Human Recognition Using Wireless Router Signal

by

S. M. A. Muksit Chowdhury
15301029
Hasinur Arefin Shawon
15301067
Tanvir Wazy Ullah Patwary
15301065
Rukshanda Rahman
15301016

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science

Department of Computer Science and Engineering
Brac University
August 2019

Declaration

It is hereby declared that

1. The thesis submitted is our own original work while completing degree at Brac University.

2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.

3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.

4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

S. M. A. Muksit Chowdhury ID: 15301029	Hasinur Arefin Shawon ID: 15301067
Tanvir Wazy Ullah Patwary ID: 15301065	Rukshanda Rahman ID: 15301016

Approval

The thesis titled "Human Recognition Using Wireless Router Signal" submitted by

1. S. M. A. Muksit Chowdhury(15301029)
2. Hasinur Arefin Shawon(15301067)
3. Tanvir Wazy Ullah Patwary(15301065)
4. Rukshanda Rahman(15301016)

Of Summer, 2019 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on September 12, 2019.

Examining Committee:

Supervisor:
(Member)

Dr. Amitabha Chakrabarty
Associate Professor
Department of Computer Science and Engineering
Brac University

Program Coordinator:
(Member)

Dr. Jia Uddin
Associate Professor
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Dr. Mahbubul Alam Majumdar
Professor and Chairperson
Department of Computer Science and Engineering
Brac University

Abstract

Human identification technology can revolutionize numerous sectors in human life and a large number of methods already exist to identify humans such as voice recognition, fingerprint identification, face recognition and so on. As WiFi devices have become an inseparable commodity in our daily life, we are presenting a system which can identify human uniquely using WiFi signals and Channel State Information(CSI). Every person has some unique moving features and gestures which can be predicted by WiFi spectrum sensing. When a person walks through a region that is emitting WiFi transmission he or she can be easily identified by our model. Every person moves in a unique manner and therefore causes unique disturbances in the WiFi signals. Using Channel State Information (CSI)of the Wi-Fi signal, we have extracted 10 uncommon characteristics that separate one human being from another. We have analyzed channel state properties of a communication link from the transmitter to receiver and their combined effects. In our database, we stored the trajectory of different people and matched them against measured trace. Our system has showcased 93% to 83% accuracy for K-NN, 94.09% to 88.15% for SVM and 96.05% to 89.84% for MLP for a group of 10 to 50 people. Our system has also shown an accuracy of 96% for K-NN, 97% for MLP in detecting gender for males from the 50 people and an accuracy of 86% for K-NN, 92% for MLP in detecting gender for female from 50 people consisting of 39 male and 11 female. However, the gender identification accuracy for both male and female were an equal 94% for KNN and 97% for MLP when the dataset consisted of 11 male and 11 female. Our proposition is that we can implement our system in residential homes and medium size offices as smart security system for identifying humans.

Keywords: Human Identification; Channel State Information; Sub-carrier Information; Channel Frequency Response, K-nearest Neighbors; Support Vector Machine; Multilayer Perceptrons.

Dedication

We want to dedicate our thesis to our loving parents, to whom we are forever grateful. We would have been lost without their extreme care, dedication and sacrifices.

Acknowledgement

To start with, all praises to Allah for enabling us to complete our thesis. After that, we want to thank and express our immense gratitude towards our supervisor Dr. Amitabha Chakrabarty for his unflinching support and guidance. This work wouldn't have been completed if it was not for his valuable feedback and helping hand. We also want to thank BRAC University IT Department for allocating us with a desktop PC. We are also grateful to Md. Nafiul Alam Nipu, an alumnus of BRAC University for taking time out of his busy schedule to come to Brac University campus and for also helping us online whenever we asked him for suggestions about our work. We also want to thank Souvik Talukder, another alumnus of Brac University, whose work on Human Identification along with Md. Nafiul Alam Nipu have motivated us in our work. Then, we want to thank each and everyone who has helped us in our thesis work. Finally, we want to acknowledge the efforts our parents put for us everyday in their lives. Without their constant care, completion of this thesis work would have been impossible for us.

Table of Contents

List of Figures

List of Tables

Nomenclature

The next list describes several symbols & abbreviation that will be later used within the body of the document

CFR Channel Frequency Response

$COTS$ Comercial off-the-shelf

CSI Channel State Information

FFT Fast Fourier Transform

KNN K-Nearest Neighbors

MLP Multilayer Perceptrons

NIC Network Interface Card

$OFDM$ Orthogonal frequency-division multiplexing

RBF Radial Basis Function

SVM Support Vector Machine

Chapter 1

Introduction

Human identification technology is indispensable for a large number of applications like health care, fitness tracker, surveillance, smartphone security, smart homes, office security and so on. The conventional technologies to detect human presence are Passive Infrared(PIR) sensors, Cameras, CO2 sensors, wearable inertial sensors, motion sensors, radio waves and so on [1] [2]. However, the Passive Infrared sensors have low range, they require multiple sensors just to cover a small area, they cannot distinguish between human and other moving animals, they are air flow and sunshine sensitive and their accuracy is unsatisfactory for moving human beings [3]. Other technologies such as radar, ultra wideband SDR-based solutions are reliable but expensive, thus making them inefficient for everywhere use. Again, the audio and visual based approach has chances of intruding people's private space. Also, the visual approaches are reliable on many things such as lighting conditions, resolution, noise, distortion and so on. Although night vision technology has been developed and researches have been made on this kind of technology. Advanced methods like Thermal Imaging has been developed but even thermal imaging doesn't work on extreme low light or extreme high lights as the night vision goggles and scope do not work perfectly in these extreme conditions [4]. The wearable sensors are really convenient, because people constantly have to wear additional devices .

Apart from the camera based approaches which achieve good accuracy but violates people's privacy, other human identification works exploit fingerprints [5], iris [6], or Sclera [7] bio-metrics for identifications, but despite having good accuracy these systems have limitations as they collect bio-metric data or they need the users to wear a certain device or wire, which causes true inconvenience for the users.

On the contrary, we use Wi-Fi devices from shops, office spaces, institutions, homes to everywhere including even rural areas. The air is filled with a spectrum of Radio Frequency signals emitting from devices that are wireless. When human beings move through the place in range of these wireless devices, they propagate the signals and therefore causing changes in Wi-Fi spectrum which we will be able to identify. By carefully examining these spreads with CSI data, basic human gaits which are unique for each person can be detected. In this method, no bio-metric data will be needed, so no misuse is possible. The subjects do not have to wear any kind of devices or get themselves wired. Therefore, we are proposing a Wi-Fi based human identification method using CSI data for its accuracy, effectiveness and conservation of people's privacy.

In our proposed system, we are using the above mentioned technique of identifying basic human gaits using Wi-Fi devices by detecting different unique perturbation using Channel State Information (CSI) data and noticing different changes in WiFi spectrum caused by different persons. We are embedding Machine Learning with our system. Machine learning is a subset of Artificial Intelligence (AI). It focuses on analyzing and interpreting patterns. Because of machine learning, our system will be able to automatically learn and become better with experience. We have used Machine Learning predictors for human identification process. The techniques or algorithms used for identifications are Support Vector Machine (SVM), K-Nearest Neighbors (KNN) and Multilayer Perceptrons (MLP).

1.1 Thesis Motivation

Security is that one thing that worries people from ancient times. People are willing to spend huge sums of money on home securities that's impenetrable for outsiders. CCTV cameras have been used extensively but even those can be hacked. The main motive of our research was to distinguish each individual with utmost accuracy and least possible cost. After much speculation and analyzing several different research papers on human identification we came up with a solution that's easy to maintain and does not violate anyone's privacy. In today's era, security is a vital issue in many offices and homes. Our main concern was to recognize a person as accurately as possible to prevent break-ins by unknown people. We wanted to develop a secured system that can protect people's home and valuable things. Coming up with different algorithms that give high precision to implement our work, we chose K-NN, SVM and MLP for best results. Our work does not stop here as we plan to implement our method on even larger group of people from a wide range of space.

1.2 Thesis Contribution

The goal of our thesis is to develop a system that can identify an individual human being with high accuracy to make people's lives easier. Also keeping the cost in mind we used Wi-Fi signals which are available in almost every home and office. The advanced technology increased the reliability of our model. Machine learning is trending in the tech world and using this for our research is a step forward in its development.

1.3 Outline of Thesis

The subsequent chapters of our paper have been divided in the following manner: In chapter 2, we have discussed about related works and existing methods to our research. Chapter 3 features the methodology for our work and an overview of CSI data. In chapter 4, our whole working process and system implementation from setting up the environment for taking CSI data to identification of human is discussed. After that, chapter 5 explores the results we have obtained after implementation and the limitations of our system. Finally, in Chapter 6, we have drawn the conclusion and talked about our future plans of expansion and further improvements.

Chapter 2

Literature Review

Several researches have been made for identifying human uniquely. Different methods and techniques have been used so far in this research. The world in recent times have seen a rise of research efforts using Channel State Information. Applications ranging from fall detection [8] to vital-sign monitoring [9] has been made possible by analysing CSI data as it provides detailed information about phase and amplitude of each subcarrier and the path the subcarriers take in propagation. Among the earlier works, multiple signal classification (MUSIC) [10], used Channel State Information from multiple antennas to approximate the angles signals take to transmit from the transmitter to the receiver. The first instance of researchers showcasing that WiFi signals can be used for identifying human is [11] and the proposed system is called Wi-Fi-ID which analyses the channel state information to extract unique features like walking style of that individual and thus identity the person. The average accuracy of identifying people is 93% to 77% among a group of people consisting of 2-6 people respectively so they implemented this device on commercial off-the-shelf devices. Furthermore, recent advancement in wireless technology has proved that the movement of human body parts will affect Channel State Information of wireless signals in the indoor environments. This technique has the ability to work in dark environments, non-line of sight detection and working without intruding people's personal space like camera-based systems [12]. The work of Liu, Jian, et al. [9], was the first work that was able to estimate heart-rate in a device-free manner. In their paper [9], Liu, Jian, et al. proposed to use CSI data to monitor both breathing rates and heart rates. Breathing rates and heart rates are two essential vital signs to assess someone's physical condition and their system utilized the CSI data in way that it was able to recognize minute movements like inhaling, exhaling, systole and diastole caused by these two vital signs. Their results showed that approximately 57% of approximation erros are lower than 2 beats per minute and more than 90% are lower than 4 beats per minute. Another study by Xin, Tong, et al. [[13]], has proposed an approach for human identification, using a combination of techniques like Dynamic Time Warping, Principal Component Analysis and Discrete Wavelet Transform. This system was executed in a smart home environment the measurement of which was 6m*5m. To evaluate, it collected data for 9 people. The identification accuracy of this experiment was encouraging as it displayed an accuracy of 94.5% to 88.9% for a user group of 2 to 6 people. That means is efficient for very small residential homes. In November 2017, Zhu Wang, Bin Guo, Zhiwen Yu, Xingshe Zhou - four professors from Northwestern Polytechnical University, China proposed a WiFi CSI

Based Behavior Recognition System [14]. In their paper, they related applications were classified into signals, actions and activities. Using COTS WiFi devices, perfectly refined respiration information can also be extracted of a person in his or her different sleeping positions[15]. In their paper [15], Liu, Xuefeng, et al. proposed a form of human identification which would use Channel State Information(CSI) around a person to identify rhythmic patterns associated with respiration and disruption in signal between the transmitter(TX) and the receiver (RX) due to body movement during sleeping. This system was able to trace respiration rate with an accuracy of more than 85% for 6 sleeping positions. CSI data can also be used in detecting finger movements. Li, Hong, et al., in their paper [16], named their system "WiFinger". This system analyses smallest movement of fingers by capturing patterns of gestures from the physical layer. It modifies the driver of Intel 5300 network interface card which follows IEEE802.11 standards. The system addressed the challenge of identifying unique gestures made by fingers by using radio-metric attributes of finger motion. The system was able to implement accurate text input in Wi-Fi devices. For 10 users, WiFinger achieved average recognition accuracy of 90.4% for finger movements of every user. For continuous input of number text its average accuracy is 82.67%. Wireless CSI data was used to detect key-stroke in the system named WiKey [17]. The system consisted of a transmitter, for example: a router and a receiver, for example: a laptop. It works on the principal that the transmitter will emit signals in a continuous manner and the receiver will receive that signal. WiKey recognizes the typed word of a user based on the received WiFi signal's Channel State Information values at the receiver side. This system had an accuracy of 93.5% in recognizing keystrokes within a typed sentence. More WiFi signal based human identification systems have already been studied, such as WiSee [18], WiHear[19], Wi-Vi [20] . These systems uses the fact that different human motions affect or distort Wi-Fi signal in different ways. WiSee uses Universal Software Radio Peripheral (USRP) catch Orthogonal Frequency Division Multiplexing (OFDM). It also uses USRP to extract the Doppler shift found in signals that are obstructed by human movement and thus it recognizes nine unique gestures.
On the contrary, the technology applied by Wi-Vi is multi-antenna technology, which analyzes the signal distortions because of motion. It can be said that, both WiSee and Wi-Vi focus on recognizing a set of coarse-grained gestures or motions, such as punching, kicking, taking forward or backward steps. Add to that, both these systems are reliant on signals that are software radios' extraction. Another recent research in passive human identification was done by Wang, Fei, et al. [21]. In their experiment, they implemented a system which they named WiPIN, which is operation-free identification system, requiring least user collaboration. The system was based on a whole new apprehension that, human body information is carried by Wi-Fi signals while passing through them. However, our proposed system of human identification is based on the insight that the distortions in WiFi signal caused by human gaits would be unique. A similar approach to this in human identification from the past is the work of Nipu, Md Nafiul Alam, et al [22]. In their experiment, they implemented a system to use CSI subcarrier information and changes in the WiFi specturm to uniquely identify humans. For a group of 2 to 5 people, they used Random Forest and Boosted Decision Tree as human identification classifiers and for 5 people, achieved an accuracy of 78% and 84% respectively.

