

Article

# WiFi-Based Gesture Recognition for Vehicular Infotainment System—An Integrated Approach

Zain Ul Abiden Akhtar  and Hongyu Wang  \*Faculty of Electronic Information and Electrical Engineering, Dalian University of Technology,  
Dalian 116023, China; zain\_akhtar@outlook.com

\* Correspondence: whyu@dlut.edu.cn

Received: 10 November 2019; Accepted: 29 November 2019; Published: 4 December 2019



**Abstract:** In the realm of intelligent vehicles, gestures can be characterized for promoting automotive interfaces to control in-vehicle functions without diverting the driver's visual attention from the road. Driver gesture recognition has gained more attention in advanced vehicular technology because of its substantial safety benefits. This research work demonstrates a novel WiFi-based device-free approach for driver gestures recognition for automotive interface to control secondary systems in a vehicle. Our proposed wireless model can recognize human gestures very accurately for the application of in-vehicle infotainment systems, leveraging Channel State Information (CSI). This computationally efficient framework is based on the properties of K Nearest Neighbors (KNN), induced in sparse representation coefficients for significant improvement in gestures classification. In this typical approach, we explore the mean of nearest neighbors to address the problem of computational complexity of Sparse Representation based Classification (SRC). The presented scheme leads to designing an efficient integrated classification model with reduced execution time. Both KNN and SRC algorithms are complimentary candidates for integration in the sense that KNN is simple yet optimized, whereas SRC is computationally complex but efficient. More specifically, we are exploiting the mean-based nearest neighbor rule to further improve the efficiency of SRC. The ultimate goal of this framework is to propose a better feature extraction and classification model as compared to the traditional algorithms that have already been used for WiFi-based device-free gesture recognition. Our proposed method improves the gesture recognition significantly for diverse scale of applications with an average accuracy of 91.4%.

**Keywords:** driver distraction; human gesture recognition; device-free localization; wireless sensor; Channel State Information; integrated classification; computational complexity; sparse representation

## 1. Introduction

Distracted driving is one of the main concerns that compromise road safety. A large number of road accidents are reported because of driver's engagement in performing conventional secondary tasks using visual-manual interfaces. With the advancements in vehicular technology and the introduction of human computer interaction (HCI), gesture-based touchless automotive interfaces are being incorporated in vehicle designs to reduce driver visual distraction. Therefore, driver gesture recognition has become the most interesting research topic in recent years. Human gestures recognition has been widely explored in the literature for a variety of applications to reduce the complexity of human interaction with computers and other digital interfaces [1–4].

Existing research on in-vehicle gesture recognition is mainly focused on sensors, radars, or cameras [5–7], which have their own limitations in practical scenarios [8,9]. Subjectively, haptified gestures with ultrasound are being incorporated in the automotive domain [10]. During the last decades, WiFi-based human gesture recognition systems have gained more attention, exploiting

Channel State Information (CSI). Because of its simplicity and easy deployment, these systems are numerously used in various human-centric applications including emergency surveillance, health monitoring and assisted living [11–15]. WiFi-based device-free approach opened a new window for the researchers to further integrate wireless human gestures recognition in vehicular technology. We are motivated to investigate WiFi-based in-vehicle human gestures to control automotive secondary functions, which is device-free and non-intrusive to driver.

This research is relevant to the ability of an intelligent vehicle to better characterize driver gestures for automotive infotainment (information and entertainment) system such as adjusting radio volume, fan speed and so on. We utilize device-free human-gestures based on the motion of the driver's hand, finger and head, as shown in Table 1. In this novel research, we particularly identify sixteen common gestures, of which ten are hand gestures (swipe left, swipe right, hand going up, hand going down, flick, grab, push hand forward, pull hand backward, rotate hand clock-wise, rotate hand anti-clock-wise), while four are finger gestures (swipe V, swipe X, swipe +, swipe –) and the remaining two are head gestures (head tilting down, head tilting right). The key challenge is how to extend the robust gesture recognition rate for the application of an in-vehicle infotainment system using WiFi signals. In order to address the problem, we are inspired to fill in this research gap using an integrated classification approach leveraging CSI measurements. In this paper, we present a device-free wireless innovative framework to address the problem of driver gesture recognition by integrating Sparse Representation based Classification (SRC) and a modified variant of the K Nearest Neighbors (KNN) algorithm for classification. To achieve this goal, we have to face several challenges because an uncontrolled in-vehicle environment is quite different from indoor environments. We perform extensive experiments and design our prototype to combat these challenges.

**Table 1.** Proposed gestures utilized for different secondary tasks.

Gesture Type	Gesture Performed	Associated Task	Gesture Label
Hand Gesture	Swipe left	(+) Change channel	SL
	Swipe right	(–) Change channel	SR
	Hand going up	(+) Volume	HU
	Hand going down	(–) Volume	HD
	Flick	Zoom in	FK
	Grab	Zoom out	GB
	Push hand forward	(+) Temperature setting	PF
	Pull hand backward	(–) Temperature setting	PB
	Rotate hand clock-wise	(+) Fan speed	RC
	Rotate hand anti-clock-wise	(–) Fan speed	RA
Finger Gesture	Swipe V	Open function	SV
	Swipe X	Close function	SX
	Swipe +	Play next track	SP
	Swipe –	Play previous track	SN
Head Gesture	Head tilting down	Pick phone call	HM
	Head tilting right	Do not pick call	HR

The classification algorithm plays a vital role in the gesture recognition framework. Most of the existing CSI-based gesture recognition systems rely on a single classification algorithm, which cannot be guaranteed for maximum performance in vehicular technology. Carefully integrating two or more classification algorithms may enhance the recognition performance of a classifier for automotive interfaces. In this context, both KNN and SRC classifiers have been efficiently used as stand-alone, to solve various classification problems in WiFi-based device-free localization and activity or gesture recognition [13,16–18].

In our proposed scheme, the superiority of SRC is applied on a nearest neighbor rule, to further improve the performance of SRC with reduced computational cost. Therefore, the advanced classification is based on the integrated properties of SRC and modified-KNN to produce efficient results. We utilize the Mean of Nearest Neighbors (MNN); a variant of KNN. The concept of the mean-based algorithm was originally proposed by Mitani and Hamamoto [19] to classify query patterns, using a local mean-based non-parametric classifier. The idea of local mean-based KNN has been successfully applied in various pattern recognition systems [20,21]. Our prototype is based on multiple variants of KNN and SRC algorithms in the literature [22–26]. Different from others, we incorporate MNN with SRC for significant improvement in device-free gesture recognition for the application of automotive infotainment system, exploiting CSI. The combination of MNN and SRC has a computational advantage over using only SRC. We integrate MNN and SRC to overcome the computational complexity of SRC.

Our proposed system leverages off-the-shelf WiFi devices to collect channel information, which are readily available in the form of CSI measurements on commercial WiFi devices. Our algorithm utilizes the change in WiFi channel information caused by driver gestures. The driver needs to perform specific gestures in the WiFi coverage area. As far as we know, this is the first attempt towards CSI-based device-free driver gesture recognition using nearest neighbor induced sparse representation for the application of automotive interface.

Our contributions can be summarized as follows:

- We present a WiFi-based device-free innovative framework to address the problem of driver gesture recognition for the application of vehicle infotainment systems leveraging CSI measurements.
- We demonstrate a novel classification model by integrating SRC and a variant of the KNN algorithm to overcome the problem of expensive computational cost.
- To evaluate the performance of our proposed framework, we perform comprehensive experiments in promising application scenarios.
- To validate the results, we compare our system performance with state-of-the-art methods.

The rest of the paper is organized as follows; Section 2, demonstrates the traditional techniques that are relevant to our work. In Section 3, we provide an overview of our proposed system. Section 4 presents the detailed system flow with the methodology of our proposed solution. Section 5 highlights the experimental settings and validates our results by way of performance evaluation. In Section 6, we discuss important aspects and limitations of the proposed framework. Finally, we conclude our work with some future suggestions in Section 7.

## 2. Related Work

In this section, we will briefly review the CSI-based activity and gesture recognition systems that exploit different classification methods including KNN and SRC algorithms. Fine-grained physical layer CSI holds pervasive indoor localization and has attracted many researchers because of its potential for accurate indoor localization. With the advancement in wireless technology and the ubiquitous deployment of indoor wireless systems, indoor localization has been permeated into a new era of modern life [27]. Recently, a WiFi-based training-free localization system has been presented with good performance [28].

WiSee [29] is a WiFi based system that can recognize nine different gestures in line-of-sight (LOS), non line-of-sight (NLOS) and through-the-wall scenarios via Doppler shifts. Through-wall motion detection was investigated by WiVi [30] using multi antenna techniques. However, traditional solutions have been prototyped with special software or hardware to capture OFDM (Orthogonal Frequency Division Multiplexing) signals. As compared to these conventional methods, our proposed solution leverages off-the-shelf WiFi devices without any change in infrastructure.

The emerging device-free localization and activity recognition relies on CSI for better characterization of WiFi signals influenced by the human activities [31–34]. In recent years, CSI

based micro-activity recognition [35] and intrusion detection [36–38] systems have emerged with good recognition results.

WiG [11] focused on WiFi-based gesture recognition for both LOS and NLOS scenarios. CSI-based gesture detection was presented in Reference [12] leveraging packet transmission and the recognition of gestures was performed by distinguishing their strengths. WiGer [39] demonstrated a WiFi-based gesture recognition system by designing a fast dynamic time warping algorithm to classify hand gestures. Recently, Reference [40] presented the writing in air with WiFi signals for virtual reality devices with more complexity and increased diversity as compared to simple gestures.

Most of existing CSI-based systems rely on single classification algorithms. During the previous decade, KNN and SRC algorithms have been numerous used in various CSI-based device-free activity recognition models, as a stand-alone classifier. WiCatch [41] proposed WiFi-based hand gesture recognition system utilizing weak signals reflected from hands and the recognition is based on support vector machines (SVMs). Wi-Key [16] recognized keystrokes via CSI-waveform using the KNN classifier. WiFinger [13] leverages WiFi signals for finger gesture recognition by examining the unique patterns of CSI using the KNN classifier. The effect of Doppler shifts on CSI for health-care applications using the SRC classification algorithm was described in Reference [17]. Human activity recognition using the SRC classifier has been successfully demonstrated in Reference [18] with high accuracy for both LOS and NLOS scenarios leveraging CSI. Different from previous work, our classification algorithm is based on the integration of modified-KNN and SRC for automotive infotainment interface.

A limited work is reported in the literature on WiFi-based driver's in-vehicle activity or gestures recognition. In recent years, WiFi-based driver fatigue detection and driver activity recognition systems have been investigated with some constraints. In this context, WiFind [42] is suitable only for fatigue detection, while WiDriver [43] leverages a driver's hand movements for better characterization of driver action recognition using WiFi signals. SafeDrive-Fi [44] demonstrated CSI-based dangerous driving recognition through body movements and gestures, using variance of CSI amplitude and phase measurements. Different from them, we have specifically focused on driver hand, finger and head gesture recognition for the application of in-vehicular infotainment system.

### 3. System Overview

In this section, we will present the overview of our proposed classification algorithm and system architecture, along with the important facts about CSI relevant to our work.

#### 3.1. Background of SRC and MNN Algorithms

Both KNN and SRC classifiers have been efficiently used in various wireless device-free localization and recognition systems [13,16–18]. Conventional SRC is time consuming in the sense that a testing sample is usually represented by all training samples. The decision rule of SRC follows that if the testing sample has a great similarity with any training sample, the sparse coefficients of that specific training sample will be larger to represent the certain testing sample. SRC is efficient in a sense that all coefficients participate well in decision making. However, the computational cost of SRC increases with the increase in size of training data.

KNN is simple yet effective classifier with optimal performance. However, KNN has the issue of neighborhood size and simple majority voting for the classification, which can degrade its performance. The traditional KNN classifier chooses the K nearest neighbors from training data and majority voting is used to decide the class. Mean Nearest Neighbor (MNN) is the variant of KNN using the mean as a prototype of associated class.

#### 3.2. Integration of SRC and MNN Algorithms

We start with traditional KNN, estimate the K nearest neighbors from all training samples of each class. Thus, we calculate the mean of K nearest neighbors within each class. We use the decision rule of SRC to supervise MNN, and sparse representation coefficients are computed from the mean vector

of nearest neighbors. Finally, the class of testing sample is decided on the residual between testing sample and the mean of nearest neighbors within each class. The more details about this integrated classification algorithm are given in Section 4.3.

