Pervasive and Mobile Computing 69 (2020) 101289

Contents lists available at ScienceDirect

Pervasive and Mobile Computing

journal homepage: www.elsevier.com/locate/pmc

DF-WiSLR: Device-Free Wi-Fi-based Sign Language Recognition

Hasmath Farhana Thariq Ahmed [a], Hafisoh Ahmad [a,*], Kulasekharan Narasingamurthi [b], Houda Harkat [c,d], Swee King Phang [a]

[a] School of Computer Science and Engineering, Taylor's University, 1, Jalan Taylor's, Subang Jaya, 47500 Selangor, Malaysia
[b] Metier Technical Leader, Simulation CoE, Valeo India Pvt Ltd., 1/396, Old Mahabalipuram Road, Navallur, Chennai, Tamil Nadu 600130, India
[c] Instituto de Telecomunicações, Instituto Superior Técnico, Av. Rovisco Pais 1, 1049-001 Lisboa, Portugal
[d] Faculty of Sciences and Technologies, University of Sidi Mohamed Ben Abdellah, Road Imouzzer Fez, BP 2626, FES 30000, Morocco

ARTICLE INFO

Article history:
Received 30 June 2020
Received in revised form 29 October 2020
Accepted 30 October 2020
Available online 8 November 2020

Keywords:
Device-free sensing
Wi-Fi
Channel state information
Gesture recognition
Learning classifiers

ABSTRACT

Recent advancements in wireless technologies enable pervasive and device free gesture recognition that enable assisted living utilizing off the shelf commercial Wi-Fi devices. This paper proposes a Device-Free Wi-Fi-based Sign Language Recognition (DF-WiSLR) for recognizing 30 static and 19 dynamic sign gestures. The raw Channel State Information (CSI) acquired from the Wi-Fi device for 49 sign gestures, with a volunteer performing the sign gestures in home and office environments. The proposed system adopts machine learning classifiers such as SVM, KNN, RF, NB, and a deep learning classifier CNN, for measuring the gesture recognition accuracy. To address the practical limitation of building a voluminous dataset, DF-WiSLR augments the originally acquired CSI values with Additive White Gaussian Noise (AWGN). Higher-order cumulant features of orders 2, 3, and 4 are extracted from the original and augmented data, as the machine learning classifiers demand manual feature extraction. To reduce the computational complexity of machine learning classifiers, an informative and reduced optimal feature subset is selected using MIFS. Whilst the pre-processed original and augmented CSI values directly fed as input to an 8-layer deep CNN, it performs auto feature extraction and selection. DF-WiSLR reported better recognition accuracies with SVM for static and dynamic gestures in both home and office environments. SVM achieved 93.4% 98.8% and 98.9% accuracies in home and office environments respectively, for static gestures. For dynamic gestures, 92.3% recognition accuracy achieved in home environment. On augmented data, the corresponding gesture recognition accuracy values reported are 97.1%, 99.9%, 99.9%, and 98.5%.

1. Introduction

Sign languages are a form of communication among the hearing impaired and children with Autism Spectrum Disorder (ASD). Sign languages are not universal, and hence every part of the globe follows a unique sign language with its grammar and lexicon and attracts research interests in assisted living environment. Device-based recognition automates the recognition scheme with the deployment of sophisticated commercial devices in the sensing environment. Such methods

* Corresponding author.
E-mail address: Hafisoh.Ahmad@taylors.edu.my (H. Ahmad).

https://doi.org/10.1016/j.pmcj.2020.101289
1574-1192/

use commercial devices like depth cameras [1], and wearable inertial or motion sensors [2–4]. However, device-based sensing methods considered to be invasive and obtrusive, hence not a preferable choice for the majority of recognition applications [5,6].

Whereas, device-free recognition paradigm, achieve non-intrusive and privacy-preserving gesture recognition gathering the reflection pattern of the signal due to human movement [7,8]. Regardless of numerous signal information, Received Signal Strength Information (RSSI) [9], and Channel State Information (CSI) [10–15] are of importance for establishing a seamless recognition. Universal Software Radio Peripheral (USRP) [16,17], Wi-Fi routers [18], and RFID [19] are the widely adopted devices that capture essential signal information for device-free recognition. The hassle-free and pervasive characteristics of Wi-Fi signals motivate the present work to adopt commercial Wi-Fi routers for performing device-free gesture recognition of sign language.

This paper proposes a Device-Free Wi-Fi-based Sign Language Recognition (DF-WiSLR) system, pioneers Wi-Fi CSI dataset acquisition for 49 Indian Sign Language (ISL) gestures (static + dynamic). Static gestures are simple signs comprising alphabets, numbers and single words whereas dynamic gestures include compounding signs involving sentences. DF-WiSLR performs the recognition task in a confined environment by gathering CSI that unveils both amplitude and phase information enabling fine-grained gesture recognition [13]. The Wi-Fi CSI values gathered from wireless routers are prone to noise, thus necessitate pre-processing to make it usable for any sensing applications. State of the art applies regression or filtering techniques for removing the noise and applies feature extraction or dimensionality reduction technique [20–22]. However, the present work eliminates the application of filtering techniques on raw CSI values as there is a chance of losing essential signal information.

DF-WiSLR adopts classical machine learning classifiers such as Support Vector Machine (SVM), K-Nearest Neighbor (KNN), Random Forest (RF), Naï ve–Bayes (NB) and a deep learning classifier, Convolutional Neural Network (CNN) for measuring the sign gesture recognition accuracy. The performance of any machine learning classifiers relies upon the handcrafted feature, whereas, deep learning classifiers perform auto feature extraction and selection. For machine learning classifiers, the present work applies a 'standard score' normalization on raw CSI and extract higher-order cumulant features of order two, three, and four. Subsequently, applies a Mutual Information Feature Selection (MIFS) algorithm on the larger set of extracted features to reduce the computational complexity [23]. Alternatively, DF-WiSLR pre-processes the raw CSI values by applying the Multiple Linear Regression (MLR) technique before feeding it as input to the CNN.

The proposed study, computes the training and testing time of the measured recognition accuracy to identify best performing classifier with reduced computational effort. Nevertheless, recognition accuracy of learning-based approach improves with increasing number of training instances. However, collecting voluminous training instances from volunteers will be a burdensome task under some practical situations. Data augmentation address this limitation by expanding the data diversity eliminating the need of physical data collection [24]. DF-WiSLR implements data augmentation by applying an Additive White Gaussian Noise (AWGN) with different Signal to Noise Ratio (SNR) values on the initially acquired signal data that expands the size of the training data.

The remainder of the paper is organized as follows: Section 2 briefly discusses the related work in gesture recognition. Section 3 provides information on materials and methods adopted in this study. Details on experimental data acquisition and implementation of the learning algorithms are presented in Section 4. Section 5 discusses the performance of the learning algorithms with recognition accuracy and computational time as evaluation measures. Section 6 summarizes and concludes the present work.

2. Related work

The Wi-Fi CSI-based human gesture recognition studies generally adopt model-based or learning-based approach [6]. A wide range of reported studies under the model-based approach adopted Fresnel [25,26] or Angle of Arrival model (AoA) [27,28] for achieving better recognition accuracy. CARM [29] and WiDraw [27] adopted an Angle of Arrival (AoA) model and achieved recognition accuracy greater than 90%. In a recent study, FingerDraw [30] tracks the on-air finger trajectories of digits, alphabets, and symbols, and achieved an accuracy of 93% using CSI-quotient model. Model-based approach achieves better recognition accuracy for limited gesture class and requisite hardware tuning. Hence, it is of preferable choice for capturing coarse-grained gestures or detection-based applications. Therefore, for classifying fine-grained gestures, learning-based approach is preferred.

Learning-based approaches adopt classical machine learning classifiers or deep learning classifiers depending on the data size and perform the recognition by mapping the pattern of Wi-Fi signal to specific gesture. Reported research work adopting machine learning extensively applies pre-processing techniques, applies Principal Component Analysis (PCA) based feature extraction with SVM for achieving better recognition accuracy [21,22,31–34]. WiHear [35] applies a band-pass filter for pre-processing and extracts wavelet features. It also adopted the Multi-Cluster Feature Selection (MCFS) algorithm for selecting optimal feature inputs and achieved 91% recognition accuracy. Gesture recognition studies like HOS-Re [36] extracted third-order cumulant features from a raw CSI dataset [37] without applying any pre-processing. It achieves overall recognition accuracy more than 95% with SVM for 276 gesture class.

Moreover, deep learning classifiers such as CNN [37], RNN [38], ResNet [39], and LSTM [40] attracts interest in gesture recognition, as it automates the feature extraction and selection step. Deep Learning Architecture for Physical Activity Recognition (DELAPAR) incorporated auto extracted high-level CNN features [41] and achieved an improvement

in recognition accuracy from 89.8% to 96.6%. A fine-tuned CNN with Mel Frequency Cepstral Coefficient (MFCC) auditory features, achieved overall recognition accuracy of 95% for nine different gestures [42]. CSI-HC [43] performed classification of XingYiQuan martial arts movements using a Restricted Boltzmann Machine (RBM) model with a modified SoftMax layer and achieves an accuracy of 85.4%.

Recently, DeepMV [44] performs a recognition task, in a hybrid fashion (Wi-Fi devices + Ultrasound devices). DeepMV generates homogeneous (only Wi-Fi devices) and heterogeneous (Wi-Fi + acoustic devices) dataset, achieving an accuracy of 83.7% and 87.9%, respectively, adopting a CNN framework. In view of the above discussion, the proposed work adopts a variety of classical machine learning classifiers (SVM, KNN, RF, NB) and a deep learning classifier (CNN) for measuring the gesture recognition performance. In particular, the proposed work prefers CNN over other deep learning classifiers, based on reported literature showing higher classification accuracy adopting it.

3. Materials and methods

A brief explanation about CSI, techniques adopted to pre-process, and augment the data are presented in this section. Insights on feature extraction and selection process, adopting the learning classifiers for the present study, are also explained in detail in this section.

