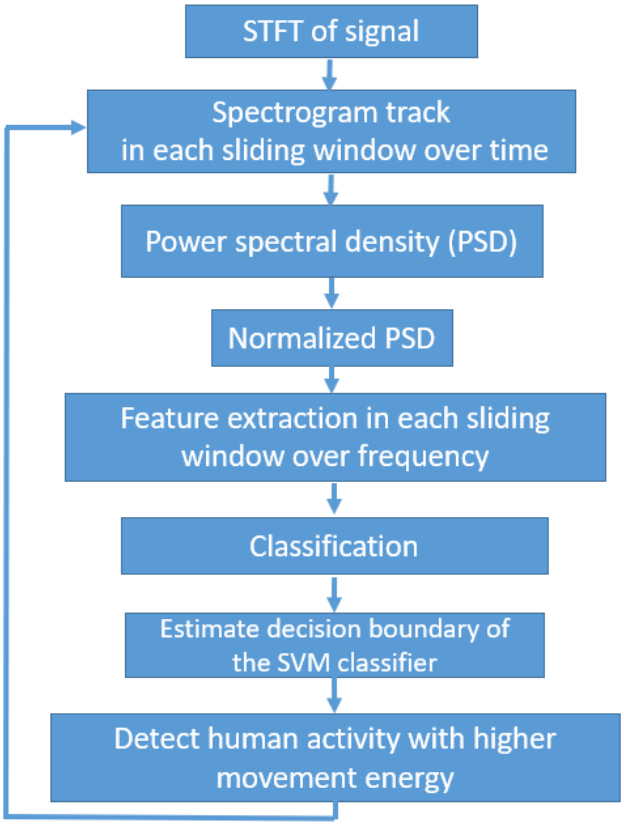


FIGURE 4.3 Flow graph of a method proposed to detect people with higher movement energy.

the frequency of movement. The band power is the average power of the signal in a specific frequency range [f1, f2] as expressed by Equation (4.2) [130].

$$band\ power = \int_{f1}^{f2} S(f)df \tag{4.2}$$

To compute the average band power, we need to estimate the PSD of the signal. Since typical human activity varies with time, we define a sliding window as illustrated in Figure 4.5 to capture near-stationary data and be able to estimate significant features. Hence, we define the frequency resolution as :

$$F_{res} = \frac{F_s}{N} = \frac{F_s}{F_s T} = \frac{1}{T} \tag{4.3}$$

where F_s is the sampling frequency of the signal, N is the total number of samples, and T is the duration in seconds. For example, if we select the full length of our data window to be 2.048 seconds, the frequency resolution is 0.488 Hz.

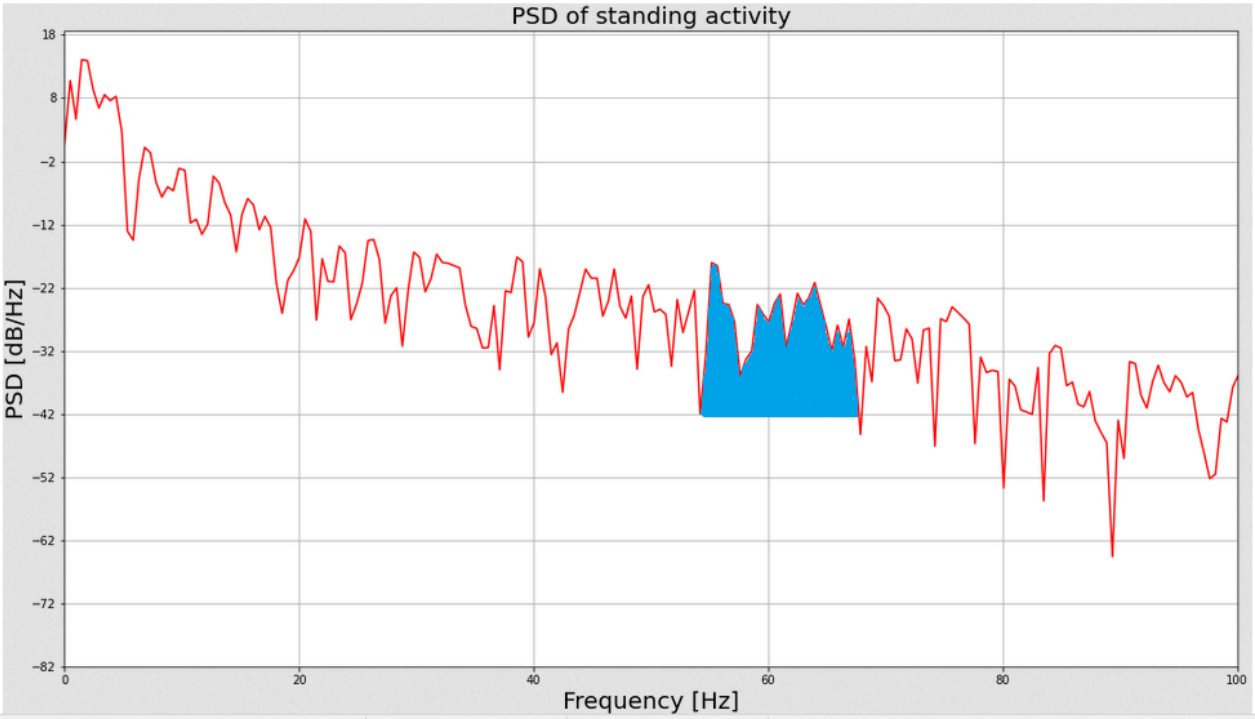


FIGURE 4.4 PSD of a signal for a standing activity illustrating the concept of band power.

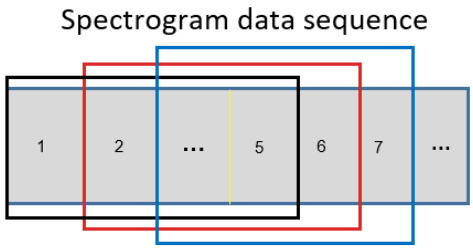


FIGURE 4.5 Sliding window applied on a spectrogram graph.

In the proposed method, we select the window size such that it contains at least two full cycles of the lowest frequency of the target human activity, which is 16 or 20 Hz based on Figure 4.2. So we choose a window size of $2/16 = 0.125$ or $2/20 = 0.1$ seconds. Based on Equation (4.3), changes in the sampling frequency do not affect the frequency resolution. By contrast, reducing the duration of the time window increases the frequency resolution. Since we use sliding windows, the delta band power is defined by Equation (4.4).

$$delta\ band\ power = \int_{f1+\Delta f}^{f2+\Delta f} S(f)df - \int_{f1}^{f2} S(f)df \tag{4.4}$$

For example, the absolute delta band power is the integral of the PSD in the blue area of Figure 4.4. Since there is no closed-form formula to integrate the blue area, we approximate it using Simpson's rule [131]. The relative delta band power is defined as the ratio of the power in a specified frequency band over the total power of the signal. After computing the absolute delta power in each sliding window, the differences in the delta power between sliding windows are calculated to obtain the maximum variation of $S(f)$.

Since WiFi-based identification of human activity is highly variable and different people perform a similar activity at different speeds and time periods, we must define an adaptive threshold α to detect them effectively. To determine that threshold, two groups of datasets are created, the first including people who are walking and the second one comprising people who are standing and people who are running. Two groups are combined and these groups are then used for training, validation, and testing of a classification model. The PSD and the area under the curve of the PSD are estimated for all instances.

Since we have two activity groups in the dataset, processing this dataset becomes a classification problem. The band power and the corresponding frequency band are considered as the inputs of the classifier. According to Equation (4.5), if the relative band power becomes greater than a threshold α, the corresponding data elements are classified as belonging to the running activity. Δf is the difference of two frequencies that defines the frequency band of interest. Δf_{max} corresponds to the frequency band for which the relative band power is maximum.

$$\Delta f_{max} = max\{\Delta f | \frac{bandPower}{totalPower} \geq \alpha\} \tag{4.5}$$

To perform that classification in two groups, we select a SVM classifier. A SVM is a robust classifier that classifies using a hyperplane defined by a few support vectors [132]. Since the SVM algorithm estimates two group classes using a threshold with a minimum error, we predict an adaptive threshold to differentiate people who are classified as running using this algorithm.

4.3 Feature Selection

In this section, we explore how to select features or variables from the dataset that are most suitable to drive the classification process. The goal is to find the best possible set of features to perform an analysis with machine learning techniques to improve accuracy. Figure 4.6 illustrates the decision boundary of the SVM classifier for one feature, the delta

band power.

In this figure, the x and y axes correspond to the delta band power feature and the labels, respectively. The red dots, corresponding to label 0, represent the delta band power for the walking and standing activities. While the red dots, corresponding to label 1, are related to the delta band power for the running activity. The blue line shows the best decision boundary of the SVM classifier for the delta band power feature. It shows how the red dots corresponding to the labels 0 and 1 are classified.

Since STFT data for each activity comprise transient movements and common data at the beginning, for delta band power values below 0, the classifier does not distinguish well the common values that are related to class 0 or class 1. To overcome this problem we added another feature at the input of the classifier to increase its performance. Figure 4.7 illustrates the decision boundary of the SVM classifier for two features : 1) delta band power and 2) frequency of movement. The threshold α is the boundary between the red and the blue regions in the range of $[0, 1]$. The red circles and the red area are related to the running activity. While the blue circles and the blue area are considered as walking and standing activities. Based on those figures, selecting two features achieves a much better separation of data compared to considering only one feature for both classes of 0 and 1.

4.4 Classification Results

Tables 4.1 and 4.2 report the SVM classifier confusion matrix when considering only the delta band power feature and two features, the delta band power and the frequency of movement, respectively. In that experiment, 80% of the data is used as a training set and 20% is used as a test set.

We considered the frequency of movement as a second dimension to the feature to improve the performance of the classifier. $Norm = True$ corresponds to the normalized confusion matrix in the range of $[0, 1]$. A confusion matrix, also called an error matrix, is used to assess the validity of a classification model. It summarizes the correct and incorrect predictions of each class relative to each other. Table 4.3 illustrates the performance measurement metrics expressed as a confusion matrix. The normalized TP is another important metric also called Sensitivity expressed by Equation (4.6). Similarly, the normalized TN is another metric of

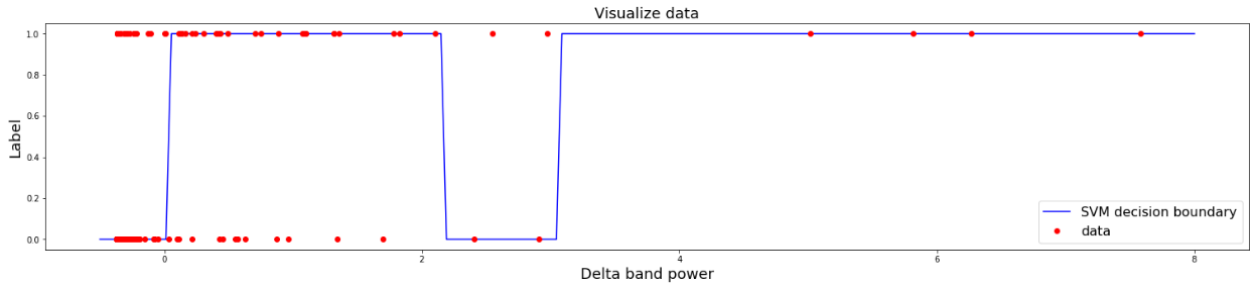


FIGURE 4.6 Decision boundary of the SVM classifier considering one feature, the delta band power.

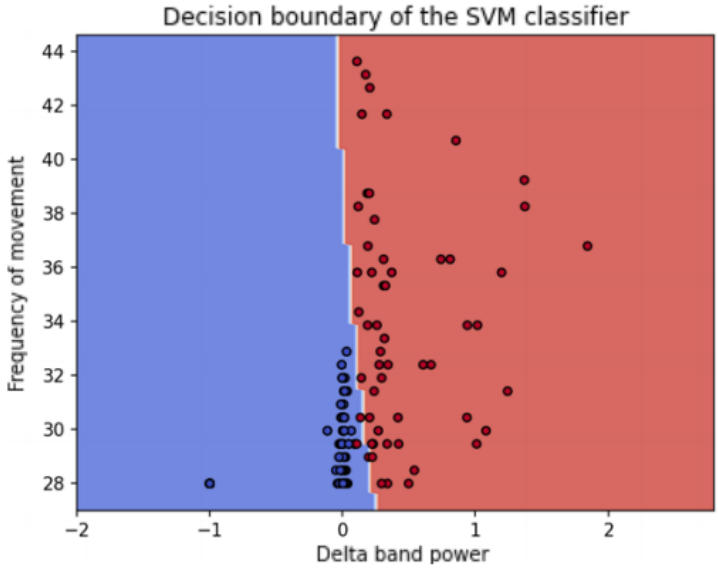


FIGURE 4.7 Decision boundary of the SVM classifier when the delta band power and the frequency of movement are used as the two features of interest.

TABLE 4.1 SVM classifier confusion matrix considering the delta band power feature for the training set and the test set.

		Delta band power							
		Training set, Norm=None		Training set, Norm=True		Test set, Norm=None		Test set, Norm=True	
True label	0	TP :119	FN :7	TP :0.94	FN :0.056	TP :28	FN :5	TP :0.85	FN :0.15
	1	FP :40	TN :25	FP :0.62	TN :0.38	FP :11	TN :4	FP :0.73	TN :0.27
		0	1	0	1	0	1	0	1
		Predicted label				Predicted label			

interest called Specificity expressed by Equation (4.7).

$$Normalized\ TP = Sensitivity = TP/(TP + FN) \tag{4.6}$$

TABLE 4.2 SVM classifier confusion matrix considering the delta band power and the frequency of movement as the two features of interest for the training set and the test set.

		Delta band power and frequency of movement							
		Training set, Norm=None		Training set, Norm=True		Test set, Norm=None		Test set, Norm=True	
True label	0	TP :122	FN :2	TP :0.98	FN :0.016	TP :32	FN :0	TP :1	FN :0
	1	FP :2	TN :42	FP :0.045	TN :0.95	FP :1	TN :10	FP :0.091	TN :0.91
		0	1	0	1	0	1	0	1
		Predicted label				Predicted label			

TABLE 4.3 Performance measurement metrics of the confusion matrix.

True Positive (TP)	Correctly classified as the class of interest
True Negative (TN)	Correctly classified as not the class of interest
False Positive (FP)	Incorrectly classified as the class of interest
False Negative (FN)	Incorrectly classified as not the class of interest

$$Normalized\ TN = Specificity = TN/(FP + TN) \tag{4.7}$$

The goal is to maximize the values of the diagonal elements of the matrix called TP and TN, and to minimize the other elements called FP and FN. Table 4.1 shows that using only the delta band power feature for the training set and the test set, class 0 is predicted correctly most of the time, with $TP = 0.94$ and 0.85, respectively. However, class 1 is classified incorrectly most of the time, with $TN = 0.38$ and 0.27. Indeed, when only one feature is considered, the two classes are not separated properly. To overcome this challenge we had to consider a second feature, the frequency of movement. As Table 4.2 shows, when we consider two features : 1) the delta band power and 2) the frequency of movement for the training set and the test set, both class 0 and 1 are classified correctly most of the times with, $TP = 0.98$ and 1 and $TN = 0.95$ and 0.91. Moreover, the non-diagonal metrics FN and FP are almost zero. Based on these results, selecting classes based on two features significantly improves the true negative classification performance when compared to considering only one feature for both classes of 0 and 1.

4.5 Comparison of the Ability to Detect Human Activity with Higher Movement Energy Using the Dataset Provided by [112]

[112] also provided a CSI-based dataset to detect human behavior. In this alternate dataset, $N_{Tx} = 2$ and $N_{Rx} = 3$. In the experiments used to produce this alternate dataset, a total of 3 different rooms were used. All the experiments for this dataset were performed in the 5 GHz frequency band with 20 MHz bandwidth channels. Human activities associated with falling and walking activities are considered. Table 4.4 summarizes the collected number of samples for falling and walking activities.

TABLE 4.4 The collected number of samples for each activity.

Activity	Fall	Walk
Number of samples	71	204

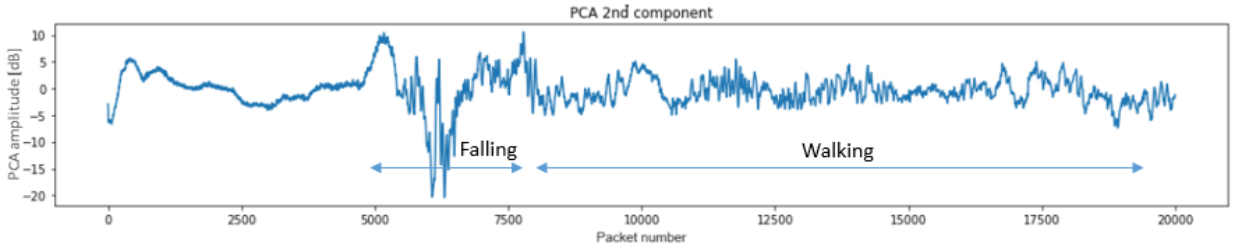


FIGURE 4.8 Received CSI amplitude versus packet numbers for the falling and walking activities.

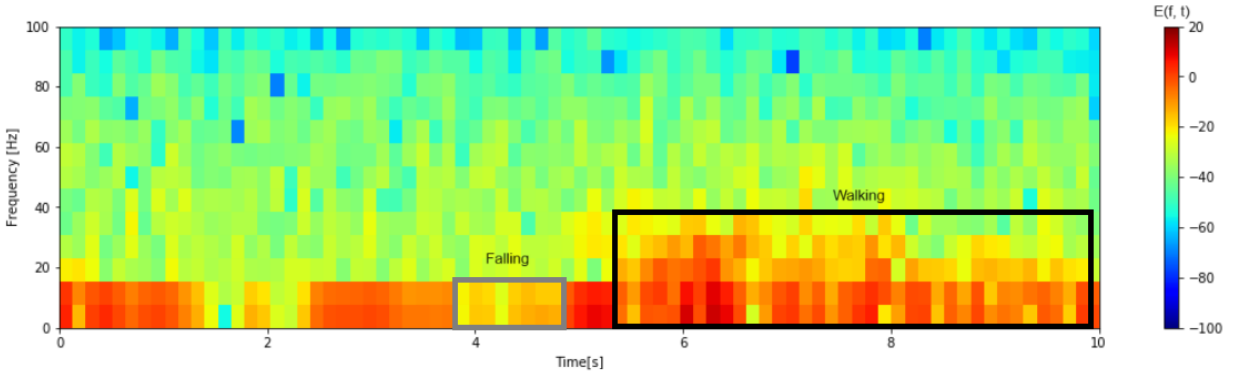


FIGURE 4.9 Time-frequency analysis for human falling and walking activities.