### 3.3. CSI Overview

Our prototype leverages WiFi ambient signal as information source to analyze the influence of a driver's gesture on a wireless channel. Existing WiFi devices that exploit IEEE 802.11n/ac protocols typically consist of multiple transmitting (Tx) and receiving (Rx) antennas and thus support the widely used Multiple-Input Multiple-Output (MIMO) technology. CSI refers to fine-grained signal containing physical layer information based on Orthogonal Frequency-Division Multiplexing (OFDM).

In our experiment, an IEEE 802.11n enabled Access Point (AP) was used as a transmitter and Intel 5300 NIC was used as a receiver to collect CSI data from the physical layer of the WiFi system, which supports 30 subcarriers for each CSI Tx-Rx antenna pair. It records the channel properties of each Tx-Rx antenna pair in OFDM subcarriers. The channel variations are readily available in the form of CSI measurements on commercial WiFi devices [45]. A typical narrowband flat-fading channel for packet index  $i$ , exploiting MIMO and OFDM technology can be modeled as:

$$Y_i = H_i X_i + \mathcal{N}_i \quad i \in [1, N], \quad (1)$$

where  $H_i$  is the CSI channel matrix for packet index  $i$ ,  $\mathcal{N}_i$  denotes the Gaussian noise vector,  $Y_i$  and  $X_i$  are the received and transmitted signals respectively,  $N$  refers to the total number of received packets.

Let  $\mathbb{N}_{Tx}$  and  $\mathbb{N}_{Rx}$  be the number of transmitting and receiving antennas respectively, then the CSI matrix consists of  $\mathbb{N}_{Tx} \times \mathbb{N}_{Rx} \times 30$  complex values for each CSI stream. CSI matrix  $H$  for each Tx-Rx antenna pair can be written as:

$$H_i = [h_1, h_2, \dots, h_{30}] \quad i \in [1, N]. \quad (2)$$

Each  $h$  is a complex value, which carries the information for both amplitude and phase response; estimated as:

$$h = |h| e^{j \sin\{\angle h\}}, \quad (3)$$

where  $|h|$  denotes the amplitude and  $\angle h$  indicates the phase information. CSI information on multiple channels is correlated, whereas all streams behave independently. We utilize both the amplitude and phase information to unleash the full potential of CSI measurements.

### 3.4. System Architecture

Our device-free driver gesture recognition system is comprised of following three basic modules—(1) CSI pre-processing module, (2) feature extraction module and (3) classification module, as illustrated in Figure 1. CSI pre-processing module; collects and pre-processes CSI measurements using basic filtering techniques. The feature extraction module is responsible for gesture detection, dimension reduction and feature extraction. Recognition is performed in the classification module, which relies on the integrated classification method. In Section 4, we will explain the function of each module in detail.

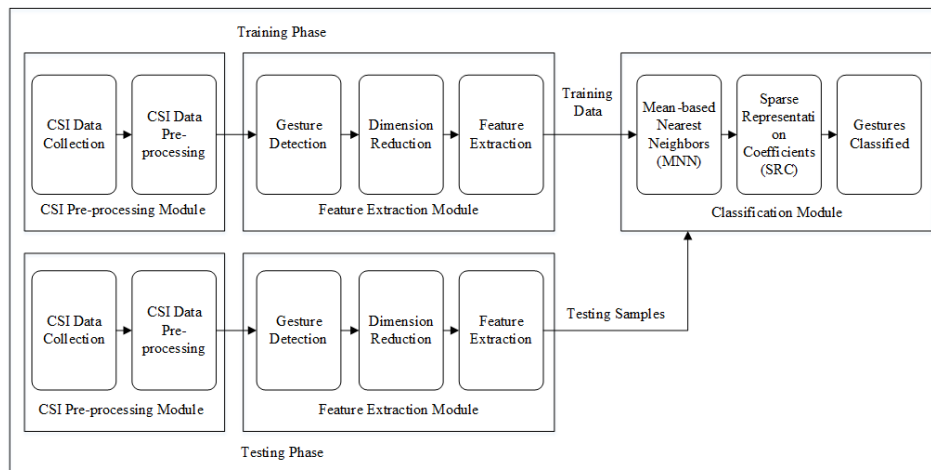


Figure 1. System architecture.

## 4. Methodology

In this section, we will explain complete flow of our system methodology.

### 4.1. CSI Pre-Processing

The CSI received signal is a combination of useful information as well as undesirable noises embedded in the signal. It is essential to filter and calibrate CSI amplitude and phase measurements, respectively. The human gestures have a relatively low frequency as compared to noise frequency. To remove high frequency noise, we apply a second order low pass Butterworth filter. We adjusted packets sampling rate ( $F_s$ ) at 80 packets/s, the same as the normalized cutoff frequency  $w_n = 2\pi f/F_s = 0.025\pi$  rad/s. CSI raw phase measurements behave extremely randomly because of the unsynchronized time clock of transmitter and receiver. In order to extract the actual phase and eliminate channel frequency offset; phase calibration and linear transformation are performed by following [46].

#### 4.1.1. Phase Calibration

CSI raw phase measurements behave extremely randomly because of the unsynchronized time clock of transmitter and receiver. The relation between measured phase and true phase can be written as:

$$\angle \hat{h}_j = \angle h_j + 2\pi \frac{n_j}{N} \Delta t + \beta + z, \quad (4)$$

where  $\angle \hat{h}_j$  indicates the measured phase of  $j$ th subcarrier and  $\angle h_j$  denotes the actual phase,  $\Delta t$  represents the time lag,  $n_j$  denotes the subcarrier index,  $N$  stands for the size of FFT,  $\beta$  indicates the unknown phase offset and  $z$  is random noise. We cannot measure the exact value of  $\beta$  and  $\Delta t$  for each packet. However, we can get the same value of  $\beta$  and  $\Delta t$  each time by using a simple transformation. Phase error  $2\pi \frac{n_j}{N} \Delta t + \beta$  is linear function of subcarrier index  $n_j$ . We can define two parameters  $a$  and  $b$  for calibration of phase error such that:

$$a = \frac{\angle \hat{h}_{j_N} - \angle \hat{h}_{j_1}}{j_N - j_1} \quad (5)$$

$$b = \frac{1}{N} \sum_{k=1}^N \angle \hat{h}_{j_k}, \quad (6)$$

subtracting  $a_{j_k} + b$  from raw phase  $\widehat{\mathcal{L}}_{h_{j_k}}$  to get the sanitized phase  $\widetilde{\mathcal{L}}_{h_{j_k}}$  as:

$$\widetilde{\mathcal{L}}_{h_{j_k}} = \widehat{\mathcal{L}}_{h_{j_k}} - a_{j_k} - b. \tag{7}$$

The phase sanitization is performed on all the subcarriers and re-assembled according to the corresponding amplitudes.

#### 4.1.2. Amplitude Information Processing

CSI amplitude measurements are very noisy for gesture extraction because of environmental noise and signal interference. We propose applying the weighted moving average (WMA) over CSI amplitude streams to eliminate the outliers and avoid false anomaly by the following procedure [47].

Let  $|h_t|$  denotes the amplitude information at time interval  $t$ , then the expression for WMA can be written as:

$$|h'_t| = \frac{[m \times |h_m| + (m - 1) \times |h_{(m-1)}| + \dots + 1 \times |h_1|]}{m + (m - 1) + \dots + 1}, \tag{8}$$

where  $|h'_t|$  is the averaged amplitude at time  $t$ . New amplitude  $|h'_t|$  has the weight value of  $m$ , which decides to what degree the current value relates to historical data. The value of  $m$  can be empirically selected based on the experiments.

Figure 2, shows the comparison of the original signal and the signal processed after WMA implementation. It is clear that WMA can make the subcarrier waveform much smoother by reducing noise level.

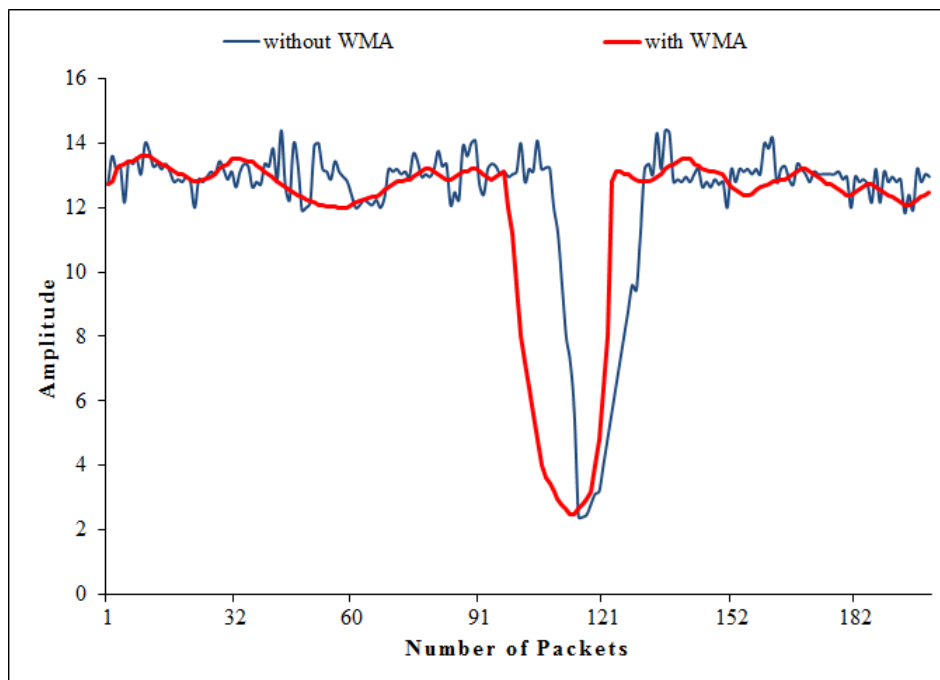


Figure 2. Implementation of weighted moving average (WMA) on single subcarrier.

#### 4.2. Gesture Detection

In the CSI data, both amplitude and phase information have the capability to be used for gesture detection [48]. Our gesture detection algorithm is based on the variance of amplitude and phase to characterize gestures from CSI filtered data. In this scheme, all subcarriers are aggregated to evaluate the variance. We normalized the variance and estimated the corresponding energy of CSI amplitude and phase over a specific time window.

Let  $v_i$  indicate the variance of a single link for all the subcarriers belonging to packet index  $i$ . For  $N$  number of received packets, variance  $v = \{v_1, v_2, \dots, v_N\}$  obeys Gaussian distribution with  $v \sim$

$N(\mu, \sigma^2)$ . We normalized  $v$  to represent normalized amplitude and phase as  $V_A$  and  $V_P$ , respectively. Let  $E_A$  and  $E_P$  are the corresponding energy of normalized amplitude ( $V_A$ ) and phase ( $V_P$ ), respectively. We can estimate the energy over a length of time window  $\tau$  and compare it with the threshold  $\eta_{th}$  as:

$$E_A = \sum_{k=1}^{length(\tau)} (V_A(k))^2 \tag{9}$$

and

$$E_P = \sum_{k=1}^{length(\tau)} (V_P(k))^2. \tag{10}$$

We set a threshold ( $\eta_{th} = 100$ ) for gesture detection based on our preliminary measurements depending on different conditions and scenarios. Corresponding to this threshold, the gesture is detected as:

$$\phi = \begin{cases} \max(V), & E \geq \eta_{th} \\ 0, & otherwise. \end{cases} \tag{11}$$

If the energy  $E$  is greater than or equal to the threshold  $\eta_{th}$ , the system takes it as a gesture and records the maximum value of normalized variance over this window, and when it goes lower than the threshold it is considered as a non-gesture.

#### 4.2.1. Dimensionality Reduction

In CSI data, some subcarriers are very sensitive to noise but non-significant for gesture sensitivity. From Figure 3, it is clear that the sensitivity of different subcarriers varies for the same gesture. Therefore, we suggest Principal Component Analysis (PCA) to reduce the dimensionality and eliminate such types of unpredictable subcarriers.