3.1. Channel state information

The channel properties of the wireless communication link, such as fading, power decay, and scattering, are estimated using RSSI and CSI. The RSSI can be gathered from almost all wireless devices without any special hardware requirements. However, the CSI can be obtained only by gaining access to the physical layer of the network through an exclusive Network Interface Card (NIC) such as Atheros [45] or Intel 5300 [46]. In comparison to RSSI, CSI reveals both phase and amplitude details of the Wi-Fi signal along with Channel Frequency Response (CFR) for accomplishing fine-grained recognition [11–15,30]. The transmission of Wi-Fi signal between the transmitter–receiver pair record different CSI values at transmitting and receiving end. Any sensing application gathers the CSI at the receiving end as it uncovers the signal reflection pattern with the presence of any moving person or objects in a closed environment.

The received CSI values usually computed at a subcarrier level with the implementation of the Orthogonal Frequency Division Multiplexing (OFDM) scheme. Thus, it enables the transmission of Wi-Fi signals with a higher data rate and capacity with a less bit error rate [47]. The time-series information of CSI values computed as a complex matrix,

$$R = HT + Noise \tag{1}$$

where R and T are the received and transmitted signals, respectively, H a complex CSI matrix. Estimation of the channel information done at regular intervals as the wireless medium is prone to frequent changes. For experiments, the proposed study deploys a TP-Link N300 wireless router (model: TL-WR840N) as transmitter and Dell Latitude E5400 laptop equipped with Intel 5300 NIC card as receiver. The 802.11n Linux CSI tool installed in the receiver extracts raw CSI information from the first 30 (out of 52) subcarriers of the Wi-Fi signal using NIC. The transmitter works at 2.4 GHz with 2-antennas at the rate of 300 Mbps.

3.2. Data pre-processing

The commercial Wi-Fi device exhibits a non-stationary signal with a trace of non-linearity present in it. The acquired Wi-Fi CSI values are prone to noise due to the hardware limitations of the commercial device and reported works adopt various pre-processing techniques [21,31,33,48]. However, the existing pre-processing methods completely ignore the non-linear behavior of the Wi-Fi signal and thus resulting in degradation of recognition accuracy. The proposed work DF-WiSLR eliminates the application of filtering or dimensionality reduction technique. The present work applies a 'standard score' normalization procedure on the absolute value of the raw complex CSI values while processing the input for machine learning classifiers. The normalization procedure subtracts the mean values from every column data, and all observations are normalized by applying the standard deviation of the data variable as follows,

$$\left.\begin{aligned} Y_1 &= \mathrm{X} - \mu(\mathrm{X}) \\ \mathrm{Y} &= Y_1 / \sigma(\mathrm{X}) \end{aligned}\right\} \tag{2}$$

where X and Y are the raw input and normalized data, respectively, with mean 'μ' and standard deviation 'σ'. Whereas, for a deep CNN, the absolute raw CSI values are pre-processed using a Multiple Linear Regression (MLR) technique to reduce the linear fitting error of the phase information. The phase offset ω across 'm' sub-carriers, with 'N_T' transmitting antennas estimated as,

$$\omega = \arg\min_{\eta} \sum_{N_T, m} (\theta_{N_T, m} + 2\pi\ (N_T - 1)\, \alpha + 2\pi f_s\, (m - 1)\, \eta + \gamma)^2 \tag{3}$$

where α, η and γ are the MLR fitting variables, θ denotes estimated CSI phase and f_s denotes the frequency spacing between m subcarriers. Finally, the pre-processed CSI values fed as input to the CNN is of the form,

$$H_{N_T, m} = \theta_{N_T, m} + 2\pi f_s\, (m - 1)\, \omega \tag{4}$$

3.3. Data augmentation

In the present work, CSI values of the Wi-Fi signal are augmented with Additive White Gaussian Noise (AWGN). AWGN augmentation is done with a universal assumption that the noise primarily occurring in the receiver end is completely independent of the received signal path [49]. DF-WiSLR prefer AWGN over other noise sources because higher-order cumulants can extract useful feature information even with the trace of WGN present in a non-Gaussian signal [50]. The mathematical representation of the AWGN (W) at time 't' is as follows,

$$W[t] = R[t] - T[t] \tag{5}$$

If transmitted signal $T = \overline{+}\mathrm{a}$, the error probability Pe is defined as,

$$\mathrm{Pe} = \mathrm{q}\left(a/\sqrt{N_0/2}\right) = \mathrm{q}(\sqrt{2SNR}) \tag{6}$$

where a - amplitude; N_0 - noise energy per symbol time and the Signal Noise Ratio SNR ($= a^2/N_0$) represents the ratio of the received signal to noise. q(.) is the complementary cumulative distribution function of a random variable N_o and determines the error probability of the channel. Therefore, the function q(.) decays exponentially with

$$\left.\begin{array}{c} \mathrm{q(T)} < e^{\frac{-T^2}{2}}, T > 0 \\ \mathrm{q}(T) > \dfrac{1}{\sqrt{2\pi}T}\left(1 - \dfrac{1}{T^2}\right)e^{\frac{-T^2}{2}}, T > 1 \end{array}\right\} \tag{7}$$

The raw and normalized CSI values are augmented with two SNR values 10 and 15, for expanding the training data, thrice the size of the original data, as with increasing SNR values, the signal error rate and the number of errors decreases.

3.4. Feature extraction and selection

DF-WiSLR extracts the second-order, third-order, and fourth-order cross-cumulant features from the normalized CSI values. These cumulants reveal the unbiased estimates of covariance (second-order), normalized skewness (third-order), and normalized kurtosis (fourth-order) respectively, of the CSI values [51]. The second-order cumulants compute the unbiased covariance estimates of the observed signal from the receiver, $x(n)$, $y(n)$ and $z(n)$. Let $x(n)$, $y(n)$ and $z(n)$ with $n = 1, \ldots, L$ denote three time-series, and N denote the samples per segment. The percentages of overlapped segments represented as $\hat{O}$ with the value $N_1 = N - N \times (\hat{O}/100)$. The time-series information of the observed signal at the receiver level is segmented into S records of N samples each, where $S = \left(L - N \times \left(\hat{O}/100\right)/N\right)$. The Sth segment of the xth sample is given by $x_s(i) = x(i + (\mathrm{s} - 1) \times N_1)$, with $i = 1, \ldots, N$ and $s = 1, \ldots, S$. Likewise $y_s(i)$ and $z_s(i)$ are defined in a similar fashion. The covariance is computed by removing the sample mean from every record as follows,

$$C_s(n) = \frac{1}{N(n)} \sum_i x_s(i)\, y_s(i+n) \tag{8}$$

where the summation over i tends from $1 + max(0, -n)$ to $N - max(0, n)$, with $N - max(0, n) - max(0, -n)$ as a normalizing parameter for unbiased estimates. The final estimate of the cumulants of S records is computed as,

$$C(n) = \frac{1}{S} \sum_{s=1}^{S} C_s(n) \tag{9}$$

The third-order cross-cumulants are computed as,

$$C_s(n, p) = \frac{1}{N(n, p)} \sum_i x_s(i)\, y_s(i+n)\, z_s(i+p) \tag{10}$$

where the summation over i tends from $1 + \max(0, -n, -p)$ to $N - \max(0, n, p)$, with $N - \max(0, n, p) - \max(0, -n, -p)$ as a normalizing parameter for unbiased estimates. The final estimate of the cumulants of S records computed as,

$$C(n, p) = \frac{1}{S} \sum_{s=1}^{S} C_s(n, p) \tag{11}$$

Similarly, the fourth-order cumulants are computed by adding the third parameter $z_s(i)$ to the third-order cumulant.

DF-WiSLR adopts MIFS, a greedy feature selection algorithm on the extracted set of cumulant features of orders 2, 3 and 4, to derive an optimal feature subset, with mutual information as a relationship measure. This helps to identify the non-linear relationship between the selected features and the corresponding output class with a regularization parameter β. The β value eliminates the redundant features and reduces the amount of uncertainty among the features within the optimal subset. This optimal feature subset is fed as input to the machine learning classifiers. The working of MIFS algorithm is as follows:

1. Initializing an empty feature set 'E'.
2. For every feature 'f' in the extracted feature set 'E_f' measure the mutual information between the features and the output class 'C'. $I(C:f)$
3. Pick the features in an increasing order that are highly mutually related to the output class and add them to the empty feature set. $E \leftarrow E\backslash\{f\}$ and $E \leftarrow \{f\}$
4. Perform a greedy selection of features for pre-defined numbers (30, 50 and 80)

    a. Compute $I(f, \hat{e})$, $\hat{e} \in E$, for every pair of features in the set 'E', if not computed already.
    b. The successive features that are highly mutually related are selected satisfying the condition,

$$I(C:f) - \beta \sum_{\hat{e} \in E} I(f, \hat{e}) \tag{12}$$

5. Finally, the set E consists of highly informative features and treated to be an optimal feature subset.

3.5. Gesture recognition classifiers

Wi-Fi CSI based device-free gesture recognition studies adopt a wide range of supervised or unsupervised learning classifiers for measuring the recognition accuracy. DF-WiSLR adopts machine learning classifiers such as SVM, KNN, RF, and NB, and also includes a CNN framework for performing deep learning task.

SVM

SVM perform classification task with kernels which help in transforming the data points and obtain an optimal boundary between the output classes. The linear and polynomial kernel of SVM separates the linear data for binary classification. However, SVM with Gaussian Radial Basis Function (RBF) kernel supports multi-class classification and non-linearly maps the samples in high dimensional space. Besides, able to handle if any non-linear relationship exists between the output gesture classes and the feature input. RBF represents the feature vectors of two samples v_1, v_2 in input domain as

$$K(v_1, v_2) = exp\left(-\frac{\|v_1 - v_2\|^2}{2\varsigma^2}\right) \tag{13}$$

where $\|v_1 - v_2\|^2$ represents the squared Euclidean distance between the two feature vectors and model fit parameter 'ς'. SVM-RBF defines Gamma and cost parameters to perform non-linear classification and measures the cost of misclassification. The present work implements the SVM-RBF classification using LIBSVM package [52].