Figure 4.8 illustrates the received CSI amplitude versus packet numbers for the falling and walking activities. Based on this figure, the time interval when the falling and walking

activities occur is roughly specified as a pair of packet numbers such as $5000 - 8000$ and $8000 - 20000$, respectively. As Figure 4.9 reflects, falling and walking activities occur in the period of gray and black rectangular shapes, respectively. According to Figure 4.9, the energy band for falling and walking activities is concentrated around $0 - 18$ Hz and $10 - 40$ Hz frequency bands, respectively. In fact, in the walking activity, the CSI energy is concentrated at higher frequencies compared to the falling activity. This helps using the SVM classifier to define the decision boundary between falling and walking activities.

FIGURE 4.10 Decision boundary of the SVM classifier when two features are used : 1) the delta band power and 2) the frequency of movement using the dataset provided by [112].

Figure 4.10 illustrates the decision boundary that can be used with the SVM classifier when two features are used : 1) the delta band power and 2) the frequency of movement when using Z. Wang et al. [112] dataset to detect falling and walking activities. The threshold α is the boundary between the red and the blue regions in the range of $[0, 1]$. The red circles and the red area are related to the walking activity with the label 1, while the blue circles and the blue area correspond to the falling activity region with the label 0. As this figure shows, walking activity occurs with a higher frequency of movement in comparison to falling activity.

Tables 4.5 and 4.6 report the SVM classifier confusion matrix considering one feature of delta band power and two features of delta band power and frequency of movement using Z. Wang et al. [112] dataset, respectively. 80% of the data is used as a training set and 20% is used as a test set. According to Table 4.5, when we consider the delta band power as

TABLE 4.5 SVM classifier confusion matrix considering delta band power feature using the dataset provided by [112].

		Delta band power							
		Training set, Norm=None		Training set, Norm=True		Test set, Norm=None		Test set, Norm=True	
True label	0	TP :36	FN :23	TP :0.61	FN :0.39	TP :6	FN :6	TP :0.50	FN :0.50
	1	FP :2	TN :159	FP :0.012	TN :0.99	FP :2	TN :41	FP :0.047	TN :0.95
		0	1	0	1	0	1	0	1
		Predicted label				Predicted label			

TABLE 4.6 SVM classifier confusion matrix obtained when considering two features, 1) the delta band power and 2) the frequency of movement using the dataset provided by [112].

		Delta band power and frequency of movement							
		Training set, Norm=None		Training set, Norm=True		Test set, Norm=None		Test set, Norm=True	
True label	0	TP :53	FN :4	TP :0.93	FN :0.07	TP :8	FN :2	TP :0.80	FN :0.20
	1	FP :7	TN :156	FP :0.043	TN :0.96	FP :0	TN :45	FP :0	TN :1
		0	1	0	1	0	1	0	1
		Predicted label				Predicted label			

the only feature, class 1 corresponding to the walking activity is predicted correctly most of the time, with $TN = 0.99$ and 0.95 for the training set and the test set. However, class 0 corresponding to the minority class of falling activity is classified incorrectly, $TP = 0.61$ and 0.50. According to Table 4.6, when we consider two features, the delta band power and the frequency of movement for the training set and the test set, both class 0 and 1 are classified correctly, $TP = 0.93$ and 0.80 and $TN = 0.96$ and 1. Moreover, the non-diagonal metrics FN and FP are almost zero. As Table 4.4 shows, the number of occurrences of the walking activity is higher than the number of occurrences of the falling activity. The means of exploiting this type of imbalanced dataset is addressed in Chapter 5. Since the falling activity occurs over a time duration shorter than the walking activity, a new robust feature adapted to the detection of this type of activity is explored in Chapter 6.

4.6 Summary

This section proposed a method that can automatically distinguish people who perform activities with higher movement energy. We discussed how to select features to achieve a better performance. The results show that adding the frequency of movement as a second feature in addition to the delta band power increases the true negative classification performance from 38% to 95% for the training set and from 27% to 91% for the test set. We

detected falling and walking activities using another dataset. The adaptive threshold was obtained by the decision boundary of the SVM classifier. The results show that extracting two features of delta band power and frequency of movement separates two classes of walking and falling properly by obtaining $TP = 80\%$ and $TN = 100\%$ for the test set.

CHAPTER 5

HANDLING DATASET IMBALANCE FOR HUMAN ACTIVITY RECOGNITION

To differentiate people who perform movement with higher energy from the others, we use an imbalanced WiFi-based dataset. Such imbalanced dataset which includes majority and minority classes in the training set affects the performance of machine learning algorithms. Unreliable samples in the majority class can perturb the minority class. This chapter proposes a method to efficiently classify imbalanced datasets in Section 5.1. Then, we explore the impact on classification performance of the imbalance observed in the WiAR dataset in relation to feature selection in a proposed machine-learning-based classification algorithm in Section 5.2. The reported results show that the proposed method improves the F1-score performance for the minority class from 46% to 95%. In Section 5.3, the proposed method is compared with the threshold adjustment classification method. A comparison of the obtained results with those obtained with another imbalanced dataset is performed in Section 5.4. Finally, a summary of the main findings and conclusions drawn from the work on imbalanced datasets are provided in Section 5.5.

5.1 Handling HAR Detection From Imbalanced Dataset

As mentioned earlier, we use an imbalanced WiAR dataset [12] generated from WiFi edge devices. In this dataset, the instances of some classes are much more prevalent than the others. As Table 3.2 represents in Chapter 3, Section 3.5, the walking activity has the largest number of samples, while the standing activity has the least.

To classify the human activities, we divide the dataset into the training and the test sets and assess the classifier's performance on the test set. A significant complexity arises when the training set is imbalanced in a way that makes one or several classes include many more samples compared to the others.

In this chapter, we apply the SVM classifier proposed in Chapter 4 to detect human activities. There are three known methods to improve the performance of an SVM : 1) using different kernels, 2) tuning the threshold, and 3) performing gradient-based optimization. In

the first method, the authors in [133] proposed an information-geometric method to modify the kernel. In the second method, the hyperplane of the SVM is shifted towards the majority class [134]. With the third method, the goal is to minimize the cost function by using a gradient-based algorithm [135]. In this research, we select the second method to tune the SVM threshold. Balancing the dataset is an alternative method to deal with an imbalanced class. The main difference between adjusting the SVM threshold and balancing the dataset is that the former relies on tuning the model output, while the latter depends on data preprocessing before the learning step. Therefore, combining two methods (adjusting the SVM threshold and balancing the dataset) as an ensemble technique is a possible means to get better estimates. In this research, we utilize the bagging technique [136] combined with SMOTE and SVM as an ensemble method. We consider adjusting the SVM threshold as a base method to compare the performance.

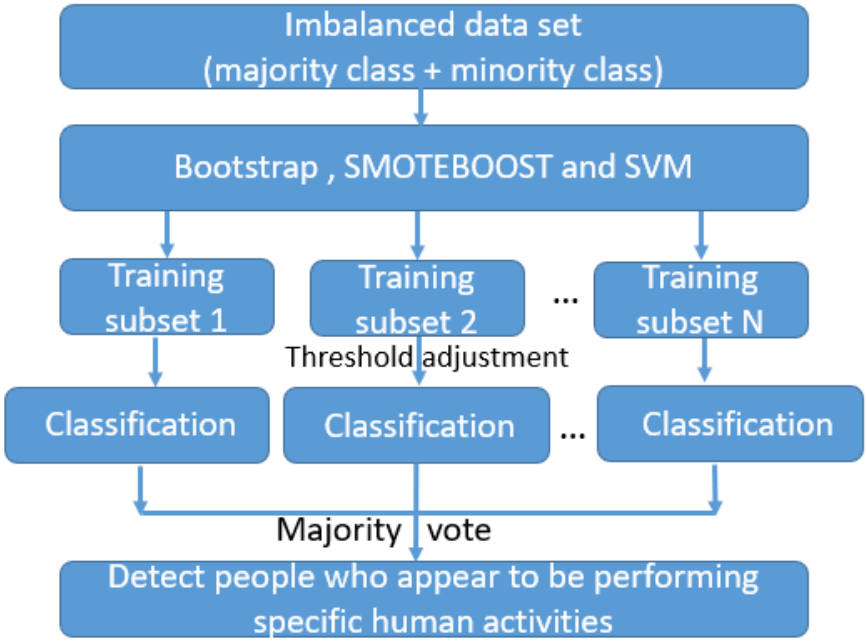


FIGURE 5.1 Ensemble method proposed to classify the imbalanced HAR dataset.

Figure 5.1 illustrates the ensemble technique proposed to improve the performance of activity detection with the imbalanced HAR dataset. Bagging and boosting are two main techniques used in ensemble learning. In our proposed method, bootstrap aggregation known as the bagging technique is used in an ensemble technique including SMOTEBoost to reduce the variance of individual models in each subset.

SMOTEBoost is based on the SMOTE algorithm to create a strong classifier from

several weak classifiers to improve prediction accuracy. The K-nearest neighbors technique is used in the SMOTE method to create synthetic samples of the minority class. In fact, in this method, multiple base classifiers are trained using the same underlying algorithm with different weighted samples from the training set. The majority voting method is performed to get a final class label and to detect human activity. In this algorithm, 100 subsets of the training set are created as weak learners. The SVM classifier is considered as a base estimator of the bagging classifier. As a result, different decision thresholds are obtained in each base classifier to predict the class label.

5.2 Performance Evaluation of the Proposed Ensemble Method

After addressing the class imbalance problem, we evaluate the classification performance using appropriate evaluation metrics such as ROCAUC, precision, recall, and F1-score. The ROC is a ranking metric to evaluate the prediction results on a test set by using a set of thresholds to interpret the true positive (TP) rate and the false positive (FP) rate of the predictions on the minority class [137]. One of the challenges in calculating the ROC is choosing a threshold in a way that results in the best balance between the TP and the FP rates. The area under the ROC curve is called ROCAUC. It reflects the performance of the model based on the ROC curve with a value between 0.5 (no-skill or random classifier) and 1 (perfect skill or perfect classifier). The highest ROCAUC score corresponds to the best ROC threshold. In this work, ROCAUC is calculated using the Python Sklearn library. Figure 5.2 illustrates an example of ROCAUC. The area under a ROC curve (AUC) corresponds to the integral of the curve TP rate values with respect to FP rate values from zero to one. A perfect classifier always correctly assigns positive class observations to the positive class and has a TP rate of 1 for all threshold values.

Precision and recall are other performance metrics for imbalanced datasets considering the positive samples from the minority class only. Precision is also called positive predictive value. It is the fraction of TPs among the sum of TP and FPs as expressed by Equation (5.1) [138].

$$Precision = TP/(TP + FP) \tag{5.1}$$

The recall metric is the fraction of TP divided by the sum of the TP and the false negatives (FN) as expressed by Equation (5.2) [138].

$$Recall = TP/(TP + FN) \tag{5.2}$$

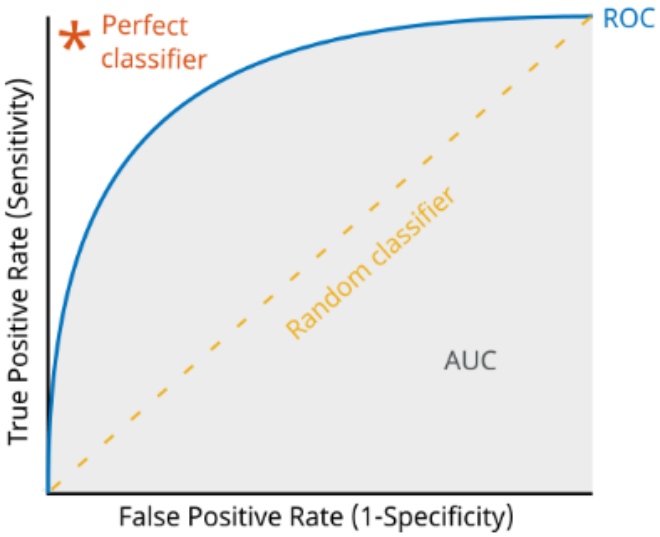


FIGURE 5.2 ROCAUC example.

Finally, the F1-score in Equation (5.3) measures the harmonic mean of precision and recall.

$$F1_{score} = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{5.3}$$

The optimal threshold is the one that produces the maximum F1 score. The threshold with the best evaluation metric is selected for the model to make predictions on the test set or when new data comes to the system. As Figure 5.3 illustrates, we validate the performance

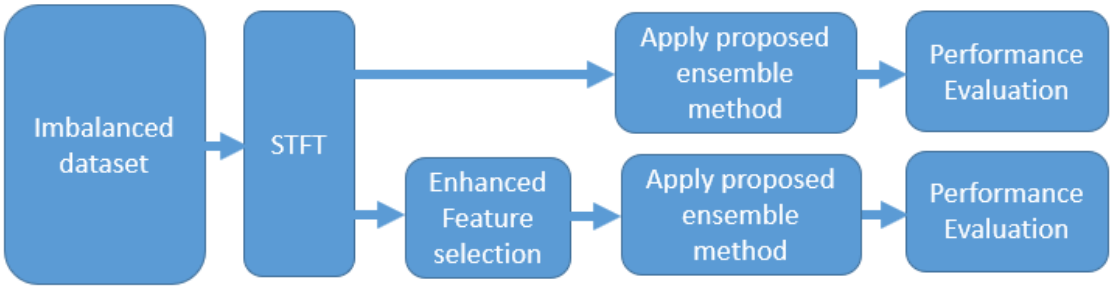


FIGURE 5.3 Steps of the method adopted to evaluate the impact of feature selection and dataset imbalance on the performance of our activity recognition method.

in two cases by 1) applying a proposed ensemble method to extract only one feature of delta band power and 2) performing feature selection before balancing and resampling the dataset to fulfill the ensemble learning model. In this step, we take into account the frequency of movement as a second feature and then classify the dataset with the proposed ensemble

method.

TABLE 5.1 Performance evaluation of the ensemble method based on the delta band power feature only.

		Precision	Recall	F1-score
SVM SMOTE Bagging	**Class 0**	**0.76**	**0.85**	**0.80**
	Class 1	**0.55**	**0.40**	**0.46**
	Accuracy		**0.70**	
	ROCAUC		**0.64**	
	Macro avg	**0.65**	**0.62**	**0.63**
	Weighted avg	**0.69**	**0.71**	**0.69**
Borderline SMOTE Bagging	**Class 0**	**0.73**	**0.79**	**0.76**
	Class 1	**0.52**	**0.40**	**0.43**
	Accuracy		**0.70**	
	ROCAUC		**0.63**	
	Macro avg	**0.62**	**0.59**	**0.60**
	Weighted avg	**0.65**	**0.67**	**0.65**

TABLE 5.2 Performance evaluation of the ensemble method based on the delta band power and the frequency of movement features.

		Precision	Recall	F1-score
SVM SMOTE Bagging	**Class 0**	**0.98**	**0.97**	**0.98**
	Class 1	**0.96**	**0.94**	**0.95**
	Accuracy		**0.96**	
	ROCAUC		**0.98**	
	Macro avg	**0.99**	**0.96**	**0.98**
	Weighted avg	**0.98**	**0.98**	**0.98**
Borderline SMOTE Bagging	**Class 0**	**0.95**	**0.92**	**0.93**
	Class 1	**0.92**	**0.89**	**0.90**
	Accuracy		**0.93**	
	ROCAUC		**0.96**	
	Macro avg	**0.96**	**0.92**	**0.94**
	Weighted avg	**0.95**	**0.93**	**0.94**

Table 5.1 presents the performance of the ensemble method described in Figure 5.1. This method applies different classifiers. The considered classifiers are the SVM SMOTE Bagging and the Borderline SMOTE Bagging exploiting the delta band power feature. In the SVM SMOTE Bagging algorithm, artificial samples of the minority class are created based on the similarities in the feature space between the existing samples using the KNN algorithm. The generated artificial samples are similar to the samples of the existing minority class, but they are not simply duplicated [139]. Borderline-SMOTE synthesizes samples near the decision border using the SMOTE technique [140]. In this table, macro-avg is the average of precision, recall, and F1-score for all classes. Weighed-avg is a type of means that gives

different weights to the values of precision, recall, and F1-score for each class. The weights are based on the number of actual occurrences of each class in the test set [141].

Table 5.2 summarizes the performance of the ensemble method described in Figure 5.1 applying SVM SMOTE Bagging and Borderline SMOTE Bagging using the delta band power and the frequency of movement features. Based on the results reported in Tables 5.1 and 5.2, the F1-score for the minority class is improved from 90% to 95%, the precision and recall are increased from 92% to 96% and 89% to 94%, respectively, by transitioning from Borderline SMOTE Bagging to SVM SMOTE Bagging algorithm. These results show that if we select the proper features to feed the input of the SVM classifier and then balance the imbalanced dataset, we achieve much better performance. However, addressing the imbalanced dataset without considering suitable features leads to no improvement. Additionally, based on the reported results, the SVM SMOTE Bagging ensemble technique achieves the highest performance compared to the Borderline SMOTE Bagging classifier. SVM SMOTE Bagging leverages SVM as the base classifier, which is known for its ability to handle complex decision boundaries. Since SVM SMOTE Bagging considers the support vectors and their neighbors to create synthetic samples, it produces synthetic samples that are better balanced.