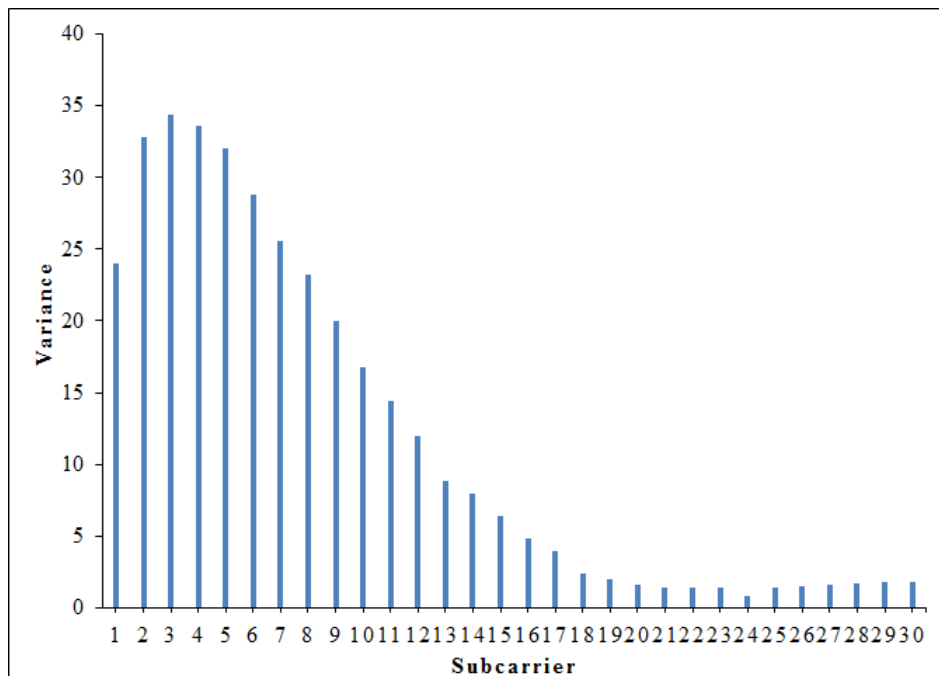


Figure 3. Variance of different subcarriers for the same gesture.

PCA is commonly used to extract most representative components and removes background noise [49]. In the PCA scheme, new variables are being generated by the linear combination of the original data, called principal components. These principal components organize the original variables in orthogonal form in order to eliminate redundant information.



In our experiment, PCA is applied to each packet received to extract  $p$  principal components. Based on our observations, we particularly select the second, third and fourth principal components that is,  $p = 3$  for both the amplitude and phase of the CSI signal. Finally, we obtain a matrix with  $p \times N_a$  dimensions, where  $N_a$  denotes the number of anomalous packets.

First, we normalized CSI matrix  $H$  and static components were removed. Let  $H_n$  represent the normalized matrix after the subtraction of the average CSI value from each column of CSI matrix  $H$ . We then calculated the corresponding correlation matrix as  $H_n \times H_n^T$ . After eigen decomposition of the correlation matrix, we compute eigen vectors  $q = q_1, q_2, \dots, q_i$  and simultaneously  $p = p_1, p_2, \dots, p_i$  principal components are structured. For  $i$ th packet, it can be expressed as:

$$p_i = H_n \times q_i, \tag{12}$$

where  $q_i$  and  $p_i$  stand for  $i$ th eigen vector and  $i$ th principal component, respectively.

#### 4.2.2. Feature Extraction

Let  $H_c(s)$  be the first CSI packet detected by our algorithm and  $S_c$  stands for the number of subcarriers between a single link of antenna pair  $Tx - Rx$ . We can extract  $N_c$  successive CSI packets for any activity profile.

Based on our preliminary experimental investigations and detailed analysis of extracted CSI, we particularly selected six statistical features that is, mean, standard deviation, median absolute deviation, maximum value, 25th percentile and 75th percentile. Mathematically, these features are defined as [50]:

(1) Mean: The mean  $\mu(j)$  is defined as the average CSI of all packets  $H_c$  belonging to  $j$ th subcarrier written as:

$$\mu(j) = \frac{[H_{cj}(s) + H_{cj}(s + 1) + \dots + H_{cj}(s + N_c)]}{N_c} \tag{13}$$

where  $N_c$  is the total number of successive CSI packets for activity profile and  $j \in [1, S_c]$ .

(2) Standard Deviation: The standard deviation  $\sigma(j)$  of  $j$ th subcarrier is basically the square root of variance. Assume  $j \in [1, S_c]$ , then  $\sigma(j)$  can be expressed as:

$$\sigma(j) = \sqrt{\frac{1}{N_c} \sum_{k=s}^{k=s+N_c} (H_{cj}(k) - \mu(j))^2}. \tag{14}$$

(3) Median Absolute Deviation: A robust way to quantify CSI variations for any activity segment is Median Absolute Deviation (MAD). Mathematically, median absolute deviation  $MAD(j)$  for  $j$ th subcarrier is defined as:

$$\begin{cases} MAD(j) = median(|H_{cj}(k) - \tilde{H}_{cj}|) \\ s.t. j \in [1, S_c] \text{ and } k \in [1, N_c] \end{cases} \tag{15}$$

where  $\tilde{H}_{cj}$  is the median of  $H_{cj}$ .

(4) Maximum Value: The maximum value is a unique number, that is the highest value of all other values in the CSI data set. Maximum value  $MAX(j)$  of  $j$ th subcarrier with  $j \in [1, S_c]$  is mathematically calculated as:

$$MAX(j) = max([H_{cj}(s)|H_{cj}(s + 1)| \dots |H_{cj}(s + N_c)]). \tag{16}$$

(5) 25th Percentile: If 25 is the ordinal rank, the 25th percentile of a data set is defined as the value at which 25 percent of distribution is below it. Mathematically, 25th percentile  $P_{25th}(j)$  for  $j \in [1, S_c]$  is explained as:

$$P_{25th}(j) = H_{cj}(\lceil \frac{25}{100} * N_c \rceil). \tag{17}$$

(6) 75th Percentile: Similarly, 75th percentile  $P_{75th}(j)$  can be formulated as:

$$P_{75th}(j) = H_{c_j}(\lceil \frac{75}{100} * N_c \rceil). \tag{18}$$

In order to differentiate between multiple gesture profiles, we integrate all features into a feature vector. We propose constituting a tuple  $\mathcal{F}$  of integrated features utilizing both magnitude and phase, defined as:

$$\mathcal{F} = \{f_1, f_2, \dots, f_n\} \subset \mathbb{F}, \tag{19}$$

where  $f_i$  is a feature and  $\mathbb{F}$  stands for the dataset consisting of all gesture features. We get twelve features in total (6-amplitude features and 6-phase features).

### 4.3. Classification Module

In the classification module, MNN and SRC algorithms were integrated. We started with traditional KNN, estimated the K nearest neighbors from all training samples of each class [26]. Thus, we calculated the mean of K nearest neighbors within each class.

#### 4.3.1. K Nearest Neighbor

Assume our classification problem has  $t$  number of classes. Let  $F_i$  be a set comprising of training samples for class index  $i$  represented as:

$$F_i = \{f_{ij} | j = 1, \dots, N_i\}, \forall i = 1, \dots, t, \tag{20}$$

where  $N_i$  is the total number of training samples for class index  $i$ . Assume  $f_{ir}$  is the prototype for  $i$ th class. For any testing sample  $y$ , let us find its nearest neighbor  $f_{ir}$  from training samples in each class. We adopt squared Euclidean distance to measure the similarity between nearest neighbor  $f_{ir}$  and testing sample  $y$ , defined as:

$$d_i(y) = \|y - f_{ir}\|^2. \tag{21}$$

The Nearest Neighbor (NN) algorithm estimates the nearest training sample based on the distance measure and designate that class to testing sample  $y$  which has the minimal distance. Mathematically,

$$class(y) = \min_i d_i(y). \tag{22}$$

The KNN algorithm is the extension of 1-NN by taking  $K$  nearest neighbors from all the training data. In KNN, the class assignment to  $y$  is based on the rule of majority voting. Assuming  $i$ th class has  $k_i$  samples, the testing sample  $y$  belongs to the class which has a maximum number of nearest neighbors as:

$$class(y) = \max_i k_i. \tag{23}$$

Such that,  $y$  is designated to the class based on maximum number of nearest neighbors. In general, the class of the testing sample is normally decided on the basis of majority voting of  $K$  nearest neighbors that are specifically chosen from training samples with a certain minimum distance.

#### 4.3.2. Mean of Nearest Neighbor (MNN)

We implemented the mean of the nearest neighbor rule in our proposed algorithm because it may be a meaningful compromise between the nearest neighbor and minimum distance. Assume  $K$  nearest neighbors of testing sample  $y$  for class index  $i$  are represented as  $F_i = \{f_{i1}, f_{i2}, \dots, f_{iK_i}\}$ . The mean vector of nearest neighbors for testing sample  $y$  in class  $i$  is defined as:

$$\mu_{y_i} = \frac{1}{K} \sum_{r=1}^K f_{ir} \tag{24}$$

After estimating the mean vector per class, we can determine the class of  $y$ . For this purpose, we find the distance from the mean vector as:

$$d_i(y) = \|y - \mu_{y_i}\|^2. \tag{25}$$

Thus, the class is estimated as follows:

$$class(y) = \min_i d_i(y). \tag{26}$$

#### 4.3.3. Sparse Representation Based Classification (SRC)

In SRC, a testing sample is expressed as a linear sparse combination of all the training samples [22]. The sparse representation coefficients can be obtained by solving  $L_1$  optimization problem. We characterise the training samples for  $i$ th class as  $F_i = \{f_{i1}, f_{i2}, \dots, f_{iN_i}\}$ , such that  $N = \sum_{i=1}^t N_i$ . Now assume a matrix  $F = \{F_1, F_2, \dots, F_t\}$  in which all training samples of entire activity classes are being concatenated.

The testing sample  $y$  is defined as  $y = Fv$  for the sparsest solution. The coefficient vector  $v$  has nonzero values only for the entries that are associated to class  $i$ . The sparse solution for coefficient vector  $v$  may be optimized with  $L_0$  norm constraint as:

$$\begin{cases} \hat{v}_0 = \arg \min \|v\|_0 \\ \text{s.t. } Fv = y. \end{cases} \tag{27}$$

Assuming the solution of  $L_0$  norm constraint is equivalent to  $L_1$  norm constraint, we obtain

$$\begin{cases} \hat{v}_1 = \arg \min \|v\|_1 \\ \text{s.t. } Fv = y. \end{cases} \tag{28}$$

After getting the optimal  $\hat{v}_1$  solution for sparsity, SRC can be designed for class specific reconstruction residual.

Let  $\delta_i(v)$  is the vector associated with  $i$ th class that selects non-zero entries corresponding to  $v$ . Based on the coefficients of  $i$ th class, we can reconstruct the test sample  $y$  as:

$$\hat{y}_i = F\delta_i(v), \tag{29}$$

where  $\hat{y}_i$  indicates the reconstructed testing sample. The residual of reconstructed class can be obtained as:

$$r_i(y) = \|y - \hat{y}_i\|_2 = \|y - F\delta_i(v)\|_2. \tag{30}$$

The decision of class is based on the following principle:

$$class(y) = \min_i r_i(y). \tag{31}$$

In our proposed framework, we used the decision rule of SRC to supervise MNN and sparse representation coefficients were computed from the mean vector of nearest neighbors.

#### 4.3.4. MNN Induced SRC (MNN-SRC)

Our proposed MNN-SRC method is much faster as compared to conventional SRC. The mean of  $K$  nearest neighbors is much smaller in comparison to the total number of training samples. In our proposed framework, the mean of  $K$  nearest neighbors is computed from all training samples. The testing sample is represented according to the mean vector of  $K$  nearest neighbors. Sparse representation based classification is applied on the mean vector of nearest neighbors instead of all

the training samples. The residuals are estimated between the testing sample and the mean vector of nearest neighbors for each class. The class of the testing sample is decided on the residual between testing sample and the mean of nearest neighbors within each class.

Suppose we have  $K_i$  nearest neighbors for  $\hat{F}_i = \{\hat{f}_{i1}, \hat{f}_{i2}, \dots, \hat{f}_{iK_i}\}$  belonging to  $i$ th class with  $K = \sum_{i=1}^I K_i$ . We find  $K$  nearest neighbors for each class corresponding to testing sample and estimate its mean. The mean vector of nearest neighbors for testing sample  $y$  in class  $i$  is defined as:

$$m = \frac{1}{K} \sum_{r=1}^K \hat{f}_{ir}. \tag{32}$$

Assume  $\hat{M}$  is a matrix comprising of mean vectors of each class. The sparse representation coefficients were calculated using  $\hat{M}$  to estimate  $y$ . Mathematically,

$$\min \|v\|_1 \text{ s.t. } y = \hat{M}v. \tag{33}$$

In some situations,  $y$  cannot be exactly equal to  $\hat{M}v$  for any coefficient. To overcome this constraint, a Lagrange factor  $\lambda$  is imposed by the following:

$$\min \|v\|_1 + \lambda \|y - \hat{M}v\|_2. \tag{34}$$

Let  $\hat{\delta}_i(v)$  be the vector associated with  $i$ th class that selects non-zero entries corresponding to  $v$ . Based on the coefficients of  $i$ th class, we can reconstruct the test sample  $y$  as:

$$\hat{y}_i = \hat{M}\hat{\delta}_i(v), \tag{35}$$

where  $\hat{y}_i$  indicates the reconstructed testing sample. The residual of the reconstructed class is obtained as:

$$r_i(y) = \|y - \hat{y}_i\|_2 = \|y - \hat{M}\hat{\delta}_i(v)\|_2. \tag{36}$$

The class  $i$  has residual  $r_i$ , which can determine the class of the testing sample. The decision for class designation is based on the following principle:

$$\text{class}(y) = \min_i r_i(y). \tag{37}$$

## 5. Experimentation and Evaluation

In this section, we will describe the experimental settings and evaluate the performance of our proposed framework.