KNN

KNN performs the classification task depending on similarity among the data points as a primary measure. KNN also referred as instance-based learning algorithm that classifies the input data by generating prediction for test instances, instead of building models. This approach considers Euclidean distance as a basic measure that locates k closely related training instances to every test instances. For example, the Euclidean distance between two points $p_1(xx_1, yy_1)$ and $p_2(xx_2, yy_2)$ is,

$$= \sqrt{(xx_2 - xx_1)^2 + (yy_2 - yy_1)^2} \tag{14}$$

Thus, perform the predictions based on these selected set of instances and through the majority voting of its neighbors. However, DF-WiSLR applies an extended version of the KNN algorithm to improve the classification accuracy, computing a local metric for each gesture class. The algorithm computes a large set of nearest neighbors, derives a locally optimal metric, and vote for the decision [53]. In the present work, KNN model selects 'inverse square distance' as the voting model for decision making.

RF

The traditional RF classifier work by creating forest in a random manner, and more accurate predictions are drawn with more number of trees. The classification step allows only a subset of randomly selected features from the best set of features derived from every node of the tree. DF-WiSLR adopts a bagging classifier, depending on the random vector values that are identical and independently distributed underlying the implementation of traditional RF. The bagging classifier considers all features to pick the best from every node of the tree, repeating the process 'bag' times as,

$$= \frac{1}{\text{Bag}} \sum_{\text{B}=1}^{\text{Bag}} f_{\text{B}}(b') \tag{15}$$

For $\text{B} = 1, \ldots,$ Bag , where f_B and b' represents the classification and predictions made considering all nodes of the tree, respectively and computes the majority voting. The random vectors are sampled with the same distribution throughout all tree structures in the forest, with each tree votes for the most popular class [54].

NB

The traditional NB algorithm often referred as probabilistic classifiers, generally applied over data that follows Gaussian distribution. Unlike traditional NB, the flexible NB classifier adopted for the present work can deal with data following a non-Gaussian distribution $P(D)$ and perform density estimation [55] as,

$$\left.\begin{aligned} P\left(u_2|u_1\right) &= P\left(D\right)/P(u_1) \\ &\propto P\left(D\right) \\ &= \prod_{d\in D} p(d|pa(d)) \end{aligned}\right\} \tag{16}$$

where, u_1 and u_2 represent the input and output variables, respectively, and $pa(d)$ is the set of parents of 'd' in the Bayesian network. The classification step adopts a simple approach and provides clear semantics for representing the probabilistic learning knowledge. The validation and parameter settings of machine classifiers are detailed in Section 4.2.2.

CNN

Deep CNN eliminates the need for manual feature extraction and performs deep learning tasks with various layers. Present work adopts an 8-layer deep CNN with input layers, convolution layer, batch normalization layer, Rectified Linear Unit (ReLU) layer, max-pooling layer, fully connected layer, SoftMax, and a classification layer. The input layer prepares the input data for the other layers. Sequentially, the convolution layer splits the input information into multiple regions and passes it through the convolutional filters or kernels that enable auto feature extraction. The batch normalization layer is defined between the convolution and ReLU layers to increase the training rate of the network and helps in normalizing the activations and gradients.

ReLU layer performs the task of forwarding the non-linear activated function or selective features to the next layer. Max pooling layer downsamples the size of the features, removes redundant information, and increases the number of filters without affecting the computational time. The fully connected layers connect the neurons of all other layers and identify the pattern of features for classifying the input data. The value assigned to the parameter 'output size' specifies the number of classes to be recognized. The output of the fully connected layers is normalized with the SoftMax layer and passed to the classification layer. Classification layer computes the loss, by assigning the input to the corresponding mutually exclusive classes, based on the probability value returned by the SoftMax layer. The implementation and parameter setting of all layers are mentioned in Section 4.2.3.

4. Implementation

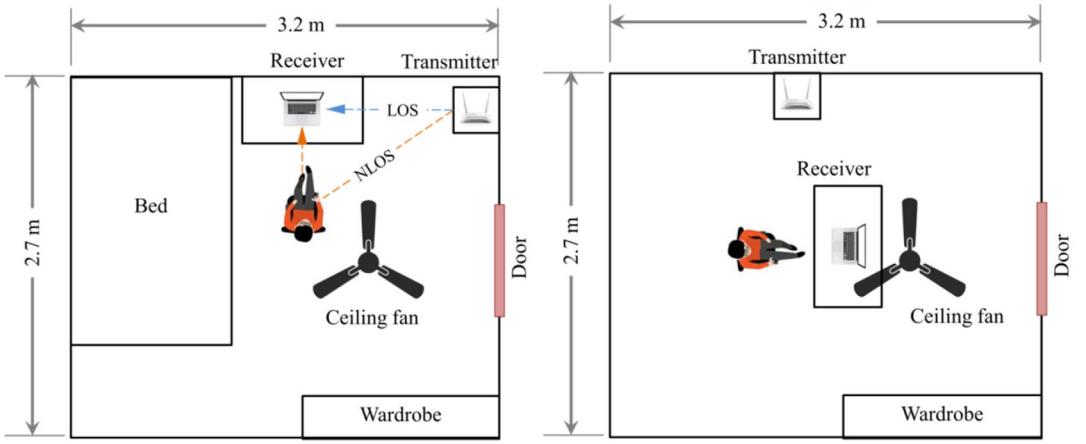
This section discusses the experimental setup with details including volunteers, gesture orientation (static and dynamic), gesture classes, and numbers of instances acquired per gesture. The implementation of the proposed methodology adopting the machine learning classifiers and deep CNN also explained in subsequent sections.

4.1. Experiment setup

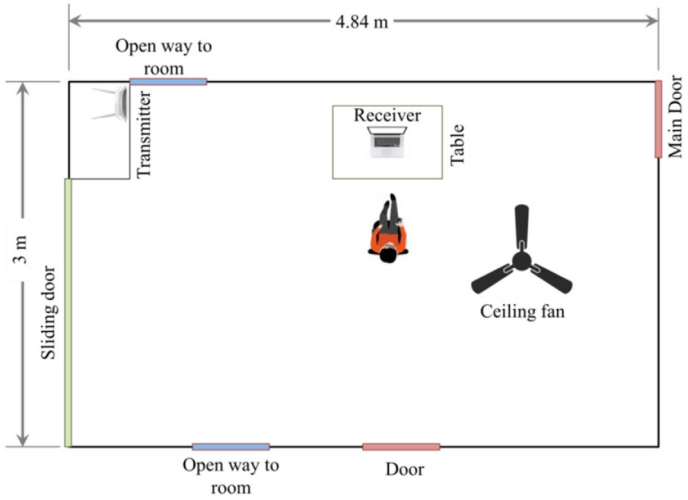
In the present study, DF-WiSLR gathers the raw CSI values for 30 static and 19 dynamic sign gesture classes of ISL, in two environment 'home' and 'office'. The home and office environments are distinctive in room dimensions, relative position of transmitter, receiver, and distance between transmitter–receiver pair and type of sign gesture (static or dynamic) recorded for the study. Therefore, the multipath reflection pattern of the Wi-Fi signal also changes depending on the interferences present in the respective environments. The raw CSI values acquired from 'single user' in the home and office environment using the hardware/tool, as mentioned in Section 3.1. The experimental layout of the home and office environment is shown in Fig. 1. The home and office environments are of dimension 3.2 m $\times$ 2.7 m and 4.84 m $\times$ 3.0 m, respectively. The home environment E1(a) is a furnished room of height 3.5 m with a bed and a table, as in Fig. 1(a). Whereas home environment E1(b) is the same room without any occlusions — an empty room, as in Fig. 1(b). The office environment E2, shown in Fig. 1(c), is also an empty room, however more spacious than the home environment.

The transmitter–receiver pair placed at a distance of 1.4 m and 1.5 m for static and dynamic gestures, respectively, in the home environment. In the office environment, this distance is 2.3 m for collecting the static gestures. The straight line connecting the transmitter–receiver pair, defined as the Line of Sight (LOS) path. In both environments, the volunteer positioned at a Non-Line of Sight (NLOS) path. The home and office environments are distinctive in room dimensions, relative position of transmitter, receiver, and distance between transmitter–receiver pair and type of sign gesture (static or dynamic) recorded for the study. Therefore, the multipath reflection pattern of the Wi-Fi signal also changes depending on the interferences present in the respective environments.

Table 1 summarizes 30 static and 19 dynamic sign gestures of ISL chosen from the vast repository of National Institute of Open Schooling (NIOS) [56]. The present study select frequently used signs with unique sign poses, to avoid similarities in the signal pattern. The 30 static signs include 10 numbers, 10 alphabets, and 10 words. The 19 dynamic signs include 5 numbers, 8 words, and 6 sentences. The CSI information gathered in a single user environment, with static gestures at E1(a), E1(b), and E2 and dynamic gestures at E1(a). The volunteers are educated about the ISL sign gesture using the NIOS videos before initiating the data gathering process.



(a) '*Home*' environment E1(a) – Furnished (b) '*Home*' environment E1(b) – Empty



(c) '*Office*' environment E2

Fig. 1. Environment layout.

Table 1
ISL gestures.

Static sign gestures			Dynamic sign gestures		
Numbers	Alphabets	Words	Numbers	Words	Sentences
0	A	Home	0	Act	Hello, what is your name?
1	B	Time	1	Ambulance	How are you?
2	C	Today	2	Namaste	I'm fine
3	D	Cold	3	Bandage	That's good
4	E	Food	4	Soap	Take care
5	F	Drink		Email	Sit down
6	G	Water		Medicine	
7	I	Sleep		Things	
8	L	Love			
9	M	Blind			

CSI values acquired after the volunteer start posing the gesture. During static gesture data collection, the volunteer freeze the pose and it is ensured that no additional movement is performed for every instance. In contrast, dynamic sign poses involve continuous movement of volunteer hand, fingers and repeated for the entire duration of data acquisition. Besides, dynamic gestures of sentences, compound one or more-word signs. For example, the sign gesture of the sentence

Table 2
Summary of data collection.

Environment		Home E1(a)	Home E1(a)	Home E1(b)	Office E2
Volunteer details	Volunteer	V1	V1	V2	V3
	Age (Years)	28	28	39	42
	Weight (kg)/Height (cm)	73/152	73/152	87/173	75/168
Gesture type		Static	Dynamic	Static	Static
Environment condition		Furnished	Furnished	Empty	Empty
Data acquisition date		11/1/2020	21/1/2020	18/1/2020	24/1/2020
No. of sign gestures		30	19	30	30
No. of instances per gesture		20	20	20	20
Total no. of gesture instances		600	380	600	600
Time taken per instance (s)		∼1.5 – 2 s	∼2 s	∼1.5 – 2 s	∼1.5 – 2 s

'I'm fine' combines signs of two words 'I'm' and 'fine'. The summary of data collection and other details regarding volunteers are shown in Table 2.