5.3 Comparison with the Threshold Adjustment Method

In this section, we explain how we tuned the SVM threshold [142] with the imbalanced HAR dataset. Additionally, we explore whether adjusting the SVM threshold could handle the imbalanced dataset. Algorithm 1 describes the threshold adjusting method based on the ROC metric [143]. To apply this technique, the delta band power and the frequency of movement are considered as the features that can be used to detect human activities.

Adding the frequency of movement feature in addition to the delta band power increased the true negative (TN) classification performance from 38% to 95% for the training set, and from 27% to 91% for the test set in Chapter 4. Proper feature selection helps reduce overfitting by focusing on the most informative features, which leads to better generalization when presented with new data. We compare the performance of the proposed ensemble method with the threshold adjustment technique [142] to explore the importance of data distribution between different classifier subsets.

In Algorithm 1, $\mathbb{A}^+$ and $\mathbb{A}^-$ are considered as the minority and the majority classes related

to the running activity as the first group, and walking and standing activities as the second group, respectively. In those experiments, 80% of the data is considered as a training set and 20% as the test set using 5-fold cross-validation.

5-fold cross-validation is a popular technique in machine learning for assessing the performance of a predictive model. The process involves partitioning the dataset into five equally sized folds or subsets. The model is then trained and evaluated five times, each time using a different fold as the test set and the remaining four folds as the training set. The goal of cross-validation is to ensure that the model's performance is consistent across different subsets of the data, helping to identify issues such as overfitting or underfitting.

The extracted delta band powers are normalized into the range $[0 - 1]$ to improve the performance of modeling. As part of this process, the model is first fitted on the training data and then the prediction is performed on the test data. The probabilities are converted to the class label. Thresholds in a range of $0 - 1$ are utilized to measure the ROCAUC. Precision, recall, and F1-score are calculated as additional evaluation metrics.

Algorithm 1 Threshold adjustment algorithm for the imbalanced HAR dataset

Input : Minority class $\mathbb{A}^+$, majority class $\mathbb{A}^-$
Classification steps :
1 : Consider 80% of the data as the training set and 20% as the test set using 5-fold cross-validation in each subset
2 : Extract the features of delta band power and corresponding frequency of movement
3 : Normalize delta band power into the range $[0 - 1]$ to train SVM classifier
4 : Fit SVM model on the training dataset
5 : Predict probabilities on the test set
6 : Measure ROCAUC to evaluate the predictions on step 5
for all Thresholds created from ROC evaluation **do**
    7 : Convert probabilities to class labels using the threshold
    8 : Evaluate class labels
    9 : If the score is better than the best score, update the threshold
end for
10 : Use the updated threshold to predict a specific human activity

Table 5.3 illustrates the performance evaluation of the threshold adjustment method when considering the delta band power and the frequency of movement features. Comparing Table 5.2 and 5.3 shows that the SVM SMOTE Bagging achieves better performance when compared with the threshold adjustment technique. The F1-score for the minority class is improved from 85% to 95% and the ROCAUC is increased from 81% to 98%. SVM SMOTE Bagging as an ensemble technique combines multiple SVM classifiers

TABLE 5.3 Performance of the threshold adjustment method considering the delta band power and the frequency of movement features.

		Precision	Recall	F1-score
Threshold adjustment algorithm	**Class 0**	**0.91**	**0.83**	**0.90**
	Class 1	**0.89**	**0.80**	**0.85**
	Accuracy	**0.83**		
	ROCAUC	**0.81**		
	Macro avg	**0.90**	**0.81**	**0.87**
	Weighted avg	**0.91**	**0.83**	**0.88**

trained on different bootstrapped samples. It distributes data on different subsets. This technique reduces overfitting and improves generalization. It results in better classification performance in comparison with a single threshold adjustment classifier.

5.4 Handling Imbalanced Dataset ; Comparison with Z. Wang's dataset [112]

In Chapter 4, we used a second imbalanced dataset provided by Z. Wang et al. [112]. This dataset, distinct from the WiAR dataset was proposed to enable research on classifiers applicable to detect and distinguish falling and walking activities. To balance Z. Wang's dataset, we apply the proposed ensemble method to improve the performance.

TABLE 5.4 Performance evaluation of the ensemble method based on the delta band power feature only using Z. Wang's dataset [112].

		Precision	Recall	F1-score
SVM SMOTE Bagging	**Class 0**	**0.75**	**0.50**	**0.60**
	Class 1	**0.78**	**0.55**	**0.64**
	Accuracy	**0.71**		
	ROCAUC	**0.65**		
	Macro avg	**0.77**	**0.53**	**0.62**
	Weighted avg	**0.65**	**0.64**	**0.64**
Borderline SMOTE Bagging	**Class 0**	**0.72**	**0.49**	**0.56**
	Class 1	**0.75**	**0.52**	**0.59**
	Accuracy	**0.69**		
	ROCAUC	**0.63**		
	Macro avg	**0.54**	**0.51**	**0.58**
	Weighted avg	**0.63**	**0.62**	**0.63**

Tables 5.4 and 5.5 summarize the performance of the proposed ensemble method in two cases. The first one considers the delta band power as the only feature, while the second one takes into account the delta band power and the frequency of movement features. For both cases, two different classifiers are considered. The first one is the SVM SMOTE Bagging,

TABLE 5.5 Performance evaluation of the ensemble method based on the delta band power and the frequency of movement features with Z. Wang's dataset [112].

		Precision	Recall	F1-score
SVM SMOTE Bagging	**Class 0**	**0.95**	**0.94**	**0.92**
	Class 1	**0.98**	**0.97**	**0.95**
	Accuracy	**0.94**		
	ROCAUC	**0.96**		
	Macro avg	**0.96**	**0.95**	**0.93**
	Weighted avg	**0.96**	**0.96**	**0.96**
Borderline SMOTE Bagging	**Class 0**	**0.93**	**0.92**	**0.90**
	Class 1	**0.95**	**0.94**	**0.93**
	Accuracy	**0.91**		
	ROCAUC	**0.93**		
	Macro avg	**0.94**	**0.93**	**0.92**
	Weighted avg	**0.94**	**0.93**	**0.94**

while the second one is the Borderline SMOTE Bagging. According to Table 5.4, balancing the imbalanced dataset without considering the proper features does not improve the classification performance. However, based on the results reported in Table 5.5 extracting the frequency of movement in addition to the delta band power feature improves the ROCAUC performance from 65% to 96% and the F1-score from 60% to 92%. This shows that the proposed classification method is not limited to classifying the WiAR data set as it applies equally well to Z. Wang's dataset [112].

5.5 Summary

This research shows that attempting to handle the imbalanced class problem without considering the proper features does not improve the classifier's performance as much as it could. However, considering the delta band power and the frequency of movement features before the ensemble classification method increased the ROCAUC from 64% to 98% and the F1-score from 46% to 98%.

Although the SVM SMOTE Bagging and Borderline SMOTE Bagging techniques both handle the class imbalance problem by oversampling the minority class, the SVM SMOTE Bagging classifier achieved better performance. This method inherits the strengths of SVM in handling decision boundaries by considering the support vectors and their neighbors. It also creates synthetic samples aligned with the SVM decision boundary. Additionally, it mitigates overfitting through ensemble learning.

CHAPTER 6

CHARACTERIZATION AND SELECTION OF IMPORTANT FEATURES FOR HUMAN ACTIVITY DETECTION

The general objective of this chapter is to explore an effective tracking algorithm related to non-deterministic target dynamics as mentioned in Chapter 3, Section 3.5.1. In this context, a target is a person whose movements are influenced by random or uncertain factors. The term "non-deterministic" implies that the exact trajectory or behavior of the target cannot be precisely determined in advance due to the influence of unpredictable elements. The method proposed in the present chapter to analyze the observed complex CSI signals proceeds by extracting chirp parameters that correspond to specific human activities and from which these activities can be detected. This feature has never been applied to the classification of human activities. The chirp signal was described in Chapter 2, Section 2.13. We estimate some other spectral and temporal STFT features as well. We characterize features by applying the PCA method before the decision-tree algorithm to select important features in order to obtain the best possible performance. Characterization and classification results prove that the chirp rate feature has high detection capability for a wide range of human movement speeds from a very low speed observed with walking subjects to a very high speed for running activity subjects.

Therefore, this chapter presents the methodology and an enhanced processing framework to characterize human activity detection parameters in Section 6.1. In Section 6.2, we describe the features of the STFT and how to estimate the chirp parameters for human activity from the incoming CSI data. In Sections 6.3 and 6.4, we present PCA and the decision-tree algorithm to select important features, respectively. In Section 6.5, we discuss the characterization results. In Section 6.6, we benchmark the performance of the selected features by estimating the confusion matrix metrics. Additionally, we recommend the features found to be most effective based on observed variability, information gain, and correlation between features. Comparison with another dataset to characterize and select important features is performed in Section 6.7. Finally, conclusions are drawn in Section 6.8.

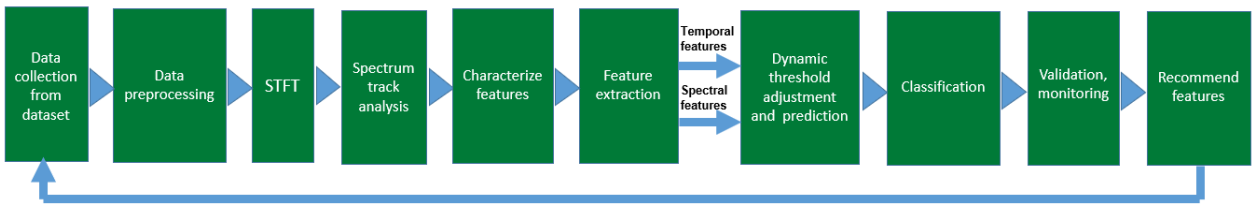


FIGURE 6.1 An enhanced processing framework to detect human activities.

6.1 Methodology

The proposed improved HAR detection framework comprises ten steps as shown in Figure 6.1. Two modules were added to the processing framework proposed in Chapter 4, Section 4.1 to characterize and recommend features based on their efficiency. The improved framework notably extracts the chirp rate and the spectral flux as additional features in addition to the delta band power and the frequency of movement.

The goal of characterizing these features is to select an appropriate subset of features that is adequate for describing a specific target. In this component, we investigate whether combining features allows gaining any significant additional information in Section 6.5. Moreover, before training a model to extract features, it is important to apply feature selection techniques and characterize it to avoid overfitting, long training time, and noisy data as input.

We apply PCA before the decision-tree algorithm to automate the decision-making about features by transforming data to directions that have the highest variability and that offer information gain. Features both in the temporal and spectral domains are extracted. Among the various temporal features, dominant features such as chirp rate and frequency of movement are predicted. Also, the spectral features such as delta band power, and spectral flux are considered. After deriving the values of features both in time and spectral domains, we estimate the dynamic threshold as a decision boundary of the classifier. We apply a 5-fold cross-validation method and confusion matrix to validate the model and monitor the data in Section 6.6 after selecting the most important features. There is a trade-off between latency, accuracy, and selecting a set of features. Thus, we recommend feature selection based on observed variability, information gain, and correlation between features.

6.2 STFT Temporal and Spectral Features

In this section, we describe temporal and spectral features related to the STFT which are important to estimating and detecting the desired characteristics of the signals. The order in which features are introduced is based on their importance according to our analysis results.

Feature 1 : $\mathcal{F}(1)$ = Chirp parameters - In this section, we describe what is a chirp signal, how to estimate the chirp parameters, and how incoming CSI could be analyzed to estimate the parameters of an equivalent chirp signal for each human activity. A chirp signal is a signal whose frequency increases or decreases with time. The chirp transform provides superior localization in both time and frequency domains compared to traditional Fourier transform methods in the case of transient signals [144]. Mathematically, the linear chirp signal is presented by Equation (6.1).

$$x(t) = \exp[j\phi_0 + 2\pi j(\alpha t + \frac{\beta}{2}t^2)] \quad 0 < t < T \tag{6.1}$$

where $j = \sqrt{-1}$, α is a starting frequency, β is a chirp rate, ϕ_0 is an initial phase and T is the duration of the chirp signal [145]. To estimate the chirp parameters, we compute the STFT of the incoming CSI data for each human activity. Mathematically, the N-point STFT of a signal at time m is described as Equation (6.2) [146].

$$X_{n,m} = \sum_{i=0}^{N-1} x_i w_{i-m} e^{-j2\pi n \frac{i}{N}}, \qquad n = 0, 1, ..., N-1 \tag{6.2}$$

where x_i is the i^{th} sample of incoming CSI data, n and m represent the corresponding frequency index and data block index, respectively. N is the number of window samples, and it is important to note that the window sequence, denoted as w_m, is not zero in the interval [0, N-1]. For each human activity, the indices of the start frequency and the end frequency are considered as n_1 and n_2, respectively. $N = n_2 - n_1 + 1$ is the number of frequency indices that cover the band of each human activity. The spectrum matrix $\mathbf{X}$ consists of complex values X_{nm} as expressed by Equation (6.3).

$$\mathbf{X} = [X_{n,m}],\ n = n_1, n_1 + 1, ..., n_2;\ m = 0, 1, ..., M-1 \tag{6.3}$$

where M is the number of samples in each time block. The magnitude squared of $X_{n,m}$ creates

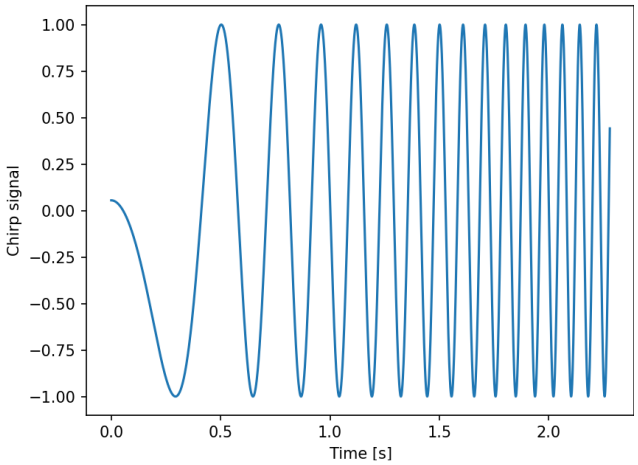


FIGURE 6.2 Chirp signal produced using Python from parameters extracted from STFT of CSI data for a standing activity. The standing activity includes a walking activity near the end.

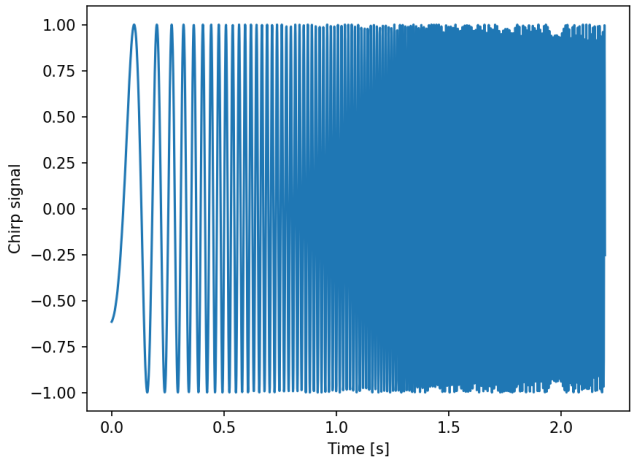


FIGURE 6.3 Chirp signal produced using Python from parameters extracted from STFT of CSI data for a running activity.

the STFT frequency time-space matrix as Equation (6.4).

$$\mathbf{S} = [|X_{n,m}|^2],\ n = n_1, n_1 + 1, ..., n_2;\ m = 0, 1, ..., M - 1 \tag{6.4}$$

We estimate chirp parameters from the STFT frequency time-space matrix **S**. Since the chirp parameters depend on the duration and the dynamic speed of each activity, it can be used to detect human activity characteristics.

For a human activity with a time-varying frequency of movement, a chirp rate would be

the slope of the frequency variation divided by the fundamental frequency.

$$\beta = \frac{\Delta f}{f_0} = \frac{f_{n2} - f_{n1}}{f_0} \tag{6.5}$$

where f_0 is the center frequency of 5300 NIC card for 5 GHz band equals 5.825 GHz and Δf is the difference between the maximum and minimum frequency for each human activity. We calculate T from the time range during which significant energy is observed according to the STFT energy threshold ξ expressed as Equation (6.6). This time range T becomes the estimated duration of each activity.

$$T = max\{\Delta t | \frac{\Delta Energy}{TotalEnergySTFT} \geq \xi\} \tag{6.6}$$

Algorithm 1 Estimate chirp parameters

1 : Collect CSI data for each human activity
2 : Apply the STFT transform
3 : Estimate chip rate β from Equation (6.5)
4 : Estimate α, T from the STFT of the incoming CSI data
5 : Estimate ϕ_0 from a spectral analysis using the Hilbert transform
6 : Calculate Equation (6.1) by estimated parameters $(\alpha, \beta, \phi_0, T)$

We consider $\alpha = f_{n1}$ for each human activity as the starting frequency. We calculate the initial phase from a spectral analysis using the Hilbert transform [147]. Algorithm 1 summarizes the method to estimate the chirp parameters from CSI data and to a corresponding illustrative chirp signal if needed. Figures 6.2 and 6.3 show the illustrative chirp signals produced with Python that were derived from CSI data for standing and running activities, respectively. The standing activity includes a walking activity near the end. The chirp signals of standing and running activities were obtained using algorithm 1. The STFT of the human activities was obtained in Chapter 3, Section 3.7.1. Chirp parameters were substituted in Equation (6.1) to synthetically generate a chirp signal using Python as per extracted parameters.

Feature 2 : $\mathcal{F}(2)$ = Delta Band Power - Algorithm 2 describes how to estimate the delta band power. It is derived from Power Spectral Density (PSD) which facilitates the calculation of cumulative power in the spectrum of a signal. We introduced and explained how to calculate the delta band power in Chapter 4. It is summarized here for completeness.