### 5.1. Experimentation Settings

We conducted experiments using 802.11n enabled off-the-shelf WiFi devices. Specifically, we used a Lenovo laptop as a receiver equipped with an Intel 5300 network interface card and an Ubuntu 11.04 LTS operating system to collect CSI data. The laptop connects to a commercial WiFi Access Point (AP); TP-Link router as transmitter operating at 2.4 GHz. The receiver can ping the AP at rate of 80 packets/s. The transmitter has single antenna, whereas the receiver has three antennas, that is,  $N_{Tx} = 1$  and  $N_{Rx} = 3$  ( $1 \times 3$  MIMO system) generating 3 CSI streams of 30 subcarriers each. We run 802.11n CSI Tool [45] on the receiver to acquire and record CSI measurements on 30 subcarriers of 20 MHz channel. The required signal processing is performed using MATLAB R2016a.

To evaluate the robustness of our proposed scheme, we choose following three scenarios:

- Scenario-I (Indoor environment)—In this scenario, all prescribed gestures are performed in an empty room of size  $11 \times 12$  feet, while sitting on a chair between Tx and Rx, separated by a distance of 2 m.

- Scenario-II (Vehicle standing in a garage)—In this scenario, all gestures are performed in a vehicle standing in a garage of size  $14 \times 16$  feet.
- Scenario-III (Actual driving)—In this scenario, all prescribed gestures are performed while driving a vehicle on a straight road of 30 km inside university campus, with average speed of 20 km/h. During gesture performance, no other activity is performed to avoid interference.

For in-vehicle scenarios, we set up our testbed in a local manufactured vehicle which was not equipped with pre-installed WiFi devices. Due to the unavailability of the WiFi access point in our test vehicle, we configured the commercial TP-Link router as AP, placed on the dashboard in front of the driver. The receiver was placed at the co-pilot's seat to collect CSI data.

In each experiment, 16 possible human gestures, as shown in Table 1, were performed by five volunteers (2-females and 3-males university students). Each volunteer repeated all gestures 20 times for each experiment, and a single gesture was performed within a window of 5 s. In total, the data set comprised of 1600 samples ( $5\text{-volunteers} \times 16\text{-gestures} \times 20\text{-times repeated}$ ) for each experiment; of which 50% were used for training and 50% for testing. In our experiments, the training data do not contain the samples from the testing data, and we keep the testing samples out for cross validation. Furthermore, we also tested the generalization of our model using an unbiased Leave-One-Participant-Out Cross-Validation (LOPO-CV) scheme.

## 5.2. Performance Evaluation

First, we tested the usefulness of our extracted features, that is, mean ( $\mu$ ), standard deviation ( $\sigma$ ), median absolute deviation (MAD), maximum value (MAX), 25th percentile ( $P_{25th}$ ) and 75th percentile ( $P_{75th}$ ). Table 2 represents some prominent values of each calculated feature extracted for different gestures. One can notice that all features are distinctively different, indicating that these features can achieve high recognition accuracy.

**Table 2.** Features test result.

Feature Type	Gesture Class															
	SL	SR	HU	HD	FK	GB	PF	PB	RC	RA	SV	SX	SP	SN	HM	HR
$\mu$	1.54	1.33	0.81	0.74	1.42	2.16	2.25	3.52	0.96	0.91	1.63	1.49	2.28	1.71	2.13	2.66
$\sigma$	0.73	0.96	0.68	0.66	1.89	1.97	1.20	1.14	0.62	0.76	1.27	1.35	1.73	1.16	1.81	1.46
MAD	0.48	0.64	0.45	0.43	1.37	1.31	0.83	0.76	0.41	0.50	0.84	0.90	1.15	0.73	1.21	0.97
MAX	3.58	3.97	5.88	6.31	8.24	6.04	7.73	9.02	8.85	6.57	4.96	7.22	5.94	6.39	7.63	5.99
$P_{25th}$	0.69	0.60	0.27	0.37	0.58	0.82	0.98	1.22	0.35	0.44	0.64	0.51	0.77	0.65	0.95	0.91
$P_{75th}$	2.11	1.83	0.86	0.92	1.76	2.57	2.93	3.71	1.01	1.34	1.95	1.55	2.35	1.98	2.88	2.73

The recognition performance of the proposed method was observed by conducting extensive experiments. For simplicity, we use abbreviated terms for our proposed prototype, that is, MNN integrated with SRC as MNN-SRC, similarly KNN with SRC as KNN-SRC.

We particularly selected a confusion matrix and recognition accuracy as metrics for performance evaluation. The occurrence of the actual gesture performed is represented by the column of the confusion matrix, whereas the occurrence of the gesture classified was represented by the rows. The confusion matrix in Figure 4 reveals the fact that our proposed method can recognize sixteen different gestures very accurately with an average accuracy of 91.4%, 90.6% and 88.7% for scenarios I, II and III respectively.

Actual Gesture Performed		SL	SR	HU	HD	FK	GB	PF	PB	RC	RA	SV	SX	SP	SN	HM	HR	
	SL	0.89	0.04	0	0	0	0	0	0	0	0.02	0.05	0	0	0	0	0	0
	SR	0.02	0.91	0	0	0	0	0	0	0.06	0	0	0	0	0.01	0	0	0
	HU	0	0	0.94	0.01	0	0	0.03	0.02	0	0	0	0	0	0	0	0	0
	HD	0	0	0.03	0.93	0	0	0.01	0.03	0	0	0	0	0	0	0	0	0
	FK	0	0	0.01	0	0.89	0.04	0.05	0.01	0	0	0	0	0	0	0	0	0
	GB	0	0	0	0	0.03	0.90	0.01	0.05	0.01	0	0	0	0	0	0	0	0
	PF	0	0	0	0.01	0.05	0	0.90	0.04	0	0	0	0	0	0	0	0	0
	PB	0	0	0	0	0.02	0.06	0.03	0.89	0	0	0	0	0	0	0	0	0
	RC	0	0.04	0	0	0	0.01	0	0	0.92	0.03	0	0	0	0	0	0	0
	RA	0.05	0.01	0	0	0.01	0	0	0	0.02	0.91	0	0	0	0	0	0	0
	SV	0	0	0	0	0	0	0	0	0	0	0.91	0.06	0.02	0.01	0	0	0
	SX	0	0	0	0	0	0	0	0	0	0.02	0.03	0.90	0.05	0	0	0	0
	SP	0	0	0	0.01	0	0	0	0	0	0.02	0.04	0.92	0.01	0	0	0	0
	SN	0	0	0	0	0	0	0	0	0	0	0.01	0.02	0.06	0.91	0	0	0
	HM	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.96	0.04	0
HR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.05	0.95	0	

(a)

Actual Gesture Performed		SL	SR	HU	HD	FK	GB	PF	PB	RC	RA	SV	SX	SP	SN	HM	HR	
	SL	0.88	0.02	0	0	0	0	0	0	0	0.03	0.07	0	0	0	0	0	0
	SR	0.01	0.91	0	0.01	0	0	0	0	0.06	0	0	0	0	0.01	0	0	0
	HU	0	0	0.88	0.02	0	0	0.05	0.03	0	0	0.02	0	0	0	0	0	0
	HD	0	0	0.02	0.93	0	0	0.01	0.04	0	0	0	0	0	0	0	0	0
	FK	0	0	0.01	0	0.87	0.04	0.05	0.01	0	0	0.01	0	0	0.01	0	0	0
	GB	0	0	0	0	0.02	0.90	0.02	0.03	0.03	0	0	0	0	0	0	0	0
	PF	0	0	0	0.02	0.04	0	0.91	0.03	0	0	0	0	0	0	0	0	0
	PB	0	0	0	0	0.02	0.06	0.03	0.89	0	0	0	0	0	0	0	0	0
	RC	0	0.02	0	0	0	0.01	0	0	0.93	0.04	0	0	0	0	0	0	0
	RA	0.04	0	0	0	0.02	0	0	0	0.02	0.92	0	0	0	0	0	0	0
	SV	0	0	0	0	0	0	0	0	0	0	0.90	0.05	0.02	0.03	0	0	0
	SX	0	0	0	0	0	0	0	0	0	0.02	0.03	0.90	0.05	0	0	0	0
	SP	0	0	0	0.02	0	0	0	0	0	0	0.03	0.03	0.91	0.01	0	0	0
	SN	0	0	0	0	0	0	0	0	0	0	0.03	0.02	0.06	0.89	0	0	0
	HM	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.94	0.06	0
HR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.07	0.93	0	

(b)

Actual Gesture Performed		SL	SR	HU	HD	FK	GB	PF	PB	RC	RA	SV	SX	SP	SN	HM	HR	
	SL	0.86	0	0	0	0	0	0	0	0	0.05	0.06	0	0	0.02	0.01	0	0
	SR	0.02	0.89	0	0.01	0	0	0	0	0.06	0	0	0	0.01	0.01	0	0	0
	HU	0	0	0.88	0.02	0	0	0.05	0.03	0	0	0.01	0.01	0	0	0	0	0
	HD	0	0	0.03	0.90	0	0	0.02	0.05	0	0	0	0	0	0	0	0	0
	FK	0	0	0	0	0.85	0.05	0.04	0.03	0	0	0	0.01	0	0.02	0	0	0
	GB	0	0	0	0	0.03	0.90	0	0.05	0.02	0	0	0	0	0	0	0	0
	PF	0	0	0	0.03	0.04	0	0.91	0.02	0	0	0	0	0	0	0	0	0
	PB	0	0	0	0	0.03	0.07	0.03	0.87	0	0	0	0	0	0	0	0	0
	RC	0	0.03	0	0	0	0.02	0	0	0.86	0.07	0	0	0.02	0	0	0	0
	RA	0.04	0	0	0	0.03	0	0	0	0.02	0.88	0	0	0.03	0	0	0	0
	SV	0.01	0.01	0	0	0	0	0	0	0	0	0.89	0.05	0.02	0.02	0	0	0
	SX	0	0.01	0	0	0	0	0	0	0	0.03	0.02	0.88	0.06	0	0	0	0
	SP	0	0	0	0.01	0	0	0	0	0	0	0.04	0.03	0.90	0.02	0	0	0
	SN	0	0.02	0	0	0	0	0	0	0	0	0.03	0.02	0.06	0.87	0	0	0
	HM	0	0	0.01	0	0	0	0	0	0	0	0	0	0	0	0.92	0.07	0
HR	0	0	0.02	0	0	0	0	0	0	0	0	0	0	0	0.05	0.93	0	

(c)

**Figure 4.** Confusion matrix of activity recognition using our proposed algorithm. (a) Scenario-I. (b) Scenario-II. (c) Scenario-III.

To ensure the reliability and efficacy of proposed framework, we analyzed the results by adopting different evaluation metrics including precision, recall and  $F_1$ -score. These evaluation metrics are presented as:

1. Precision is defined as positive predictive value, mathematically described as:

$$Precision = \frac{TP}{TP + FP} \tag{38}$$

where *TP* and *FP* are true positive and false positive respectively. True Positive (TP) is the probability that a model correctly predicts the positive class. Whereas, False Positive (FP) is the probability of that a model incorrectly predicts the positive class.

2. Recall is defined as the True Positive Rate (TPR) and measures the sensitivity of system as follows:

$$Recall = \frac{TP}{TP + FN} \tag{39}$$

where *FN* is a false negative and is defined as the probability that the model incorrectly predicts the negative class. We will also evaluate our proposed method with False Negative Rate (FNR), defined as:

$$FNR = \frac{FN}{TP + FN} \tag{40}$$

3. F-measure or *F*<sub>1</sub>-score is defined as the weighted average of precision and recall, calculated as:

$$F_1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{41}$$

Figure 5 shows the results related to precision, recall and *F*<sub>1</sub>-score for all three scenarios, using our proposed scheme. Table 3 summarizes the performance of our MNN-SRC algorithm using TPR and FNR, for each gesture. In general, the MNN-SRC algorithm has an average TPR of over 88.9%, with average FNR less than 11.1% for all three scenarios.