DF-WiSLR captures the raw CSI values from volunteer 1 (V1) in the home environment E1(a). In the home environment E1(b), and office environment E2 data gathered from volunteer 2 (V2) volunteer 3 (V3), respectively. Note that static sign gestures are highly fine-grained than the dynamic sign gestures, and the complexity of recognizing static signs is higher than dynamic signs. The video of 49 ISL gestures reported in the present study, clipped from the original NIOS videos, is made available in a public repository[1] for quick reference.

4.2. DF-WiSLR methodology

The systematic process of sign gesture recognition adopted by DF-WiSLR is explained in Fig. 2, representing static gesture data collection from E1(a) with 600 instances, for illustration purpose. Linux 802.11n CSI tool initiates ample collection of CSI values at the receiver end of structure (struct in short) data type of size $2 \times 30 \times 3$, obtained from 2 transmitting antenna (N_T) with 30 subcarrier (m) and 3 receiving channels (N_c). The noise prone characteristic of the acquired signal necessitates trimming the front and rear segment of the gathered CSI values. In the present work, 75 number of struct from the useful portion of the collected data is considered for assembling the raw CSI dataset. Therefore, every gesture instance comprise 75 rows of information that forms the raw data of size $150 \times 30 \times 3$ ($= (75 \times 2) \times 30 \times 3$).

The individual raw data files obtained from each instance assembled to form a 4-dimensional original raw CSI dataset. For 30 static gestures with 20 instances per gesture, the size of the raw CSI dataset will be $150 \times 30 \times 3 \times 600$. Likewise, for 19 dynamic gestures with 20 instances per gesture, the dataset size is $150 \times 30 \times 3 \times 380$. A 'standard score' normalization and MLR technique applied to the absolute value of the raw complex CSI data before passing it to the machine learning classifiers and deep CNN, respectively as explained in Section 3.2.

The size of the normalized data retained the same as that of raw data assembly. With the intent of increasing the training data thrice the size of the originally acquired data, the AWGN augmentation technique is applied on both raw and normalized data. As a result, a matrix of size $150 \times 30 \times 3 \times 1800$ and $150 \times 30 \times 3 \times 1140$, obtained for static and dynamic gestures, respectively. The subsequent section briefly explains the feature extraction, selection and gesture recognition process using the machine learning classifier and deep CNN, adopted in the present study.

4.2.1. Feature extraction and selection

A distinctive set of second-order, third-order, and fourth-order cross-cumulant features are extracted and mutually informative optimal features are selected as shown in Fig. 3. The raw CSI input data obtained for each instance is of size $150 \times 30 \times 3$. The assembled raw dataset for 30 gestures with 20 instances each will be of size $150 \times 30 \times 3 \times 600$. This assembled dataset normalized using a 'standard score' normalization procedure to generate a normalized dataset, prior feature extraction step. The normalized original data split into chosen quantity of sample segments ($N = 128$) with a window overlap of 1% between successive sampling windows. The maxlag value mainly limits the total number of cross-cumulant features extracted from the input data.

The cross cumulant features of order 2, 3 or 4 (covariance, skewness or kurtosis respectively) are extracted by assigning the maxlag value as 2. It extracts first 5 cumulant coefficients from 3 receiving channels (N_c) with 30 subcarriers (m) resulting the formation of a extracted feature set comprising 450 features ($f = 5 \times 30 \times 3$). For illustration purposes, the contour plots of samples, static sign gesture – '5' and dynamic sign gesture – 'I am fine', are shown. It represents the symmetrical behavior of the unbiased cumulant estimates, plotted with first and second time lag. Hence, a feature matrix of size 600×451 and 380×451 generated for static and dynamic sign gestures consisting of 600 and 380 instances, respectively. The gesture labels are manually added to the last column of the feature matrix. Correspondingly

[1] https://github.com/hasmaththariq/DF-WiSLR.

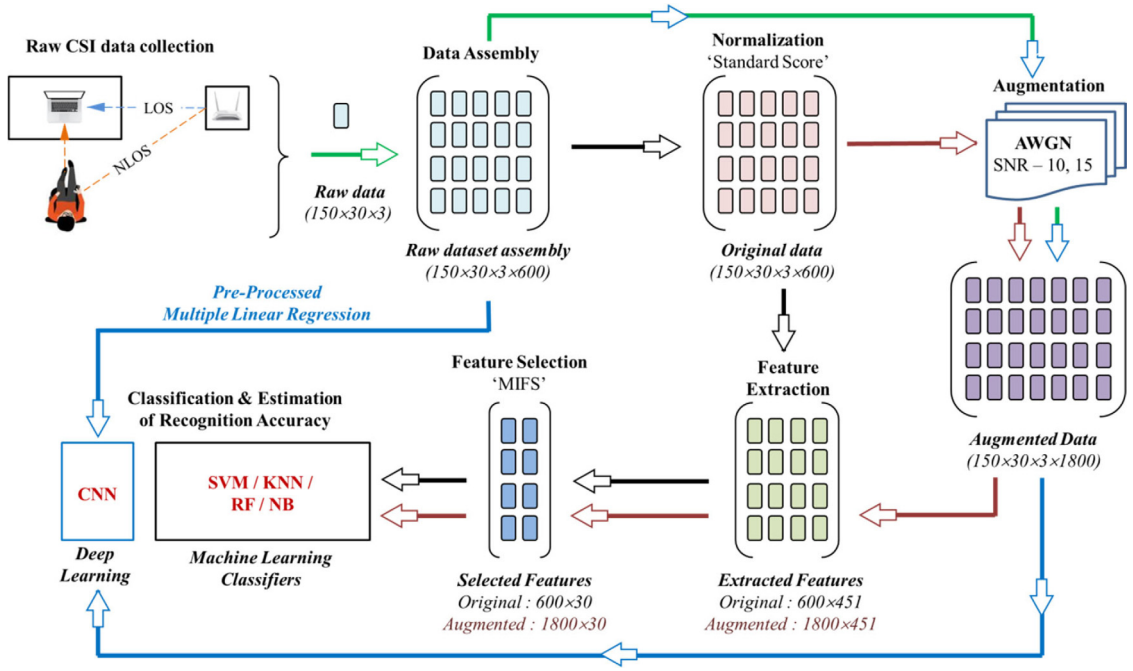


Fig. 2. DF-WiSLR gesture Recognition Methodology.

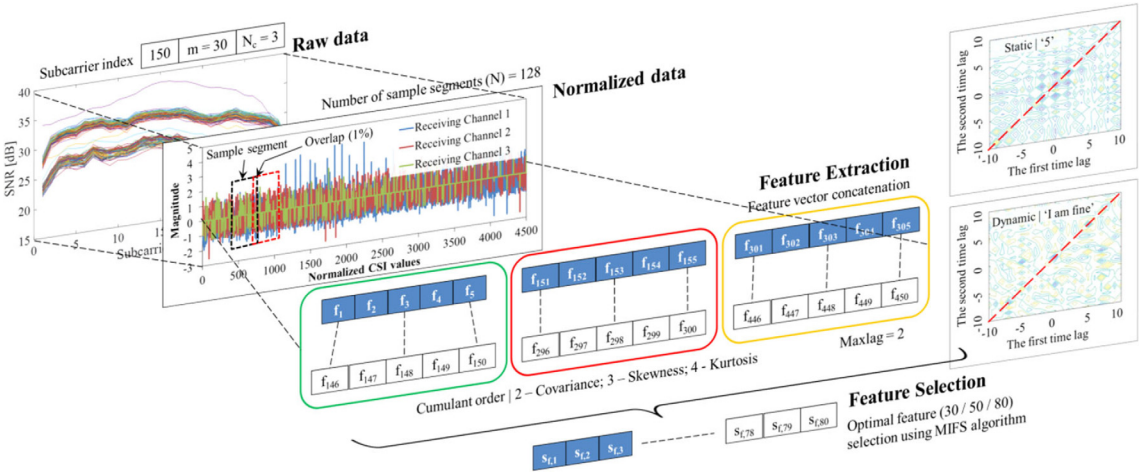


Fig. 3. Feature extraction and selection.

a feature matrix of size 1800 × 451 and 1140 × 451 are formed by AWGN augmentation for static and dynamic gestures respectively (refer Fig. 2). The cumulant features are extracted using Higher Order Spectral Analysis (HOSA) [51] toolbox with MATLAB.

Subsequently, MIFS algorithm picks the optimal subset of mutually informative features as explained in Section 3.4. MIFS optimally selects a feature subset consisting of 30 ($s_{f,1}$to $s_{f,30}$) or 50 ($s_{f,1}$to $s_{f,50}$) or 80 ($s_{f,1}$to $s_{f,80}$) features from the extracted set of 450 features ($f_1, f_2, \ldots, f_{450}$). These optimal features 's_f' are selected separately for each of second, third, and fourth-order features with β values ranging between 0 and 1 ($\beta = 0.1, 0.3, 0.5, 0.8$ and 1). Finally, the optimal feature matrices of the original and augmented data (600 × 30/50/80 and 1800 × 30/50/80 respectively) serve as input to the machine learning classifiers (as in Fig. 2).