Algorithm 2 Delta band power

1 : Divide the dataset into segments
2 : Estimate PSD for each segment
3 : Compute the band power using the PSD estimate in a specific frequency range $[f1, f2]$
4 : Calculate the change of band power in the frequency range of
$[f1 + \Delta f, f2 + \Delta f]$ and $[f1, f2]$

Feature 3 : $\mathcal{F}(3) =$ **Spectral Flux** - The spectral flux is computed as the difference of the squared values between the normalized magnitudes of the spectra of two successive frames. More precisely, it calculates the L2-norm, known as the Euclidean distance, between the two normalized spectra [148].

$$SF_p = \sum_{k=1}^{N/2} (|X_p[k]| - |X_{p-1}[k]|)^2 \tag{6.7}$$

where p is the frame index, k is the frequency index for the discrete Fourier transform, $X_p[k]$ and $X_{p-1}[k]$ are the normalized magnitude of the spectrum of the present and previous frames, respectively. In fact, spectral flux indicates the change in the power spectrum of the signal within the frames.

Feature 4 : $\mathcal{F}(4) =$ **Frequency of movement** - Multi-path reflection model is used to estimate the frequency of movement described in Chapter 3, Section 3.3.

6.3 PCA

PCA is used for dimension reduction of large datasets, by transforming a large set of variables into a smaller one that still contains most of the information and minimizes the projection error. It creates a set of linearly uncorrelated axes called principal components (PCs) in a way that the first principal component captures the maximum variance in the data, the second principal component captures the second largest variance (while being orthogonal to the first), and so on [119]. An important question with PCA is finding how many dimensions we need to analyze the data. The first metric to answer the question is the redundancy in the dataset. Ideally, the basis vectors of the PCA should not create redundant data. Redundant data refers to data features that do not contribute significantly to the overall variance in the dataset. Redundant features are highly correlated with one another and with other features in the dataset, meaning they carry essentially the same information as other variables.

Figure 6.4 illustrates a spectrum of possible redundancies in a dataset. The data in Figure 6.4 (a) are uncorrelated because one cannot predict one from the other. Conversely,

FIGURE 6.4 A range of potential redundancies within a dataset a) Uncorrelated dimensions, b) Partly correlated dimensions, and c) Strongly correlated dimensions.

the data in Figure 6.4 (c) are highly correlated indicating highly redundant data. The data in Figure 6.4 (b) are partially correlated. Removing redundant and correlated basis vectors reduces the dataset dimensionality. The second metric to identify redundancy is the covariance measure. A large positive value of covariance shows a high correlation between two features. A large negative value represents that two features are negatively correlated. It describes when two variables tend to move in opposite sizes and directions from one another, such that when one increases the other variable decreases, and vice-versa. The covariance value for uncorrelated features is zero. If we consider features as a column vector of H, where h_i represents the i^{th} feature, the covariance matrix C_H is represented by Equation (6.8).

$$C_H = \frac{1}{N} H H^T \tag{6.8}$$

Eigenvectors of covariance matrix $H \times H^T$ with dimension of $N \times N$ are calculated, where N is the number of samples. The diagonal elements of the covariance matrix represent the variance of feature vectors, and large values correspond to significant features. On the other hand, the non-diagonal terms indicate the covariance between features, and large values correspond to high redundancy in the data.

Algorithm 3 PCA Algorithm [149]

1: Normalize input data
2: Calculate the covariance matrix of input data
3: Calculate eigenvectors and eigenvalues
4: Sort the eigenvectors according to eigenvalues in a decreasing order
5: Choose first k eigenvectors that will be the new k dimensions

Algorithm 3 summarizes PCA analysis. Computing the eigenvectors and sorting them

in descending order of eigenvalues allows us to create the principal components in order of significance. Discarding components with lesser eigenvalues, we can generate a matrix of feature vectors. The columns of this matrix are the eigenvectors of the principal components. Moreover, the i^{th} diagonal value of C_H is the variance of H along h_i or feature i.

The eigenvectors of the covariance matrix are the directions of the axes where there is the most variance called principal components. Eigenvalues are simply the coefficients attached to eigenvectors, which give the amount of variance carried in each principal component. The largest value of variance presents the most important feature of data and is associated with the highest signal-to-noise ratio (SNR). The ordered set of features is orthogonal to each other when the covariance matrix is a diagonal matrix.

Mathematically, PCA is computed using singular value decomposition (SVD) as Equation (6.9) [150].

$$H = U\Sigma V^T \tag{6.9}$$

where $V = \{\hat{v}_1, \hat{v}_2, ..., \hat{v}_m\}$ is orthonormal eigenvectors with corresponding eigenvalues of λ for symmetric matrix of $H^T H$ as $(H^T H)\hat{v}_i = \lambda_i \hat{v}_i$, $\sigma_i \equiv \sqrt{\lambda_i}$. Diagonal matrix Σ is the rank-ordered set of singular values σ_i, U is an $m \times m$ unitary matrix, $\hat{u}_i \equiv \frac{1}{\sigma_i} X\hat{v}_i$, $XV = U\Sigma$ [150]. Based on Equation (6.8), the principal components of H are the eigenvectors of HH^T. Having covariance matrix C_H, we can calculate the Pearson correlation coefficient (PCC) matrix R as Equation (6.10) [151].

$$R_{ij} = \frac{C_{ij}}{\sqrt{C_{ii}C_{jj}}} \tag{6.10}$$

PCC is used to recognize the degree of correlation between features. Meanwhile, PCA is utilized to quantify the relative importance of each dimension by measuring the variance along each principle component. In fact, we can characterize and select the important parts of our original dataset by choosing the orthogonal directions or the principal components along which data has a variance.

The explained variance is a statistical measure that expresses how much variation in a dataset can be assigned to each principal component (eigenvector) generated by the PCA method. Specifically, the explained variance is the total variance "explained" by each component, which allows ranking components following an order of importance [152]. The importance of each principal component is reflected by the magnitude of the corresponding values in the eigenvectors. A higher score of explained variance means that the component

is more significant. The explained variance for each eigenvector (principal component) $\hat{v}_i$ is expressed as the ratio of its eigenvalue λ_i to the sum of all eigenvalues as expressed by (6.11). The sum of all explained variance is equal to one and the explained variance for partly redundant features is small. A small eigenvalue can thus be used as a measure of redundancy.

$$Explained\ variance(\lambda_i) = \frac{\lambda_i}{\lambda_1 + \lambda_2 + ... + \lambda_m} \tag{6.11}$$

To answer the question of how many principal components to consider, we characterize it with performance metrics presented in Sections 6.5 and 6.6. Since PCA transforms the data in the directions that have the highest variance, it inherits the property of the highest information gain. In the next section, we describe the decision-tree algorithm that can be used to visualize features based on the highest information gain.

6.4 Decision-Tree and Entropy

In this section, we describe how to use entropy and information gain to build a decision-tree that allows to select the most relevant features in order to achieve better performance. In a decision-tree, all data elements are considered as root nodes. One feature is chosen and is split into two groups. Then another feature is chosen to split data further.

The recursive splitting continues until a stopping criterion is met. This criterion could be a predefined maximum depth of the tree, a minimum number of samples required to split a node or the purity or error of the resulting subsets falling below a threshold. Ideally, a good stopping criterion prevents overfitting and ensures that the tree does not become too complex. In this research, the stopping criterion that we used is the maximum depth of the tree. The last node is called the terminal/leaf node. The nodes between the root node and the leaf node are called decision nodes. The decision node contains a condition to split the data. The condition is defined by the feature index and the threshold value for that feature. The leaf node helps to predict the class to which a new data point belongs. A key question is to decide which feature to use for splitting the data. Algorithm 4 describes the Iterative Dichotomiser 3 (ID3) method that we used to build a decision-tree and to select features based on information gain values [153]. This algorithm learns which features and the corresponding threshold values to use to optimally split the data. The model selects the split that maximizes the information gain. Once the tree-building process is complete, the samples end up in the leaf nodes. For classification tasks, the majority class of samples in a leaf node becomes the

predicted class for any new sample that falls into that node [153].

Algorithm 4 Steps in the Iterative Dichotomiser 3 algorithm [153]

1 : It begins with the original set S (our dataset) as the root node
2 : It iterates through each feature of the set S and calculates entropy ($\mathscr{H}$) and information gain (I) of features
3 : It selects the feature that has the smallest entropy or the largest information gain
4 : The set S is then split by the selected feature to produce a subset of data
5 : The algorithm continues to recur on each subset

The main advantage of the decision-tree algorithm is that it uses different feature subsets and decision rules at different stages. In fact, it is able to capture descriptive decision-making knowledge from training sets. The goal of machine learning models is to use entropy to reduce uncertainty in order to build a decision-tree.

Shannon defined the entropy $\mathscr{H}$ of a discrete random variable X when it takes values from an alphabet χ that is distributed according to $p : \chi \rightarrow [0, 1]$ [154].

$$\mathscr{H}(X) = E[I(X)] = E[-log_b\ p(X)] \tag{6.12}$$

where E is the expected value operator, b is the base of the logarithm used and I is the information content of X that is a random variable. The entropy can be expressed as :

$$\mathscr{H}(X) = -\sum_{x\epsilon\chi} p(x) log_b\ p(x) \tag{6.13}$$

For binary data classification, entropy is expressed as :

$$\begin{aligned} \mathscr{H}(S) &= -\sum_{i=1}^{c} p_i\ log_2(p_i) \\ &= -p_1\ log_2(p_1) - p_2\ log_2(p_2) \end{aligned} \tag{6.14}$$

where p_i is the probability of class i in our dataset S and c is the number of classes which is 2 in our case, $p_1 = \frac{N_{PeopleWalkAndStand}}{N_{Total}}$ and $p_2 = \frac{N_{PeopleRun}}{N_{Total}}$. As described in Chapter 4, we detect human activities with higher and lower movement energy, resulting in the creation of two distinct classes. The first class is related to people who are standing and walking, and the second class is related to people who are running and have a higher speed and movement energy. Information gain is the reduction in entropy produced by transforming a dataset and it is often used when training decision-trees. It is calculated by comparing the entropy of the dataset before and after a transformation and it is the basic criterion to decide whether

a feature should be used to split a node or not [155]. It is also called mutual information $I(Y;X)$ in information theory.

$$I(Y;X) = \mathscr{H}(Y) - \mathscr{H}(Y|X) \tag{6.15}$$

As Equation (6.15) describes, we simply subtract the entropy of Y given X from the entropy of just Y to calculate the reduction of uncertainty about Y given an additional information X about Y [156]. In our case, when we build a decision-tree, we want to pick up the feature that will give us the greatest reduction of uncertainty. This decision-tree is the one offering the largest information gain.

6.5 Characterization Results

Using PCA to reduce dimensionality and computational complexity, we need to decide which features and how many features to use. Table 6.1 illustrates the most important features in decreasing order based on the explained percentage of contributed variance. This table shows that the chirp rate is the most important feature among the four features considered in the present chapter (chirp rate, delta band power, spectral flux, and frequency of movement).

TABLE 6.1 The most important features using PCA.

PC1	**chirp-rate**
PC2	**delta-band-power**
PC3	**spectral-flux**
PC4	**frequency-of-movement**

We use "Scree/Elbow plot" [157] to decide how many features to use. Using this method, we plot the percentage of explained variance versus the number of features. As Figure 6.5 shows, the first feature explains over 64% of the variance. And the next ones explain a smaller percentage of the variance. This graph shows that at the 4^{th} feature, the variance decays to an almost insignificant level. As a result, we use the first three features to analyze the data. The total variance using these three components is 91%. The corresponding PCA explained variance ratio for those three features is : $[0.64, 0.16, 0.11]$. PC1 explains 64% of variability, PC2 16% and PC3 11%.

Another way to determine how many features to choose is to define a threshold like 85% or 95% that PCA explained-variance-ratio requires to meet. Figure 6.6 illustrates PCA

visualization for all four features. In this figure, blue and red colors correspond to the class 0 and 1, respectively. It illustrates the variance of each principal component using a covariance matrix and eigenvectors and ranks them by their relevance (explained variance/eigenvalues). Moreover, it shows the features required to easily separate two different classes. It shows that PC1 contains the highest explained variance and is the most important feature in separating two classes. The combination of PC1 with other principal components adds more information and increases the performance scores discussed in Section 6.6.

Table 6.2 presents the Pearson correlation coefficients for four features of chirp rate, delta band power, spectral flux, and frequency of movement. Pearson correlation coefficient measures the strength and direction of the linear relationship between two quantitative variables. Specifically, we can test if there is a significant relationship between the two features. If the non-diagonal elements are zero, it indicates that those features are orthogonal. This table shows that the features are partially orthogonal. It is found that the most significant linear relationship exists between the chirp rate and the frequency of movement. The frequency of movement is a subset of the chirp rate and is a partly redundant feature. We use a covariance matrix when the variability of the data is important. We can use a correlation matrix when the angle of basis and the orthogonality is important. The angle of basis typically refers to the angle between two basis vectors in a vector space. A basis is a set of linearly independent vectors that span the entire space, and the angle between basis vectors can provide insights into the geometry and orientation of the vector space. If the basis vectors are orthogonal, meaning the angle between them is 90

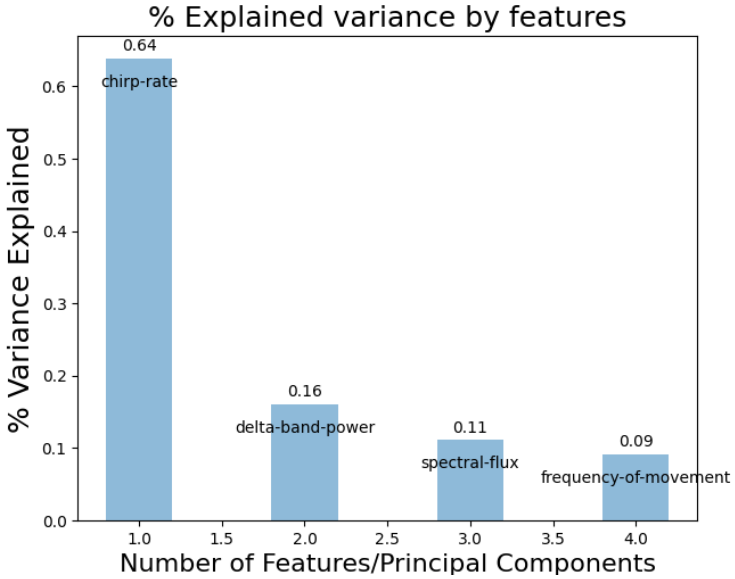


FIGURE 6.5 Explained variance of the considered features.

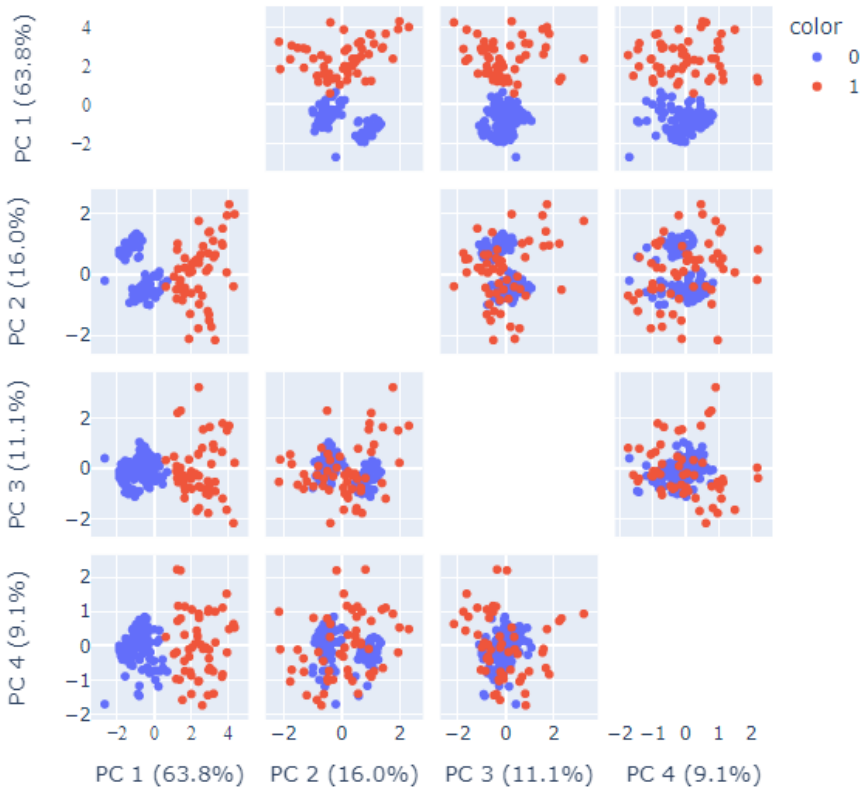


FIGURE 6.6 Graphical representation of how data points are separated according to the most important PCA components in order of decreasing importance : 1) chirp rate, 2) delta band power, 3) spectral flux, and 4) frequency of movement.

degrees and they are linearly independent.

TABLE 6.2 Pearson correlation coefficients.

	Chirp rate	Delta band power	Spectral flux	f movement
Chirp rate	**1**	**0.54**	**0.538**	**0.598**
Delta band power	**0.54**	**1**	**0.411**	**0.418**
Spectral flux	**0.538**	**0.411**	**1**	**0.57**
f movement	**0.598**	**0.418**	**0.57**	**1**

The Python Scikit-learn package was used to build and optimize the decision-tree classifier. Figure 6.7 illustrates the ID3 algorithm to build the decision-tree before applying PCA. In each node, an entropy greater than zero implies that samples contained within that node belong to different classes. An entropy value of zero means that the node is pure and only a single class of samples exists. The sample value in each node indicates the number of samples in the corresponding feature. The value list represents how many samples at the given node fall into each category. The first element of the list shows the number of samples that belong to the class 0, and the second element of the list shows the

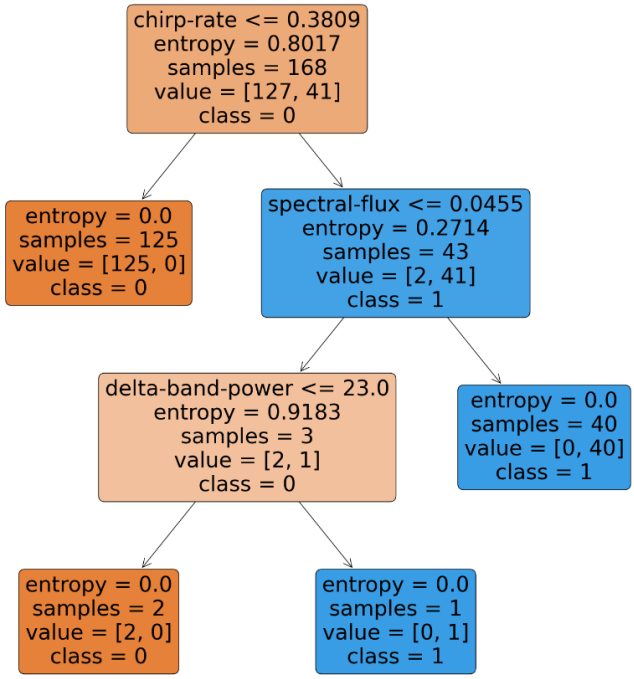


FIGURE 6.7 Decision-tree algorithm based on entropy for features of chirp rate, delta band power, spectral flux, and frequency of movement before applying PCA.

number of samples that belong to the class 1. The class value shows the prediction of a given node and it is determined from the value list. This method uses color to indicate the majority of the class and the extremity of values for splitting and creating multiple output branches. The leaf node indicates the predicted value. This value is the majority class of the leaf node. It determines the class of a new data point if the data point ends up in this particular leaf node [158].