Chapter 3

Methodology

3.1 Machine Learning

We have implemented three machine learning algorithms for our experiment.

3.1.1 Support Vector Machine

Support Vector Machine (SVM) is a supervised machine learning algorithm which can be used for both classification and regression which are called SVC (Support Vector classification) and SVR (Support Vector Regression) respectively. The main objective behind SVM is finding a hyperplane that separates features into different domains or regions in the best possible way. In this algorithm, each data item is plotted as a point in an n-dimensional space (n = number of features) with the value of each feature being the value of a particular coordinate. Then, classification is performed by finding the hyper-plane that differentiate or separate the two classes in a correct way.

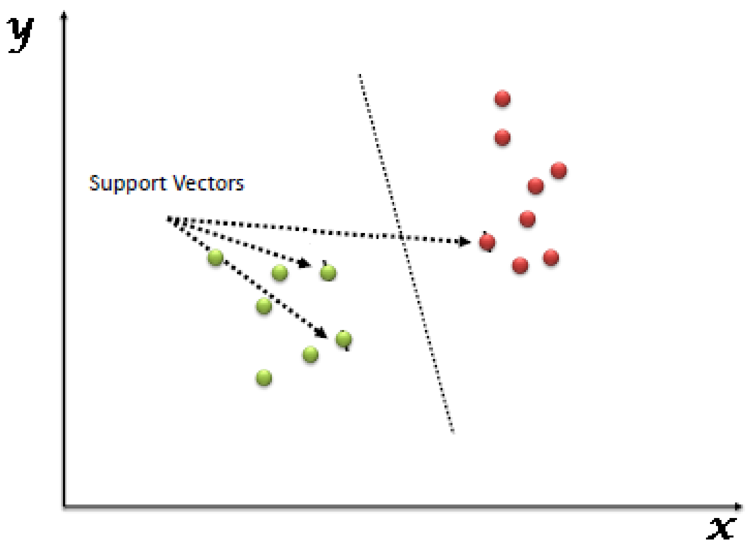

Figure 3.1: Support Vector Machine
[23]

Role of Intuition in SVM

Assume, someone is getting emails from a stalker. So, he or she wants to design a function (hyperplane) which will precisely differentiate between the two cases, so that in case of he or she receiving an email from the stalker, that very email will be classified or differentiated as a spam and the other emails would be classified as "not spam". In figure 3.2 two cases of hyperplane being drawn is showed. That person is definitely going to pick the hyperplane that is dentoed as (a), because it looks more classified. In this way, SVM is primarily composed of the idea of determining an optimal hyperplane that will be able to precisely perform classification between the unique classes.

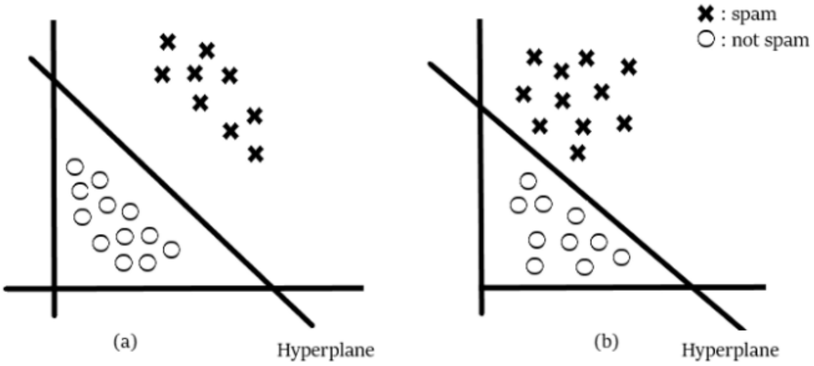

Figure 3.2: Role of Intuition in SVM
[24]

SVM Terminologies

Different types of terminologies like support vector points, margin and so on are used in SVM. It is necessary to know the what each terminology refers to. To start with, support vector points refer to the points which are the closest points to they hyperplane. Margin is the term used to define the distance of the support vectors from the hyperplane.
To determine the hyperplane, support vector points are very crucial as the placement of hypeprlane is dependent on the position of the support vector points.

Hyperplane Determination

The hyperplane refers to a function that is used in Support Vector Machine algorithm to classify or differentiate between the features that are available. The function that is used to classify between features in 2-D is called a line. However, this function is called a plane in 3-D space. Similarly, hyperplane is the function which classifies or differentiate between the points in higher dimension. Assuming there are m dimensions, we can put the equation of the hyperplane in the 'M' dimension as:

$$\begin{aligned} y &= w_0 + w_1x_1 + w_2x_2 + w_3x_3 \ldots \\ &= w_0 + \textstyle\sum_{i=1}^{m} w_ix_i \\ &= w_0 + w^TX \\ &= b + w^TX \end{aligned} \tag{3.1}$$

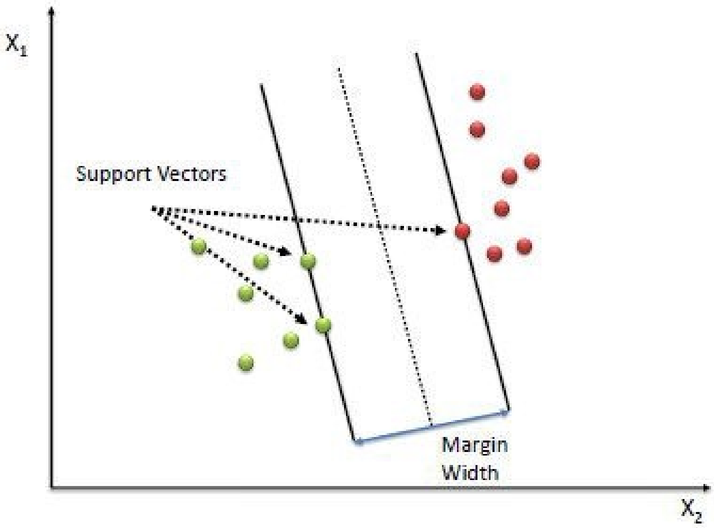

Figure 3.3: SVM Terminologies with Example
[25]

Here,
$w_i = vectors(w_0, w_1, w_2, w_3......w_m)$
b = biased term (w_0)
x = variables.

Soft Margin SVM and Hard Margin SVM

Hard Margin SVM allows softness that is allowance of error making while fitting the model in the training dataset. However, Soft Margin SVM doesn't allow any model to fit with errors. Allowing errors can benefit in making the model more generalized for new datsets. On the other hand, forcing strict margins may result in better performance for the training dataset,but may overfit when it is applied to a new dataset.

Types of SVM

There are mainly two kinds of SVM:

1. **Linear SVM**: Linear SVM is mainly used for classifying linear datasets. It has linear decision boundaries and it is efficient in dealing with large datasets with multiclass classification. It performs faster than non-linear SVM. However, it is not suitable for non-linear complex dataset.

2. **Non-Linear SVM**: Though Linear SVM performs faster and gives good result in linear datasets, there are a lot of datasets which are not linearly separable. Therefore, non-linear SVM is used for classifying these kinds of problems. Non-linear boundaries are used to deal with these kinds of problems. This is known as the kernel trick of SVM. These functions covert low dimensional input space to higher dimensional space for separating non-linear separation problem. It transforms extremely complex data, and then separates the data based on the defined labels or outputs.

Kernel Trick

Kernel tricks are the most famous part of SVM. The kernel computes the dot products of two vectors (x and y) in a feature space which is very high dimensional. This is the reason why kernel functions are occasionally called or known as "generalized dot product".

$$\text{maximize} \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j K\left(X_i^T.X_j\right) \tag{3.2}$$

$$\text{s.t } 0 \leq \alpha_i \leq C \text{ For all } i = 1, 2, \ldots., n \text{ and } \sum_{i=1}^{n} \alpha_i y_i = 0 \tag{3.3}$$

When kernel trick is applied, it means kernel function simply substituting the dot product of the two vectors.

Polynomial Kernel

The polynomial kernel is commonly used in SVM and other kernelized models. It induces space of polynomial combinations of the features up to certain degree. The polynomial kernel is defined as:
Usually, the polynomial kernel can be defined as shown in equation 3.4 :

$$K\left(X_1, X_2\right) = \left(a + X_1^T X_2\right)^b \tag{3.4}$$

Here, b = degree of kernel & a = constant term.

The dot product vector of the two vectors are simply calculated by increasing the power of the kernel, in case of polynomial kernel.

Gaussian Kernel

Gaussian Kernel or Radial Basis Function (RBF) Kernel is another popular Kernel method used in Support Vector Machine models. RBF kernel is a kind of function, the value of which depends on its from its origin or from some other point. The formula for RBF kernel is given in equation 3.5:

$$K\left(A_1, A_2\right) = \text{exponent}\left(-\gamma \left\|A_1 - A_2\right\|^2\right) \tag{3.5}$$

Here, $||A_1 - A_2||$ = Euclidean distance between $A_1 A_2$
By making use of the distance in the original space we calculate the dot product (similarity) of A_1 and A_2.

3.1.2 K-Nearest Neighbors

K-Nearest Neighbors (KNN) is a basic supervised machine learning algorithm that stores all available cases and characterizes new cases which are dependent on a closeness measurement like the distance functions. It is an algorithm that is used for both classification and regression. Although, it is used more frequently in solving the classification issues.

K is denoted as the number of nearest or closest neighbors in KNN. K is the key deciding factor in this algorithm. The algorithm is simply known as nearest neighbor algorithm in the cases where K=1. K is generally an odd number.

Suppose, the label for which prediction is needed is A1. Then we have to find the k nearest points to A1. After that the points need to be classified by majority vote of its k neighbors. Each of the object will vote for their own class and the class which will have greater number of votes will be considered as the prediction. To find out the closest similar points, distance between points has to be calculated using measures like Euclidean distance, Manhattan distance, Hamming distance and Minkowski distance. Step wise, we can divide KNN into three steps in the following order:

1. The distance between point calculation.
2. Determining the closest neighbors the found distance.
3. Voting for labels.

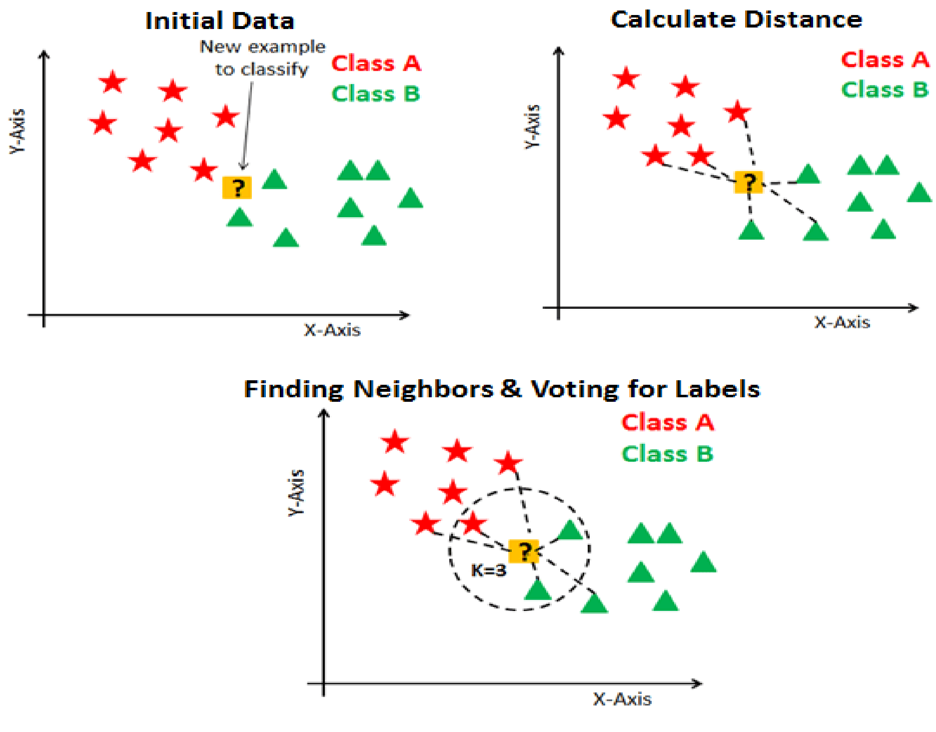

Figure 3.4: K-Nearest Neighbor Algorithm
[26]

The distance formulas for K-NN are given below:

- Euclidean Distance:

$$\sqrt{\sum_{i=1}^{k} (x_i - y_i)^2} \tag{3.6}$$

- Manhattan Distance:

$$\sum_{i=1}^{k} |x_i - y_i| \tag{3.7}$$

- Minkowski Distance:

$$\left(\sum_{i=1}^{k} (|x_i - y_i|)^q \right)^{1/q} \tag{3.8}$$

However, this three equations are used for continuous variables. For categorical variables, Hamming distance function is used. The formula for Hamming distance is:

$$D_H = \sum_{i=1}^{k} |x_i - y_i| \tag{3.9}$$

When, x=y; D=0 and when, x$\neq y; D = 1$. The example for Hamming Distance in given in table 3.1.