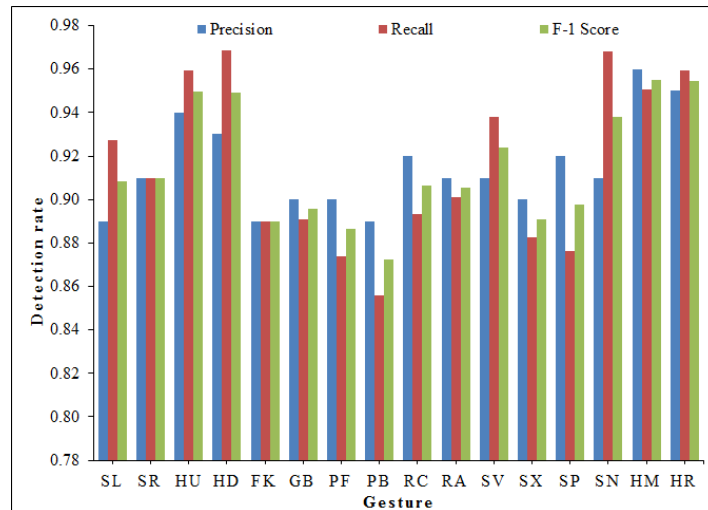
**Table 3.** Performance evaluation for each gesture.

Experiment	Gesture	SL	SR	HU	HD	FK	GB	PF	PB	RC	RA	SV	SX	SP	SN	HM	HR
Scenario-I	TPR (%)	92.7	91.0	95.9	96.9	89.0	89.1	87.4	85.6	89.3	90.1	93.8	88.2	87.6	96.8	95.0	96.0
	FNR (%)	7.3	9.0	4.1	3.1	11.0	10.9	12.6	14.4	10.7	9.9	6.2	11.8	12.4	3.2	5.0	4.0
Scenario-II	TPR (%)	94.6	95.8	96.7	93.0	89.7	89.1	85.0	86.4	86.9	87.6	88.2	90.0	87.5	93.7	93.1	93.9
	FNR (%)	5.4	4.2	3.3	7.0	10.3	10.9	15.0	13.6	13.1	12.4	11.8	10.0	12.5	6.3	6.9	6.1
Scenario-III	TPR (%)	92.5	92.7	93.6	92.8	86.7	86.5	86.7	82.9	85.1	84.6	89.9	88.0	80.4	91.6	94.8	93.0
	FNR (%)	7.5	7.3	6.4	7.2	13.3	13.5	13.3	17.1	14.9	15.4	10.1	12.0	19.6	8.4	5.2	7.0

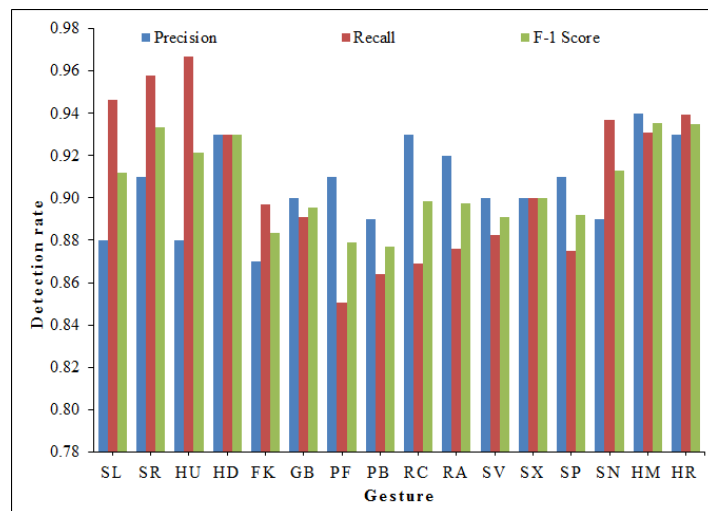
In order to determine the efficacy of the integrated classification algorithm, some experiments were performed with each classifier separately and the results are presented in Figure 6. As can be seen, the average recognition accuracy of stand-alone MNN (Mean Nearest Neighbor) or SRC (Sparse Representation based Classification) method is less as compared to the integrated MNN-SRC algorithm.

We have compared the performance of MNN-SRC with KNN-SRC as shown in Figure 7. It is obvious that the average recognition accuracy of MNN-SRC algorithm is higher as compared to KNN-SRC.

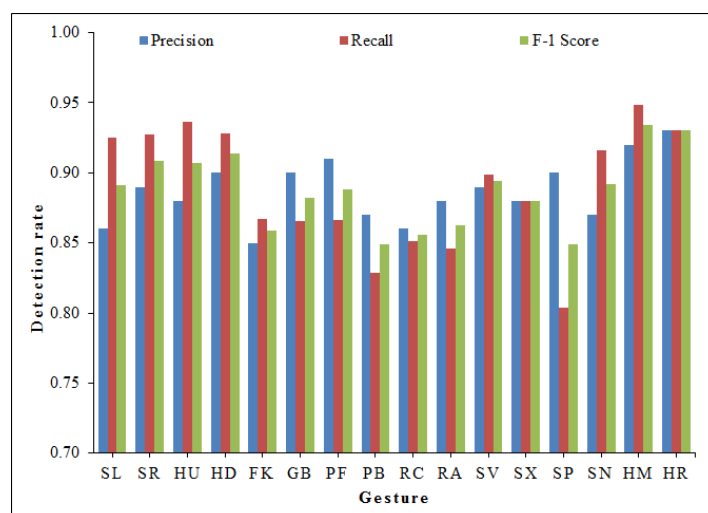
Figure 8 ensures the reliability of our feature extraction method. The model was evaluated using state-of-the-art classifiers including SVM, NB, KNN, MNN and SRC algorithms. One can notice that all classifiers show a satisfactory recognition results. However, our proposed MNN-SRC classification model outperforms other classification algorithms.



(a)



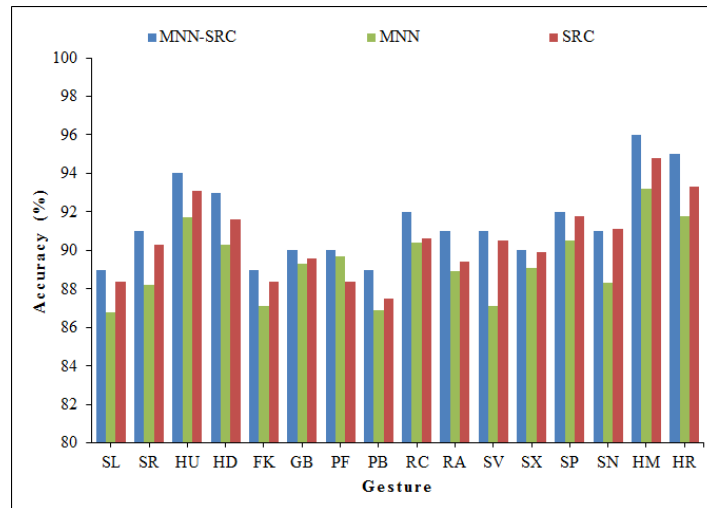
(b)



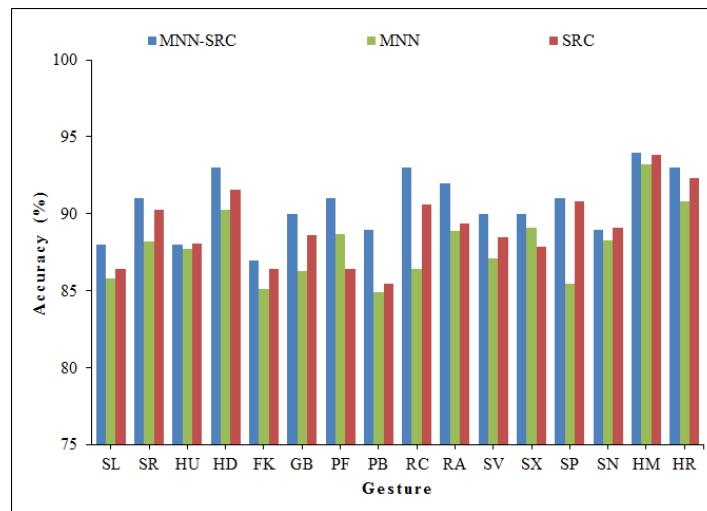
(c)

Figure 5. Precision, recall and F-1 score. (a) Scenario-I. (b) Scenario-II. (c) Scenario-III.

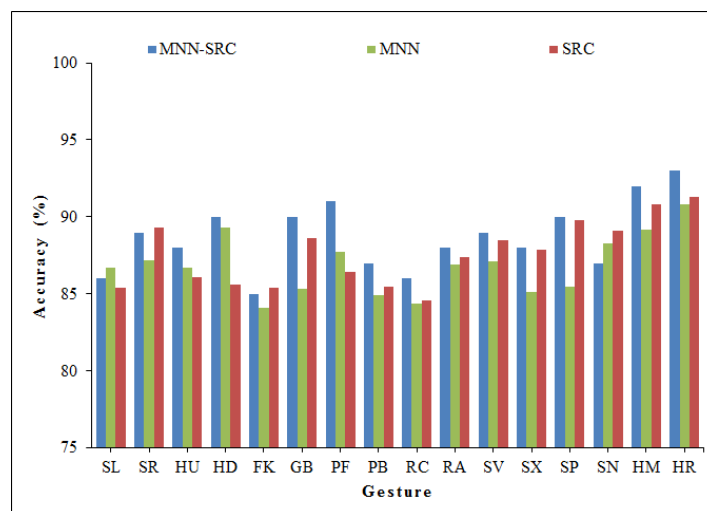




(a)

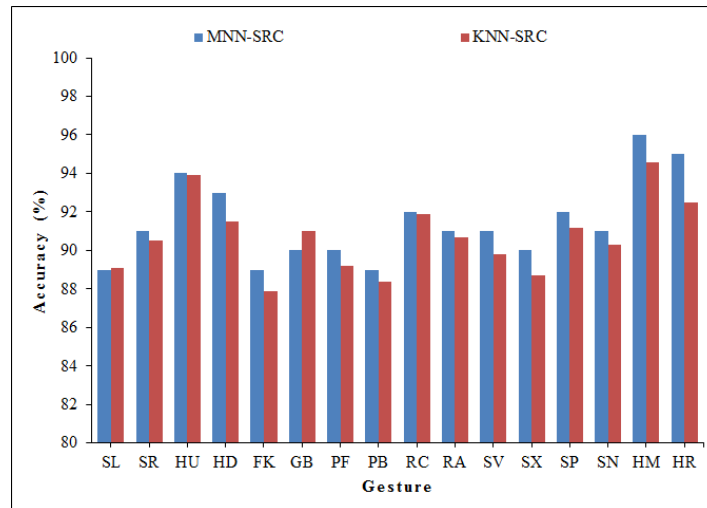


(b)

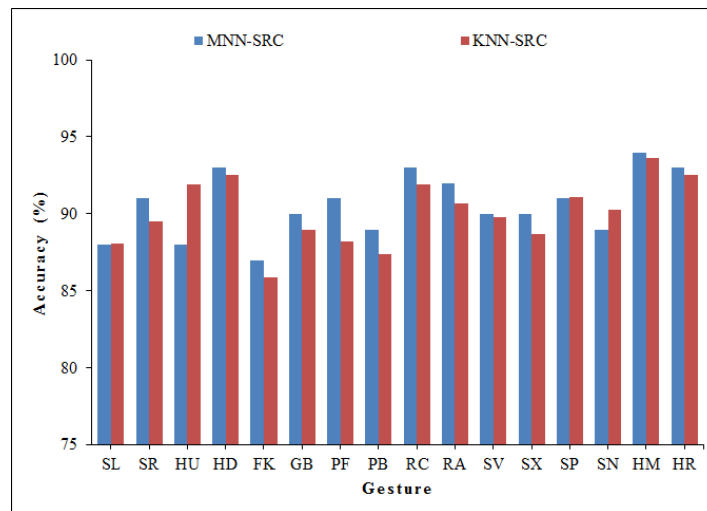


(c)

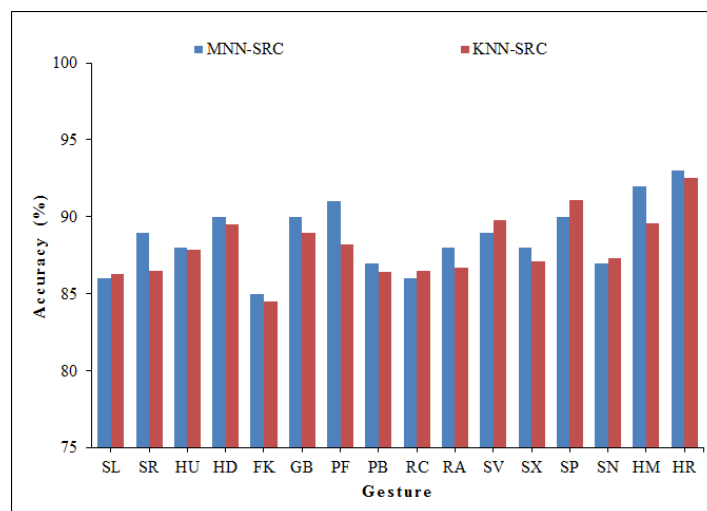
**Figure 6.** Comparison of accuracy using SRC and MNN separately. (a) Scenario-I. (b) Scenario-II. (c) Scenario-III.