The root node is determined by the feature with a condition that produces a split with the purest subsets. In fact, the feature with the highest information gain is considered a root node. To calculate the information gain of each feature in a particular decision node, the Python Scikit-learn package with its decision-tree classifier calculates the entropy of the whole data set and the entropy of the individual feature values using Equations (6.14) and (6.15). Once the root node is selected, the same process is repeated for the other nodes in the tree. The algorithm looks at the subsets and measures their impurity by estimating the information gain and considering multiple thresholds. It determines the threshold that creates the best split for the given feature. This is repeated for all the features in the training set. The samples on the leaf node belong to only one class.

The ID3 algorithm loops through all four features to find the best split. The decision-

tree algorithm aims to find the best features and thresholds for splitting the data at each internal node. It evaluates splitting criteria such as entropy for classification tasks. The goal is to minimize the impurity or error in the resulting subsets after the split. As we see in Figure 6.7, the decision-tree-classifier function of the Scikit-learn package gives the criterion equal to chirp-rate $\leq$ 0.3809 to split data for the first step. The corresponding entropy of this criterion is 0.802. The chirp rate is selected as the first node, because it creates the most information gain among the other features and creates the pure left leaf node with a condition of chirp-rate $\leq$ 0.3809. Now the algorithm should split data from the right branch with the condition of spectral-flux $\leq$ 0.0455 to the left and to the right. In this way, the right child is a pure leaf node. The algorithm splits data further with those conditions to build the decision-tree. Finally, delta band power with a condition of delta-band-power $\leq$ 23.0 creates two pure leaf nodes that do not require more split. The values between nodes are called split points. The separation of one class from the others is performed by those values. The values of condition and split points are chosen by the algorithm to include the most information gain. In this figure, the depth of a tree is three.

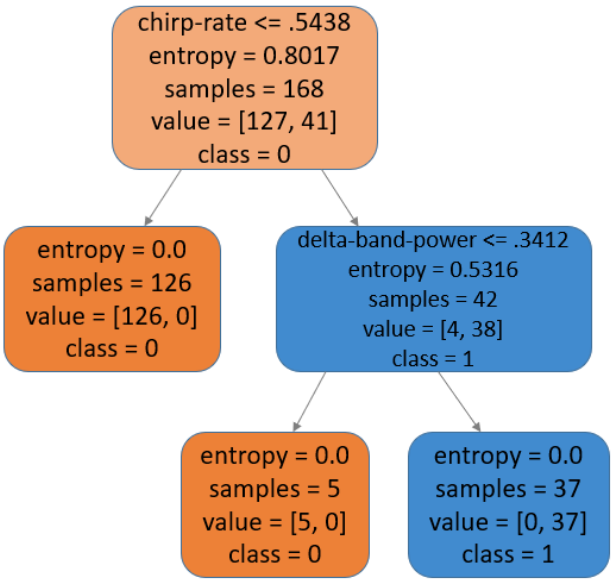


FIGURE 6.8 Decision-tree algorithm based on entropy for features of chirp rate, delta band power, spectral flux, and frequency of movement after applying PCA.

Figure 6.8 illustrates the results of the ID3 algorithm used to build the decision-tree after applying PCA to those four features. Applying PCA before the decision-tree algorithm allows effective feature characterization. PCA is used to select features based on the highest eigenvalues, which leads to minimizing the total number of features and amplify the classification performance by transforming data to the directions that have the highest

variability. Additionally, it makes the decision-tree learn faster and achieve a higher accuracy with the least number of features. Since decision-tree is applied on the aligned data with the highest variability, after splitting the data with a condition of chirp-rate $\leq$ 0.544, delta band power creates the most information gain and two pure leaf nodes. Therefore, it reduces the complexity of computation from three features to two.

6.6 Selection of the Most Important Features

To ensure that we choose the right number of features, we cross-validate the classification performance providing the confusion matrix as Tables 6.3, 6.4, 6.5 and 6.6. The computation and prediction are performed using an Intel Xeon $X86$ processor. Python libraries of Scipy, Numpy, and Sklearn are used for estimating and characterizing important features. Among the total 3400 available samples of different human activities using the WiAR dataset, we estimated running, standing, and walking activities. The features of the chirp rate, the delta band power, the spectral flux, and the frequency of movement with different combinations are used as inputs to the SVM classifier based on supervised machine learning. A 5-fold cross-validation approach was applied, where 80% of the data is used as a training set and 20% is used as a test set.

Tables 6.3, 6.4, and 6.5 report SVM classifier confusion matrix for three use cases of 1) chirp rate, 2) chirp rate and delta band power), and 3) chirp rate, delta band power and spectral flux. Norm = True presents the normalized confusion matrix in the range of [0, 1]. The rows of the table show the true classes of the human activities , whereas the columns show the predicted classes as estimated by the SVM linear-kernel classifier. An ideal result is to obtain near 100% classification on the diagonal of the confusion matrices, whereas near zero on the non-diagonal elements related to misclassification results.

Table 6.6 illustrates the performance of the SVM model in terms of the classification metrics for the test set. The performance metrics include accuracy, precision, recall, and F1-score based on the following equations [159].

$$Accuracy = (TP + TN)/(TP + FP + TN + FN) \tag{6.16}$$

$$Precision = TP/(TP + FP) \tag{6.17}$$

$$Recall = TP/(TP + FN) \tag{6.18}$$

TABLE 6.3 SVM classifier confusion matrix for the training set and the test set.

		chirp-rate							
		Training set, Norm=None		Training set, Norm=True		Test set, Norm=None		Test set, Norm=True	
True label	0	TP :118	FN :6	TP :0.95	FN :0.048	TP :30	FN :2	TP :0.94	FN :0.062
	1	FP :12	TN :32	FP :0.27	TN :0.73	FP :1	TN :10	FP :0.091	TN :0.91
		0	1	0	1	0	1	0	1
		Predicted label				Predicted label			

TABLE 6.4 SVM classifier confusion matrix for the training set and the test set.

		chirp-rate and delta-band-power							
		Training set, Norm=None		Training set, Norm=True		Test set, Norm=None		Test set, Norm=True	
True label	0	TP :122	FN :0	TP :1	FN :0	TP :32	FN :1	TP :0.97	FN :0.03
	1	FP :7	TN :39	FP :0.15	TN :0.85	FP :0	TN :10	FP :0	TN :1
		0	1	0	1	0	1	0	1
		Predicted label				Predicted label			

TABLE 6.5 SVM classifier confusion matrix considering important features for the training set and the test set.

		chirp-rate, delta-band-power and spectral-flux							
		Training set, Norm=None		Training set, Norm=True		Test set, Norm=None		Test set, Norm=True	
True label	0	TP :127	FN :1	TP :0.99	FN :0.0078	TP :32	FN :0	TP :1	FN :0
	1	FP :1	TN :43	FP :0.03	TN :0.97	FP :1	TN :10	FP :0.046	TN :0.95
		0	1	0	1	0	1	0	1
		Predicted label				Predicted label			

TABLE 6.6 Classification metrics.

	Test set					Training set
	chirp-rate	chirp-rate, Δband-power	Δband-power, f-mov	chirp-rate, Δband-power, spectral-flux	all-4-features	all-4-features
Accuracy	**0.93**	**0.98**	**0.98**	**0.98**	**0.72**	**0.98**
Precision	**0.83**	**1**	**0.91**	**1**	**0.64**	**1**
Recall	**0.91**	**0.9**	**1**	**0.91**	**0.69**	**0.97**
F1-score	**0.87**	**0.95**	**0.95**	**0.98**	**0.66**	**0.98**

$$F1_{score} = 2\frac{Precision \times Recall}{Precision + Recall} \tag{6.19}$$

Confusion matrix metrics of TP, TN, FP, and FN are true positive, true negative, false positive, and false negative, respectively [27]. Equation (6.16) represents the accuracy of the model. It is the ratio of TP and TN to all positive and negative observations. Equation (6.17) describes the precision metric which measures the proportion of positively predicted labels that are correct. Equation (6.18) represents the recall metric which measures the extent to which the positives are correctly predicted. This equation shows that FN impacts the recall score. The lower FN results in the higher recall score. The recall score, also known as the

true positive rate or sensitivity, primarily measures the model's ability to correctly identify positive samples while potentially sacrificing the accuracy of negative sample identification. It focuses on evaluating the extent to which false negatives are correctly predicted. Equation (6.19) represents F1-score which is the harmonic mean of precision and recall.

According to the results in Tables 6.3 and 6.4, using only chirp rate as a feature, we achieve $TP = 94\%$, $TN = 91\%$, accuracy = 93%, precision = 83% and F1-score = 87%. When we add the delta band power, the second most important feature, it improves the performance to $TP = 97\%$, $TN = 100\%$, accuracy = 98%, precision = 100%, and F1-score = 95% by giving more information gain. Based on Tables 6.5 and 6.6, when we consider the set of three most important features (1- chirp rate, 2- delta band power, and 3- spectral flux), the information gain increases more and it improves the performance to $TP = 100\%$, F1-score = 98%. Therefore, we obtain the most information gain, the highest performance, and the best scores by adding the second and third most important features.

To compare the proposed method with the results reported in Chapter 4, Table 6.6 shows that the classification metrics and performance obtained with the delta band power and frequency of movement are roughly as high as the ones for chirp rate and delta band power. However, based on Figure 6.5 and Table 6.2, frequency of movement results in the lowest explained variance and the most correlation with chirp rate feature. The reason is that the frequency of movement is a subset of the chirp rate. Additional parameters such as duration of activity, rate of frequency changes, minimum and maximum frequency of movement are extracted and estimated using chirp rate as a feature which makes the model robust and proper for non-deterministic environments. The lesser selection of features makes the lesser consumption of resources and computational time.

Based on Table 6.6, considering all four features causes the overfitting problem, higher computational complexity, and higher latency. Overfitting occurs when a model performance on the training dataset is improved at the cost of worse performance on data not seen during the training, such as a test set or new data. This table shows that using all four features in the training set results in high performance, whereas for the test set, we obtain F1-score = 66% and precision = 64%. In fact, it is the place where the algorithm is broken down.

Characterizing features by applying PCA before the decision-tree algorithm allows the selection of the most important features based on the highest eigenvalues and information gain. Evaluating classification metrics helps us understand the strengths and limitations of our model when making predictions in new situations. Based on the results obtained in this

section, chirp rate contributes the highest variance and information gain among the other features.

In terms of classification metrics, there is a trade-off between false positives and false negatives. That is why recall and precision have to be considered together. If the goal is to reduce false negatives, a model has to be used with high precision. On the contrary, if the goal is to decrease false positives, a model with a high recall is employed. Moreover, a low recall score specifies that the model is not good at identifying TP and FN examples. Based on Table 6.6, if precision = 83% and F1-score = 87% are considered, estimating chirp rate as a feature meets the requirements. If high precision is required, we should select the chirp rate and the delta band power features. If high recall is required, using the delta band power and the frequency of movement features is the best choice. If the highest F1-score is required, we should select all three most important features.

6.7 Characterizing and Selecting Important Features; Comparison with Z. Wang's Dataset [112]

In Chapter 4, we used Z. Wang et al. [112] dataset to detect falling and walking activities. Since falling activity occurs in a shorter time in comparison with walking activity, chirp rate is extracted as a robust feature to address detecting activities with different time durations and different rates of motion. To answer the question of how many features should be used for effective human fall detection, the explained percentage of contributed variance is calculated for each feature. Figure 6.9 illustrates the most important features in descending order based on the explained variance versus features. As this figure shows, the chirp rate, the delta band power and the spectral flux explain roughly 68%, 15%, 10% of the variance, respectively. The frequency of movement feature explains a smaller percentage of the variance. This graph shows that at the 4^{th} feature, the variance decays to an almost insignificant level, roughly 0.1 maximum variance. As a result, we use the first three features to detect the falling activity.

Decision-tree algorithm and the PCA are used as the other way to determine how many features to choose. Figure 6.10 illustrates decision-tree algorithm based on entropy for features of chirp rate, delta band power, spectral flux, and frequency of movement before applying PCA using Z. Wang's dataset [112]. Python Scikit-learn package [160] is used to build and optimize the decision-tree algorithm. The decision-tree-classifier function of the Scikit-learn package is based on the ID3 algorithm that gives the criterion defined to be to chirp-rate

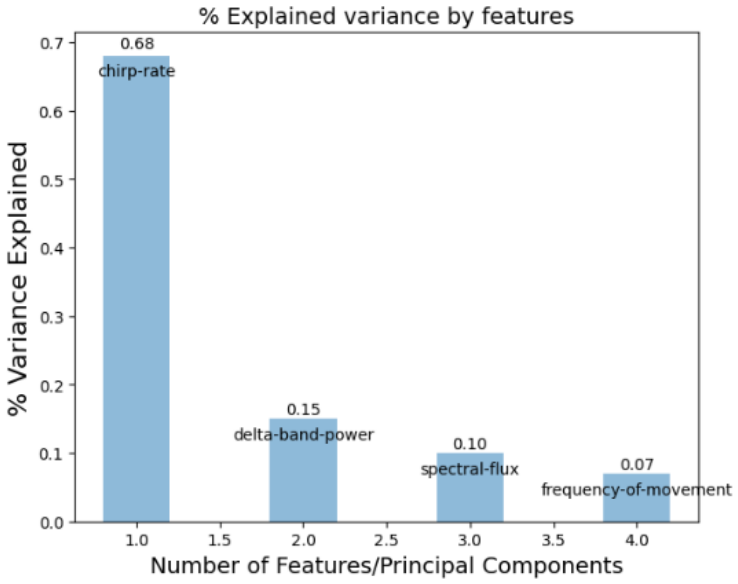


FIGURE 6.9 Explained variance of the considered features using Z. Wang et al. [112] dataset.

≤ 0.27 to split data for the root node. The entropy that corresponds to this criterion is equal to 0.82. The right-most leaf node is created with this condition. Now the algorithm should split data from the left branch with the condition of delta-band-power ≤ 0.34 to the left and to the right. In this way, the right child is a pure leaf node. The algorithm splits data further with those conditions to build the decision-tree. Finally, the spectral flux with a condition of spectral-flux ≤ 0.245 creates two pure leaf nodes that do not require more split. In this figure, the depth of a tree is three.

Figure 6.11 illustrates the results of the ID3 algorithm used to build a decision-tree after applying PCA to those four features. Applying PCA before the decision-tree algorithm allows effective feature characterization. Since the decision-tree is applied to the aligned data with the highest variability, after splitting the data with a condition of chirp-rate ≤ 0.29, delta band power creates the most information gain and two pure leaf nodes. Therefore, it reduces the number of features from three to two.

TABLE 6.7 SVM classifier confusion matrix for the training set and the test set using Z. Wang's dataset [112].

		chirp-rate							
		Training set, Norm=None		Training set, Norm=True		Test set, Norm=None		Test set, Norm=True	
True label	0	TP :56	FN :3	TP :0.95	FN :0.051	TP :15	FN :2	TP :0.88	FN :0.12
	1	FP :5	TN :156	FP :0.031	TN :0.97	FP :0	TN :38	FP :0	TN :1
		0	1	0	1	0	1	0	1
		Predicted label				Predicted label			

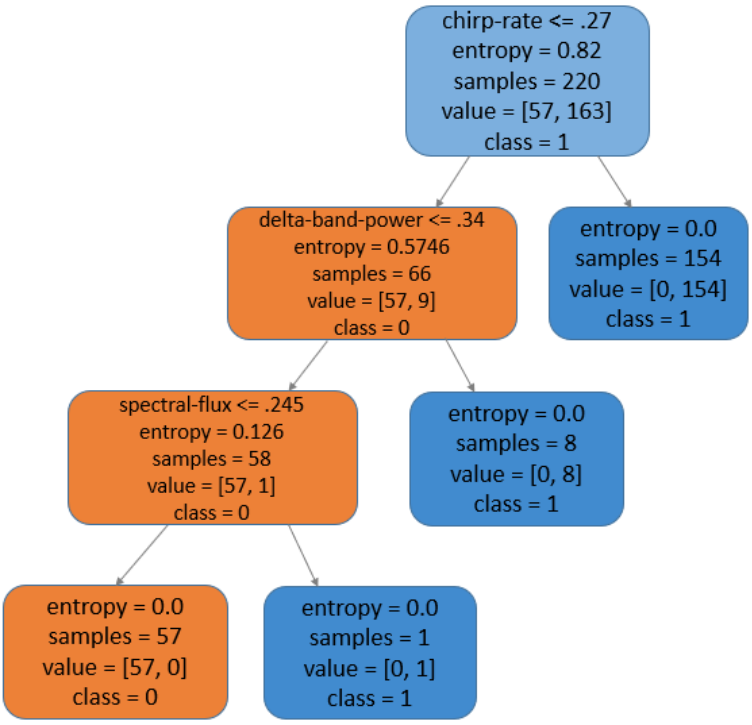


FIGURE 6.10 Decision-tree algorithm based on entropy for features of chirp rate, delta band power, spectral flux, and frequency of movement before applying PCA using Z. Wang's dataset [112].