4.2.2. Machine learning classifiers

DF-WiSLR normalizes the raw CSI data for measuring the overall recognition accuracy using machine learning classifiers such as SVM, KNN, RF, and NB. The training and testing conditions for a learning classifier can be defined in different

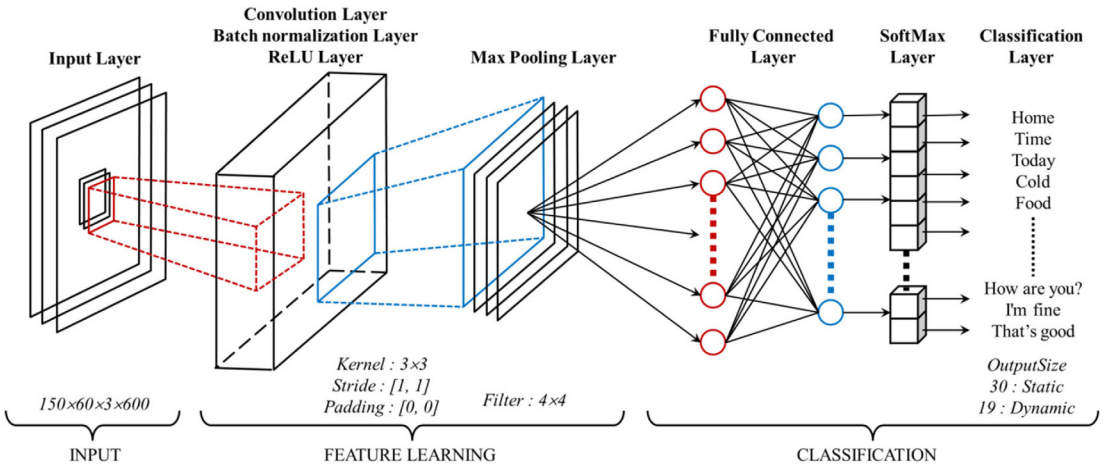


Fig. 4. Classification and gesture recognition with deep CNN.

scenarios as explained in [57,58]. For the present study, environmental bounded ('Trained' approach) training and testing conditions are defined. Thus, the training and testing of instances are done with respect to every environment in which the data is collected. For the 10 × 5 cross-fold validation of the dataset using SVM, the recognition accuracy values are measured by partitioning the dataset obtained from each of the environment, as 80% for training and 20% for testing. For experimental analysis, the Gamma and cost parameter of SVM-RBF takes values following the correlation $0.001e^{1.1513j}$ where j range from 1 to 12 in steps of 1. Other classifiers such as KNN, NB, and RF take an input of batch size equal to 100, and the validation strategy is simple 5-fold cross-validation without repetition. For KNN, the number of nearest neighbors used to vote is equal to 1, whereas the size of the local set used to induce local metric assigned as 100. The chosen type of distance measure used for the attributes is City and Simple Value Difference (City-SVD), which combines City-block Manhattan and Simple Value Difference metric for numerical and symbolic attributes, respectively. The corresponding vicinity size of the density-based metric set to 200 with a 'Distance-based' weighting method for measuring the recognition performance.

For the present study, RF classifier uses a seed equal to 1 and the size of each bag, as a percentage of the training set size, fixed as 100. The maximum depth of the tree is unlimited, and the number of randomly chosen attributes set to the value: *int* ($\log_2$ (*predictors*) + 1). The number of execution slots deployed for constructing the ensemble set to 1 and measures the recognition accuracy, assigning the number of trees in the model as 100. Lastly, the NB classifier predicts the output gesture class of test instances following a normal distribution with no supervised induction. The implementation of SVM is done in MATLAB, whereas the implementation of other classifiers performed using the WEKA tool [59].

4.2.3. Deep CNN

Applying deep learning technique for small datasets apparently affects the convergence of the network with mediocre outcomes. In such scenario, popular techniques like data augmentation, k-fold cross-validation, synthetic data generation and transfer learning will be of preferred choice to deal with limited data. In view of this, the present work adopted data augmentation and k-fold cross-validation with the aim of studying the influence of dataset size with respect to the convergence of deep CNN. Details on data augmentation are already explained in Section 3.3. Fig. 4 represents the architecture of an 8-layer Deep CNN adopted in the present study. The input layer initially prepares the pre-processed input data applying MLR of size 150 × 30 × 3 × 600, as explained in Section 3.2. Every sign gestures consist of 75 CSI values per transmission channel, and a CSI matrix of dimension 150 × 30 × 3 computed per gesture class. Later, the input layer transforms it as a tensor matrix of size 150 × 60 × 3 × 600. For convolutional layer, the kernel of size 3 × 3, is adopted with stride 1 and zero paddings along rows and columns of the input data. A [4, 4] rectangular region filter is deployed by the training function to scan the inputs in Max Pooling layer. In the present study, the OutputSize in the fully connected layer is defined as 30 and 19 for classifying static and dynamic sign gestures, respectively. SoftMax layer normalizes the output from the fully connected layer. The adopted learning rate is equal to 0.01, and the implementation is done using MATLAB for 5-fold cross-validation.

5. Evaluation

DF-WiSLR evaluates the sign gesture recognition with two performance criteria of the learning classifiers: (1) Recognition accuracy and (2) Training and testing time consumption, for accurately classifying the gesture class. The following sections briefly discuss the performance criteria in accordance with other factors that influence the recognition accuracy. Lastly, compares the performance of the present work with the related work reported in the literature.

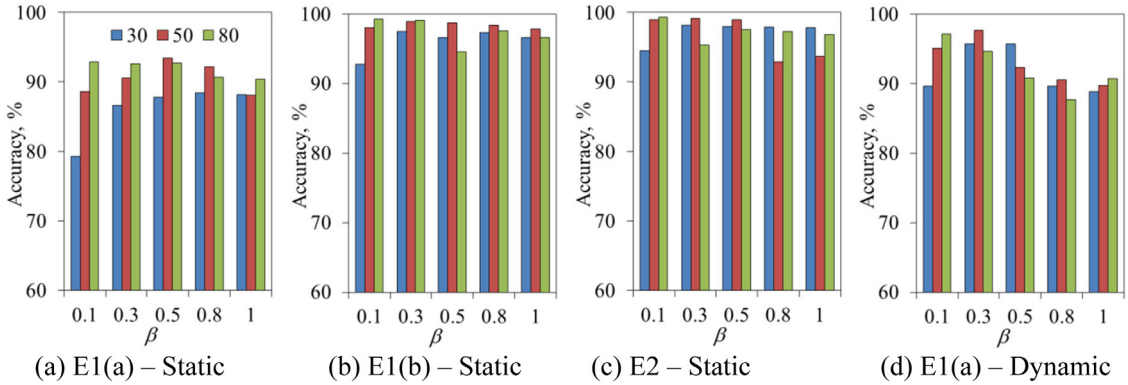


Fig. 5. Comparison of recognition accuracy with SVM classifier — Original data.

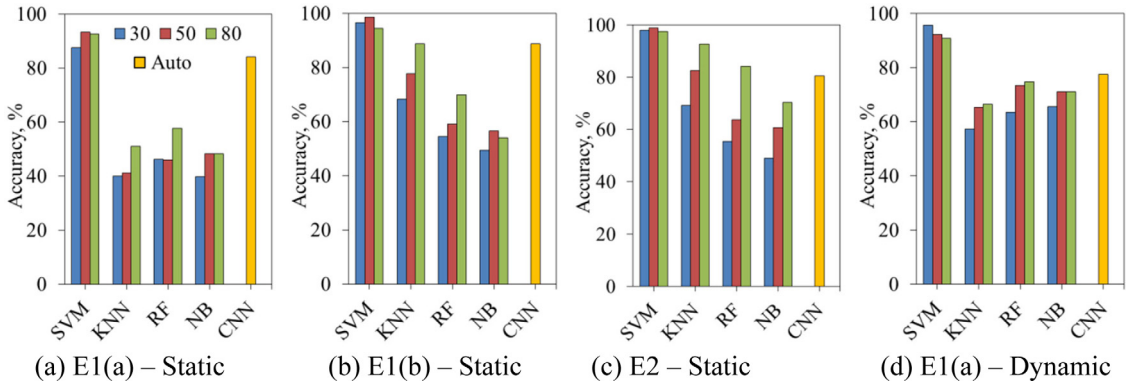


Fig. 6. Comparison of recognition accuracy with different classifiers — Original data.

5.1. Gesture recognition accuracy

The ratio of the correctly recognized sign gestures to the total number of tested sign instances of the same gesture is defined as gesture recognition accuracy. DF-WiSLR determines the static and dynamic gesture recognition performance. Besides, analyze the influence of cumulant order, optimal feature subset, data augmentation, gesture orientation, and environmental factors on gesture recognition accuracy. The measured accuracy values are compared between machine learning classifiers and a deep learning classifier to identify the best performing classifier.

5.1.1. Classifier performance

The machine learning classifiers considers 30, 50 and 80 optimal features of MIFS algorithm as input to the classification task. The recognition performance evaluated for second-order cumulant features of original data using SVM with five different regularization parameter $\beta = 0.1, 0.3, 0.5, 0.8$ and 1, as shown in Fig. 5. SVM reported better accuracy values of 93.4% and 98.8% for static gesture in home environments E1(a) and E1(b) respectively at $\beta = 0.5$ with 50 optimal features. In office environment E2, an accuracy of 99.2% observed at $\beta = 0.3$ and with $\beta = 0.5$ the accuracy value dropped slightly to 98.9% with 50 optimal features. In E1(a), dynamic gestures report 97.7% accuracy at $\beta = 0.3$ with 50 optimal features.

A comparative analysis of gesture recognition accuracy performed within SVM, KNN, RF, NB, and CNN shown in Fig. 6. The classifiers are evaluated with 30, 50, and 80 optimal features of original data for $\beta = 0.5$, as this value observed to report nearly better performance with SVM. Among the static gesture environments, KNN and RF classifier achieved the best performance in office environment E2, which reports 92.7% and 84.2% accuracies, respectively, with 80 optimal selected features. With dynamic gestures, the performance of KNN and RF observed declining to 66.6% and 74.7%, respectively. NB reported its best accuracy value of 71.1% for dynamic gestures and 70.3% for static gestures in office (E2) with 80 features. This indicates NB classifier achieves the least performance compared to all other classifiers, even with 80 optimal features. CNN with auto feature extraction achieved accuracies of 84.2%, 88.8%, 80.5% for static gestures in E1(a), E1(b), E2 and 77.6% for dynamic gestures in E1(a).