(a)

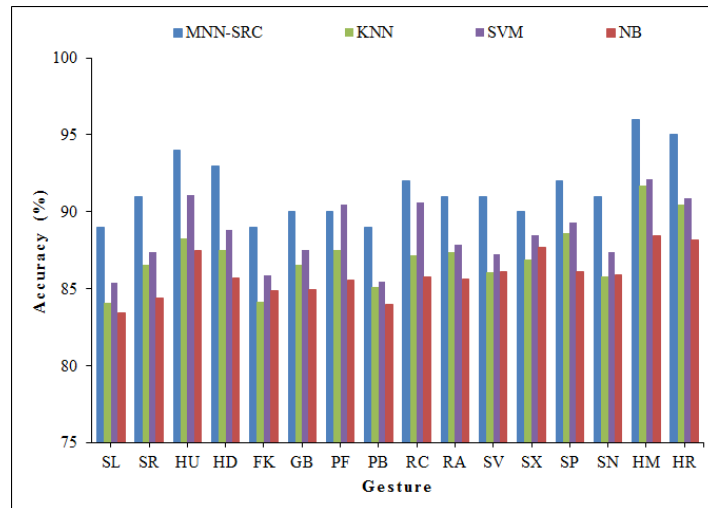


(b)

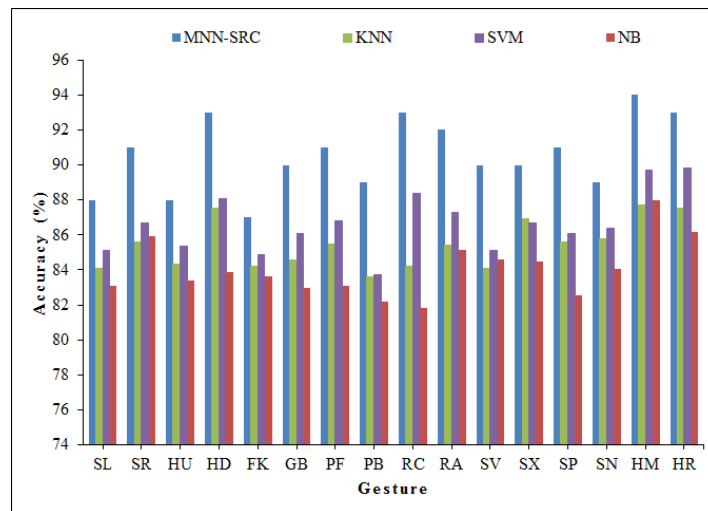


(c)

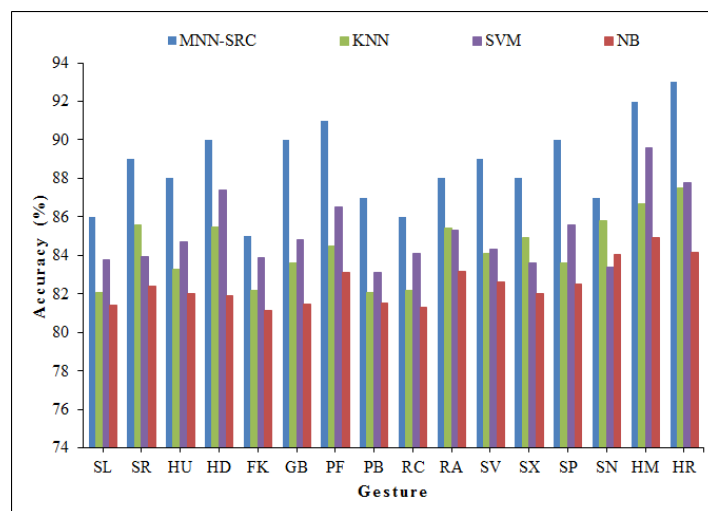
**Figure 7.** Comparison of accuracy using KNN-SRC and MNN-SRC. (a) Scenario-I. (b) Scenario-II. (c) Scenario-III.



(a)



(b)



(c)

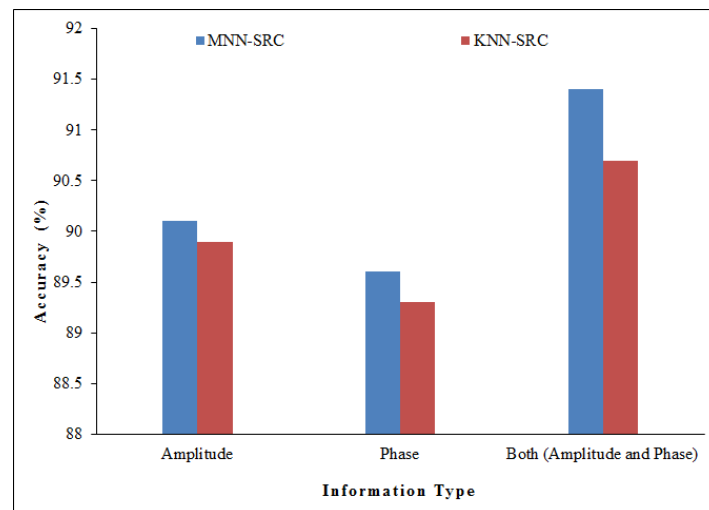
**Figure 8.** Comparison of accuracy using state-of-the-art classifiers. (a) Scenario-I. (b) Scenario-II. (c) Scenario-III.

The overall performance comparison of particular state-of-the-art classifiers is illustrated in Table 4. From the experimental results, it can be concluded that our proposed MNN-SRC method is much better as compared to KNN-SRC or stand-alone conventional classification methods including MNN, KNN, SRC, SVM and NB.

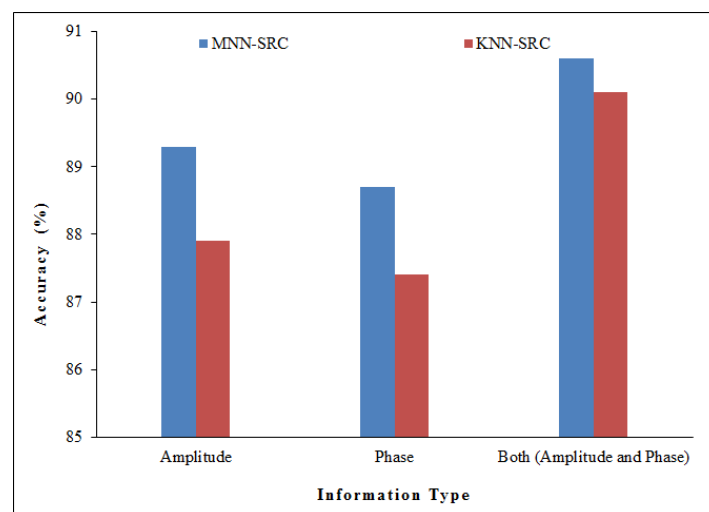
**Table 4.** Comparison of accuracy with state-of-the-art classification methods.

Experiment	Average Recognition Accuracy (%)						
	NB	SVM	KNN	MNN	SRC	KNN-SRC	MNN-SRC
Scenario-I	85.9	88.5	87.1	87.5	89.5	90.7	91.4
Scenario-II	84.1	86.6	85.4	86.2	88	90.1	90.6
Scenario-III	82.5	85.1	84.3	84.5	86.6	88.1	88.7

From Figure 9, we can observe that the proposed system could achieve reasonable performance even using only the amplitude or phase information. However, the recognition accuracy will be significantly better when we are combining both the amplitude and phase information.

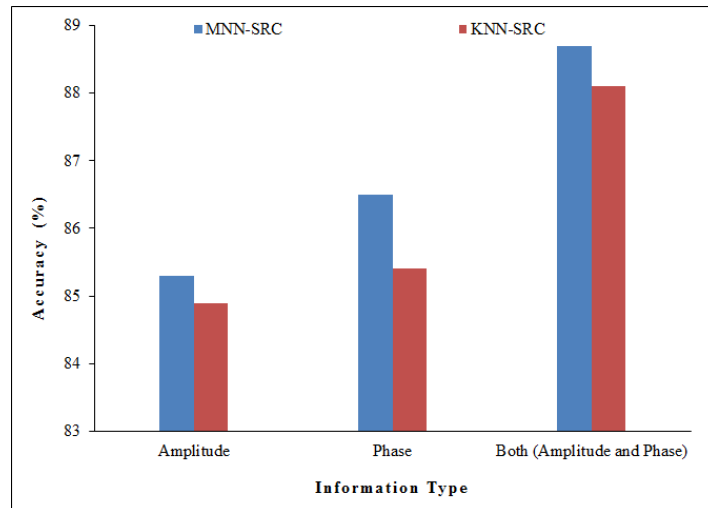


(a)



(b)

**Figure 9.** Cont.

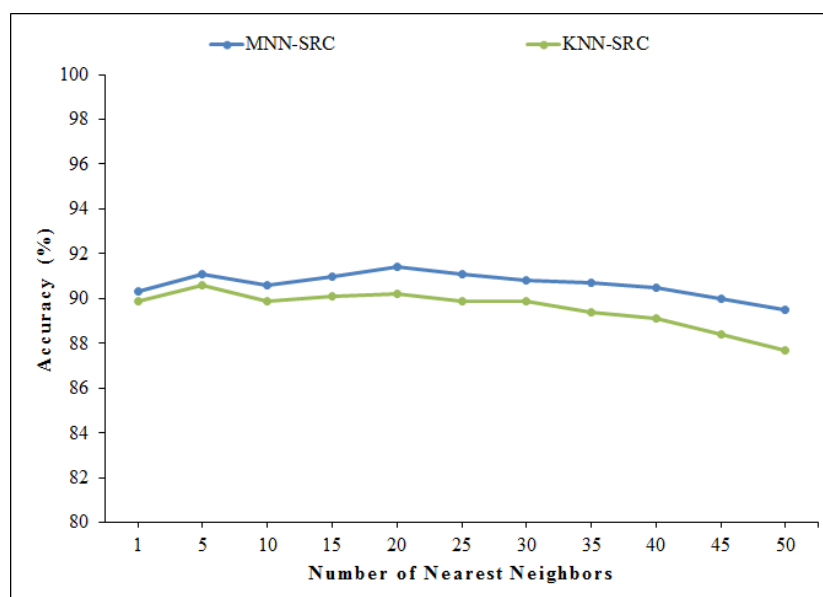


(c)

**Figure 9.** Average accuracy using amplitude and phase information. (a) Scenario-I. (b) Scenario-II. (c) Scenario-III.

To study the impact of nearest neighbors, we performed experiments with varying  $K$  values. From Figure 10, it is clear that MNN-SRC outperforms KNN-SRC for almost all  $K$  values. However, optimal values  $K = 20$  and  $K = 5$  were used for MNN and KNN, respectively, throughout our experiments. The detailed results with different  $K$  values of MNN-SRC are illustrated in Table 5.