TABLE 6.8 SVM classifier confusion matrix considering important features for the training set and the test set using Z. Wang's dataset [112].

		chirp-rate, delta-band-power							
		Training set, Norm=None		Training set, Norm=True		Test set, Norm=None		Test set, Norm=True	
True label	0	TP :51	FN :1	TP :0.98	FN :0.019	TP :18	FN :1	TP :0.95	FN :0.053
	1	FP :1	TN :167	FP :0.006	TN :0.99	FP :0	TN :36	FP :0	TN :1
		0	1	0	1	0	1	0	1
		Predicted label				Predicted label			

TABLE 6.9 Classification metrics using Z. Wang's dataset [112].

	Test set	
	chirp-rate	chirp-rate, Δband-power
Accuracy	0.931	0.978
Precision	0.85	0.94
Recall	0.91	0.99
F1-score	0.87	0.97

According to the results on Tables 6.7, 6.8 and 6.9, using only chirp rate as a feature, we

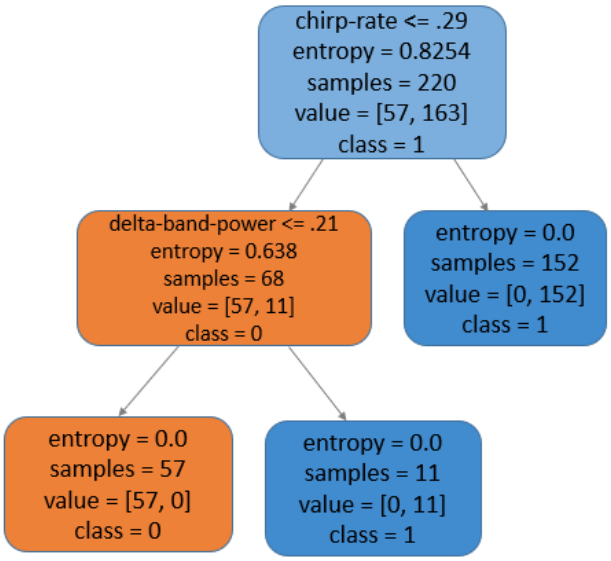


FIGURE 6.11 Decision-tree algorithm based on entropy for features of chirp rate, delta band power, spectral flux, and frequency of movement after applying PCA using Z. Wang's dataset [112].

achieve $TP = 88\%$, $TN = 100\%$, accuracy $= 93\%$, recall $= 91\%$ and F1-score $= 87\%$. When we add the second important feature of delta band power, it improves the performance to $TP = 95\%$, accuracy $= 97.8\%$, recall $= 99\%$ and F1-score $= 97\%$ by giving more information. Therefore, we obtain the most information gain, the highest performance and scores by adding a second important feature.

6.8 Summary

Characterizing features using the PCA method before the decision-tree algorithm, we conclude that the chirp rate explains the highest variability and produces the highest information gain. Moreover, measuring the SVM classification metrics confirms that using only chirp rate as a feature, we achieve $TP = 94\%$, $TN = 91\%$, precision $= 83\%$ and F1-score $= 87\%$. Considering delta band power as an additional feature adds more information and achieves a higher performance and scores to $TP = 97\%$, $TN = 95\%$, precision $= 100\%$, and F1-score $= 95\%$. Adding each principal component increases information and improves the performance scores. The Pearson correlation coefficient confirms that the information obtained by the frequency of movement feature is a subset of that obtained by the chirp feature.

There is a clear trade-off between latency, accuracy, and feature selection. Thus, we recommended a feature selection based on observed variability, information gain, and correlation between features. The results show that by using PCA we reduced the number of features from four to three. By applying the decision-tree algorithm after PCA, the dimension of features was decreased to two which allows for reducing the computational time and the complexity.

CHAPTER 7

GENERAL DISCUSSION

Human activity recognition can indeed be a vital component of smart city platforms. By accurately detecting and understanding human activities within an urban environment, smart city data processing systems can gain valuable insights that enable better decision-making, resource allocation, and the optimization of various services. As an enabling step toward detecting human activities, we set the main goal of this research to explore methods that could be used to track effectively human movements using measured characteristics of reflected RF signals.

This led us to propose a data processing framework for human activity recognition (HAR) in Chapter 3. In this framework, 1) human activity was characterized based on PCA and spectrogram analysis, 2) CSI data was mapped to different kinds of human activities, 3) different human activities were categorized by applying a time series classification algorithm based on the LSTM-RNN model, 4) a method was proposed to distinguish walking and running speeds through CSI changes. To validate the proposed methods and framework, we used the public domain WiAR dataset [12] that was generated using an Intel 5300 wireless NIC card and its associated CSI tool. We also confirmed with Z. Wang's dataset [112] that the proposed methods are generally applicable for HAR based on CSIs. These methods provide a versatile and effective approach to accurately identify and categorize human activities using the information derived from CSIs.

7.1 Interpretation of Results

The framework first proposed in Chapter 3 based on LSTM-RNN produced promising results, with an accuracy detection of 94% $\pm$1.4% when trying to predict true human activities available as labels to measured CSI streams. Building on these promising developments, a method was proposed to automatically detect some specific human activities based on the maximum variance of movement energy within time-series data to detect higher movement-energy activities in Chapter 4. The proposed method is based on estimating a threshold adaptively to track features in a spectrogram using two sliding windows to detect dynamically

the maximum variance of movement energy within time-series data. A first sliding window was defined over time to obtain the power spectral density (PSD) of WiFi signals and a second sliding window was obtained over frequency to calculate features of the PSD signal. This led to the estimation of PSD features, expressed as delta band power, which had never been applied before for the classification of human activities.

In another method reported in Chapter 4 [27] that was proposed before the framework of Chapter 3 [17], using adaptive-threshold estimation, it was shown that selecting the right features is vital to classification and to the effectiveness of machine learning algorithms. Adding the frequency of movement feature in addition to the delta band power increased the true negative (TN) classification performance from 38% to 95% for the training set, and from 27% to 91% for the test set.

One of the limitations of the methods and associated results reported in Chapters 3 and 4 was that an imbalanced WiFi-based dataset was used, which affected the performance of the proposed machine learning algorithms. Thus, in Chapter 5, we proposed a method to efficiently classify imbalanced datasets. Then the impact on classification performance of the imbalance observed in the WiAR dataset in relation to feature selection for a proposed machine-learning-based classification algorithm was explored. In Chapter 5, it was also shown that attempting to handle the imbalanced class problem without considering the proper features does not improve the classifier's performance. By contrast, when considering the delta band power and the frequency of movement features before the proposed imbalanced classification method, which are the best features that could be used in the proposed method with the available dataset, the ROCAUC increased from 64% to 98% and the F1-score from 46% to 98%.

The other limitation of this research was that there are several sources of uncertainty and factors in a practical system utilizing WiFi transceivers that could affect the quality and reliability of wireless communication. These sources of uncertainty include 1) multi-path propagation, 2) physical obstructions such as walls, buildings, and other structures that can block or attenuate wireless signals, 3) signal attenuation, especially at longer distances, 4) environmental factors like humidity, temperature, and atmospheric conditions that can affect signal propagation, and finally 5) the movement of people, vehicles, and objects within the environment.

To mitigate the effects of uncertainty and improve the reliability of wireless communication in real systems, in Chapter 6, the chirp parameters were estimated and

extracted as a robust feature including temporal and spectral information to effectively detect human activity. This feature has never been applied to the classification of human activities. Some other spectral and temporal STFT features were estimated as well. Characterization and classification results proved that the chirp rate feature has high detection capability for a wide range of human movement speeds from a very low speed of walking to a very high speed of running activity with any time duration.

Additional research questions that were explored relate to 1) determining how many features to use, 2) what number of dimensions to use in the PCA, and 3) assessing the impact of correlated and uncorrelated features on performance. To answer those questions, features were characterized by applying the PCA method before the decision-tree algorithm to select the features in order of importance and to obtain the best possible performance. This method allowed us to identify the number of dimensions and their priority in analyzing data. Finally, based on the obtained results, the most effective features were recommended for the explored detection task according to observed variability, information gain and correlation between features.

7.2 Comparison to Existing Literature

Recent research has shown the rapid development of wireless sensing and machine learning-based methods due to their potential to enable various applications and advances in technology.

7.2.1 Performance Comparison ; Human Activity Recognition Using Andrii and Z. Wang's Datasets [125][112]

In Chapter 3, Section 3.10, the performance of our classification method on the WiAR dataset was compared with those obtained on another CSI-based dataset provided by Andrii et al. [125]. The results showed that the average accuracy for the training set and the test set of Andrii et al. dataset [125] is 91.3% and 51.4%, respectively. The main reason for the low accuracy of the test set was that this dataset was highly imbalanced and there was insufficient amount of data. The model favored the majority class and performed poorly on the minority class. In this research, the random oversampling technique was applied to handle this imbalanced dataset. It improved the average accuracy from 51.4% to 90.4%. However, even better accuracy detection of 94% $\pm$1.4% was achieved with the WiAR dataset.

In Chapter 4, Section 4.5, we detected falling and walking activities using another dataset provided by Z. Wang et al. [112]. The adaptive threshold was obtained by the decision boundary of the SVM classifier. The results showed that extracting two features of delta band power and frequency of movement separates two classes of walking and falling properly by obtaining $TP = 80\%$ and $TN = 100\%$ for the test set. In Chapter 5, Section 5.4, we applied a proposed ensemble method to improve the ROCAUC performance from 65% to 96% and the F1-score from 60% to 92% to detect falling activity. In Chapter 6, Section 6.7, we characterized and selected important features to detect human falling activity using Z. Wang et al. [112] dataset. The utilization of human fall detection for intelligent daily activity monitoring in modern healthcare applications has become increasingly popular, especially for monitoring elderly individuals.

7.2.2 Performance Comparison of Human Activity Recognition Using Deep Learning Methods

In Chapter 3, Section 3.11, a comparative analysis of the performance with different deep-learning-based algorithms was discussed for human activity detection using the WiAR dataset. In this section, it was concluded that the CNN-based model is designed for spatial feature extraction from grid-like data, such as images. Although CNNs are commonly used in computer vision, they require substantial computing resources to perform motion detection. KNN models consider local similarity but struggle with CSI's high-dimensional and sequential nature. The Multilayer Perceptron (MLP) model is commonly used for classification tasks. However, when applied to sequential activities with CSI, it exhibits limitations in capturing temporal dependencies. By contrast, Long Short Term Memory (LSTM) networks, a class of recurrent neural networks (RNN) are well-suited for time-series data analysis. They can capture long-term dependencies in the data, while remembering and utilizing the context over time, thus making them suitable for tasks involving sequences and temporal patterns.

7.2.3 Disadvantages of Some Feature Extraction Methods in the Literature on Human Activity Detection

Authors in [26] used some preprocessing methods for WiFi-based crowd estimation by extracting 13 spectral descriptors. In total, 13 statistical features were considered : 1) mean, 2) standard deviation, 3) centroid, 4) spread factors, 5-6) 1^{st} and 2^{nd} order moments, 7) mean sigma ratio, 8) number of peaks, 9) number of valleys, 10) maximum width between

peaks, 11) maximum width between valleys, and 12-13) width/height between maximum peak and minimum valley are used to extract channel information from WiFi signals reflected by the presence of human bodies. However, estimating all those statistics is a computationally expensive process. Additionally, since there is a source of uncertainty in real systems due to the stochastic nature of the environment, this technique is not as robust and reliable as extracting chirp parameters, which is robust for a non-deterministic environment.

Hidden Markov Model (HMM) based feature extraction from incoming CSI signals was proposed by W. Wang et al. [12]. The drawback of this technique is that it is unable to capture complex movements with a sequence of micro-activities. HMMs assume that the future state of the system depends only on its current state and is independent of its previous states. This assumption may not hold for complex human activities, where the current action may depend on a history of prior actions and environmental conditions. Additionally, it requires determining the appropriate number of hidden states and setting initial parameters, which is a challenging task.

7.2.4 Features Selection

Both the chirp rate and the delta band power tracking method for human activity detection are considered in this research for the first time. However, the chirp rate was previously used in radar systems for detecting SAR images, but it was never used in a context similar to or comparable to the applications explored in this thesis.

7.3 Addressing Research Objectives/Hypotheses

The degree to which this thesis addresses the formulated research objectives/hypotheses can be summarized as follows :

1. **Propose a processing framework to detect human activity**
   The human activity recognition (HAR) data processing framework of Chapter 3 meets this first objective. Our framework was inspired by the MAPE-K loop method [19] that was proposed to analyze human motions.
2. **Propose a method based on estimating an adaptive threshold by tracking spectrogram in time and frequency domains to detect automatically high movement-energy human activities**
   This objective led to the work reported in Chapter 4. In this chapter, the impact of

feature selection on classification performance was explored. This chapter also raised the question of how many features should be used for effective human activity detection. Chapter 6 is an attempt to answer this question.

3. **Propose a method to efficiently classify imbalanced datasets and explore the impact on classification performance of the imbalanced dataset in relation to feature selection**
   In Chapter 5, an ensemble method combining the bagging technique with SMOTE and a support vector machine (SVM) method was proposed to balance the dataset. In addition to balancing the imbalanced dataset to improve the representation of the minority class, the impact of feature selection was explored to propose more effective machine learning algorithms.
4. **Propose how to estimate the chirp rate that was found to be quite effective to analyze incoming CSI data in the public domain WiAR dataset**
   In Chapter 6, chirp parameters were extracted to effectively detect human motions. This feature has never been applied to the classification of human activities.
5. **Apply the principal component analysis (PCA) before the decision-tree algorithm to efficiently characterize features**
   One of the challenges in this research was to explore the impact of correlated and uncorrelated features in terms of performance. To tackle this challenge, in Chapter 6, the PCA was applied before the decision-tree algorithm to automate the feature decision-making process and to improve the model accuracy. Finally, feature selection based on observed variability, information gain, and correlation between features was recommended. By applying the decision-tree algorithm after PCA, the data was transformed to the directions that offer the highest information gains. It made the decision-tree learn faster and achieve a higher accuracy with the least number of features.

7.4 Future Directions

In this thesis, HAR data processing framework was proposed to detect human activities using a public domain WiAR dataset based on $IEEE$ 802.11n CSI measurements. Human activities were inferred by analyzing the changes in WiFi signals caused by human movements. Additionally, a practical approach was proposed to characterize and select features for human activity detection. The following summarizes key points to consider with respect to WiFi-based human motion detection :

— Identifying congestion areas : In WiFi-based smart public area subject to congestion, this issue could be addressed by : 1) channel selection and configuration to improve performance, 2) load balancing to distribute the traffic evenly across multiple access points to prevent congestion 3) traffic monitoring and analysis to identify areas of congestion and to optimize network resources.
— Privacy protection : In this research, the proposed detection method is based on reflected signals emitted by the WiFi system. It is not related to any device that belongs to the detected persons such as a smartphone. It only detects the presence and the movement of humans in the environment. Therefore, by itself, it can hardly introduce any privacy issues.
— Real-world application scenario : Validation of the proposed concept with a real-world system would be very interesting and highly valuable. However, the present work is based on and was enabled by the public domain WiAR dataset, and thus validation in a real-world is beyond its scope.

Since the amplitude of CSI data was used for human activity detection in this research, it is worth receiving phase information of the incoming CSI data and estimating the angle of arrival (AoA) and time of flight (ToF) to estimate the direction of movement in WiFi-based human activity detection. Here are key points to detect the direction of movement in WiFi-based human activity detection :

— Estimation of AoA : The WiFi signals received from different access points could provide the direction of the incoming signal. AoA can be estimated by analyzing the variations in signal strength across multiple access points to infer the direction of movement of people.
— Triangulation technique : Tracking the position and the movement of people could be estimated by comparing the signal strengths and ToF of WiFi signals received at different access points using the triangulation technique.

CHAPTER 8

CONCLUSIONS AND RECOMMENDATIONS

8.1 Summary of Main Findings

In this research, a HAR data processing framework was proposed to detect human activities based on a deep learning approach. Datasets based on $IEEE$ 802.11n CSI measurements were used for estimation and activity detection. The processing framework was inspired by the MAPE-K loop to analyze human motions. In this framework, the challenge of mapping channel state information (CSI) data to different kinds of human activities was addressed. Path length change rate and frequency of movement based on received multi-path signals were analyzed. Doppler shift was used to detect walking and running speed from CSI changes. By calculating the speed of movement a method to detect human activities, based on adaptive thresholds was proposed to track spectrograms using two sliding windows to detect dynamically the maximum variance of movement energy within time-series data. The impact of the feature selection and the imbalanced dataset on the performance of the classifier was explored as well. An ensemble method combining the bagging technique with SMOTE and a support vector machine (SVM) method was proposed to balance the dataset. This method creates a set of classifiers to distribute the dataset and each subset learns from the resampled balanced data and uses different SVM decision thresholds to automatically shift the decision threshold to the majority class. It was concluded that without selecting proper features, balancing the dataset does not affect the performance of the classifier. However, handling the imbalanced dataset after selecting proper features improves the performance.

The performance of the proposed method was evaluated using two ensemble classifier algorithms, namely, the SVM SMOTE Bagging and the Borderline SMOTE Bagging. The results showed that although the SVM SMOTE Bagging and the Borderline SMOTE Bagging techniques both handle the class imbalance problem by oversampling the minority class, the SVM SMOTE Bagging classifier achieved a better performance. This technique inherits the strengths of SVM in handling decision boundaries by considering the support vectors and their neighbors. It creates synthetic samples aligned with the SVM decision boundary. Additionally,

it decreases overfitting through ensemble learning. Then, a non-deterministic characteristic of the environment was explored.

Since the main goal of this research was to explore an effective tracking algorithm in a non-deterministic environment, the chirp rate was considered as a feature in an environment subject to a great deal of uncertainty. Besides considering the chirp rate as a feature, some other spectral and temporal STFT features were taken into account to obtain the best possible performance. Then a research question was raised concerning the number of dimensions required to analyze the data most effectively. To tackle this challenge, the PCA method was applied before the decision-tree algorithm to improve the classification performance by transforming data into the directions characterized by the highest variability and information gain.