Table 3.1: Hamming Distance Example

A	B	D
Boy	Boy	0
Boy	Girl	1

3.1.3 Neural Network and Multilayer Perceptrons

Neural network algorithms are one of machine learning algorithms, they follow a model based on human brain and nervous system and based on linear algebra. They consist of different layers for data analysis, learning and pattern recognition. The pattern neural network algorithms identify are numerical which are accommodated in vectors. All real life data of any kind such as audio, video, letters and so on has to be converted into numerical vectors for neural network algorithms to recognize pattern.

A neural network has at least three layers of neurons: The input layer, the hidden layer(s) and the output layer. The hidden layer or layers are placed in between the input and the output layer. In case of multiple hidden layers, all hidden layers are interconnected. As the neural network gets "trained" the data of the neurons are processed in a way so that the network comes up with an accurate prediction. Figure 3.5 shows different types of neural network layers.
There are usually two kinds of neural networks:

1. Classic Neural Network

2. Deep Neural Network

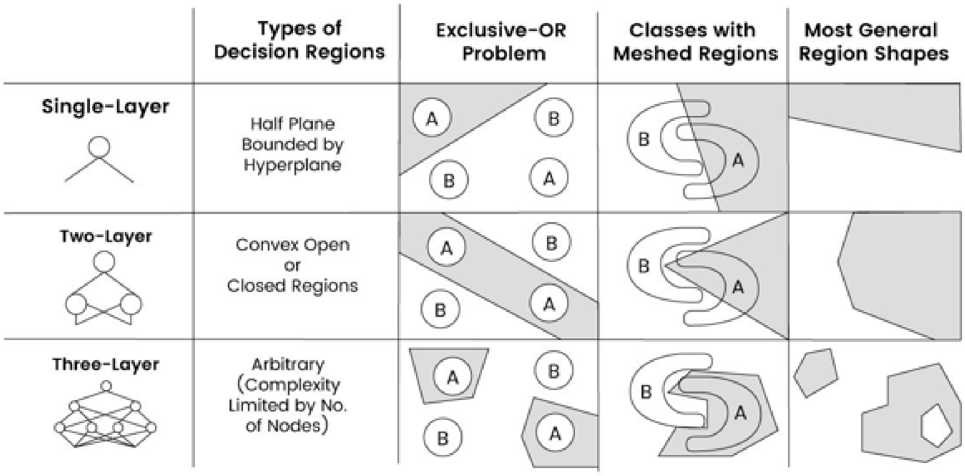

Figure 3.5: Different Neural Network Layers
[27]

Classic Neural Networks or Multilayer Perceptrons (MLP)

In MLPs, data transmits in one direction through multiple layers, which are interconnected. MLPs generally consist of one or more layers of neurons. Informtaion is fed into the input layer, there maybe one or multiple layers between where the data will get fine-tuned, and finally the output layer, which is also called the visible layer will give the prediction.

MLPs are effective in classification problems, i.e, where the input data set has a class or label. They also perform well in regression prediction. They predict a real-world valued quantity in regression problems given a set of input data in tabular form in a spreadsheet or in a CSV format file.

MLPs are supervised machine learning algorithms. They are trained on sets of input-output pairs and eventually learn model the inter-dependency between the pairs. To train the data parameters or the biases have to be adjusted so that the error percentage remains minimum. The technique used for these adjustments is called "Backpropagation".

Deep Neural Network

Deep neural networks are the complex neural networks with many layers (as many as one thousand). Each layer also has more neurons. Because of its number of layers and neurons it can handle more complex tasks. However, the data may take a longer period to be "trained". With the rapid increase in the capacity of modern GPUs, deep neural network has become extremely feasible in recent times.

3.2 CSI Data Overview

Channel properties in communications that doesn't require a wire are known as CSI. It supports IEEE802.11 and following standards. It portrays a finer grained wireless links in amplitude and phase, in comparison to RSSI [28]. CSI basically conveys how a signal travels to the receiver from the transmitter. The amplitudes and transitions of each subcarrier that are portrayed through channel measurements is calculated using the equation -

$$H(f_k) = ||H(f_k)||e^{j\angle H(f_k)} \tag{3.10}$$

In equation (3.10), $H(f_k)$ represents the complex value of channel frequency response in the format of CSI with central frequency of f_k, $||H(f_k)||$ is the amplitude and $\angle H(f_k)$ is the phase shift on the k-th path caused by a propagation delay [29]. Therefore, a set of CSIs $H(f_k)$ reveals K samples of Channel Frequency Responses at subcarrier level. These sample CFRs have been used in wireless communication studies like because of its dependability and improved efficiency [30]. Typically, the CSI is assessed by the receiver then gives evaluation to the sender. CSI can be categorized into two types – instantaneous and statistical. It can also be classified as CSI in the transmitter (SCIT) and CSI at the receiver (CSIR). Changes in wireless channel are constantly under surveillance by WiFi NICs which classifies the frequency response of the wireless channel. Let us assume A(f, t) as the frequency domain of transmitted signal and B(f, t) be that of received signal with carrier frequency f. These two signals can be expressed as –

$$\mathrm{B(f,t) = H(f,t) \times A(f,t)} \tag{3.11}$$

Here H(f, t) refers to the complex channel frequency response (CFR) values. These complex CFR values are for carrier frequency f calculated at a certain time denoted by t. These CFR values are contained by CSI measurements. Let's consider N_{TX} as the number of transmitting antennas and N_{RX} be that of receiving antennas. CSI is generally calculated on 30 chosen OFDM subcarriers for a received frame that abides by IEEE 802.11 standards. Each frame consists of 30 matrices whose dimensions can be denoted as N_{TX} X N_{RX}. The CFR value between two antennas at a fixed OFDM subcarrier frequency at a particular time is represented by a matrix in every entry. Hence, CSI streams refer to the time-series of CFR values for OFDM subcarrier and the two antennas. Consequently, 30 X N_{TX} X N_{RX} CSI streams are present in each time-series of Channel State Information values.

Chapter 4

System Implementation and Design

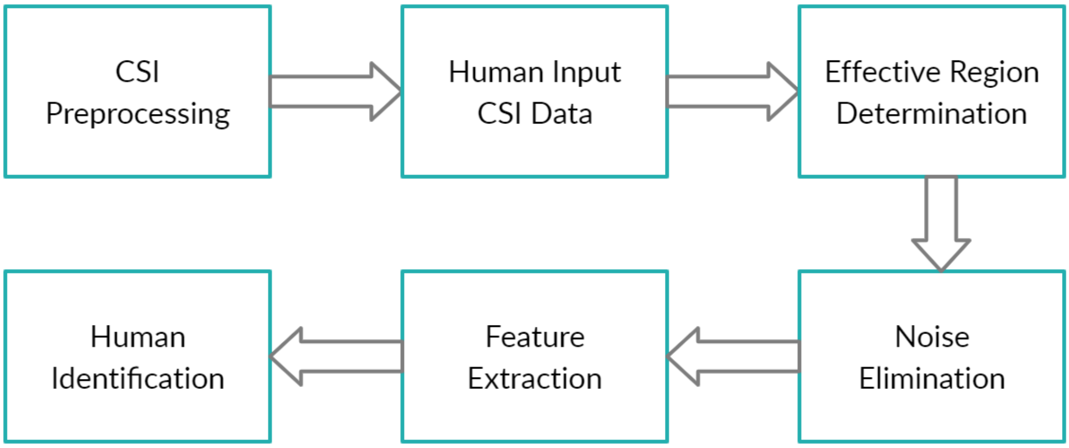

Figure 4.1: Proposed Work Flow of the System

Figure 4.1 showcases the proposed flow of work procedures which we followed. These implementation procedures along with the methodology and overview of CSI data are discussed in this chapter.

4.1 CSI Preprocessing

We have to process our data before taking any input. CSI preprocessing is one of the major challenges of our thesis. We needed to create an environment so that we could obtain CSI data. Channel State Information is a physical layer information. From past works we know that, it is possible to acquire Channel State Information values from COTS Wi-Fi network interface cards like Intel 5300 NIC [31] and Atheros 9390 wireless chipsets [32]. However, for our system we have used a desktop PC with INTEL core i3 processor and 4 gigabytes of RAM and Intel 5300 Network Interface Card (NIC) installed in its motherboard as the receiver card and a TENDA router as the transmitter. Our network interface card has 3 antennas and the router has 1 antenna. So, we used Single Input Multiple Output(SIMO) communication and 20

MHz channel width and thus we got 30 subcarrier information. After we were done with the hardware setup, we installed the build tools, Linux development headers and the GIT client. Then, we installed the modified wireless driver on our Linux OS Ubuntu 14.04.1. The Kernel tag we used is : Ubuntu 3.13.0-32.57. After that, we installed the modified firmware and next we defined the data rate to get CSI data. Then, we modified the wireless driver to obtain CSI data. After that, we installed the modified driver into our module updates directory and also obtained the CSI Tool supplementary materials. And at last, we built "log_to_file", a command line tool that writes CSI obtained via the driver to a file. During our whole installation process, the Wi-Fi network was kept open, the router was not password protected. The admin panel of the router was put in HT(high throughput) mode. The above procedures were done on Ubuntu terminal with the help of Halperin, Daniel, et al.[31] and Sen, Souvik, et al.[32]. Figure 4.2 shows the CSI data received in UBUNTU terminal after building the command line tool "log_to_file" and using the "ping" command at a fixed datarate.

