Device-Free User Authentication, Activity Classification and Tracking using Passive Wi-Fi Sensing: A Deep Learning Based Approach

Vinoj Jayasundara, *Member, IEEE,* Hirunima Jayasekara, *Member, IEEE,* Tharaka Samarasinghe, *Member, IEEE,* Kasun T. Hemachandra *Member, IEEE*

***Abstract*—Growing concerns over privacy invasion due to video camera based monitoring systems have made way to non-invasive Wi-Fi signal sensing based alternatives. This paper introduces a novel end-to-end deep learning framework that utilizes the changes in orthogonal frequency division multiplexing (OFDM) sub-carrier amplitude information to simultaneously predict the identity, activity and the trajectory of a user and create a user profile that is of similar utility to a one made through a video camera based approach. The novelty of the proposed solution is that the system is fully autonomous and requires zero user intervention unlike systems that require user originated initialization, or a user held transmitting device to facilitate the prediction. Experimental results demonstrate over 95% accuracy for user identification and activity recognition, while the user localization results exhibit a $\pm$12cm error, which is a significant improvement over the existing user tracking methods that utilize passive Wi-Fi signals.**

***Index Terms*—Activity Classification, Bidirectional Gated Recurrent Unit (Bi-GRU), Tracking, Long Short-Term Memory (LSTM), User Authentication, Wi-Fi.**

I. Introduction

USER identification, behaviour analysis, localization and user activity recognition have become crucial tasks due to the increasing popularity of fully automated facilities such as cashierless stores, and many applications associated with smart cities. Current state-of-the-art techniques for passive user authentication [1], re-identification [2], activity classification [3] and tracking [4], [5] are primarily based on video feed analysis. Growing concerns on privacy invasion related to video surveillance have made way to many non-invasive alternatives. These alternatives utilize passive visible light positioning and artificial potential fields [6], electric potential changes in human bodies [7], body frequency absorption signatures [8], acoustics [9], and kinect sensing [10]. However, a better alternative is ambient Wi-Fi signals, due to the wide availability and easy accessibility.

V. Jayasundara, H. Jayasekara and K.T. Hemachandra are with the Department of Electronic and Telecommunication Engineering, University of Moratuwa, Sri Lanka (e-mail: {vinojjayasundara, nhirunima}@gmail.com, kasunh@uom.lk).

T. Samarasinghe is with the Department of Electronic and Telecommunication Engineering, University of Moratuwa, Sri Lanka, and the Department of Electronic and Electrical Engineering, University of Melbourne, Australia (e-mail: tharakas@uom.lk).

This work is supported by the Senate Research Council, University of Moratuwa, Sri Lanka, under grant SRC/LT/2018/2.

In this paper, we introduce a fully autonomous, non invasive, Wi-Fi based alternative, which can carry out user identification, activity recognition and tracking, simultaneously, similar to a video camera based approach.

A. Related Works

*1) **User Authentication**:* A majority of the wireless aided user authentication systems in the literature require the user to carry or wear a device to facilitate the authentication process [11]. A device-free method which eliminates the necessity of a user to carry a wireless transmitting device for active user sensing is deemed more suitable practically. Several device-free authentication mechanisms are popular in the literature, including video for appearance-based user authentication [1], gait pattern based user authentication, and hybrid (appearance-based and gait pattern) models [12]. To this end, WiWho [13] and Wi-Fi-ID [14] utilize conventional signal processing techniques to create a gait profile for each user, which is subsequently used for identification. Research focus, however, has recently shifted towards learning based techniques [15], [16], but being able to handle only registered users is considered a major limitation in such systems, *e.g.*, NeuralWave [15]. WiAU [16] focuses on alleviating this issue by introducing a system that is robust to unauthorized users via training their model with both authorized and unauthorized (unregistered) user data. However, training a model for limitless potential unauthorized users is practically infeasible. Our system focuses on providing a robust device-free solution for this limitation.

*2) **Activity Recognition**:* Wireless-aided activity recognition is a well studied area in the literature [17]–[21]. It has been already established that deep learning based techniques do outperform the conventional signal processing based techniques [17]–[19] with regards to activity recognition (see [20] and references therein). However, the existing deep learning based systems face difficulties in deployment due to their omission of the constantly re-occurring periods without any activities in their models. Thus, these systems require the user to invoke the system by conducting a predefined action, or a sequence of actions. This limitation is addressed in our work to introduce a fully autonomous system.

*3) **User Tracking**:* Wi-Fi based localization systems that utilize deep learning approaches are well studied in the literature [22]–[24]. However, most of these systems have the compulsory requirement of user to carry a Wi-Fi capable

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSEN.2020.2987386, IEEE Sensors Journal

device for active user sensing. Hence, existing device-free Wi-Fi based approaches have attracted considerable research interest [25]–[27]. Most existing deep learning based device free localization systems can predict the position of the user out of a set of pre-determined positions [9], but lack the ability to continuously output user co-ordinates, which is mandatory for continuous tracking. This is another gap in the literature that is bridged in our paper.