It has been concluded that the chirp rate explains the highest variability and produces the highest information gain by applying the PCA method before the decision-tree algorithm. Moreover, measuring the SVM classification metrics confirmed that using only chirp rate as a feature, $TP = 94\%$, $TN = 91\%$, precision = 83% and F1-score = 87% are achieved. Considering delta band power as an additional feature adds more information and achieves a higher performance and scores to $TP = 97\%$, $TN = 95\%$, precision = 100%, and F1-score = 95%. Adding each principal component increases information and improves the performance scores. Finally, feature selection based on observed variability, information gain, and correlation between features was recommended.

8.2 Recommendations

This research focused on 1) designing a HAR method to recognize human activities in the human activity prediction method, 2) estimating an adaptive threshold to track the spectrogram using two sliding windows to detect dynamically the maximum variance of movement energy within time-series data, 3) exploring the impact of an imbalanced dataset and selecting features, 4) proposing a practical approach to characterize and select features. However, there are several directions to further extend the present work.

Enhance the ensemble method : A promising future work would be enhancing the ensemble method considering non-deterministic characteristics of an environment, and exploring the impact on classification performance of the imbalanced dataset in relation to the selection of important features such as the chirp rate.

Cancel interfering signals from other people near a receiver : In real-time scenarios, HAR suffers from a near-far problem that originates from other people near a receiver. The reflections from interfering people who are closer to the receiver could create more power compared to a target person.

Calculate direction of movement : The angle of arrival (AoA) and time of flight (ToF) are required to be estimated to calculate the angle of movement from the receiver.

Wavelet transform : Wavelet transforms can be explored as a potential avenue for future work in human activity recognition. These transforms exploit mathematical techniques that allow for multi-resolution analysis of signals. This means it can analyze signals at different scales and provide insights into both high and low-frequency components.

REFERENCES

[1] A. Rejeb, K. Rejeb, S. Simske, H. Treiblmaier, and S. Zailani, "The big picture on the internet of things and the smart city : a review of what we know and what we need to know," *Internet of Things*, vol. 19, p. 100565, 2022.

[2] B. Dash and P. Sharma, "Role of artificial intelligence in smart cities for information gathering and dissemination (a review)," *Academic Journal of Research and Scientific Publishing*, vol. 4, no. 39, 2022.

[3] D. R. Beddiar, B. Nini, M. Sabokrou, and A. Hadid, "Vision-based human activity recognition : a survey," *Multimedia Tools and Applications*, vol. 79, no. 41, pp. 30 509–30 555, 2020.

[4] S. Zhang, Y. Li, S. Zhang, F. Shahabi, S. Xia, Y. Deng, and N. Alshurafa, "Deep learning in human activity recognition with wearable sensors : A review on advances," *Sensors*, vol. 22, no. 4, p. 1476, 2022.

[5] H. M. Do, K. C. Welch, and W. Sheng, "Soham : A sound-based human activity monitoring framework for home service robots," *IEEE Transactions on Automation Science and Engineering*, 2021.

[6] C. Yin, J. Chen, X. Miao, H. Jiang, and D. Chen, "Device-free human activity recognition with low-resolution infrared array sensor using long short-term memory neural network," *Sensors*, vol. 21, no. 10, p. 3551, 2021.

[7] L. M. Dang, K. Min, H. Wang, M. J. Piran, C. H. Lee, and H. Moon, "Sensor-based and vision-based human activity recognition : A comprehensive survey," *Pattern Recognition*, vol. 108, p. 107561, 2020.

[8] C. Wang, S. Chen, Y. Yang, F. Hu, F. Liu, and J. Wu, "Literature review on wireless sensing-wi-fi signal-based recognition of human activities," *Tsinghua Science and Technology*, vol. 23, no. 2, pp. 203–222, 2018.

[9] S. Liu, Y. Zhao, F. Xue, B. Chen, and X. Chen, "Deepcount : Crowd counting with wifi via deep learning," *arXiv preprint arXiv :1903.05316*, 2019.

[10] R. Alazrai, A. Awad, A. Baha'A, M. Hababeh, and M. I. Daoud, "A dataset for wi-fi-based human-to-human interaction recognition," *Data in Brief*, p. 105668, 2020.

[11] L. Guo, L. Wang, C. Lin, J. Liu, B. Lu, J. Fang, Z. Liu, Z. Shan, J. Yang, and S. Guo, "Wiar : A public dataset for wifi-based activity recognition," *IEEE Access*, vol. 7, pp. 154 935–154 945, 2019.

[12] W. Wang, A. X. Liu, M. Shahzad, K. Ling, and S. Lu, "Understanding and modeling of wifi signal based human activity recognition," in *Proceedings of the 21st annual international conference on mobile computing and networking*, 2015, pp. 65–76.

[13] S. Zhu, J. Xu, H. Guo, Q. Liu, S. Wu, and H. Wang, "Indoor human activity recognition based on ambient radar with signal processing and machine learning," in *2018 IEEE international conference on communications (ICC)*. IEEE, 2018, pp. 1–6.

[14] D. Yang and J. Huangfu, "Improving activity recognition for multiple-node wireless sensor network system based on compressed sensing," *International Journal of Sensor Networks*, vol. 34, no. 3, pp. 162–171, 2020.

[15] I. Nirmal, A. Khamis, M. Hassan, W. Hu, and X. Zhu, "Deep learning for radio-based human sensing : Recent advances and future directions," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 2, pp. 995–1019, 2021.

[16] D. Halperin, W. Hu, A. Sheth, and D. Wetherall, "Tool release : Gathering 802.11 n traces with channel state information," *ACM SIGCOMM computer communication review*, vol. 41, no. 1, pp. 53–53, 2011.

[17] R. Alizadeh, Y. Savaria, and C. Nerguizian, "Human activity recognition and people count for a smart public transportation system," in *2021 IEEE 4th 5G World Forum (5GWF)*. IEEE, 2021, pp. 182–187.

[18] H. Li, X. He, X. Chen, Y. Fang, and Q. Fang, "Wi-motion : A robust human activity recognition using wifi signals," *IEEE Access*, vol. 7, pp. 153 287–153 299, 2019.

[19] E. Rutten, N. Marchand, and D. Simon, "Feedback control as mape-k loop in autonomic computing," in *Software Engineering for Self-Adaptive Systems III. Assurances*. Springer, 2017, pp. 349–373.

[20] I. A. Lawal and S. Bano, "Deep human activity recognition using wearable sensors," in *Proceedings of the 12th ACM International Conference on PErvasive Technologies Related to Assistive Environments*, 2019, pp. 45–48.

[21] T. Huynh and B. Schiele, "Analyzing features for activity recognition," in *Proceedings of the 2005 joint conference on Smart objects and ambient intelligence : innovative context-aware services : usages and technologies*, 2005, pp. 159–163.

[22] A. Bulling, U. Blanke, and B. Schiele, "A tutorial on human activity recognition using body-worn inertial sensors," *ACM Computing Surveys (CSUR)*, vol. 46, no. 3, pp. 1–33, 2014.

[23] E. Swiercz, D. Janczak, and K. Konopko, "Detection of lfm radar signals and chirp rate estimation based on time-frequency rate distribution," *Sensors*, vol. 21, no. 16, p. 5415, 2021.

[24] J. Chen, H. Yu, G. Xu, J. Zhang, B. Liang, and D. Yang, "Airborne sar autofocus based on blurry imagery classification," *Remote Sensing*, vol. 13, no. 19, p. 3872, 2021.

[25] G. Yu, J. Liang, W. Fan, H. C. So, and D. Zhou, "Autofocus algorithms with phase error correction for synthetic aperture radar imagery," *Digital Signal Processing*, vol. 130, p. 103692, 2022.

[26] O. Oshiga, H. U. Suleiman, S. Thomas, P. Nzerem, L. Farouk, and S. Adeshina, "Human detection for crowd count estimation using csi of wifi signals," in *2019 15th International Conference on Electronics, Computer and Computation (ICECCO)*. IEEE, 2019, pp. 1–6.

[27] R. Alizadeh, Y. Savaria, and C. Nerguizian, "Automatic detection of people getting into a bus in a smart public transportation system," in *2022 29th IEEE International Conference on Electronics, Circuits and Systems (ICECS)*. IEEE, 2022, pp. 1–4.

[28] J. C. Soto, I. Galdino, E. Caballero, V. Ferreira, D. Muchaluat-Saade, and C. Albuquerque, "A survey on vital signs monitoring based on wi-fi csi data," *Computer Communications*, vol. 195, pp. 99–110, 2022.

[29] Z. Yang, K. Qian, C. Wu, Y. Zhang, Z. Yang, K. Qian, C. Wu, and Y. Zhang, "Inferring motion direction with wi-fi," *Smart Wireless Sensing : From IoT to AIoT*, pp. 157–182, 2021.

[30] S. Chen, W. Yang, Y. Xu, Y. Geng, B. Xin, and L. Huang, "Afall : Wi-fi-based device-free fall detection system using spatial angle of arrival," *IEEE Transactions on Mobile Computing*, 2022.

[31] J. Jung, H.-C. Moon, J. Kim, D. Kim, and K.-A. Toh, "Wi-fi based user identification using in-air handwritten signature," *IEEE Access*, vol. 9, pp. 53 548–53 565, 2021.

[32] D. Bhattacharya and K. A. Acharya, "A platform for free weight exercise monitoring using passive tags," in *Advances in Control Instrumentation Systems : Select Proceedings of CISCON 2019*. Springer, 2020, pp. 205–214.

[33] R. H. Venkatnarayan, S. Mahmood, and M. Shahzad, "Wifi based multi-user gesture recognition," *IEEE Transactions on Mobile Computing*, vol. 20, no. 3, pp. 1242–1256, 2019.

[34] Y. Wang, Y. Tian, and R. Peng, "Position and orientation independent wireless gesture recognition," in *2022 14th International Conference on Wireless Communications and Signal Processing (WCSP)*. IEEE, 2022, pp. 466–471.

[35] C. Chen, X. Chen, D. Das, D. Akhmetov, and C. Cordeiro, "Overview and performance evaluation of wi-fi 7," *IEEE Communications Standards Magazine*, vol. 6, no. 2, pp. 12–18, 2022.

[36] N. Damodaran and J. Schäfer, "Device free human activity recognition using wifi channel state information," in *2019 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*. IEEE, 2019, pp. 1069–1074.

[37] H. Lee, C. R. Ahn, and N. Choi, "Fine-grained occupant activity monitoring with wi-fi channel state information : Practical implementation of multiple receiver settings," *Advanced Engineering Informatics*, vol. 46, p. 101147, 2020.

[38] Y. Xie, Z. Li, and M. Li, "Precise power delay profiling with commodity wifi," in *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*, 2015, pp. 53–64.

[39] A. T. Parameswaran, M. I. Husain, S. Upadhyaya *et al.*, "Is rssi a reliable parameter in sensor localization algorithms : An experimental study," in *Field failure data analysis workshop (F2DA09)*, vol. 5. IEEE Niagara Falls, NY, USA, 2009.

[40] P. E. Pedersen and H. Nysveen, "Understanding mobile apps as platform-based services in multisided markets," *The Oxford Handbook of Mobile Communication and Society*, p. 425, 2020.

[41] G. Retscher, "Fundamental concepts and evolution of wi-fi user localization : An overview based on different case studies," *Sensors*, vol. 20, no. 18, p. 5121, 2020.

[42] R. Du, H. Xie, M. Hu, Y. Xin, S. McCann, M. Montemurro, T. X. Han, J. Xu *et al.*, "An overview on ieee 802.11 bf : Wlan sensing," *arXiv preprint arXiv :2207.04859*, 2022.

[43] I. . W. L. W. Group *et al.*, "IEEE standard for information technology–telecommunications and information exchange between systems local and metropolitan area networks–specific requirements part 11 : Wireless lan medium access control (mac) and physical layer (phy) specifications amendment 1 : Enhancements for high-efficiency wlan," *IEEE Std*, vol. 802, pp. 1–767, 2021.

[44] R. L. Haupt, *Wireless Communications Systems : An Introduction*. John Wiley & Sons, 2019.

[45] S. K. Memon, K. Nisar, M. H. A. Hijazi, B. Chowdhry, A. H. Sodhro, S. Pirbhulal, and J. J. Rodrigues, "A survey on 802.11 mac industrial standards, architecture, security & supporting emergency traffic : Future directions," *Journal of Industrial Information Integration*, vol. 24, p. 100225, 2021.

[46] E. G. Varthis and D. I. Fotiadis, "A comparison of stop-and-wait and go-back-n arq schemes for ieee 802.11 e wireless infrared networks," *Computer communications*, vol. 29, no. 8, pp. 1015–1025, 2006.

[47] K. Nisar, "Voice priority queue scheduling system models for voip over wlans," *International Journal of Information Communication Technologies and Human Development (IJICTHD)*, vol. 5, no. 1, pp. 36–59, 2013.

[48] T. Nabetani and H. Harada, "Full duplex for ieee 802.11 wireless lans," 2022.

[49] K. HOCINI and M. Yazid, "Full duplex radio communications in high efficiency wlans : Study and comparison of the main mac protocols," *International Journal of Informatics and Applied Mathematics*, vol. 3, no. 1, pp. 1–21, 2020.

[50] H. Sharma, A. Haque, and F. Blaabjerg, "Machine learning in wireless sensor networks for smart cities : a survey," *Electronics*, vol. 10, no. 9, p. 1012, 2021.

[51] M. Cheffena, "Propagation channel characteristics of industrial wireless sensor networks [wireless corner]," *IEEE Antennas and Propagation Magazine*, vol. 58, no. 1, pp. 66–73, 2016.

[52] T. T. S. Rappaport, "Wireless communications - principles and practice," 2001.

[53] J. Wang, J. Xiong, H. Jiang, K. Jamieson, X. Chen, D. Fang, and C. Wang, "Low human-effort, device-free localization with fine-grained subcarrier information," *IEEE Transactions on Mobile Computing*, vol. 17, no. 11, pp. 2550–2563, 2018.

[54] L. Yang, Y. Qiu, F. Lin, B. Wang, and X. Wu, "Optimized algorithm for rfid-based activity recognition of the elderly," *IEEE Access*, vol. 7, pp. 136 777–136 782, 2019.

[55] Y. Wang and Y. Zheng, "Tagbreathe : Monitor breathing with commodity rfid systems," *IEEE Transactions on Mobile Computing*, vol. 19, no. 4, pp. 969–981, 2020.

[56] Y. Zhang, X. Liu, Z. Yang, Z. Li, X. Zhang, and B. Yuan, "Capturing features and performing human detection from human gaits using rfid," *Sensors*, vol. 22, no. 21, p. 8353, 2022.

[57] C. Wu, F. Zhang, B. Wang, and K. R. Liu, "mmtrack : Passive multi-person localization using commodity millimeter wave radio," in *IEEE INFOCOM 2020-IEEE Conference on Computer Communications*. IEEE, 2020, pp. 2400–2409.

[58] H. Liu, A. Zhou, Z. Dong, Y. Sun, J. Zhang, L. Liu, H. Ma, J. Liu, and N. Yang, "M-gesture : Person-independent real-time in-air gesture recognition using commodity millimeter wave radar," *IEEE Internet of Things Journal*, vol. 9, no. 5, pp. 3397–3415, 2021.

[59] Z. Meng, S. Fu, J. Yan, H. Liang, A. Zhou, S. Zhu, H. Ma, J. Liu, and N. Yang, "Gait recognition for co-existing multiple people using millimeter wave sensing," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 01, 2020, pp. 849–856.

[60] F. Zhang, C. Wu, B. Wang, and K. R. Liu, "mmeye : Super-resolution millimeter wave imaging," *IEEE Internet of Things Journal*, vol. 8, no. 8, pp. 6995–7008, 2020.

[61] B. Wei, H. Song, J. Katto, and T. Kikkawa, "Rssi-csi measurement and variation mitigation with commodity wifi device," *IEEE Internet of Things Journal*, 2022.

[62] Y. Sasiwat, D. Buranapanichkit, K. Chetpattananondh, K. Sengchuai, N. Jindapetch, and A. Booranawong, "Human movement effects on the performance of the rssi-based trilateration method : adaptive filters for distance compensation," *Journal of Reliable Intelligent Environments*, vol. 6, pp. 67–78, 2020.

[63] Z. Ma, B. Wu, and S. Poslad, "A wifi rssi ranking fingerprint positioning system and its application to indoor activities of daily living recognition," *International Journal of Distributed Sensor Networks*, vol. 15, no. 4, p. 1550147719837916, 2019.

[64] N. Chuku and A. Nasipuri, "Rssi-based localization schemes for wireless sensor networks using outlier detection," *Journal of Sensor and Actuator Networks*, vol. 10, no. 1, p. 10, 2021.

[65] W. Dib, K. Ghanem, A. Ababou, and B. M. Eskofier, "Human activity recognition based on the fading characteristics of the on-body channel," *IEEE Sensors Journal*, vol. 22, no. 8, pp. 8094–8103, 2022.

[66] C. Wu, W. Zhang, J. Yang, and S. Deng, "Wi-fi indoor 3d localization algorithm based on multi-classifier fusion," *Engineering Research Express*, vol. 4, no. 3, p. 035042, 2022.

[67] S. Depatla and Y. Mostofi, "Crowd counting through walls using wifi," in *2018 IEEE International Conference on Pervasive Computing and Communications (PerCom)*. IEEE, 2018, pp. 1–10.

[68] S. Depatla, A. Muralidharan, and Y. Mostofi, "Occupancy estimation using only wifi power measurements," *IEEE Journal on Selected Areas in Communications*, vol. 33, no. 7, pp. 1381–1393, 2015.

[69] M. El Gharbi, R. Fernández-García, and I. Gil, "Wireless communication platform based on an embroidered antenna-sensor for real-time breathing detection," *Sensors*, vol. 22, no. 22, p. 8667, 2022.

[70] N. Jarvis, J. Hata, N. Wayne, V. Raychoudhury, and M. O. Gani, "Miamimapper : Crowd analysis using active and passive indoor localization through wi-fi probe

monitoring," in *Proceedings of the 15th ACM International Symposium on QoS and Security for Wireless and Mobile Networks*, 2019, pp. 1–10.