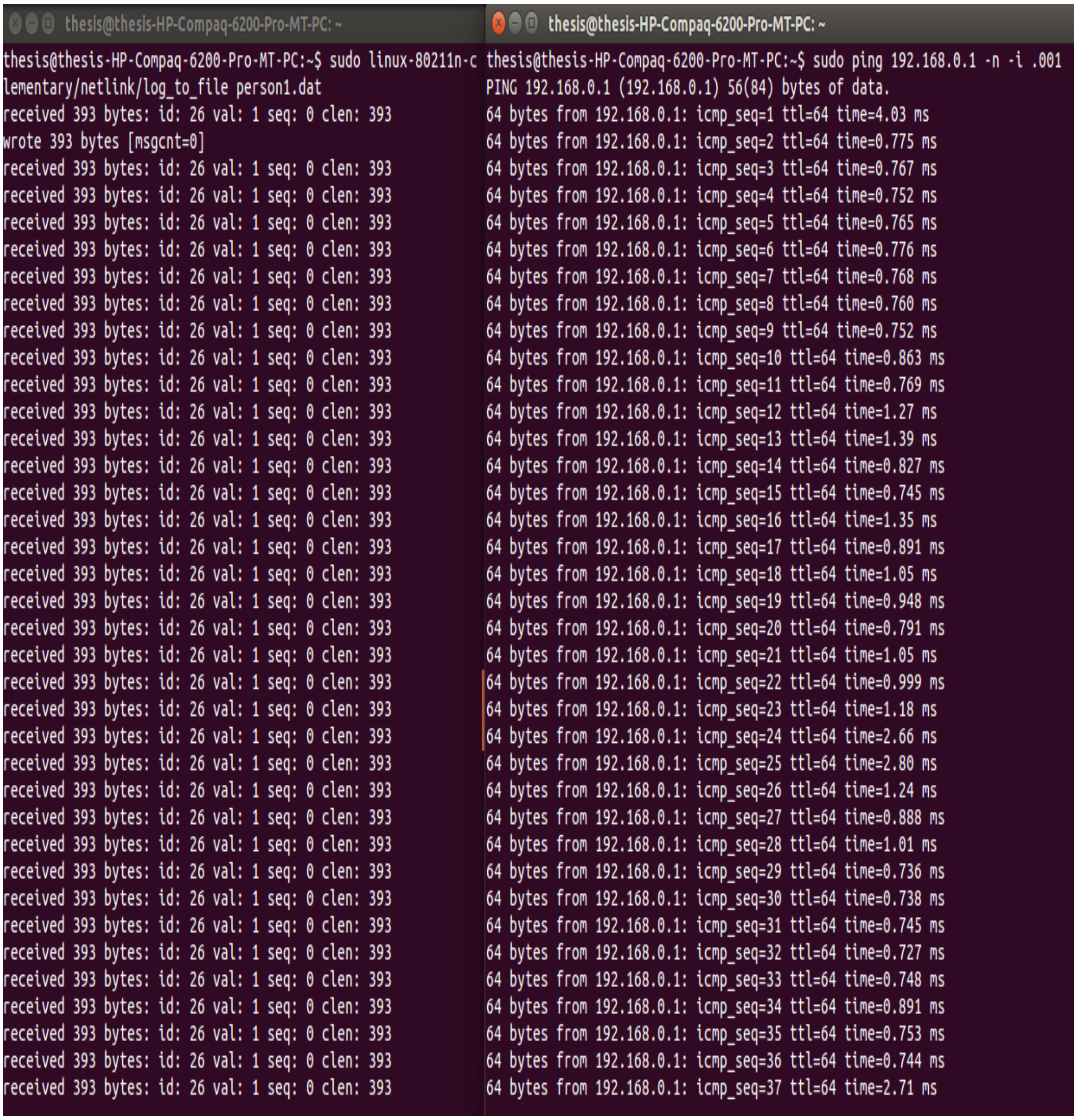

Figure 4.2: CSI Data Receiving and Ping Command in Ubuntu Terminal

4.2 Experimental Setup and Human Input CSI Data

After creating the environment for taking CSI data, we took the human inputs for identification purpose. There were two core devices for our system experiment. They are: Tenda F3 Router and Intel 5300 NIC in our motherboard. Both the devices were kept 80cm above the ground and were placed parallely with a distance of 150cm feet between them. The two devices were put inside a room and the human subject walked in an open space between the devices. In the setup, the transmitter sent packets and the receiver(Network Interface Card) received them via the 3 antennas it has. When a person walked through the path between the router(transmitter) and the NIC(receiver), the signal got affected by the human body and there was a change in the signals received from the signal transmitted. The fundamental idea was, as gait, walking style, body shape of each human being is different, the change in the signal will vary from person to person too. Thus,it is possible to identify human being analyzing those signals. We managed 50 person for our system experiment and asked them to walk between the path between the router (T_x) and the NIC (R_X). The transmission rate would be 1000 packets per second from the T_X to R_X and to do that we used ping command. We took 25 walking samples of each person. The CSI data of each sample gave us amplitude and phase information. we later used these CSI data for further processing. Also, another thing worth-mentioning is that, during our experiment all WiFi devices were fully functioning. Figure 4.3 depicts the experimental setup we have organized for our experiment.

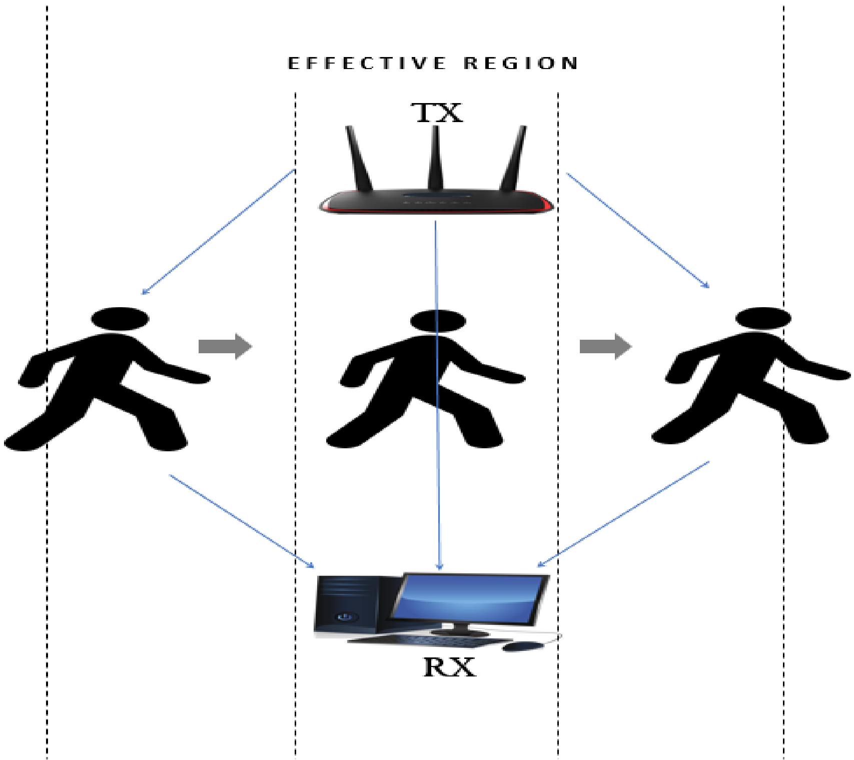

Figure 4.3: Experimental Setup

4.3 Effective Region Determination

To get the effective region, we recommended the distance that people will cover while walking between the router and the transmitter, so that we can get the actual walking sample of the person and get the values of the effective region. The R_x and T_x are directly connected in a straight line and it is also known as the focal area. This region is the best possible way to identify human uniquely. Before selecting the region, we need to consider the duration of the effective region in the time series. A lot of information will be missed if we keep the time-series duration too small and we will get unwanted misinformation if the duration is too large. Also, it was really crucial for us to be able to determine the start point and the end point very precisely.

In our approach for effective region selection, we followed the procedures done by Nipu, Md Nafiul Alam, et al.[22]. The effective region will be same for all receiver antennas. So, to determine the effective region, the CSI values from one receiver antenna was considered. To start with, a person's walking sample was taken. We then divided the whole frame into short frames. After that, the energy for each frame was calculated and for that, we considered one frame at a time. Each frame had 50 packets in them. Among those 50 packets there are 30 sub-carrier values. We get the energy of a frame by taking the mean value. After calculating the energy of all frames, the whole sample's mean energy was calculated.

The energy of a frame had to be compared to the mean or average energy of the whole sample to decide the start point. If the mean energy of the whole sample is less than the energy of a particular frame, we then assumed that the frame is in the effective region. After that, the effective region's start point's packet number was calculated.
To determine the endpoint, the entire sample's half length was considered and then the start or beginning point was summed with this half length. Throughout our experiment, we considered the effective region to be at least 50% of the whole sample. Figure 4.4, 4.5 and 4.6 show our startpoint and endpoint determination process for 3 random persons from our dataset.

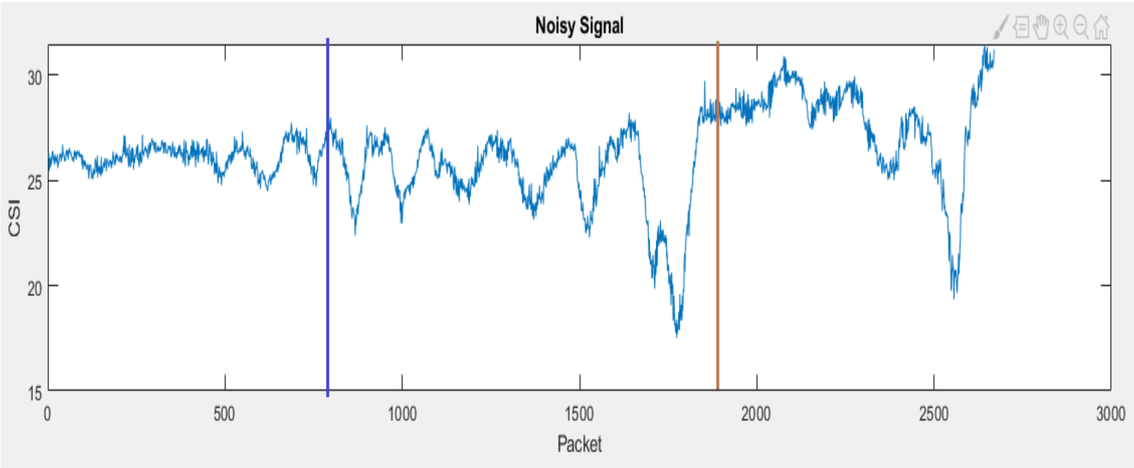

Figure 4.4: Effective Region Selection (Person1)

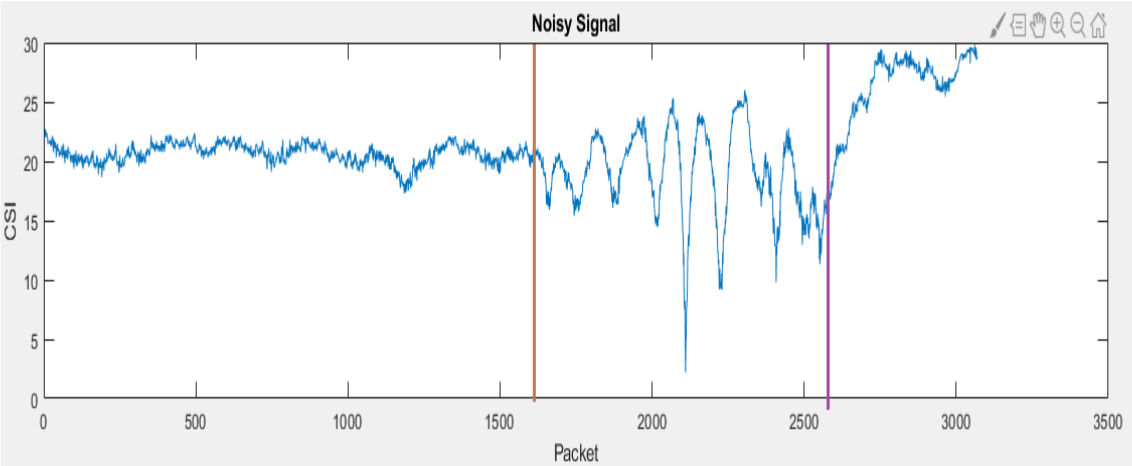

Figure 4.5: Effective Region Selection (Person2)

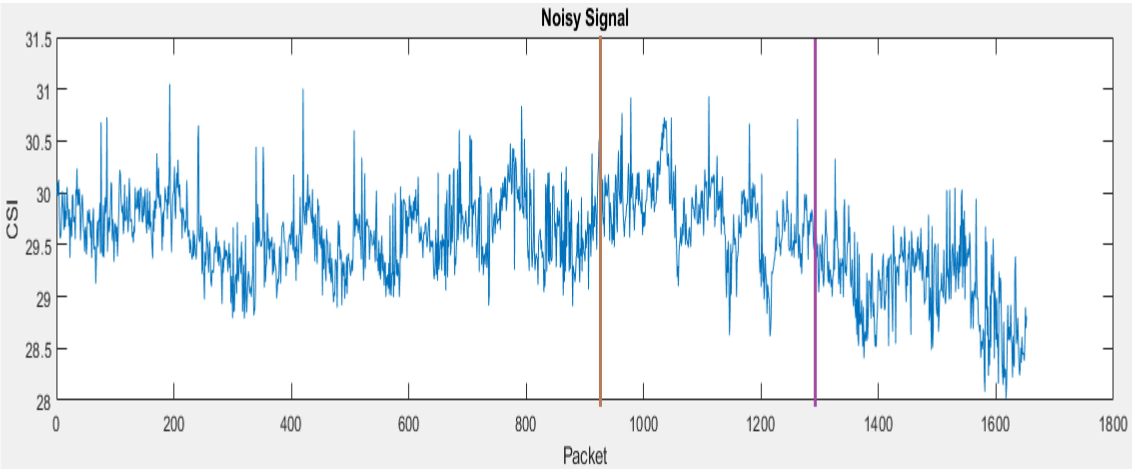

Figure 4.6: Effective Region Selection (Person3)

4.4 Noise Elimination

After finding the effective region our challenge is to remove the noise. The CSI signal data that we get is full of noise. Add to that, the CSI data will capture ambient noise from other radio frequency transmissions from nearby if nobody is passing through the route. These data are useless and have to be removed for the betterment of accuracy in identification. Therefore, we need to remove the noise to get a better result. We have followed the noise removal procedure from the work of Nipu, Md Nafiul Alam, et al. [22] We have used Butterworth Low Pass Filter for filtering our signal and removing high frequency noise. Butterworth is used for noisy signals. Its objective is to achieve an utmost flat frequency response in the passband. Human walking causes variation in the frequency of CSI time series and according to [13], the frequency for this variation is around 10hz. So, motivated by them, the cutoff frequency we have used is also 10 Hz. We have used second order filter and we have used MATLAB for processing our input data for identification. In MATLAB, the butterworth filter was created by using the command below:

$$[b,a] = \text{butter}(n,wn,ftype);$$

Where:

n = order number of the filter,
wn = denotes the cutoff frequency,
ftype = determines whether it is a low pass filter or a high pass filter.