The comparative analysis between classification algorithms adopted in the present study implies that SVM reports better recognition performance on original data with 50 optimal features. Therefore, in the subsequent section, the recognition performance of second-order features is further compared with third and fourth-order cumulant features

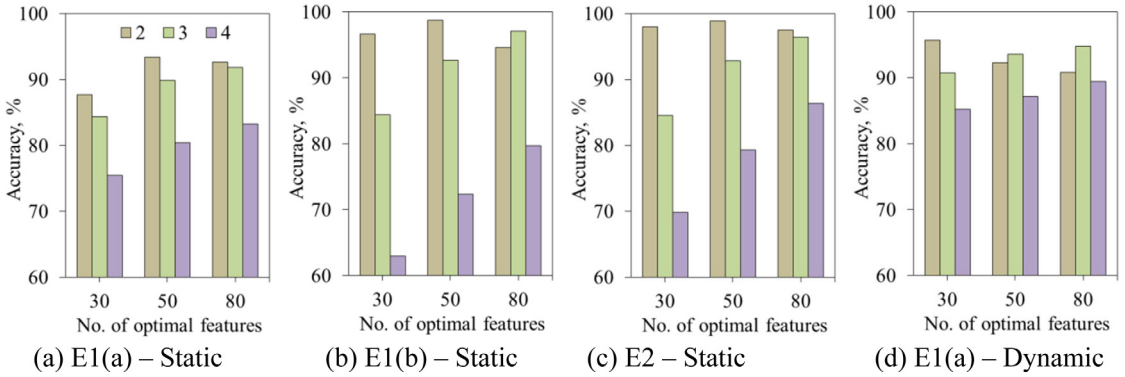


Fig. 7. Comparison of recognition accuracy with Cumulant order — Original data.

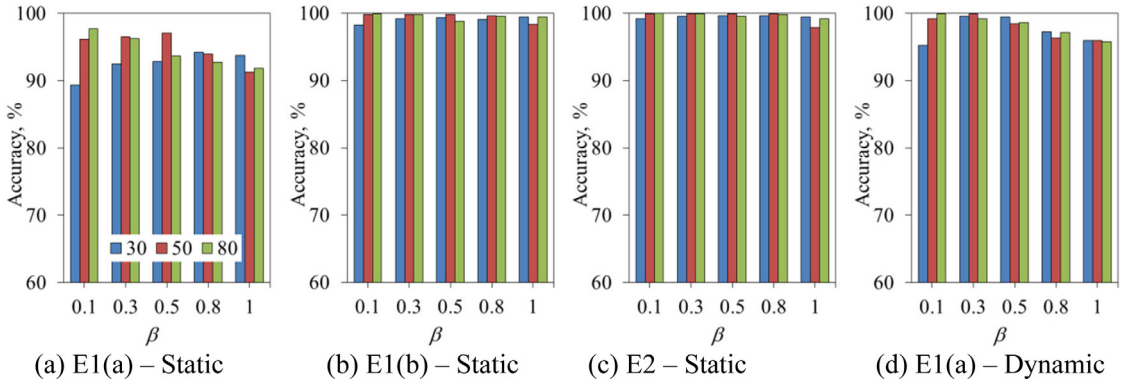


Fig. 8. Comparison of recognition accuracy with SVM classifier — Augmented data.

using SVM. It is to be noted that, the present study did not use any aggregation or normalization of the results because WEKA software deployed to access the performance using probabilistic classifiers (except for SVM and CNN). Thus, it may not be appropriate to normalize the output of different methods, as the objective is to compare the variation in the output of different classifiers with plausible parameters, for the same input.

5.1.2. Cumulant order

The measured accuracies in the home — E1(a), E1(b), and office — E2 with 30, 50, and 80 optimal feature subset shown in Fig. 7, for all cumulant orders. The static gesture recognition performance observed to improve with an increasing number of optimal feature selection, across all environments irrespective of the cumulant order. However, in environments that acquired static sign gesture, second-order cumulants attained better recognition accuracy than third and fourth-order, for all selected features.

In a similar fashion, second-order cumulants achieved exceptional recognition performance with limited optimal features, i.e., 30 for dynamic gestures. In comparison with second and third-order, the fourth-order cumulant features achieved the least performance in all environments. Third-order cumulant features performed well with 50 and 80 optimal feature subsets, yet not consistent as second-order cumulant features, in different scenarios. Hence, DF-WiSLR prefers second-order cumulant features over third and fourth-order for achieving better gesture recognition accuracy.

5.1.3. Data augmentation and optimal feature selection

DF-WiSLR extends the performance analysis on augmented data using second-order cumulant features having better results on original datasets. Fig. 8 compares the performance of second-order cumulant features of augmented data, with 30, 50, and 80, optimal feature subset for different β values using SVM. In home environments E1(a), E1(b), and office environment E2, SVM achieved 97.1%, 99.9% and 99.9% recognition accuracy respectively at $\beta = 0.5$ for static gestures with 50 optimal features. For dynamic gestures, the corresponding recognition accuracy values observed to be 98.5% in the home environment E1(a). The accuracy values indicate that data augmentation enhances the recognition performance compared to that of the original datasets. Results from Figs. 5 and 8 confirm that SVM achieves overall recognition accuracy greater than 92% on all four datasets with 50 optimal features.

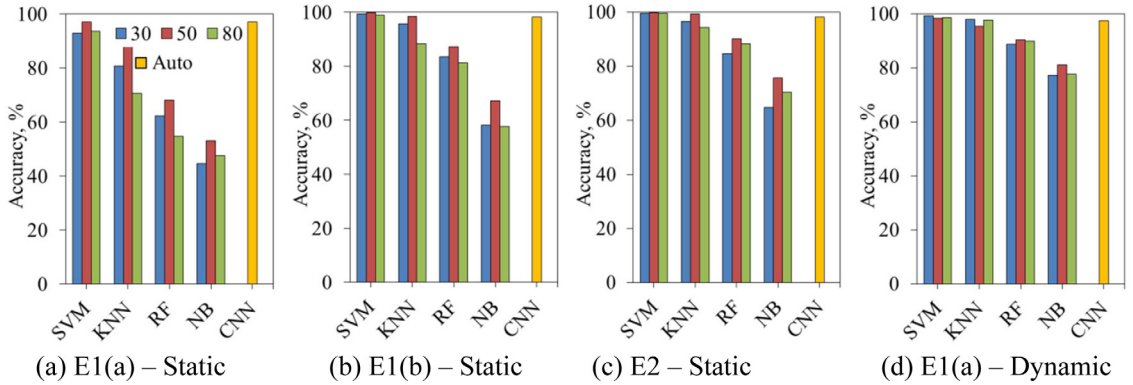


Fig. 9. Comparison of recognition accuracy with different classifiers — Augmented data.

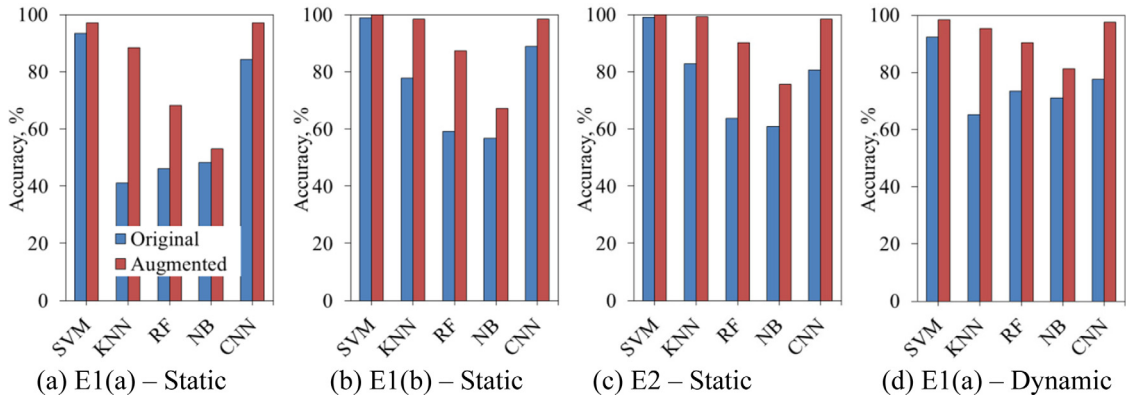


Fig. 10. Comparison of recognition accuracy of classifiers — Original vs. Augmented.

Fig. 9 compares the recognition performance on augmented data, using different machine learning classifiers and CNN. KNN report best accuracy values of 88.3%, 98.4%, 99.3% in E1(a), E1(b), E2 for static gestures with 50 optimal features and 98.0% in E1(a) for dynamic gestures with 30 optimal features. RF and NB achieve accuracy values less than 90% in all four environments, even with 80 optimal features. Compared to other classifiers, SVM attained higher recognition accuracies with augmented data, as observed with original data. Even with higher training size, RF and NB show a marginal improvement in accuracy with augmentation, however not very impressive. Consistency in performance as in SVM not observed with KNN, RF, and NB. These observations indicate that the augmentation technique improved the recognition performance without altering the signal characteristics of the originally acquired data. Besides, it facilitates expanding the training data size that alternates the need for huge experimental efforts in collecting physical data.

Fig. 10 compares the gesture recognition accuracy of various classifiers tested in the present study, with original and augmented data, obtained at home and office environments. The accuracy values reported are of second-order cumulant features with 50 selected optimal features and $\beta = 0.5$. Comprehensively, all learning classifiers show an improvement in the recognition performance with increasing training data. With original data, SVM outperformed all other classifiers with a good margin, in all the environments, for both static and dynamic gestures. KNN show performance closer to SVM, in home environment E1(b) and office environment E2 with static gestures and in home environment E1(a) with dynamic gesture. Compared to SVM and KNN, the reported accuracies of RF and NB are significantly less on both original and augmented data. Therefore, DF-WiSLR manifests SVM and second-order cumulants with 50 optimal features as a plausible choice for attaining better recognition performance.

5.1.4. Gesture orientation and environment

The accuracy values of static and dynamic gestures compared in home environment E1(a) on original data, referring to Fig. 5, analyze the impact of gesture orientation on recognition performance. The environmental factors and the granularity level of sign gesture profoundly influence the performance. The presence of interference or objects in the sensing environment provokes multiple reflections of the acquired signal that degrade the recognition accuracy. The impact of impediments in home environment E1(a) affects the static gesture recognition performance reporting 93.4%

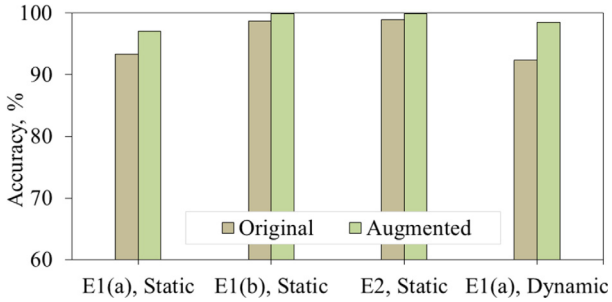


Fig. 11. Influence of gesture orientation and environment.