[71] Z. Yang, Z. Zhou, and Y. Liu, "From rssi to csi : Indoor localization via channel response," *ACM Computing Surveys (CSUR)*, vol. 46, no. 2, pp. 1–32, 2013.

[72] J. Liu, H. Liu, Y. Chen, Y. Wang, and C. Wang, "Wireless sensing for human activity : A survey," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 3, pp. 1629–1645, 2019.

[73] T. Xing, Q. Yang, Z. Jiang, X. Fu, J. Wang, C. Q. Wu, and X. Chen, "Wifine : Real-time gesture recognition using wi-fi with edge intelligence," *ACM Transactions on Sensor Networks*, vol. 19, no. 1, pp. 1–24, 2022.

[74] P. Duan, J. Li, C. Jiao, Y. Cao, and J. Kong, "Wibfall : A device-free fall detection model for bathroom," in *Mobile Networks and Management : 11th EAI International Conference, MONAMI 2021, Virtual Event, October 27-29, 2021, Proceedings*. Springer, 2022, pp. 182–193.

[75] Z. Lin, Y. Xie, X. Guo, Y. Ren, Y. Chen, and C. Wang, "Wieat : Fine-grained device-free eating monitoring leveraging wi-fi signals," in *2020 29th International Conference on Computer Communications and Networks (ICCCN)*. IEEE, 2020, pp. 1–9.

[76] D. Wu, R. Gao, Y. Zeng, J. Liu, L. Wang, T. Gu, and D. Zhang, "Fingerdraw : Sub-wavelength level finger motion tracking with wifi signals," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 4, no. 1, pp. 1–27, 2020.

[77] B. Korany, C. R. Karanam, H. Cai, and Y. Mostofi, "Xmodal-id : Using wifi for through-wall person identification from candidate video footage," in *The 25th Annual International Conference on Mobile Computing and Networking*, 2019, pp. 1–15.

[78] X. Li, L. Chang, F. Song, J. Wang, X. Chen, Z. Tang, and Z. Wang, "Crossgr : Accurate and low-cost cross-target gesture recognition using wi-fi," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 5, no. 1, pp. 1–23, 2021.

[79] C. Li, L. Liu, Z. Cao, and M. Zhang, "Wivelo : Fine-grained walking velocity estimation for wi-fi passive tracking," in *2022 19th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*. IEEE, 2022, pp. 172–180.

[80] Y. Zhang, Y. Zheng, G. Zhang, K. Qian, C. Qian, and Z. Yang, "Gaitid : robust wi-fi based gait recognition," in *Wireless Algorithms, Systems, and Applications : 15th International Conference, WASA 2020, Qingdao, China, September 13–15, 2020, Proceedings, Part I 15*. Springer, 2020, pp. 730–742.

[81] WiKi, "Modulation and coding schemes," https://en.wikipedia.org/wiki/IEEE_802.11n-2009, Oct. 2020.

[82] Z. C. Lipton, J. Berkowitz, and C. Elkan, "A critical review of recurrent neural networks for sequence learning," *arXiv preprint arXiv :1506.00019*, 2015.

[83] A. Sherstinsky, "Fundamentals of recurrent neural network (rnn) and long short-term memory (lstm) network," *Physica D : Nonlinear Phenomena*, vol. 404, p. 132306, 2020.

[84] B. V. Gnedenko, *Theory of probability*. Routledge, 2018.

[85] A. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE transactions on Information Theory*, vol. 13, no. 2, pp. 260–269, 1967.

[86] A. Graves, G. Wayne, and I. Danihelka, "Neural turing machines," *arXiv preprint arXiv :1410.5401*, 2014.

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

[88] S. Siuly and Y. Li, "Improving the separability of motor imagery EEG signals using a cross correlation-based least square support vector machine for brain–computer interface," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 20, no. 4, pp. 526–538, 2012.

[89] Z. Sun, Q. Song, X. Zhu, H. Sun, B. Xu, and Y. Zhou, "A novel ensemble method for classifying imbalanced data," *Pattern Recognition*, vol. 48, no. 5, pp. 1623–1637, 2015.

[90] J. Li, Q. Zhu, Q. Wu, Z. Zhang, Y. Gong, Z. He, and F. Zhu, "Smote-nan-de : Addressing the noisy and borderline examples problem in imbalanced classification by natural neighbors and differential evolution," *Knowledge-Based Systems*, vol. 223, p. 107056, 2021.

[91] T. Pan, J. Zhao, W. Wu, and J. Yang, "Learning imbalanced datasets based on smote and gaussian distribution," *Information Sciences*, vol. 512, pp. 1214–1233, 2020.

[92] A. Sivarajan, E. Sivasankar *et al.*, "Balancing of an imbalanced dataset by applying smote variants and predicting neonatal mortality using ensemble learning techniques," in *2022 International Conference on Innovative Trends in Information Technology (ICITIIT)*. IEEE, 2022, pp. 1–6.

[93] X. Jiang and Z. Ge, "Data augmentation classifier for imbalanced fault classification," *IEEE Transactions on Automation Science and Engineering*, vol. 18, no. 3, pp. 1206–1217, 2020.

[94] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *Journal of computer and system sciences*, vol. 55, no. 1, pp. 119–139, 1997.

[95] S. Ando, "Classifying imbalanced data in distance-based feature space," *Knowledge and Information Systems*, vol. 46, no. 3, pp. 707–730, 2016.

[96] Z. Liu, H. Cao, X. Chen, Z. He, and Z. Shen, "Multi-fault classification based on wavelet svm with pso algorithm to analyze vibration signals from rolling element bearings," *Neurocomputing*, vol. 99, pp. 399–410, 2013.

[97] S. Datta and S. Das, "Near-bayesian support vector machines for imbalanced data classification with equal or unequal misclassification costs," *Neural Networks*, vol. 70, pp. 39–52, 2015.

[98] N. V. Chawla, A. Lazarevic, L. O. Hall, and K. W. Bowyer, "Smoteboost : Improving prediction of the minority class in boosting," in *European conference on principles of data mining and knowledge discovery*. Springer, 2003, pp. 107–119.

[99] L. Nanni, C. Fantozzi, and N. Lazzarini, "Coupling different methods for overcoming the class imbalance problem," *Neurocomputing*, vol. 158, pp. 48–61, 2015.

[100] A. Subbaswamy and S. Saria, "From development to deployment : dataset shift, causality, and shift-stable models in health ai," *Biostatistics*, vol. 21, no. 2, pp. 345–352, 2020.

[101] B. Kovalerchuk, "Enhancement of cross validation using hybrid visual and analytical means with shannon function," in *Beyond Traditional Probabilistic Data Processing Techniques : Interval, Fuzzy etc. Methods and Their Applications*. Springer, 2020, pp. 517–543.

[102] L. Yijing, G. Haixiang, L. Xiao, L. Yanan, and L. Jinling, "Adapted ensemble classification algorithm based on multiple classifier system and feature selection for classifying multi-class imbalanced data," *Knowledge-Based Systems*, vol. 94, pp. 88–104, 2016.

[103] M. Kepesi and L. Weruaga, "Adaptive chirp-based time–frequency analysis of speech signals," *Speech communication*, vol. 48, no. 5, pp. 474–492, 2006.

[104] Wiki, "Chirp," https://en.wikipedia.org/wiki/Chirp, March 2023.

[105] M. Barbu, E. J. Kaminsky, and R. E. Trahan, "Fractional fourier transform for sonar signal processing," in *Proceedings of OCEANS 2005 MTS/IEEE*. IEEE, 2005, pp. 1630–1635.

[106] A. Gómez-Echavarría, J. P. Ugarte, and C. Tobón, "The fractional fourier transform as a biomedical signal and image processing tool : A review," *Biocybernetics and Biomedical Engineering*, vol. 40, no. 3, pp. 1081–1093, 2020.

[107] M. Soumekh, *Synthetic aperture radar signal processing*. New York : Wiley, 1999, vol. 7.

[108] K. Wang, X. Cheng, Y. Chen, C. She, K. Sun, and P. Zhao, "Heart sound model based on cascaded and lossless acoustic tubes," *Journal of Mechanics in Medicine and Biology*, vol. 19, no. 05, p. 1950031, 2019.

[109] Y. Sun, T. Fei, F. Schliep, and N. Pohl, "Gesture classification with handcrafted micro-doppler features using a fmcw radar," in *2018 IEEE MTT-S International Conference on Microwaves for Intelligent Mobility (ICMIM)*. IEEE, 2018, pp. 1–4.

[110] J. R. Klauder, A. Price, S. Darlington, and W. J. Albersheim, "The theory and design of chirp radars," *Bell System Technical Journal*, vol. 39, no. 4, pp. 745–808, 1960.

[111] K. Kaboutari and V. Hosseini, "A compact 4-element printed planar mimo antenna system with isolation enhancement for ism band operation," *AEU-International Journal of Electronics and Communications*, vol. 134, p. 153687, 2021.

[112] Z. Wang, K. Jiang, Y. Hou, W. Dou, C. Zhang, Z. Huang, and Y. Guo, "A survey on human behavior recognition using channel state information," *Ieee Access*, vol. 7, pp. 155 986–156 024, 2019.

[113] "Ieee standard for information technology–telecommunications and information exchange between systems local and metropolitan area networks–specific requirements part 11 : Wireless lan medium access control (mac) and physical layer (phy) specifications ame," *IEEE Standard for Information Technology–Telecommunications and Information Exchange between Systems Local and Metropolitan Area Networks–Specific Requirements Part 11 : Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Ame*.

[114] D. Halperin, W. Hu, A. Sheth, and D. Wetherall, "Predictable 802.11 packet delivery from wireless channel measurements," *ACM SIGCOMM Computer Communication Review*, vol. 40, no. 4, pp. 159–170, 2010.

[115] Y. Xiao, "Ieee 802.11 n : enhancements for higher throughput in wireless lans," *IEEE Wireless Communications*, vol. 12, no. 6, pp. 82–91, 2005.

[116] D. Vasisht, S. Kumar, and D. Katabi, "Decimeter-level localization with a single wifi access point," in *13th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 16)*, 2016, pp. 165–178.

[117] A. V. Oppenheim, J. Buck, M. Daniel, A. S. Willsky, S. H. Nawab, and A. Singer, *Signals & systems*. Pearson Educación, 1997.

[118] S. Yousefi, H. Narui, S. Dayal, S. Ermon, and S. Valaee, "A survey on behavior recognition using wifi channel state information," *IEEE Communications Magazine*, vol. 55, no. 10, pp. 98–104, 2017.

[119] S. L. Brunton and J. N. Kutz, *Data-driven science and engineering : Machine learning, dynamical systems, and control*. Cambridge University Press, 2019.

[120] Matplotlib, "Specgram," https://en.wikipedia.org/wiki/Short-time_Fourier_transform, Oct. 2023.

[121] ——, "Specgram," https://matplotlib.org/3.5.0/api/_as_gen/matplotlib.pyplot.specgram.html, Oct. 2023.

[122] A. Virmani and M. Shahzad, "Position and orientation agnostic gesture recognition using wifi," in *Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services*, 2017, pp. 252–264.

[123] P. Chavan, "LSTM Parameters," https://www.researchgate.net/post/How-to-decide-the-number-of-hidden-layers-and-nodes-in-a-hidden-layer, June 2021.

[124] M. E. Van Valkenburg, *Reference data for engineers : radio, electronics, computers and communications*. Elsevier, 2002.

[125] A. Zhuravchak, O. Kapshii, and E. Pournaras, "Human activity recognition based on wi-fi csi data-a deep neural network approach," *Procedia Computer Science*, vol. 198, pp. 59–66, 2022.

[126] R. C. Staudemeyer and E. R. Morris, "Understanding lstm–a tutorial into long short-term memory recurrent neural networks," *arXiv preprint arXiv :1909.09586*, 2019.

[127] M. T. Manry, H. Chandrasekaran, and C.-H. Hsieh, "Signal processing using the multilayer perceptron," in *Handbook of Neural Network Signal Processing*. CRC Press, 2018, pp. 2–1.

[128] P. Asghari, E. Soleimani, and E. Nazerfard, "Online human activity recognition employing hierarchical hidden markov models," *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, pp. 1141–1152, 2020.

[129] M. T. Sadiqa, S. Siulyb, and A. U. Rehmanc, "Evaluation of power spectral and machine learning techniques for the development of subject-specific BCI," *Artificial Intelligence-Based Brain-Computer Interface*, p. 99, 2022.

[130] S. Pahuja, K. Veer *et al.*, "Recent approaches on classification and feature extraction of EEG signal : A review," *Robotica*, pp. 1–25, 2022.

[131] Wikipedia, "Simpson rule," en.wikipedia.org/wiki/Simpson%27s_rule, Jan. 2022.

[132] S. E. Roshan and S. Asadi, "Improvement of bagging performance for classification of imbalanced datasets using evolutionary multi-objective optimization," *Engineering Applications of Artificial Intelligence*, vol. 87, p. 103319, 2020.

[133] H. Yang, Z. Wang, and K. Song, "A new hybrid grey wolf optimizer-feature weighted-multiple kernel-support vector regression technique to predict tbm performance," *Engineering with Computers*, pp. 1–17, 2020.

[134] Scikit-learn, "SVM Tuning Parameters," https://towardsdatascience.com/hyperparameter-tuning-for-support-vector-machines, April 2022.

[135] S. H. Haji and A. M. Abdulazeez, "Comparison of optimization techniques based on gradient descent algorithm : A review," *PalArch's Journal of Archaeology of Egypt/Egyptology*, vol. 18, no. 4, pp. 2715–2743, 2021.

[136] L. Breiman, "Bagging predictors," *Machine learning*, vol. 24, pp. 123–140, 1996.

[137] H. He and Y. Ma, "Imbalanced learning : foundations, algorithms, and applications," 2013.

[138] Dr. Guanglinang Chen, "Machine Learning Classification Evaluation Criteria," https://www.sjsu.edu/faculty/guangliang.chen/Math251/lec6eval-criteria.pdf, Dec. 2021.

[139] L. Demidova and I. Klyueva, "Svm classification : Optimization with the smote algorithm for the class imbalance problem," in *2017 6th Mediterranean conference on embedded computing (MECO)*. IEEE, 2017, pp. 1–4.

[140] Q. Wang, Z. Luo, J. Huang, Y. Feng, and Z. Liu, "A novel ensemble method for imbalanced data learning : bagging of extrapolation-smote svm," *Computational intelligence and neuroscience*, vol. 2017, 2017.

[141] Scikit-learn, "Classification Report," https://scikit-learn.org/stable/modules/generated/sklearn.metrics.classification_report.html, June 2023.

[142] C. Esposito, G. A. Landrum, N. Schneider, N. Stiefl, and S. Riniker, "Ghost : adjusting the decision threshold to handle imbalanced data in machine learning," *Journal of Chemical Information and Modeling*, vol. 61, no. 6, pp. 2623–2640, 2021.

[143] Scikit-learn, "ROC," https://scikit-learn.org/stable/modules/generated/sklearn.metrics.roc_curve.html, March 2022.

[144] J. Cui and D. Wang, "Biosignal analysis with matching-pursuit based adaptive chirplet transform," *arXiv preprint arXiv :1709.08328*, 2017.

[145] Y. Lu, R. Demirli, G. Cardoso, and J. Saniie, "A successive parameter estimation algorithm for chirplet signal decomposition," *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, vol. 53, no. 11, pp. 2121–2131, 2006.

[146] J. S. Lim and A. V. Oppenheim, *Advanced topics in signal processing*. Prentice-Hall, Inc., 1987.

[147] H. Chen, Z. Liu, B. Wu, and C. He, "A technique based on nonlinear hanning-windowed chirplet model and genetic algorithm for parameter estimation of lamb wave signals," *Ultrasonics*, vol. 111, p. 106333, 2021.

[148] E. Priya, P. S. Reshma, S. Sashaank *et al.*, "Temporal and spectral features based gender recognition from audio signals," in *2022 International Conference on Communication, Computing and Internet of Things (IC3IoT)*. IEEE, 2022, pp. 1–5.

[149] B. M. S. Hasan and A. M. Abdulazeez, "A review of principal component analysis algorithm for dimensionality reduction," *Journal of Soft Computing and Data Mining*, vol. 2, no. 1, pp. 20–30, 2021.

[150] J. Shlens, "A tutorial on principal component analysis," *arXiv preprint arXiv :1404.1100*, 2014.

[151] Numpy, "Corrcoef," https://numpy.org/doc/stable/reference/generated/numpy.corrcoef.html, Jan. 2023.

[152] B. Cael, K. Bisson, E. Boss, and Z. K. Erickson, "How many independent quantities can be extracted from ocean color ?" *Limnology and Oceanography Letters*, 2023.

[153] B. Charbuty and A. Abdulazeez, "Classification based on decision tree algorithm for machine learning," *Journal of Applied Science and Technology Trends*, vol. 2, no. 01, pp. 20–28, 2021.

[154] WiKi, "Entropy," https://en.wikipedia.org/wiki/Entropy_(information_theory), Sep 2023.

[155] B. T. Jijo and A. M. Abdulazeez, "Classification based on decision tree algorithm for machine learning," *evaluation*, vol. 6, p. 7, 2021.

[156] J. Liu, C. Wang, W. Chi, G. Chen, and L. Sun, "Estimated path information gain-based robot exploration under perceptual uncertainty," *Robotica*, vol. 40, no. 8, pp. 2748–2764, 2022.

[157] Wiki, "Scree plot," https://en.wikipedia.org/wiki/Scree_plot, Feb. 2023.

[158] Scikit-learn, "Decision-Tree," https://scikit-learn.org/stable/modules/generated/sklearn.tree.plot_tree.html, April 2023.

[159] A. Tharwat, "Classification assessment methods," *Applied computing and informatics*, vol. 17, no. 1, pp. 168–192, 2021.

[160] S. Raschka, Y. H. Liu, V. Mirjalili, and D. Dzhulgakov, *Machine Learning with PyTorch and Scikit-Learn : Develop machine learning and deep learning models with Python*. Packt Publishing Ltd, 2022.