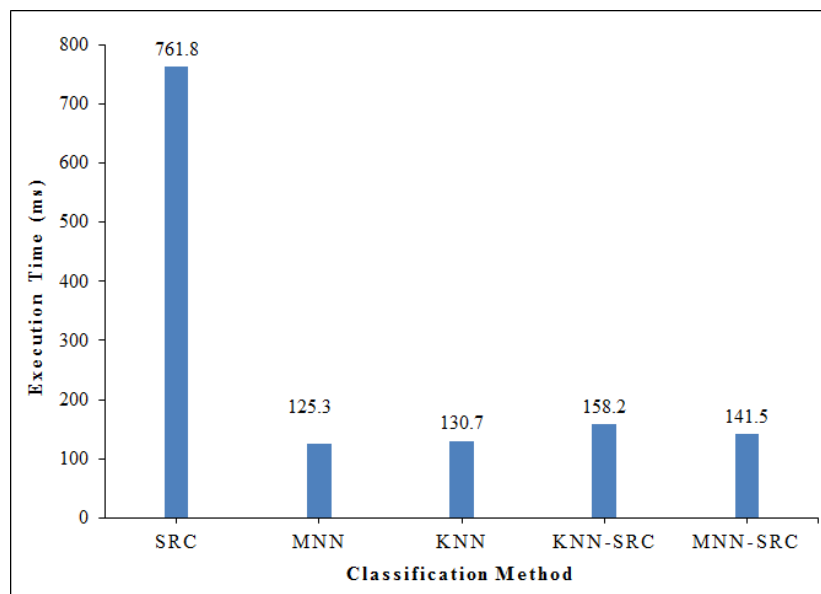
We have proven our results by calculating the execution time as illustrated in Figure 11. It is clear that the computational cost of our proposed algorithm MNN-SRC is much less with an execution time of 141.5 ms as compared to the SRC alone with an execution time of 761.8 ms. Furthermore, the execution time of MNN-SRC is less as compared to KNN-SRC. This is due to the fact that we are using the mean of nearest neighbors, which takes much less time to calculate its sparse coefficients as compared to traditional KNN. Although the execution time of MNN-SRC is a little higher as compared to MNN alone; however, as a compromise the recognition accuracy of MNN-SRC is comparatively better. The execution time of KNN is 130.7 ms, which is again higher as compared to MNN.



**Figure 10.** Effect of nearest neighbors.

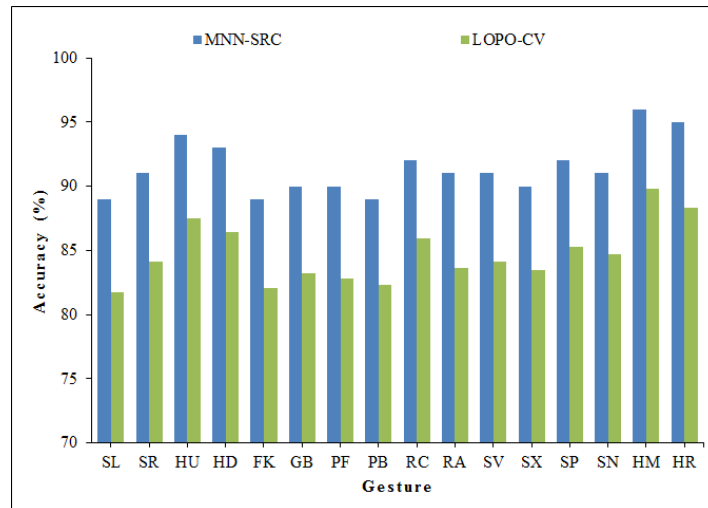
**Table 5.** Classification accuracy (%) of MNN-SRC with varying K.

K Value	Gesture Class															
	SL	SR	HU	HD	FK	GB	PF	PB	RC	RA	SV	SX	SP	SN	HM	HR
K = 1	88.3	90.5	90.4	89.3	87.1	87.7	88.3	87.9	91.4	89.3	90.6	90.1	92.3	90.3	96.0	94.5
1 < K < 5	<88.8	<90.8	<91.8	<90.5	<87.7	<88.6	<89.2	<88.2	<91.8	<90.4	<89.5	<90.3	<91.9	<90.1	<95.8	<94.7
K = 5	89.2	91.0	93.1	91.7	88.3	89.4	90.1	88.5	92.1	91.5	90.1	90.2	92.1	90.2	95.9	94.6
5 < K < 10	<89.2	<90.8	<92.8	<91.7	<88.7	<89.1	<89.8	<88.2	<92.0	<90.9	<90.4	<89.7	<92.0	<89.5	<94.7	<94.1
K = 10	89.1	90.6	92.4	91.6	89.1	88.8	89.5	87.9	91.8	90.2	89.3	89.5	91.2	89.1	95.3	94.0
10 < K < 15	<88.7	<90.9	<92.7	<92.0	<89.0	<89.3	<89.8	<87.7	<92.1	<90.4	<90.0	<90.0	<91.9	<89.8	<95.5	<94.4
K = 15	88.2	91.1	93.0	92.3	88.9	89.8	90.1	87.5	92.3	90.5	90.2	90.1	92.0	89.8	95.7	95.1
15 < K < 20	<88.6	<91.3	<93.4	<92.7	<89.1	<89.8	<90.2	<88.1	<92.4	<90.6	<90.7	<89.8	<92.2	<89.5	<95.6	<95.2
K = 20	88.9	91.4	93.7	93.1	89.2	89.7	90.3	88.6	92.4	90.7	91.1	89.6	92.3	89.6	95.9	95.2
20 < K < 25	<89.4	<91.3	<93.4	<93.0	<89.4	<89.4	<90.5	<87.9	<92.2	<90.5	<90.2	<89.9	<92.6	<90.1	<96.0	<94.6
K = 25	89.8	91.1	93.1	92.8	89.6	89.1	90.6	87.2	92.0	90.4	90.2	89.7	92.0	89.8	95.6	94.6
25 < K < 30	<89.0	<91.1	<93.0	<92.4	<89.7	<88.8	<89.9	<87.5	<92.1	<90.3	<90.1	<89.9	<92.2	<89.7	<95.0	<94.8
K = 30	88.2	91.0	92.8	91.9	89.7	88.5	89.1	87.7	92.2	90.3	90.2	89.9	92.1	89.8	95.4	94.7
30 < K < 40	<88.5	<90.9	<92.7	<91.8	<89.4	<88.3	<88.8	<87.5	<92.4	<90.3	<89.9	<89.6	<91.8	<89.8	<95.6	<94.3
K = 40	88.7	90.8	92.6	91.7	89.1	88.1	88.5	87.3	92.5	90.3	88.3	89.1	91.6	89.5	95.3	93.9
40 < K < 50	<88.5	<90.5	<91.5	<91.0	<88.3	<87.6	<88.1	<87.2	<91.5	<89.7	<88.1	<89.0	<91.3	<88.7	<94.9	<94.6
K = 50	88.3	90.1	90.3	90.2	87.4	87.1	87.7	87.0	90.4	89.1	88.2	88.7	91.1	88.3	94.5	93.7

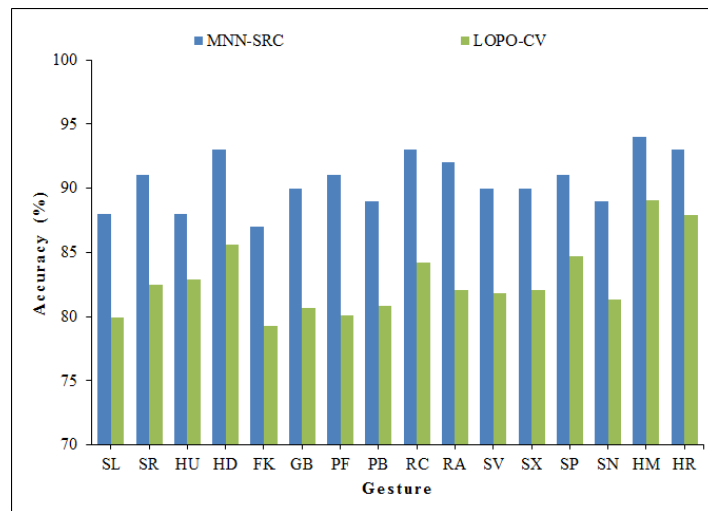


**Figure 11.** Comparison of execution time.

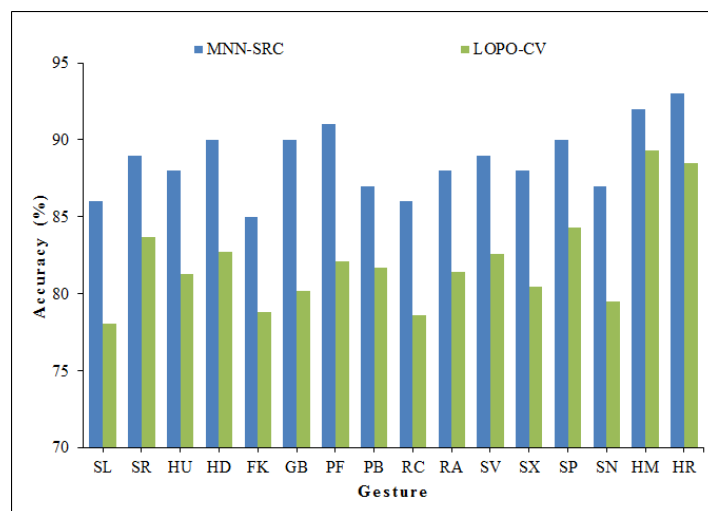
In order to present the practical performance of our proposed framework, we perform a user independence test. We particularly adopt the Leave-One-Participant-Out Cross-Validation (LOPO-CV) scheme, in which the training data do not know about the test user. LOPO-CV is an effective technique for evaluating the generalization of results for unseen data [51]. In this experiment, all data are treated as the training data set, except a particular person’s data that is selected as the test data. This process is repeated for each person. Figure 12, reveals the fact that our proposed method has acceptable performance even using LOPO-CV scheme with a recognition accuracy of 84.7%, 82.8% and 82.1% for scenarios I, II and III respectively. It can be concluded that our presented model is capable for the generalization of new users. The unbiased LOPO-CV estimator is difficult to implement due to its large amount of computation. However, it is suggested to acquire a large amount of training data from a variety of entities to get more better results.



(a)



(b)



(c)

**Figure 12.** Accuracy test using leave-one-participant-out cross-validation (LOPO-CV) scheme. (a) Scenario-I. (b) Scenario-II. (c) Scenario-III.

## 6. Discussion

In this section, we discuss the limitations and potential results obtained from the experiments. We observe that all gestures are classified with a very good recognition performance using our MNN-SRC classification algorithm, however several factors may influence the accuracy. In this context, some gestures have a great resemblance to each other, such as flick, which has great resemblance with grab and push hand forward. Similarly, push hand backward has great resemblance with grab and push hand forward which degrades the recognition accuracy. Although, all these types of limitations degrade the system performance; however, the overall performance of MNN-SRC is still better as compared to other algorithms.

Although CSI-based systems can achieve a reasonable performance, but there are still some limitations. Firstly, the CSI measurements are much more sensitive to moving objects. As a result, the recognition accuracy may suffer a degradation in performance when there is any other vehicle's motion in the testing area. Moreover, the system is designed by considering only a single person, that is, the driver. However, in real vehicle scenarios there may exist more than one person which can degrade system performance accordingly by making the recognition much more complex. In general, other vehicles on the road and people outside the vehicle may have a very slight influence [42]. Thus, additional signal processing may overcome these issues which we will consider in future.

Despite these limitations, our CSI-based device-free driver gesture recognition system is more scalable and easy to deploy as compared to other models. It should be noted that our proposed classification method is a general solution to solve any device-free localization and activity or gesture recognition problem. In this paper, we have utilized this method for in-vehicle driver gesture recognition.

## 7. Conclusions

In this paper, we have presented a novel framework for device-free robust driver gesture recognition. It can be concluded that the recognition rate is significantly improved by leveraging an integrated classification algorithm. Experimental results show that the mean of nearest neighbors based sparse representation coefficients framework can achieve remarkable performance in terms of gestures recognition and execution time. Our proposed integrated classifier is a promising algorithm for driver gesture recognition in the field of automotive vehicle infotainment systems.

This integrated classification approach opens a new direction for a diverse scale of potential applications. There are still several aspects that need be considered. In the future, we are interested in more complex driving scenarios based on the findings presented in this paper.

**Author Contributions:** Conceptualization, Z.U.A.A., H.W.; methodology, Z.U.A.A., H.W.; software, Z.U.A.A., H.W.; validation, Z.U.A.A., H.W.; formal analysis, Z.U.A.A.; investigation, Z.U.A.A.; resources, H.W.; data curation, Z.U.A.A., H.W.; writing—original draft preparation, Z.U.A.A.; writing—review and editing, H.W.; visualization, Z.U.A.A., H.W.; supervision, H.W.; project administration, H.W.; funding acquisition, H.W.