Table 3
Training and testing time consumption per Instance (ms).

Environment/Gesture	SVM				CNN			
	Original		Augmented		Original		Augmented	
	Train	Test	Train	Test	Train	Test	Train	Test
E1(a)/Static	0.35	0.14	0.59	0.40	63.66	65.47	93.38	95.35
E1(b)/Static	0.32	0.10	0.53	0.29	60.47	62.22	102.67	104.36
E2/Static	0.25	0.11	0.60	0.38	61.80	63.48	89.58	91.25
E1(b)/Dynamic	0.13	0.07	0.21	0.16	72.77	74.68	81.09	82.83
Average time	**0.26**	**0.10**	**0.48**	**0.31**	64.67	66.46	91.68	93.45

accuracy with 50 optimal features at $\beta = 0.5$. Dynamic gestures being relatively coarse-grained, in the same environment E1(a), reported a higher accuracy of 97.7% with 50 optimal features at $\beta = 0.3$.

With fewer impediments in E1(b) and E2, the reported recognition accuracies of static gestures on original data, are observed to be consistent at all values of β, and observed to be better than that of E1(a). Despite the fact that room dimensions and the distance between transmitter–receiver pair in E2 measuring larger than E1, the former report better recognition accuracy than the latter. The preceding analysis infers that the existence of impediments in the environment impacts the recognition accuracy to a considerable degree. Similar trends observed with augmented data on gesture recognition performance.

Fig. 11 compares the recognition accuracy achieved on original and augmented data, for static and dynamic sign gestures acquired in all three environments. The plotted accuracy values are estimated using SVM with second-order cumulant and 50 optimal features for $\beta = 0.5$. This observation substantiates the earlier discussions that the recognition accuracy improves with lesser impediments in the sensing environment. Thus, the environment E1 (b) and E2 with lesser impediments report better recognition than E1 (a) having more impediments. Under all scenarios, the recognition accuracy considerably improves with augmented data.

Nevertheless, dynamic gesture recognition involves compounding words, whereas static gesture recognition deals only with single word signs. The preceding discussion on the experimental results implies that, with SVM, DF-WiSLR achieves significant recognition performance in all scenarios.

5.2. Training and testing time

Table 3 compares the training and testing time consumption for analyzing the performance of the machine learning classifier with a deep learning classifier. It presents the computational time consumption per instance (in milliseconds) of SVM and CNN on original and augmented data. The training and testing time consumption of SVM for static gestures are observed to be higher than that of dynamic gestures, both on original and augmented data. For CNN, the training and testing time consumption of static gestures are relatively lesser than that of dynamic gestures on original data. No unique trend observed for augmented data.

The average time spent by SVM for training and testing on original data is 0.26 ms and 0.10 ms per gesture instance, respectively. For augmented data, it takes 0.48 ms and 0.31 ms for training and testing, respectively. CNN takes an average time of 64.67 ms for training and 66.46 ms for testing on original data, and the corresponding values reported for augmented data are 91.68 ms and 93.45 ms. The training and testing times are measured using a laptop with Intel i5-6200U CPU processor, operating at 2.30 GHz with 12 GB RAM.

SVM spend 13% more time during testing than training, averaged over all environments. Compared to the time taken by original data, on an average, the augmented data takes about 5.5 times more time for training and about 8.6 times more time for testing with SVM. The average time taken by CNN observed to be similar for training and testing on both original and augmented datasets. The reported values indicate that SVM consumed minimal time on measuring the accuracy values compared to CNN. The comparison, as mentioned above, clearly indicates that SVM excels in the recognition performance with reduced computational efforts.

Table 4
Comparison of present work with related work.

Reference	Number of gestures	Accuracy
[60]	12 hand and finger gestures, 50 instances per instance for each environment = 2×50×12 = 1200	CNN-SVM — 97% Fine-tuned-CNN — 98%
[42]	CARM dataset [29] - 9 activities and CERTH/ITI dataset (5 activities with 50 samples per activity)	CARM-MFCC-CNN 95% precision
CSI-HC [43]	1000 training samples	85.4%
FingerDraw [30]	Digits, alphabets, and symbols	93%
DeepMV [44]	9 activities, 4 rounds for 51 s	Homogeneous Wi-Fi data (Wi-Fi only) —83.7% Heterogeneous Wi-Fi data (Wi-Fi + Acoustic) – 87.9%
WiGer [11]	7 finger/hand gestures	Five scenarios 97.28%, 91.8%, 95.5%, 94.4% and 91%
WiCatch [12]	9 finger/hand gestures	Trajectory recognition efficiency — 95%
Wi-Finger [13]	8 finger gestures	95%
WiKey [15]	37 key strokes	Minimum 77.4%, maximum 93.4%
Mudra [14]	9 finger gestures	96%
SignFi [37]	276 gestures of American Sign Language (ASL)	98.01% - lab 276, 98.91% - home 276, 94.81% - lab and home 276, and 86.66% - lab 150
HOS-Re [36]	SignFi dataset — 276 ASL gestures	97.84% - lab 276, 98.26% - home 276, 96.34% - lab and home 276, and 96.23% - lab 150
DF-WiSLR (Present work)	49 ISL gestures (Static sign + dynamic sign)	SVM — Original; Static-93.4%, 98.8%, 98.9%; Dynamic-92.3%; SVM — Augmented; Static-97.1%, 99.9%, 99.9%; Dynamic-98.5%

5.3. Comparison of DF-WiSLR with existing systems

Table 4 shows related work that utilizes Wi-Fi CSI for recognizing finger/hand gestures and specifies the number of gestures with reported accuracy in comparison to the present work. When compared to other reported works, DF-WiSLR handles more number of gestures next to SignFi [37], to the best of our knowledge. Furthermore, the present work achieved better recognition performance even with limited physical data collection and reduced experimentation efforts. Also, the data augmentation technique adopted in the present work increases the size of the dataset, suitable and solves the data adequacy need of classifiers like CNN.

6. Conclusions

This paper proposed a device-free WiFi-CSI based sign language recognition, DF-WiSLR, utilizing Wi-Fi signals for sign gesture recognition. DF-WiSLR performs the recognition task by acquiring CSI of Wi-Fi signals and adopt machine learning classifiers such as SVM, KNN, RF, NB, and a deep learning classifier — an 8-layer CNN, as classification algorithms. The distinctive cross-cumulant features of order two, three, and four are extracted from the input data and applied MIFS algorithm for optimal feature selection. The optimal feature subset will serve as input to the machine learning classifiers. The pre-processed input data are directly fed as input to the 8-layer CNN, as it can extract and select features automatically.

The observations on the results obtained indicate that gesture orientation and environmental impediments highly influence the recognition performance. DF-WiSLR with SVM reported robust performance in recognizing static and dynamic gesture on both original and augmented data. With fewer impediments in the sensing environment, the reported recognition accuracies improve. Besides, the distance between the transmitter–receiver pair and the position of the volunteer in different environment show a relatively lesser impact on recognition performance. Data augmentation triples the training data size, and better results were reported without altering the signal characteristics of the originally acquired data. SVM measured better recognition accuracy with second-order cumulant features than the third and fourth-order features. KNN and RF achieved the best performance in a specific combination of parameters. NB shows the least performance compared to all other classifiers in all environments. The performance of deep learning classifier CNN also improved with augmentation. However, the overall accuracy of all classifiers falls back than the corresponding value reported by SVM, under all scenarios with reduced computational efforts.

Earlier work on sign gesture recognition using WiFi-CSI performed the recognition task of static gestures consisting of only single words. Whereas, DF-WiSLR attained better recognition accuracies for dynamic gesture comprising of compounding word signs. Also, reported exceptional performance on fine-grained static signs consisting of alphabets, numbers, and words. The present work performs the recognition task in offline manner. Online recognition and translation of sign poses into text and audio format is left for future consideration. Besides, further improvement on recognition accuracy of compounding dynamic signs involving complex sentences will be extended as future work. Future scope also includes adopting transfer learning, in which the learning classifier is trained with instances of one environment and tested for instances acquired across different environments.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This work was supported by Taylor's University, Malaysia through its TAYLOR'S PhD SCHOLARSHIP Programme.