B. Contributions of the Paper

We consider a distributed single-input-multiple-output (SIMO) system that consists of a Wi-Fi transmitter, and a multitude of fully synchronized multi-antenna Wi-Fi receivers, placed in the sensing area. The samples of the received signals are fed forward to a data concentrator, where channel state information (CSI) related to all OFDM sub carriers are extracted and pre-processed, before feeding them into the deep neural networks. The key features of the proposed system are as follows:

- The system is self-sustaining, device-free, non-invasive, and does not require any user interaction at commencement or otherwise, and can be deployed with existing infrastructure.
- The system consists of a novel black-box technique that produces a standardized annotated vector for authentication, activity recognition and tracking with pre-processed CSI streams as the input for any event.
- With the aid of the three annotations, the system is able to fully characterize an event, similar to a camera video.

State-of-the-art deep learning techniques are the key enabler of the proposed system. With the advanced learning capabilities of such techniques, complex mathematical modelling required for the process of interest can be conveniently learned. To the best of our knowledge, this is the first attempt at proposing an end-to-end system that predicts all these three in a multi-task manner. Then, to address the limitations in already available systems; firstly, for authentication, we propose a novel prediction confidence-based thresholding technique to filter out unauthorized users of the system, without the necessity of any training data from them. Secondly, we introduce a *no activity* (NoAc) class to characterize the periods without any activities, which we utilize to make the system fully autonomous. Finally, we propose a novel deep learning based approach for device-free passive continuous user tracking, which enables the system to completely characterize an event similar to a camera video, but in a non-invasive manner. The performance of the proposed system is evaluated through experiments, and the system achieves accurate results even with only two single-antenna Wi-Fi receivers.

The rest of the paper is organized as follows: in Sections II, III and IV, we present the system overview, methodology on data processing, and the proposed deep neural networks, respectively. Subsequently, we discuss our experimental setup in Section V, followed by results and discussion in Section VI. Section VII concludes the paper.

II. System Overview

Consider a distributed SIMO system that consists of a single-antenna Wi-Fi transmitter, and M Wi-Fi receivers having N antennas each. The transmitter and the receivers are placed in the sensing area, and an example scenario is illustrated in Fig. 1. The receivers are fully synchronized, with a sampling frequency of f_s, and connected to a data concentrator for centralized processing. The received signal at the n-th antenna of the m-th receiver, where $n \in \{1, \ldots, N\}$ and $m \in \{1, \ldots, M\}$, is given by

$$y_{m,n}(t) = \sum_{i=1}^{S} h_{m,n,i,t}\ x_i \cos(2\pi f_i t + \theta_{m,n,i,t}) + \eta(t), \quad (1)$$

where S is the number of subcarriers in the transmitted OFDM signal. Moreover, for the i-th subcarrier, $h_{m,n,i,t}$ and $\theta_{m,n,i,t}$ denote the amplitude and the phase values of the random channel between the transmitter and the n-th antenna of the m-th receiver, respectively, and $\eta(t)$ represents the random noise in the received signal. We assume that at a given time t, the data concentrator has access to all received signals (samples), which can be achieved through a feedforward mechanism.

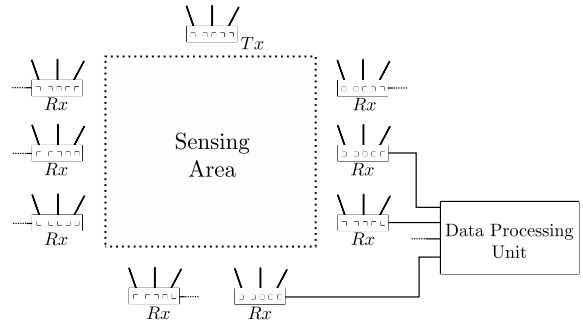

Fig. 1. Generic Wi-Fi transmitter and receiver configuration for the proposed system, where Rx and Tx refer to receivers and transmitters, respectively.

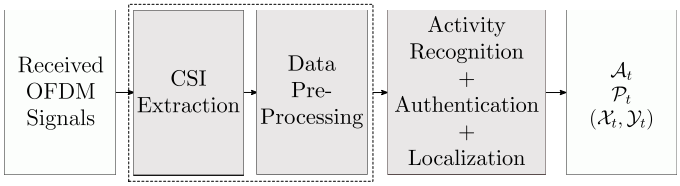

Fig. 2. The system architecture, where at time t, $\mathcal{A}_t$ is the predicted activity, $\mathcal{P}_t$ is the predicted person and $(\mathcal{X}_t, \mathcal{Y}_t)$ are the Cartesian coordinates of the location.