In our case, order number, n = 2, wn = 10, filter type = LOW pass. The transfer function co-efficients are denoted by b and a.
We can express the transfer function for digital filter in terms of b and a, as shown in equation 4.1:

$$G(z) = \frac{B(z)}{A(z)} = \frac{b_0 + b_1 z + b_2 z^M + ... + b_m z^M}{a_0 + a_1 z + a_2 z^M + ... + a_m z^M} \tag{4.1}$$

For instance, let us assume that we want to filter the data z, the command we would be needing for this is:

y = filter(b,a,z);

The equation for the above command line is:

$$y_j = \sum_{i=1}^{N} b_i X_{j-1} + \sum_{i=1}^{M} a_i Y_{j-1} \tag{4.2}$$

Below in figure 4.7 to figure 4.12, we are providing three instances of signal state before and after noise removal:

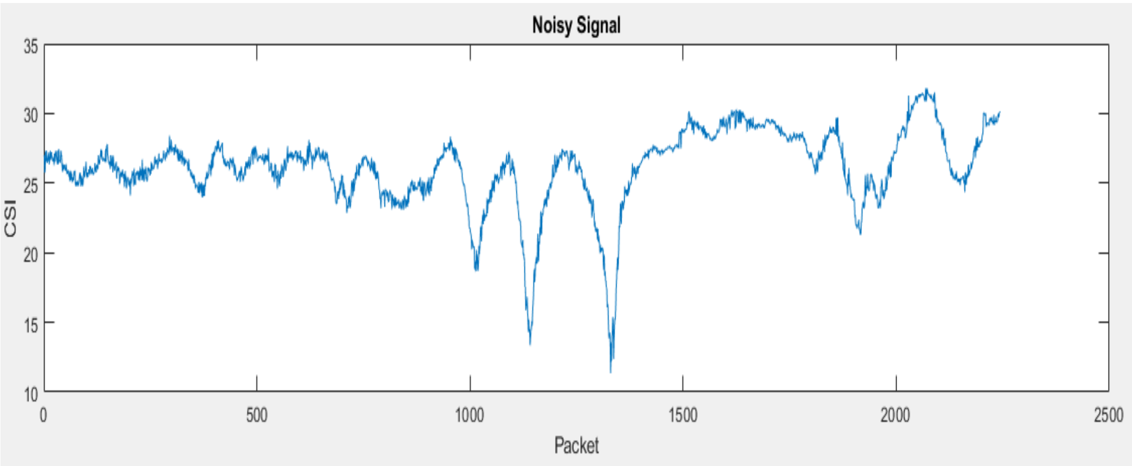

Figure 4.7: Noisy Signal(Person1)

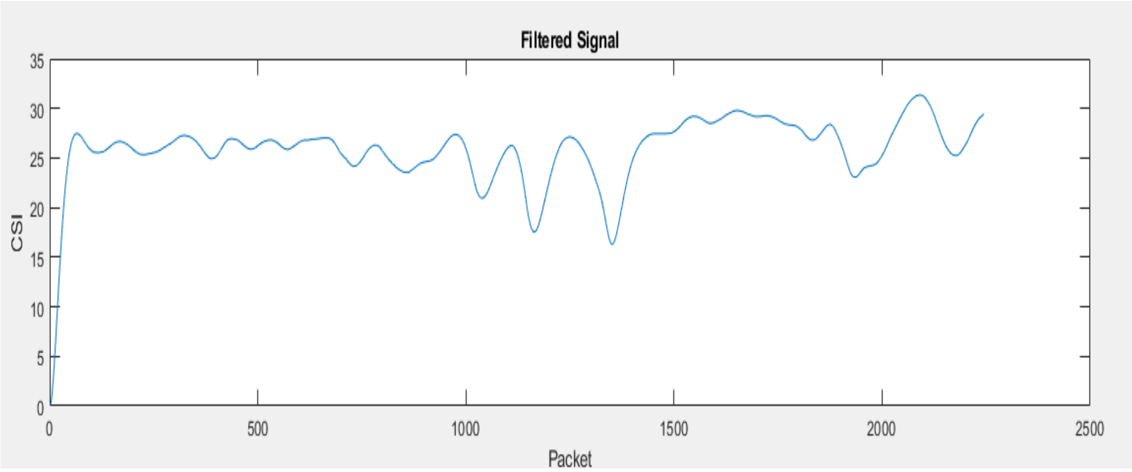

Figure 4.8: Filtered signal (Person1)

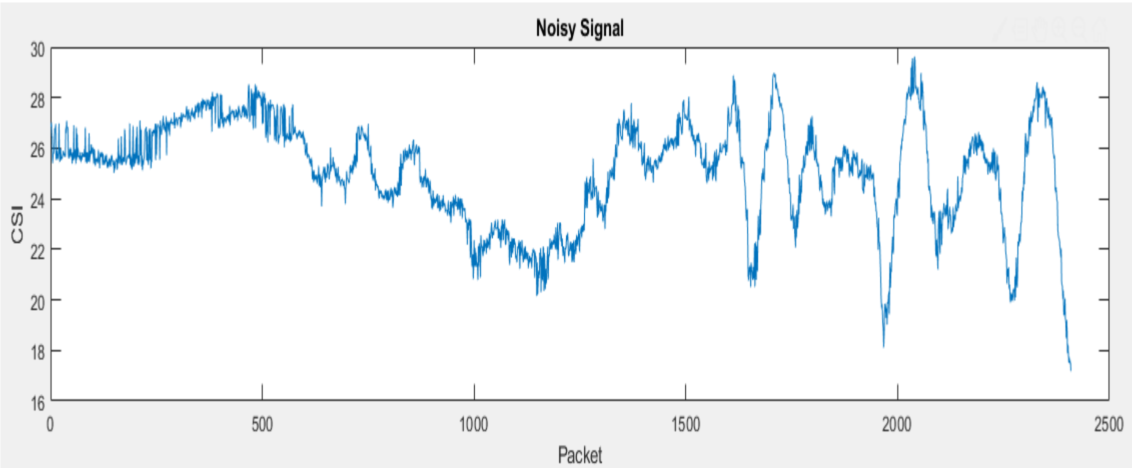

Figure 4.9: Noisy Signal(Person2)

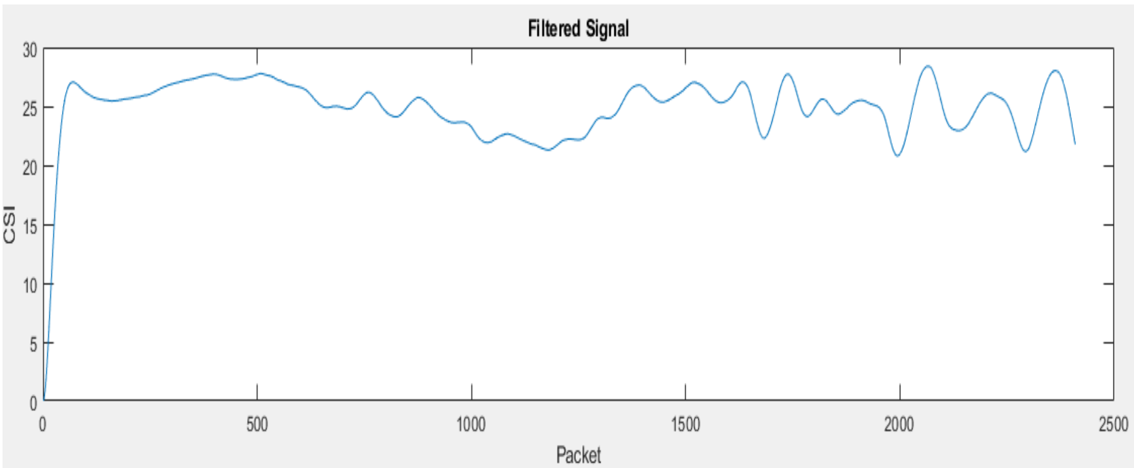

Figure 4.10: Filtered signal (Person2)

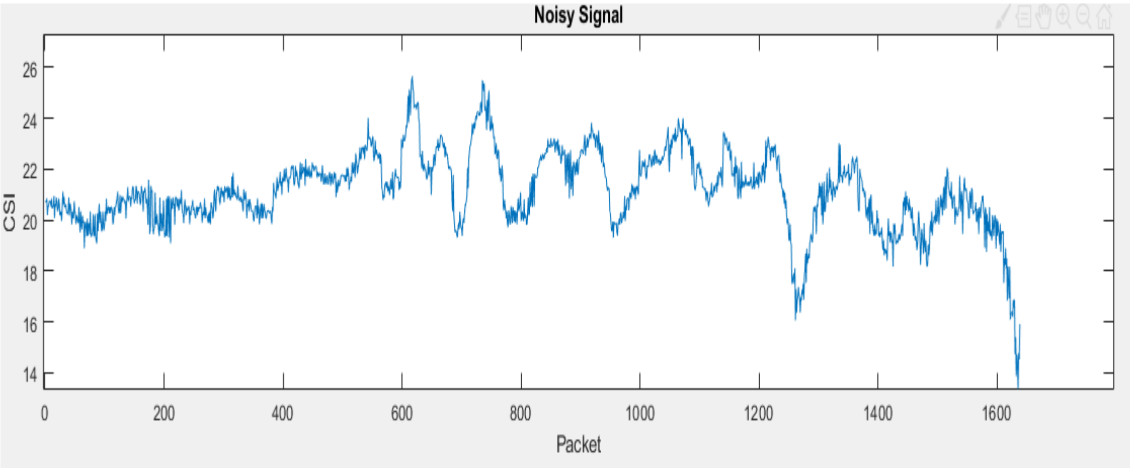

Figure 4.11: Noisy Signal(Person3)

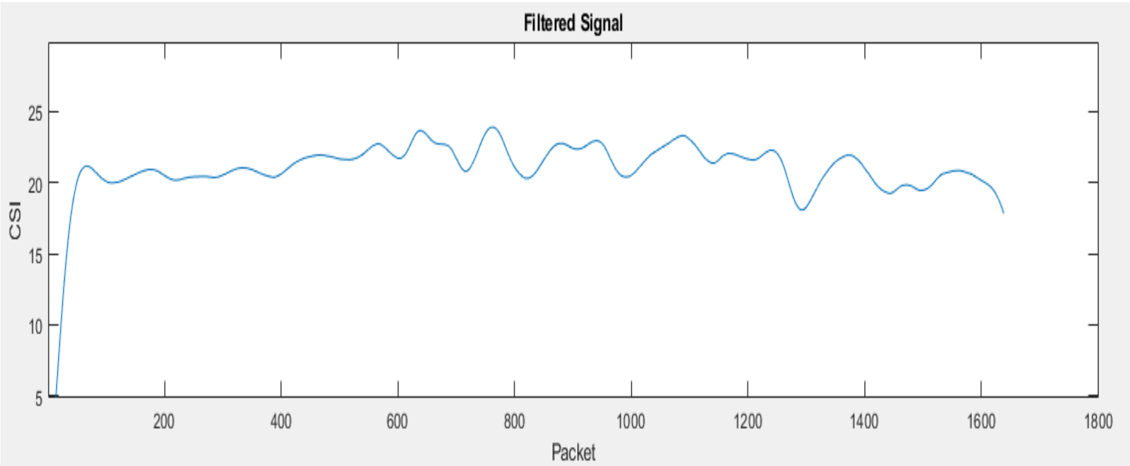

Figure 4.12: Filtered signal (Person3)

4.5 Feature Extraction

After Noise Elimination, we needed to select features that would give us information about each individual's unique walking style and gait analysis so that that they can be identified distinctively. We extracted 10 unique features which would helped us in identify humans uniquely. As discussed earlier, there are 1x3x30=90 sub-carrier information for each one of our packets. The statistical features that we have extracted for each sample are extracted from each and every 90 data streams collected from each packet. We extracted both time-domain features and frequency-domain features. The time-domain features we have extracted includes skewness, maximum, minimum, median and kurtosis. Highest FFT and peaks and kurtosis are our frequency domain extracted features. We have used these features as human identification classifiers in our implemented algorithms. Name and definitions of all of our features are given below in table 4.1.

Table 4.1: Feature and their descriptions

Features	Definition
Skewness	Measurement of the asymmetry of the amplitude data around the sample mean
Mean	The average of amplitude value
Maximum	The maximum from the sub-carrier amplitude values among all sub-carrier packets
Minimum	The minimum from the sub-carrier amplitude values among all sub-carrier packets
Median	The specific amplitude value dividing the data into two halves: higher and lower.
Kurtosis	The sharpness of the peak of the amplitude distribution of CSI data
Energy	Calculation of total energy in every frequency. It is denoted as: $$E = \sum_{i=1}^{N/2} (\text{magnitude})^2 \quad (4.3)$$
Highest FFT and Peaks	Highest frequencies in the signal with their amplitude value
Standard Deviation	The difference between maximum signal value and the mean signal value
Variance	The average of the squared differences from the mean value

4.6 Human Identification Classifiers

The last step of our system implementation is human identification with classifiers. We have taken 25 walking samples for each of the 50 persons and after extracting 10 distinctive features for each of them, we trained the classifiers with our dataset. We have chosen three machine learning based algorithms and they are - SVM, KNN, and MLP. The reason behind choosing these three algorithms is, they suit our supervised non-linear dataset. For example, if we used decision trees, increasing number of nodes would have resulted in overfitting and accuracy would have been less as the tree for our data would have been deep. On the other hand, algorithms like Naïve Bayes and Logistic Regression works better with linear dataset. The tools we used for our experiments are:

- Tensorflow: A library built with joint collaboration of Google and Brian Team. Used in Machine Learing for almost every Google application, it is a computational library for writing new algorithms. It represents data in matrices of N-dimesnion

- Scikit-learn: One of the best Python libraries for complex data. It is related to NumPy and SciPy.

- Numpy: One of the most prominent and used libraries for machine learning that are available in Python. Tensorflow uses Numpy internally. Its Array interface is considered its best feature.

- Pandas: Another library in Python which is known for its high-level data structure and vast variety of analysis tools. It has numberos in-built methods for data manipulation and processeing. It also has time-series functinality.