**Funding:** This work is supported by the National Natural Science Foundation of China under grant number 61671103.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Haseeb, M.A.A.; Parasuraman, R. Wisture: Touch-Less Hand Gesture Classification in Unmodified Smartphones Using Wi-Fi Signals. *IEEE Sens. J.* **2019**, *19*, 257–267. [[CrossRef](#)]
2. Li, Y.; Huang, J.; Tian, F.; Wang, H.A.; Dai, G.Z. Gesture interaction in virtual reality. *Virtual Real. Intell. Hardw.* **2019**, *1*, 84–112. [[CrossRef](#)]
3. Vuletic, T.; Duffy, A.; Hay, L.; McTeague, C.; Campbell, G.; Grealy, M. Systematic literature review of hand gestures used in human computer interaction interfaces. *Int. J. Hum.-Comput. Stud.* **2019**, *129*, 74–94. [[CrossRef](#)]



4. Grifa, H.S.; Turc, T. Human hand gesture based system for mouse cursor control. *Procedia Manuf.* **2018**, *22*, 1038–1042. [[CrossRef](#)]
5. Suchitra, T.; Brinda, R. Hand Gesture Recognition Based Auto Navigation System for Leg Impaired Persons. *Int. J. Adv. Res. Electr. Electron. Instrum. Eng.* **2017**, *6*, 251–257.
6. Ahmed, S.; Khan, F.; Ghaffar, A.; Hussain, F.; Cho, S.H. Finger-Counting-Based Gesture Recognition within Cars Using Impulse Radar with Convolutional Neural Network. *Sensors* **2019**, *19*, 1429. [[CrossRef](#)]
7. Zengeler, N.; Kopinski, T.; Handmann, U. Hand Gesture Recognition in Automotive Human-Machine Interaction Using Depth Cameras. *Sensors* **2019**, *19*, 59. [[CrossRef](#)]
8. Khan, F.; Leem, S.K.; Cho, S.H. Hand-Based Gesture Recognition for Vehicular Applications Using IR-UWB Radar. *Sensors* **2017**, *17*, 833. [[CrossRef](#)]
9. Cao, Z.; Xu, X.; Hu, B.; Zhou, M.; Li, Q. Real-time gesture recognition based on feature recalibration network with multi-scale information. *Neurocomputing* **2019**, *347*, 119–130. [[CrossRef](#)]
10. Large, D.R.; Harrington, K.; Burnett, G.; Georgiou, O. Feel the noise: Mid-air ultrasound haptics as a novel human-vehicle interaction paradigm. *Appl. Ergon.* **2019**, *81*, 102909. [[CrossRef](#)]
11. He, W.; Wu, K.; Zou, Y.; Ming, Z. WiG: WiFi-Based Gesture Recognition System. In Proceedings of the 24th International Conference on Computer Communication and Networks (ICCCN), Las Vegas, NV, USA, 3–6 August 2015.
12. Xiong, H.; Gong, F.; Qu, L.; Du, C.; Harfoush, K. CSI-based Device-free Gesture Detection. In Proceedings of the 12th International Conference on High-Capacity Optical Networks and Enabling/Emerging Technologies (HONET), Islamabad, Pakistan, 21–23 December 2015.
13. Li, H.; Yang, W.; Wang, J.; Xu, Y.; Huang, L. WiFinger: Talk to Your Smart Devices with Finger-grained Gesture. In Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing, Heidelberg, Germany, 12–16 September 2016; pp. 250–261.
14. Abdelnasser, H.; Harras, K.; Youssef, M. A Ubiquitous WiFi-based Fine-Grained Gesture Recognition System. *IEEE Trans. Mob. Comput.* **2018**, *18*, 2474–2487. [[CrossRef](#)]
15. Zhou, Q.; Xing, J.; Chen, W.; Zhang, X.; Yang, Q. From Signal to Image: Enabling Fine-Grained Gesture Recognition with Commercial Wi-Fi Devices. *Sensors* **2018**, *18*, 3142. [[CrossRef](#)] [[PubMed](#)]
16. Ali, K.; Liu, A.X.; Wang, W.; Shahzad, M. Recognizing Keystrokes Using WiFi Devices. *IEEE J. Sel. Areas Commun.* **2017**, *35*, 1175–1190. [[CrossRef](#)]
17. Tan, B.; Chen, Q.; Chetty, K.; Woodbridge, K.; Li, W.; Piechocki, R. Exploiting WiFi Channel State Information for Residential Healthcare Informatics. *IEEE Commun. Mag.* **2018**, *56*, 130–137. [[CrossRef](#)]
18. Al-qaness, M.A.A.; Li, F.; Ma, X.; Zhang, Y.; Liu, G. Device-Free Indoor Activity Recognition System. *Appl. Sci.* **2016**, *6*, 329. [[CrossRef](#)]
19. Mitani, Y.; Hamamoto, Y. A local mean-based nonparametric classifier. *Pattern Recognit. Lett.* **2006**, *27*, 1151–1159. [[CrossRef](#)]
20. Gou, J.; Ma, H.; Ou, W.; Zeng, S.; Rao, Y.; Yang, H. A generalized mean distance-based k-nearest neighbor classifier. *Expert Syst. Appl.* **2019**, *115*, 356–372. [[CrossRef](#)]
21. Gou, J.; Qiu, W.; Yi, Z.; Xu, Y.; Mao, Q.; Zhan, Y. A Local Mean Representation-based K-Nearest Neighbor Classifier. *ACM Trans. Intell. Syst. Technol.* **2019**, *10*, 29. [[CrossRef](#)]
22. Zou, J.; Li, W.; Du, Q. Sparse Representation-Based Nearest Neighbor Classifiers for Hyperspectral Imagery. *IEEE Geosci. Remote Sens. Lett.* **2015**, *12*, 2418–2422.
23. Yang, J.; Zhang, L.; Yang, J.Y.; Zhang, D. From classifiers to discriminators: A nearest neighbor rule induced discriminant analysis. *Pattern Recognit.* **2011**, *44*, 1387–1402. [[CrossRef](#)]
24. Bulut, F.; Amasyali, M.F. Locally adaptive k parameter selection for nearest neighbor classifier: One nearest cluster. *Pattern Anal. Appl.* **2017**, *20*, 415–425. [[CrossRef](#)]
25. Zhang, S.; Li, X.; Zong, M.; Zhu, X.; Wang, R. Efficient kNN Classification With Different Numbers of Nearest Neighbors. *IEEE Trans. Neural Netw. Learn. Syst.* **2018**, *29*, 1774–1785. [[CrossRef](#)] [[PubMed](#)]
26. Zhang, N.; Yang, J. K Nearest Neighbor Based Local Sparse Representation Classifier. In Proceedings of the 2010 Chinese Conference on Pattern Recognition (CCPR), Chongqing, China, 21–23 October 2010.

27. Xiao, J.; Zhou, Z.; Yi, Y.; Ni, L.M. A Survey on Wireless Indoor Localization from the Device Perspective. *ACM Comput. Surv. (CSUR)* **2016**, *49*, 25. [[CrossRef](#)]
28. Zhang, L.; Gao, Q.; Ma, X.; Wang, J.; Yang, T.; Wang, H. DeFi: Robust Training-Free Device-Free Wireless Localization With WiFi. *IEEE Trans. Veh. Technol.* **2018**, *67*, 8822–8831. [[CrossRef](#)]
29. Pu, Q.; Gupta, S.; Gollakota, S.; Patel, S. Whole-home gesture recognition using wireless signals. In Proceedings of the 19th Annual International Conference on Mobile Computing & Networking, Miami, FL, USA, 30 September–4 October 2013.
30. Adib, F.; Katabi, D. See through walls with WiFi. *ACM SIGCOMM Comput. Commun. Rev.* **2013**, *43*, 75–86. [[CrossRef](#)]
31. Liu, M.; Zhang, L.; Yang, P.; Lu, L.; Gong, L. Wi-Run: Device-free step estimation system with commodity Wi-Fi. *J. Netw. Comput. Appl.* **2019**, *143*, 77–88. [[CrossRef](#)]
32. Li, F.; Al-qaness, M.A.A.; Zhang, Y.; Zhao, B.; Luan, X. A Robust and Device-Free System for the Recognition and Classification of Elderly Activities. *Sensors* **2016**, *16*, 2043. [[CrossRef](#)]
33. Dong, Z.; Li, F.; Ying, J.; Pahlavan, K. Indoor Motion Detection Using Wi-Fi Channel State Information in Flat Floor Environments Versus in Staircase Environments. *Sensors* **2018**, *18*, 2177. [[CrossRef](#)]
34. Yang, X.; Xiong, F.; Shao, Y.; Niu, Q. WmFall: WiFi-based multistage fall detection with channel state information. *Int. J. Distrib. Sens. Netw.* **2018**, *14*, 1550147718805718. [[CrossRef](#)]
35. Al-qaness, M.A.A. Device-free human micro-activity recognition method using WiFi signals. *Geo-Spat. Inf. Sci.* **2019**, *22*, 128–137. [[CrossRef](#)]
36. Lv, J.; Man, D.; Yang, W.; Gong, L.; Du, X.; Yu, M. Robust Device-Free Intrusion Detection Using Physical Layer Information of WiFi Signals. *Appl. Sci.* **2019**, *9*, 175. [[CrossRef](#)]
37. Wang, T.; Yang, D.; Zhang, S.; Wu, Y.; Xu, S. Wi-Alarm: Low-Cost Passive Intrusion Detection Using WiFi. *Sensors* **2019**, *19*, 2335. [[CrossRef](#)] [[PubMed](#)]
38. Ding, E.; Li, X.; Zhao, T.; Zhang, L.; Hu, Y. A Robust Passive Intrusion Detection System with Commodity WiFi Devices. *J. Sens.* **2018**, *2018*, 8243905. [[CrossRef](#)]
39. Al-qaness, M.A.A.; Li, F. WiGeR: WiFi-Based Gesture Recognition System. *ISPRS Int. J. Geo-Inf.* **2016**, *5*, 92. [[CrossRef](#)]
40. Fu, Z.; Xu, J.; Zhu, Z.; Liu, A.X.; Sun, X. Writing in the Air with WiFi Signals for Virtual Reality Devices. *IEEE Trans. Mob. Comput.* **2019**, *18*, 473–484. [[CrossRef](#)]
41. Tian, Z.; Wang, J.; Yang, X.; Zhou, M. WiCatch: A Wi-Fi Based Hand Gesture Recognition System. *IEEE Access* **2018**, *6*, 16911–16923. [[CrossRef](#)]
42. Jia, W.; Peng, H.; Ruan, N.; Tang, Z.; Zhao, W. WiFind: Driver Fatigue Detection with Fine-Grained Wi-Fi Signal Features. *IEEE Trans. Big Data (Early Access)* **2018**. [[CrossRef](#)]
43. Duan, S.; Yu, T.; He, J. WiDriver: Driver Activity Recognition System Based on WiFi CSI. *Int. J. Wirel. Inf. Netw.* **2018**, *25*, 146–156. [[CrossRef](#)]
44. Arshad, S.; Feng, C.; Elujide, I.; Zhou, S.; Liu, Y. SafeDrive-Fi: A Multimodal and Device Free Dangerous Driving Recognition System Using WiFi. In Proceedings of the 2018 IEEE International Conference on Communications (ICC), Kansas City, MO, USA, 20–24 May 2018.
45. Halperin, D.; Hu, W.; Sheth, A.; Wetherall, D. Tool release: Gathering 802.11n traces with channel state information. *ACM SIGCOMM Comput. Commun. Rev.* **2011**, *41*, 53. [[CrossRef](#)]
46. Sen, S.; Radunovic, B.; Choudhury, R.R.; Minka, T. You are facing the Mona Lisa: Spot localization using PHY layer information. In Proceedings of the ACM 10th International Conference on Mobile Systems, Applications, and Services, Lake District, UK, 25–29 June 2012.
47. Wang, Y.; Wu, K.; Ni, L.M. WiFall: Device-Free Fall Detection by Wireless Networks. *IEEE Trans. Mob. Comput.* **2016**, *16*, 581–594. [[CrossRef](#)]
48. Feng, C.; Arshad, S.; Yu, R.; Liu, Y. Evaluation and Improvement of Activity Detection Systems with Recurrent Neural Network. In Proceedings of the 2018 IEEE International Conference on Communications (ICC), Kansas City, MO, USA, 20–24 May 2018.
49. Wang, W.; Liu, A.X.; Shahzad, M.; Ling, K.; Lu, S. Device-Free Human Activity Recognition Using Commercial WiFi Devices. *IEEE J. Sel. Areas Commun.* **2017**, *35*, 1118–1131. [[CrossRef](#)]

50. Arshad, S.; Feng, C.; Liu, Y.; Hu, Y.; Yu, R.; Zhou, S. Wi-chase: A WiFi based human activity recognition system for sensorless environments. In Proceedings of the 2017 IEEE 18th International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM), Macau, China, 12–15 June 2017.
51. Wu, C.T.; Dillon, D.G.; Hsu, H.C.; Huang, S.; Barrick, E.; Liu, Y.H. Depression Detection Using Relative EEG Power Induced by Emotionally Positive Images and a Conformal Kernel Support Vector Machine. *Appl. Sci.* **2018**, *8*, 1244. [[CrossRef](#)]



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).