Funding

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

References

[1] N. Zengeler, T. Kopinski, U. Handmann, Hand gesture recognition in automotive human–machine interaction using depth cameras, Sensors 19 (2019) 1–27.
[2] T. Yanay, E. Shmueli, Air-writing recognition using smart-bands, Pervasive Mob. Comput. (2020) 101183.
[3] L. Xiao, K. Wu, X. Tian, J. Luo, Activity-specific caloric expenditure estimation from kinetic energy harvesting in wearable devices, Pervasive Mob. Comput. (2020) 101185.
[4] R. San-Segundo, J.D. Echeverry-Correa, C. Salamea-Palacios, S.L. Lutfi, J.M. Pardo, I-vector analysis for gait-based person identification using smartphone inertial signals, Pervasive Mob. Comput. 38 (2017) 140–153.
[5] T. Sztyler, H. Stuckenschmidt, W. Petrich, Position-aware activity recognition with wearable devices, Pervasive Mob. Comput. 38 (2017) 281–295.
[6] H.F.T. Ahmed, H. Ahmad, C. Aravind, Device free human gesture recognition using Wi-Fi CSI: A survey, Eng. Appl. Artif. Intell. 87 (2020) 1–19.
[7] R. Zhou, X. Lu, P. Zhao, J. Chen, Device-free presence detection and localization with SVM and CSI fingerprinting, IEEE Sens. J. 17 (2017) 7990–7999.
[8] L. Li, C. Guo, Y. Liu, L. Zhang, X. Qi, Y. Ren, B. Liu, F. Chen, Accurate device-free tracking using inexpensive RFIDs, Sensors 18 (2018) 2816.
[9] S. Sigg, M. Scholz, S. Shi, Y. Ji, M. Beigl, RF-sensing of activities from non-cooperative subjects in device-free recognition systems using ambient and local signals, IEEE Trans. Mob. Comput. 13 (2013) 907–920.
[10] T.F. Sanam, H. Godrich, Comute: a convolutional neural network based device free multiple target localization using csi, 2020, pp. 1–18, arXiv preprint arXiv:2003.05734.
[11] M.A.A. Al-qaness, F. Li, WiGeR: WiFi-based gesture recognition system, ISPRS Int. J. Geo-Inf. 5 (2016) 1–17.
[12] Z. Tian, J. Wang, X. Yang, M. Zhou, WiCatch: A Wi-Fi based hand gesture recognition system, IEEE Access 6 (2018) 16911–16923.
[13] S. Tan, J. Yang, WiFinger: leveraging commodity WiFi for fine-grained finger gesture recognition, in: Proceedings of the 17th ACM International Symposium on Mobile Ad Hoc Networking and Computing, ACM, Paderborn, Germany, 2016, pp. 201–210.
[14] O. Zhang, K. Srinivasan, Mudra: User-friendly fine-grained gesture recognition using WiFi signals, in: Proceedings of the 12th International on Conference on Emerging Networking Experiments and Technologies, ACM, Irvine, California, USA, 2016, pp. 83–96.
[15] K. Ali, A.X. Liu, W. Wang, 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.
[16] Q. Pu, S. Gupta, S. Gollakota, S. Patel, Whole-home gesture recognition using wireless signals, in: Proceedings of the 19th Annual International Conference on Mobile Computing & Networking, ACM, Miami, Florida, USA, 2013, pp. 27–38.
[17] M. De Sanctis, E. Cianca, S. Di Domenico, D. Provenziani, G. Bianchi, M. Ruggieri, Wibecam: Device free human activity recognition through wifi beacon-enabled camera, in: Proceedings of the 2nd Workshop on Workshop on Physical Analytics, ACM, 2015, pp. 7–12.
[18] N. Damodaran, E. Haruni, M. Kokhkharova, J. Schäfer, Device free human activity and fall recognition using WiFi channel state information (CSI), CCF Trans. Pervasive Comput. Interact. 2 (2020) 1–17.
[19] A. Jayatilaka, D.C. Ranasinghe, Real-time fluid intake gesture recognition based on batteryless UHF RFID technology, Pervasive Mob. Comput. 34 (2017) 146–156.
[20] J. Zhao, L. Liu, Z. Wei, C. Zhang, W. Wang, Y. Fan, R-DEHM: CSI-based robust duration estimation of human motion with WiFi, Sensors 19 (2019) 1–17.
[21] F. Hong, X. Wang, Y. Yang, Y. Zong, Y. Zhang, Z. Guo, WFID: Passive device-free human identification using WiFi signal, in: Proceedings of the 13th International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services, ACM, Hiroshima, Japan, 2016, pp. 47–56.
[22] H. Zhu, F. Xiao, L. Sun, R. Wang, P. Yang, R-TTWD: Robust device-free through-the-wall detection of moving human with WiFi, IEEE J. Sel. Areas Commun. 35 (2017) 1090–1103.
[23] R. Battiti, Using mutual information for selecting features in supervised neural net learning, IEEE Trans. Neural Netw. 5 (1994) 537–550.
[24] R.S. Sinha, S.-M. Lee, M. Rim, S.-H. Hwang, Data augmentation schemes for deep learning in an indoor positioning application, Electronics 8 (2019) 1–19.
[25] D. Wu, D. Zhang, C. Xu, Y. Wang, H. Wang, WiDir: walking direction estimation using wireless signals, in: Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing, ACM, Heidelberg, Germany, 2016, pp. 351–362.

[26] F. Zhang, D. Zhang, J. Xiong, H. Wang, K. Niu, B. Jin, Y. Wang, From fresnel diffraction model to fine-grained human respiration sensing with commodity wi-fi devices, Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. 2 (2018).
[27] L. Sun, S. Sen, D. Koutsonikolas, K.-H. Kim, Widraw: Enabling hands-free drawing in the air on commodity wifi devices, in: Proceedings of the 21st Annual International Conference on Mobile Computing and Networking, ACM, Paris, France, 2015, pp. 77–89.
[28] X. Li, D. Zhang, Q. Lv, J. Xiong, S. Li, Y. Zhang, H. Mei, IndoTrack: Device-free indoor human tracking with commodity Wi-Fi, Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. 1 (2017) 72:1-72:22.
[29] W. Wang, A.X. Liu, M. Shahzad, K. Ling, S. Lu, Device-free human activity recognition using commercial WiFi devices, IEEE J. Sel. Areas Commun. 35 (2017) 1118–1131.
[30] D. Wu, R. Gao, Y. Zeng, J. Liu, L. Wang, T. Gu, D. Zhang, Fingerdraw: Sub-wavelength level finger motion tracking with WiFi signals, Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. 4 (2020) 1–27.
[31] B. Fang, N.D. Lane, M. Zhang, A. Boran, F. Kawsar, BodyScan: Enabling radio-based sensing on wearable devices for contactless activity and vital sign monitoring, in: Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services, 2016, pp. 97-110.
[32] S. Palipana, D. Rojas, P. Agrawal, D. Pesch, FallDeFi: Ubiquitous fall detection using commodity Wi-Fi devices, Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. 1 (2018).
[33] W. Jia, H. Peng, N. Ruan, Z. Tang, W. Zhao, Wifind: Driver fatigue detection with fine-grained wi-fi signal features, IEEE Trans. Big Data (2018) 1–14.
[34] S.W. Shah, S.S. Kanhere, Smart user identification using cardiopulmonary activity, Pervasive Mob. Comput. 58 (2019) 101024.
[35] G. Wang, Y. Zou, Z. Zhou, K. Wu, L.M. Ni, We can hear you with wi-fi!, IEEE Trans. Mob. Comput. 15 (2016) 2907–2920.
[36] H. Farhana Thariq Ahmed, H. Ahmad, S.K. Phang, C.A. Vaithilingam, H. Harkat, K. Narasingamurthi, Higher order feature extraction and selection for robust human gesture recognition using CSI of COTS Wi-Fi devices, Sensors 19 (2019) 1–23.
[37] Y. Ma, G. Zhou, S. Wang, H. Zhao, W. Jung, Signfi: sign language recognition using wifi, Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. 2 (2018) 1–21.
[38] Y. Xu, M. Chen, W. Yang, S. Chen, L. Huang, Attention-based walking gait and direction recognition in Wi-Fi networks, 2018, arXiv preprint arXiv:1811.07162.
[39] Z. Wang, Z. Gu, J. Yin, Z. Chen, Y. Xu, Syncope detection in toilet environments using Wi-Fi channel state information, in: Proceedings of the 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, ACM, Singapore, 2018, pp. 287–290.
[40] F. Wang, W. Gong, J. Liu, K. Wu, Channel selective activity recognition with WiFi: A deep learning approach exploring wideband information, IEEE Trans. Netw. Sci. Eng. (2018).
[41] M. Gil-Martín, R. San-Segundo, F. Fernández-Martínez, J. Ferreiros-López, Improving physical activity recognition using a new deep learning architecture and post-processing techniques, Eng. Appl. Artif. Intell. 92 (2020) 103679.
[42] T. Tegou, A. Papadopoulos, I. Kalamaras, K. Votis, D. Tzovaras, Using auditory features for WiFi channel state information activity recognition, SN Comput. Sci. 1 (2020) 1–11.
[43] Z. Hao, Y. Duan, X. Dang, T. Zhang, CSI-HC: A WiFi-based indoor complex human motion recognition method, Mob. Inf. Syst. 2020 (2020) 1–20.
[44] H. Xue, W. Jiang, C. Miao, F. Ma, S. Wang, Y. Yuan, S. Yao, A. Zhang, L. Su, DeepMV: Multi-view deep learning for device-free human activity recognition, Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. 4 (2020) 1–26.
[45] Y. Xie, Z. Li, M. Li, Precise power delay profiling with commodity Wi-Fi, IEEE Trans. Mob. Comput. 18 (2018) 1342–1355.
[46] D. Halperin, W. Hu, A. Sheth, D. Wetherall, Linux 802.11 n CSI tool ACM SIGCOMM, Comput. Commun. Rev. 41 (2010) 53.
[47] Y. Chapre, A. Ignjatovic, A. Seneviratne, S. Jha, CSI-MIMO: An efficient wi-fi fingerprinting using channel state information with MIMO, Pervasive Mob. Comput. 23 (2015) 89–103.
[48] Z. Zhang, S. Ishida, S. Tagashira, A. Fukuda, Danger-pose detection system using commodity Wi-Fi for bathroom monitoring, Sensors 19 (2019) 884.
[49] D. Tse, P. Viswanath, Fundamentals of Wireless Communication, Cambridge university press, 2005.
[50] A. Swami, J. Mendel, C. Nikias, Higher-Order Spectral Analysis Toolbox User's Guide, The Math Works Inc, 1998.
[51] A. Swami, J.M. Mendel, C.L. Nikias, Higher order spectral analysis toolbox, for use with MATLAB, The MathWorks, 1998.
[52] C.-C. Chang, C.-J. Lin, LIBSVM: A library for support vector machines, ACM Trans. Intell. Syst. Technol. 2 (2011) 1–27.
[53] A. Wojna, L. Kowalski, RSESLIB Programmer's Guide, Faculty of Mathematics, Informatics and Mechanics, University of Warsaw, 2017.
[54] L. Breiman, Random forests, Mach. Learn. 45 (2001) 5–32.
[55] G.H. John, P. Langley, Estimating continuous distributions in Bayesian classifiers, 2013, arXiv preprint arXiv:1302.4964.
[56] National Institute of Open Schooling, Ministry of HRD, Govt. of India, 2010 [Accessed on 2019 Nov 24].
[57] S. Di Domenico, M. De Sanctis, E. Cianca, F. Giuliano, G. Bianchi, Exploring training options for RF sensing using CSI, IEEE Commun. Mag. 56 (2018) 116–123.
[58] L. Zhang, Q. Gao, X. Ma, J. Wang, T. Yang, H. Wang, DeFi: Robust training-free device-free wireless localization with WiFi, IEEE Trans. Veh. Technol. 67 (2018) 8822–8831.
[59] Weka : the workbench for machine learning, 2019, [Accessed on 08 November 2019].
[60] Q. Bu, G. Yang, X. Ming, T. Zhang, J. Feng, J. Zhang, Deep transfer learning for gesture recognition with WiFi signals, Pers. Ubiquitous Comput. (2020) 1–12.