- Mathplotlib: A very pouplar library for plotting in 2D and is used for designing figures in numerous formats and it provides cross-platform compatibility for these formats.

- Keras: Another machine learning library in Python which supports all models of a Neural Network like recurrent, embedding and so on. In its backend, Keras usese one of Tensorflow or Theano.

- Seaborn: It is a library in Python which was designed based on Matplotlib and is used in visualization of complext statistical models.

4.6.1 K-nearest Neighbors

The core deciding factor for KNN is determining the value of K. There is no strict rule for determining the value of 'K'. A small value of K has a higher influence and flexibility on the result. It will give a bias which is low and a variance which is high. However, it could result in over fitting. On the contrary, higher value of 'K' gives more voters to each prediction. As a result, it is more resilient to outliers and have boundaries displaying smoothness for decision making. Mean error was calculated for all the predicted values for selecting the best value of K for our experiment. The value of K here varies from 1 to 40. The mean error graph in figure 4.13 showed that for our dataset, K=1 gives better accuracy than the higher values of 'K'.

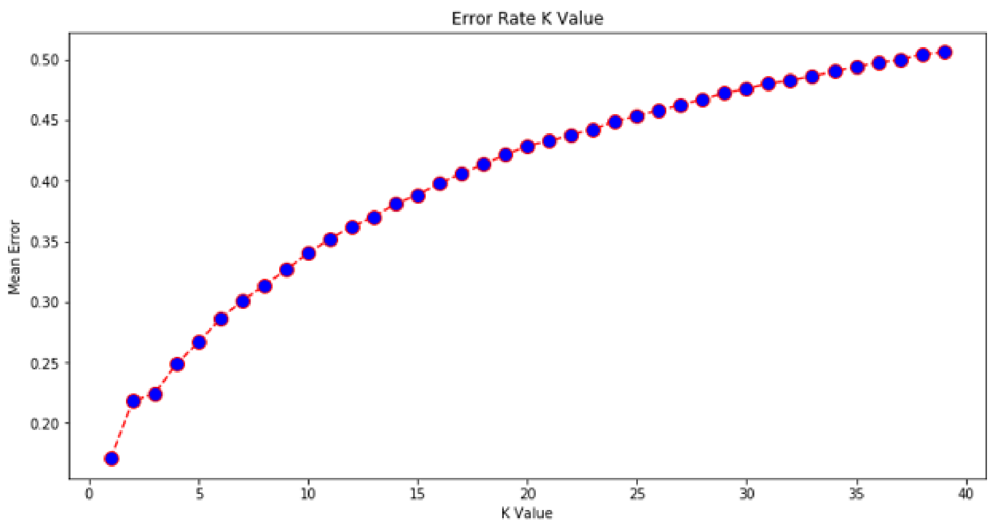

Figure 4.13: Determining the value of k

We have used Minkowski distance formula for finding distance between points in our model.

4.6.2 Support Vector Machine

As our dataset is non-linear, linear SVM algorithms would not have worked well with it. Therefore, we decided to use Gaussian/RBF kernel SVM for our human identification classification. There are two parameters for Gaussian SVM and they are gamma and C. The Gamma (γ) parameter defines the distance for closest training set example from the hyperplane. It is a parameter for non-linear hyperplanes. The increasing value of gamma results in overfitting and its decrease causes the model to underfit. The C parameter dictates the exchange of smooth margin and correctness of training set example classification. The increase in value for C results in overfitting and the model underfits when C is decreasing. The parameters value we have used are given below in table 4.2.

Table 4.2: Parameters of SVM

Parameters	Value
Number of features	10
Gamma	1.0
C	10000
Kernel	RBF

We have selected these parameters by tuning them with GridSearchCV from the sci-kit learn library as shown in figure 4.14.

```
In [12]: from sklearn.model_selection import GridSearchCV
parameter_candidates = [
  {'C': [1, 10, 100, 1000, 10000], 'gamma': [0.01, 0.1, 1, 10, 100], 'kernel': ['rbf']},
]
# Create a classifier object with the classifier and parameter candidates
clf = GridSearchCV(estimator=svm.SVC(), param_grid=parameter_candidates, n_jobs=-1)

# Train the classifier on data1's feature and target data
clf.fit(X_train, y_train)
print('Best score for data1:', clf.best_score_)
print('Best C:',clf.best_estimator_.C)
print('Best Gamma:',clf.best_estimator_.gamma)

Best score for data1: 0.9204596162427487
Best C: 10000
Best Gamma: 1
```

Figure 4.14: Tuning Parameters with GridSearchCV

4.6.3 Multilayer Perceptrons

A single layer neural network can only be used for linearly separable datasets. However, a Multilayer Perceptron can be used to represent convex regions. Therefore, Multilayer Perceptron is suitable for our non-linear dataset. Neural network cannot process categorical data. Thus, we have used label encoding to transform our string values to integer values. However, integer values have a natural ordered relationship between each other. These could lead to some problems when there is no ordinal relationship. Therefore, label encoding is not enough to use our data in neural network. Thus, we had to use One-Hot Encoding to which gives the network more expressive power to learn a probability for each possible label value. As we have used One-Hot Encoding for the output variable, it offers a better prediction than a single label. Then, we have scaled our dataset with Standard Scaler which standardize a dataset by rescaling the distribution of values setting the mean of observed value to 0 and standard deviation to 1. We have used one input layer consisting 10 nodes as our number of features are 10. Then, we have used four densely connected hidden layers with 512 nodes in each layer. We have selected ReLU as our activation function in each hidden layers. Our output layer consists of 50 nodes as we are classifying 50 people in our dataset. In this layer we have used "Softmax" as our activation function. After selecting these hyperparameters, we have selected our batch size as 350 and epoch as 100 for the model.

ReLU

ReLU is short form for 'Rectifier linear unit'. It is the most popular activation function for hidden layers of a neural network. ReLU suits non-linear dataset and that's why we have used it as it can enable us in using multiple layers and help us in backpropagating errors. The equation for ReLU is:

$$A(x_0) = max(0, x) \tag{4.4}$$

For an output the value of x has to be positive otherwise it will give zero.

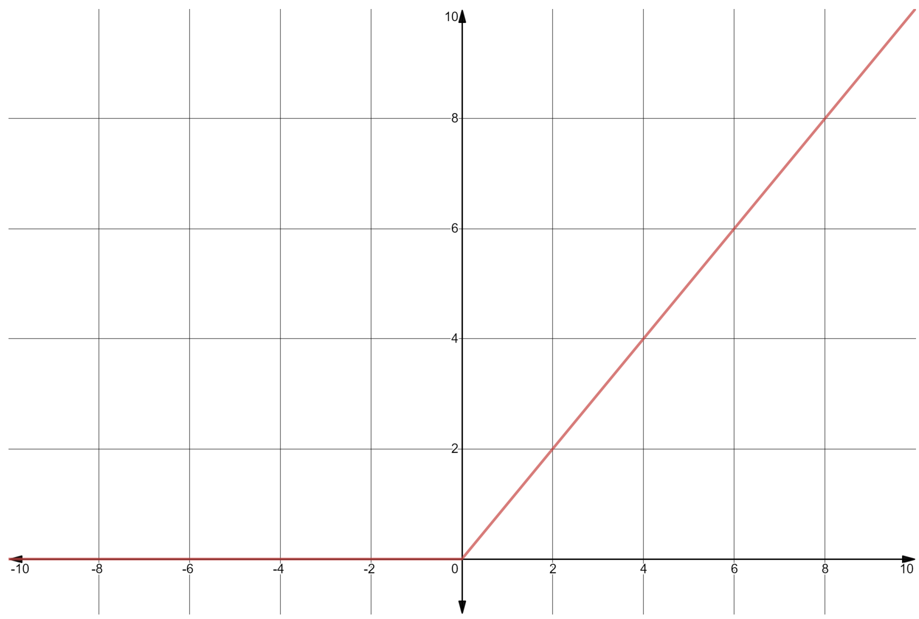

Figure 4.15: ReLU Activation Function Graph

Softmax

Softmax is one kind of sigmoid function which is really useful for multi-class classification dataset like ours. It is helpful in classification as it would give us an output for each class as 0 and 1. This is the reason for us choosing Softmax function. The Softmax function's equation is given below:

$$\sigma(\mathbf{z})_j = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}} \text{ for } j = 1, \ldots, K \tag{4.5}$$

Chapter 5

Results and Discussion

5.1 Human Identification

5.1.1 Performance Analysis

In this section, the performance of the proposed human identification model is evaluated. As mentioned earlier, we have used KNN, SVM and MLP to identify human uniquely. Among the implemented algorithms, we got accuracy of 96.05% in MLP, 94.09% in SVM and 93% in KNN for a group of 10 people. Every person's height, body shape, gesture is unique. That's why as the group size of people increases, the dataset become more complex and as a result the accuracy for each algorithm decreases. Here, MLP performs better than other implemented algorithms and it shows 89.84% of accuracy for 50 people. However, it has higher complexity than other two algorithms, which takes more time to train the dataset and show result. In this perspective, KNN performs faster but has lower accuracy than SVM and MLP.

Table 5.1: Result Accuracy of Algorithms

Number of People	KNN	SVM	MLP
10	93%	94.09%	96.05%
20	88%	91.64%	92.06%
30	85%	90.21%	91.12%
40	83%	88.69%	90.44%
50	83%	88.15%	89.84%

Table 5.1 illustrates the accuracy of the classifiers in identifying human from the dataset. Here, KNN shows the lowest accuracy of 83% for group of 50 people among the implemented classifiers but it has lower complexity and it can generate result quickly compared to other algorithms. KNN is preferable for small group of people as it has 93% accuracy for group for 10 people and complexity is low which is quite good for a smart home security system. On the other hand, for large group of people, SVM and MLP perform better and give 88.15% and 89.44% of accuracy respectively for 50 people. However, we need to compromise the complexity in this scenario.

5.1.2 Comparative Study

Figure 5.1 is the graphical bar chart representation of accuracy (%) among the implemented three algorithms. From the chart we can clearly see that the accuracy of different group size like 10, 20, 30, 40 and 50 people.

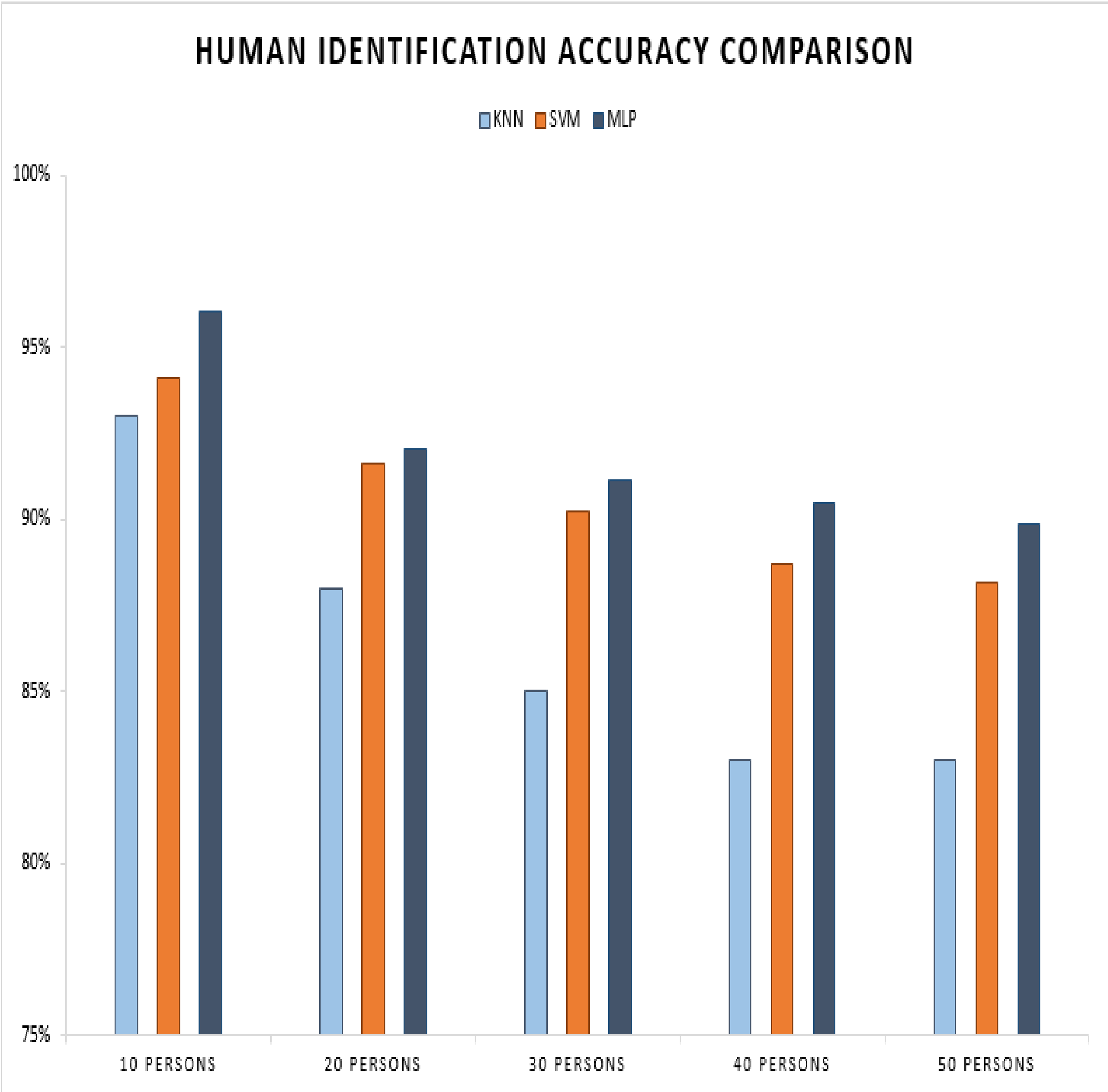

Figure 5.1: Human identification accuracy comparison chart

When the group size is 10 people, KNN gives accuracy of 93%, whereas SVM and MLP show an accuracy of 94.09% and 96.05% respectively. When we increase the group size to 20 people the accuracies decrease as follows: for KNN it becomes 88%, SVM drops to 91.64% and MLP becomes 92.06%. Moreover, when the group size is 30 people the accuracy of KNN goes to 85%, SVM becomes 90.21% and MLP goes to 91.12%. When the group size is 40 and 50 people, KNN gives 83% and 83%, SVM gives 88.69% and 88.15%, MLP gives 90.44% and 89.84% accuracy respectively.

Table 5.2: Classification matrix for a group of 20 people

Person	KNN(%)	SVM(%)	MLP(%)
Person01	88	87	92
Person02	89	91	94
Person03	93	95	96
Person04	86	91	90
Person05	91	91	92
Person06	99	99	99
Person07	85	88	95
Person08	81	83	93
Person09	91	94	93
Person10	83	89	90
Person11	86	91	91
Person12	87	89	85
Person13	88	93	98
Person14	88	91	94
Person15	90	95	96
Person16	84	93	88
Person17	80	90	88
Person18	89	95	89
Person19	93	94	97
Person20	90	96	96

For identification purpose, we gathered 25 samples per person and for 50 people we collected 1250 data samples. We fed the dataset to the classifiers after extracting the features. Table 5.2 shows the classification matrix for different classifiers. In this case, in table 5.2 we have shown classification matrix for 20 randomly picked person from our dataset. This table illustrates the difference between the algorithm in how accurately they identify each person. For example, KNN can identify Person1 88 times, SVM identifies 87 times and MLP identifies 92 times out of 100 times.

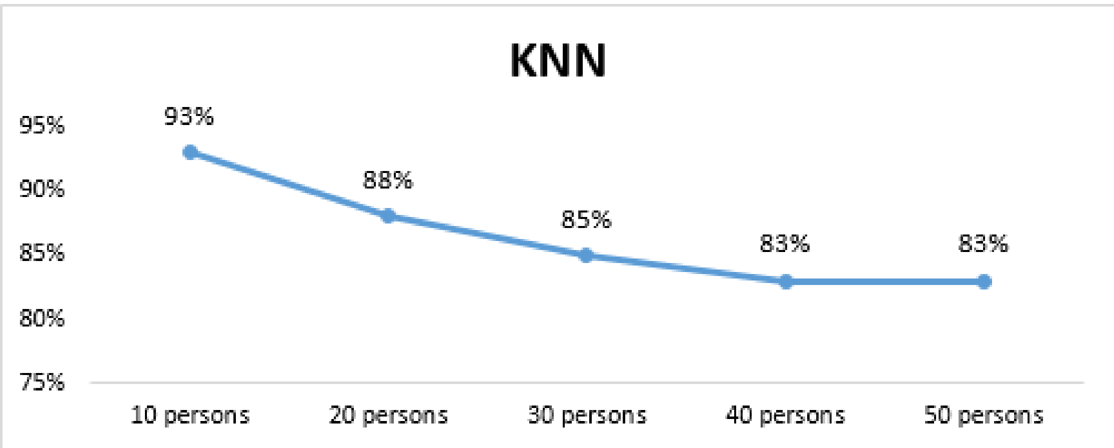

Figure 5.2: : Human identification accuracy line curve using KNN

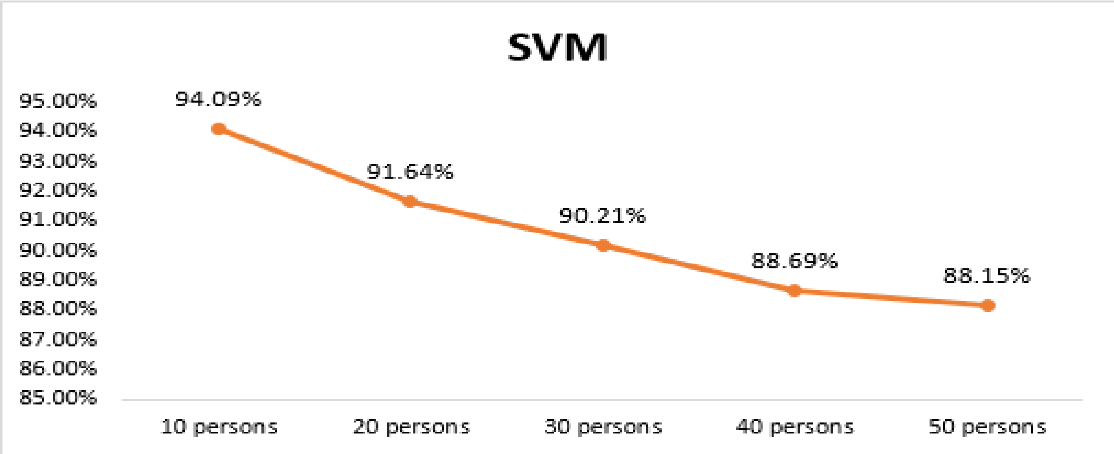

Figure 5.3: : Human identification accuracy line curve using SVM

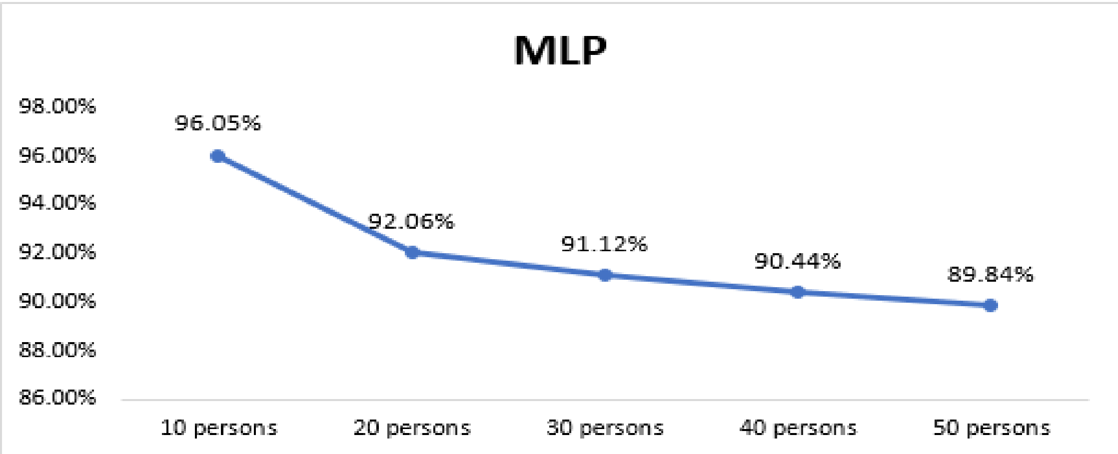

Figure 5.4: : Human identification accuracy line curve using MLP

Figure 5.2, 5.3 and 5.4 illustrate the graph of human identification accuracy using the three implemented algorithm. Here, X-axis represents the number of people and Y-axis represents the accuracy level. In these figures, we can see that accuracy declines gradually as the number of people increases. However, for KNN the drop of accuracy is lower after group of 30 people compared to the group of 10 to 20 people. The accuracy drop from 10 to 20 numbers of people is 5%. On the contrary, accuracy drops just 2% for group of 30 to 50 persons. From this, we can assume that, accuracy of the algorithm will drop slowly if we increase the number of people from 50. However, in SVM the drop of accuracy is lower after group of 40 people compared to the group of 10 to 30 people. The accuracy drops from 10 to 30 numbers of people is 3.88%. On the other hand, for 40 to 50 people the change in accuracy is fractional, which is 88.69% to 88.15%. Moreover, in MLP the accuracy drastically drops for 10 to 20 people, which is around 4%. However for 20 to 50 people the accuracy changes from 92.06% to 89.84%, which is around 3% and it is better compared to KNN and SVM. From this, we can assume that, accuracy of the algorithm will drop slowly if we increase the number of people from 50 to a higher number. Although, if we increase the dataset, then more unique body shape, height, gesture of different persons will be added in the dataset thus will result in a slight but gradual decline in the accuracy.

5.2 Gender Identification

In this section, the performance of gender-wise identification is evaluated. We have used the same KNN and MLP to identify gender from the same datasets. We implemented the algorithm in our collected data of 50 persons consisting of 39 male and 11 female. The results for gender identification is showed in table 5.3.

Table 5.3: Gender Detection Accuracy for Uneven Number of Male and Female

Gender	KNN	MLP
Male(39)	96%	97%
Female(11)	86%	92%
Overall Accuracy	94%	96%

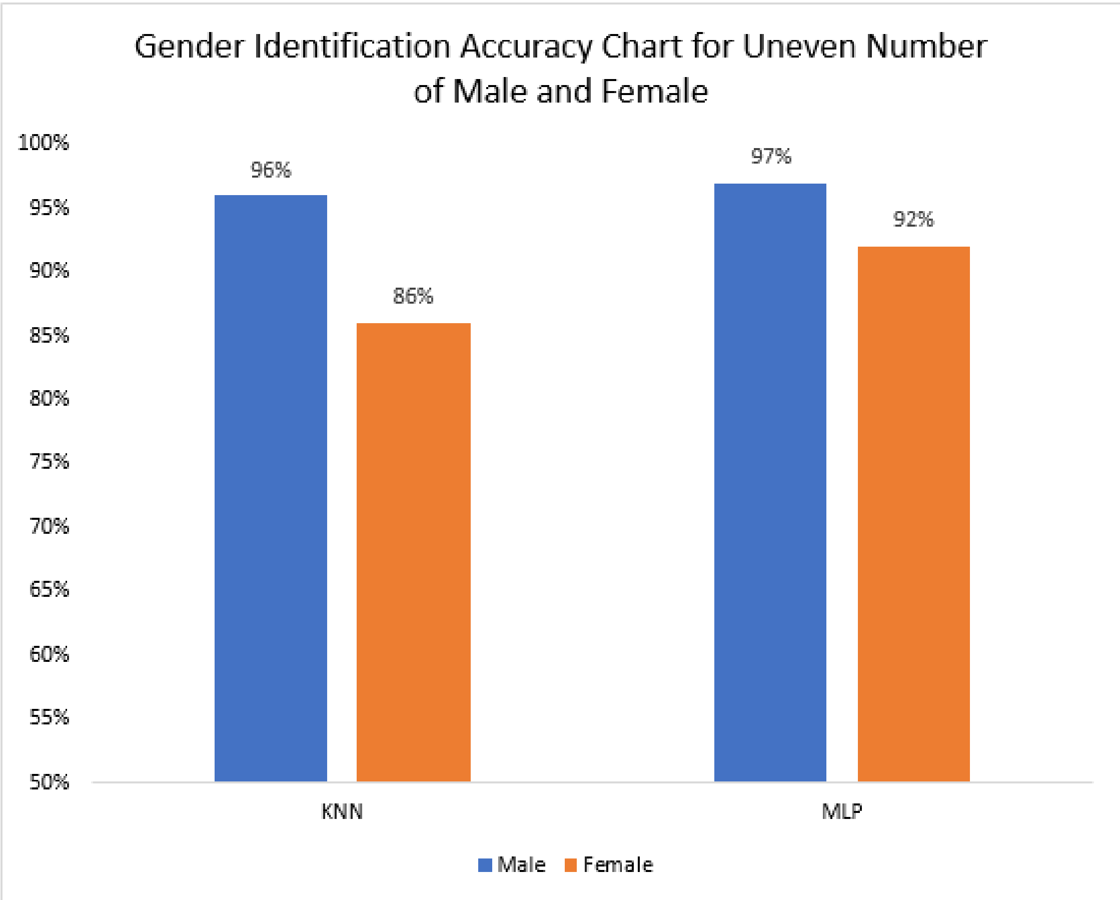

Figure 5.5: Gender detection accuracy comparison

From table 5.3 and figure 5.5 we can see that KNN gives an accuracy of 96% for male and 86% for female for group of 50 people. For MLP the accuracy is 97% for identifying male and 92% for female. Moreover, overall accuracy for classifying gender, KNN gives 94% accuracy and MLP gives 96% accuracy. We can clearly see that for identifying gender, MLP performs better than KNN in our dataset. Although KNN performs faster, it has lower accuracy than MLP.

However, if we scale down the uneven number of male and female, the accuracy becomes similar for gender identification of both male and female. Previously, we saw that there was a difference in accuracy when we used data samples of 39 male and 11 female. But, we scaled down the number of males to match that of female. As there were 11 females so we choose data of 11 random male and implemented the algorithms again.

Table 5.4: Gender Detection Accuracy for Equal Number of Male and Female

Gender	KNN	MLP
Male(11)	94%	97%
Female(11)	94%	97%

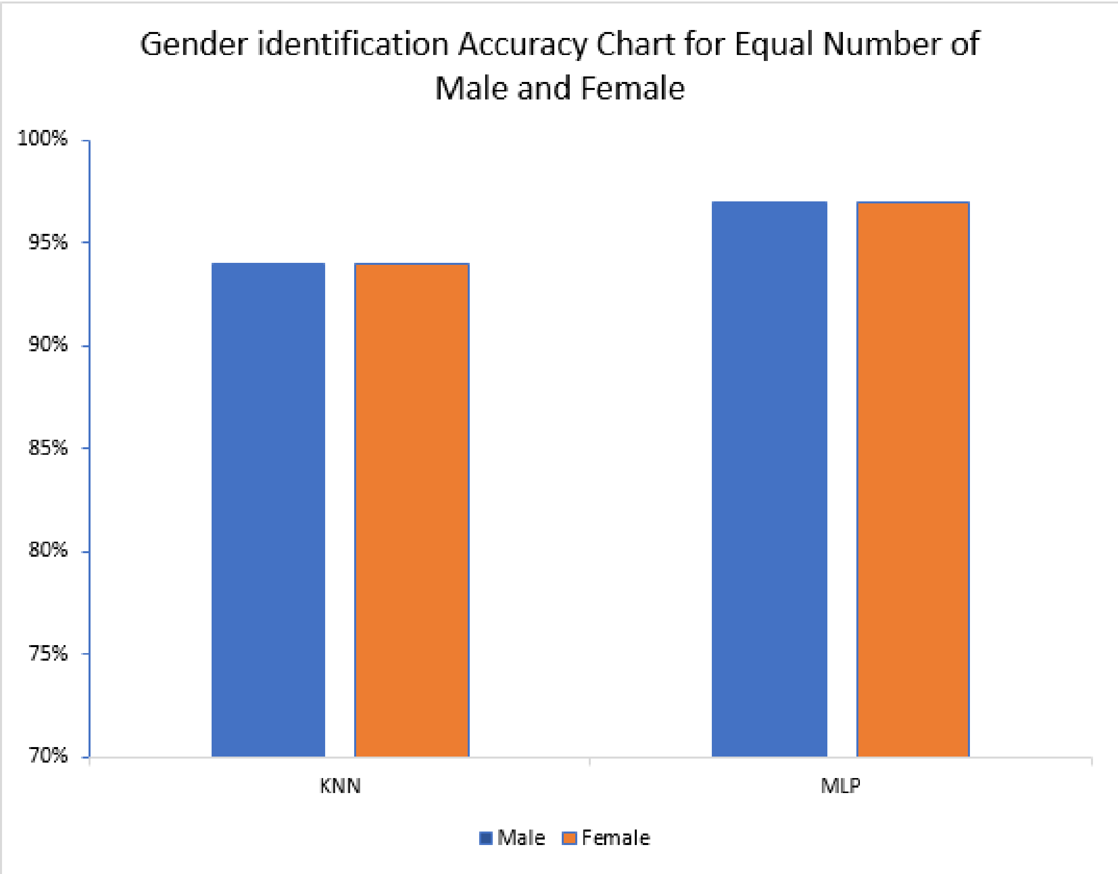

Figure 5.6: Gender detection accuracy comparison for equal number of male and female.

Table 5.4 and figure 5.6 show that KNN gives an accuracy of 94% for both male and female when we use the randomly picked 11 males and 11 females. For MLP the accuracy is 97% for identifying both male and female. We can clearly see that for identifying gender, MLP performs better than KNN in our dataset. However, KNN performs faster but has lower accuracy than MLP. As the dataset gets shorter, the gender identifying data does not get very complex. So, we get better accuracy.

5.3 Limitations

Despite some encouraging results, our system is not perfect. It has its own set of limitations too. We have listed few limitations of our system below. We hope to work more on this limitations in our future work.

- As CSI data is ambient or environment dependent, any changes in the environment would cause fluctuation in CSI data too. Therefore, we had to make sure that our environment was static.

- We have tested our system for 50 people. From our results, we have noticed that our system accuracy has an inverse relationship with the size of sample. However, the decrease in accuracy is really minimum for for every added ten people.

- We have taken 25 walking samples of each person in our search for better accuracy. However, our walking sample was predetermined. We haven't tested our system against people in wheelchair, a crouch, or somebody who have lost their limb or limbs.

Chapter 6

Conclusion and Future Works

To conclude, human identification for a larger number of people with absolute precision without violating their private spaces or without acquiring their bio-metric data which are vulnerable to misuse, still remains a big challenge. However, we know that each individual have their own distinctive body shape and manner of walking. So, in our work we have implemented a system that makes use of the unique distractions caused in the signals sent by the router by every person when walking through the effective region that we selected for them and we have processed these CSI data to identify them uniquely without violating their personal spaces. During our work, we have found that tree-based classifications like Random Forest and Boosted Decision Trees work less than SVM, MLP and K-NN for larger data groups. For K-NN, we got accuracy of 93% to 83%, we got upto 88.15% accuracy for SVM for 50 people and we got up to 89.84% identification accuracy for MLP. So, we can conclude that our results have been really encouraging and our system can be implemented in residential home setup and medium size offices.

We are hoping to work with a bigger group of people (at least 150 people) in future. We are interested to see our system work in bigger space, which we couldn't afford this time. We implemented our system inside a confined room and we are interested in expanding our system setup for a larger open space like a certain residential area or a campus of any educational institutes and test the accuracy of our system with human identification classifiers. We want to test the identification accuracy of our system with other existing machine learning algorithms too. Also, as handicapped persons are really important part of our society, another thing we plan to implement is, testing this system's accuracy with a dataset that includes people in wheelchair and people without limb or limbs. Add to that, we have separated our dataset into two parts : male and female and found similar results for gender identification when the dataset has equal number of male and female. However, in future we want to see if our system has a pattern of recognizing a certain gender more accurately than the other when we have a much larger dataset.

References

[1] T. Kivimäki, T. Vuorela, P. Peltola, and J. Vanhala, "A review on device-free passive indoor positioning methods", *International Journal of Smart Home*, vol. 8, no. 1, pp. 71–94, 2014.

[2] T. Teixeira, G. Dublon, and A. Savvides, "A survey of human-sensing: Methods for detecting presence, count, location, track, and identity", *ACM Computing Surveys*, vol. 5, no. 1, pp. 59–69, 2010.

[3] Y. Benezeth, H. Laurent, B. Emile, and C. Rosenberger, "Towards a sensor for detecting human presence and characterizing activity", *Energy and Buildings*, vol. 43, no. 2-3, pp. 305–314, 2011.

[4] S. K. Sharma, R. Agrawal, S. Srivastava, and D. K. Singh, "Review of human detection techniques in night vision", in *2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)*, IEEE, 2017, pp. 2216–2220.

[5] C. Wang, J. Zhang, L. Wang, J. Pu, and X. Yuan, "Human identification using temporal information preserving gait template", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 11, pp. 2164–2176, 2012.

[6] L. Wang, T. Tan, H. Ning, and W. Hu, "Silhouette analysis-based gait recognition for human identification", *IEEE transactions on pattern analysis and machine intelligence*, vol. 25, no. 12, pp. 1505–1518, 2003.

[7] P. Airey and J. Verran, "A method for monitoring substratum hygiene using a complex soil: The human fingerprint", *Fouling, cleaning and disinfection in food processing*, 2006.

[8] H. Wang, D. Zhang, Y. Wang, J. Ma, Y. Wang, and S. Li, "Rt-fall: A real-time and contactless fall detection system with commodity wifi devices", *IEEE Transactions on Mobile Computing*, vol. 16, no. 2, pp. 511–526, 2016.

[9] J. Liu, Y. Wang, Y. Chen, J. Yang, X. Chen, and J. Cheng, "Tracking vital signs during sleep leveraging off-the-shelf wifi", in *Proceedings of the 16th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, ACM, 2015, pp. 267–276.

[10] R. Schmidt, "Multiple emitter location and signal parameter estimation", *IEEE transactions on antennas and propagation*, vol. 34, no. 3, pp. 276–280, 1986.

[11] J. Zhang, B. Wei, W. Hu, and S. S. Kanhere, "Wifi-id: Human identification using wifi signal", in *2016 International Conference on Distributed Computing in Sensor Systems (DCOSS)*, IEEE, 2016, pp. 75–82.

[12] M. S. Aljumaily and G. A. Al-Suhail, "Towards ubiquitous human gestures recognition using wireless networks", *International Journal of Pervasive Computing and Communications*, vol. 13, no. 4, pp. 408–418, 2017.

[13] T. Xin, B. Guo, Z. Wang, M. Li, Z. Yu, and X. Zhou, "Freesense: Indoor human identification with wi-fi signals", in *2016 IEEE Global Communications Conference (GLOBECOM)*, IEEE, 2016, pp. 1–7.

[14] Z. Wang, B. Guo, Z. Yu, and X. Zhou, "Wi-fi csi-based behavior recognition: From signals and actions to activities", *IEEE Communications Magazine*, vol. 56, no. 5, pp. 109–115, 2018.

[15] X. Liu, J. Cao, S. Tang, J. Wen, and P. Guo, "Contactless respiration monitoring via off-the-shelf wifi devices", *IEEE Transactions on Mobile Computing*, vol. 15, no. 10, pp. 2466–2479, 2015.

[16] H. Li, W. Yang, J. Wang, Y. Xu, and L. Huang, "Wifinger: Talk to your smart devices with finger-grained gesture", in *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, ACM, 2016, pp. 250–261.

[17] K. Ali, A. X. Liu, W. Wang, and M. Shahzad, "Keystroke recognition using wifi signals", in *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*, ACM, 2015, pp. 90–102.

[18] Q. Pu, S. Gupta, S. Gollakota, and S. Patel, "Whole-home gesture recognition using wireless signals", in *Proceedings of the 19th annual international conference on Mobile computing & networking*, ACM, 2013, pp. 27–38.

[19] G. Wang, Y. Zou, Z. Zhou, K. Wu, and L. M. Ni, "We can hear you with wi-fi!", *IEEE Transactions on Mobile Computing*, vol. 15, no. 11, pp. 2907–2920, 2016.

[20] F. Adib and D. Katabi, *See through walls with WiFi!*, 4. ACM, 2013, vol. 43.

[21] F. Wang, J. Han, Z. Dai, H. Ding, and D. Huang, "Wipin: Operation-free person identification using wifi signals", *arXiv preprint arXiv:1810.04106*, 2018.

[22] M. N. A. Nipu, S. Talukder, M. S. Islam, and A. Chakrabarty, "Human identification using wifi signal", in *2018 Joint 7th International Conference on Informatics, Electronics & Vision (ICIEV) and 2018 2nd International Conference on Imaging, Vision & Pattern Recognition (icIVPR)*, IEEE, 2018, pp. 300–304.

[23] *Understanding support vector machines algorithm (along with code)*, https://www.analyticsvidhya.com/blog/2017/09/understaing-support-vector-machine-example-code/, (Accessed on 08/05/2019),

[24] *Support vector machines(svm) - towards data science*, https://towardsdatascience.com/support-vector-machines-svm-c9ef22815589, (Accessed on 08/05/2019),

[25] *Support vector machine*, https://www.saedsayad.com/support_vector_machine.htm, (Accessed on 08/05/2019),

[26] *Classifying heart disease using k-nearest neighbors*, https://towardsdatascience.com/implement-k-nearest-neighbors-classification-algorithm-c99be8f14052, (Accessed on 08/05/2019),

[27] *Machine learning algorithms: What is a neural network?*, https://bit.ly/2YtI1Jx, (Accessed on 08/05/2019),

[28] D. Giustiniano and S. Mangold, "Caesar: Carrier sense-based ranging in off-the-shelf 802.11 wireless lan", in *Proceedings of the Seventh COnference on emerging Networking EXperiments and Technologies*, ACM, 2011, p. 10.

[29] Y. Gu, J. Zhan, Y. Ji, J. Li, F. Ren, and S. Gao, "Mosense: An rf-based motion detection system via off-the-shelf wifi devices", *IEEE Internet of Things Journal*, vol. 4, no. 6, pp. 2326–2341, 2017.

[30] A. Bhartia, Y.-C. Chen, S. Rallapalli, and L. Qiu, "Harnessing frequency diversity in wi-fi networks", in *Proceedings of the 17th annual international conference on Mobile computing and networking*, ACM, 2011, pp. 253–264.

[31] D. Halperin, W. Hu, A. Sheth, and D. Wetherall, "802.11 with multiple antennas for dummies", *ACM SIGCOMM Computer Communication Review*, vol. 40, no. 1, pp. 19–25, 2010.

[32] S. Sen, J. Lee, K.-H. Kim, and P. Congdon, "Avoiding multipath to revive inbuilding wifi localization", in *Proceeding of the 11th annual international conference on Mobile systems, applications, and services*, ACM, 2013, pp. 249–262.