Fig. 2 presents an overview of the system. The first stage, which is implemented at the data concentrator, focuses on extracting CSI from the received signals. CSI is considered to be more stable and robust to complex environmental effects compared to Received Signal Strength Indicator (RSSI) values [28], and since the CSI related to each subcarrier of the OFDM signal can be extracted, the system will have more information for effective learning. Moreover, the effect of activities in a sensing area on the CSI has been recently studied in [29]. This stage is explained in Section III-A. The second stage is pre-processing the extracted CSI information, which is elaborated in Section III-B. The pre-processed data is fed into the deep neural networks, which include the activity recognizer, the authenticator and the tracker. This is considered to be the third stage in the system architecture, and it is discussed in Section IV. The deep neural networks output three annotations per data segment of the form $[\mathcal{A}_t, \mathcal{P}_t, (\mathcal{X}_t, \mathcal{Y}_t)]$, where at time t, $\mathcal{A}_t$ is the predicted activity, $\mathcal{P}_t$ is the predicted

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSEN.2020.2987386, IEEE Sensors Journal

person performing the activity and $(\mathcal{X}_t, \mathcal{Y}_t)$ are the Cartesian coordinates of the person's location, relative to a pre-defined coordinate frame. With the aid of the three annotations, we can sufficiently characterize the proceedings within a T second window, similar to a camera video.

III. Data Processing

A. Extracting Channel State Information (CSI)

Current OFDM implementations (including 802.11a, g, n and ac) use the information available in pilot sub-carriers of the OFDM signal to estimate the channel behaviour and the multi-path disturbances caused by the environment. Channel estimation in OFDM systems is well-studied in the literature [30]. The data concentrator performs the channel estimation in the proposed system. We note that the channel estimation can alternatively be performed at the receivers, and the estimated CSI can then be forwarded to the data concentrator for further processing. However, most commercial Wi-Fi devices do not provide access to the estimated CSI data, and hence, we propose a more general architecture for wider adaptability.

Both amplitude and phase values are finer-grained descriptors of the wireless channel [31]. However, the phase values are affected by several sources of error, including the carrier frequency offset (CFO) and the sampling frequency offset (SFO) [20]. Although these errors can be eliminated using a calibration technique termed data sanitization in [32], we avoid phase values in the learning process to reduce the computational and implementational complexity. Thus, at time t, for each $m \in \{1, \ldots, M\}$ and $n \in \{1, \ldots, N\}$, the data concentrator estimates $\boldsymbol{h}_{m,n,t} = [h_{m,n,1,t} \ \ldots \ h_{m,n,S,t}]^\top$, which is an S-by-1 vector that consists of amplitude values of the channel between the transmitter and the n-th antenna of the m-th receiver. The concentrator then forwards the estimated amplitude values to the next stage for further processing.

B. CSI Preprocessing

We start the CSI preprocessing by a sparsity reduction operation that aids the learning of the network. The coarse frequency offset correction done in channel estimation usually leads to some elements in $\boldsymbol{h}_{m,n,t}$ to have negligibly small values (quantitatively, amplitude < 0.05), as evident from Fig. 8, irrespective of the setup, trial, activity or the person. Furthermore, this observation holds irrespective of the application, distance between the Wi-Fi devices, and the position of the person. These elements do not provide any useful information and in fact hinders learning. Hence, they are removed from the estimated channel amplitude vectors. We denote the respective sparsity reduced amplitude vector of $\boldsymbol{h}_{m,n,t}$ by $\hat{\boldsymbol{h}}_{m,n,t}$. As an example, if $\boldsymbol{h}_{1,1,t} = [0.1 \ 0.2 \ 0.0 \ 0.4 \ 0.0003]^\top$, we have $\hat{\boldsymbol{h}}_{1,1,t} = [0.1 \ 0.2 \ 0.4]^\top$. Moreover, $\hat{\boldsymbol{h}}_{m,n,t}$ is a $\bar{S}$-by-1 vector, where $\bar{S} \leq S$. Note that for a given m and n, the dimensionality of $\hat{\boldsymbol{h}}_{m,n,t}$ is fixed at $\bar{S}$ for all t values since the sparsity is caused by the coarse frequency offset correction. Next, for each $m \in \{1, \ldots, M\}$ and $n \in \{1, \ldots, N\}$, we concatenate the sparsity reduced amplitude vectors along the temporal axis to produce the $\bar{S}$-by-T_{jk} matrix $\hat{\boldsymbol{C}}_{m,n} = [\hat{\boldsymbol{h}}_{m,n,1} \ \cdots \ \hat{\boldsymbol{h}}_{m,n,T_{jk}}]$. Here, T_{jk} denotes the duration of the j-th trial of the k-th activity, signifying the variability in the duration of activities, as well as the variability in the duration of different trials of the same activity.

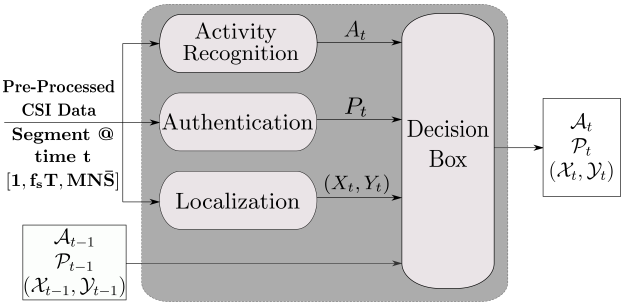

Fig. 3. Generating annotations using deep neural networks. During the inference phase, the decision box aids to distinguish between static and dynamic activities, to predict $[\mathcal{A}_t, \mathcal{P}_t, (\mathcal{X}_t, \mathcal{Y}_t)]$ correctly, where at a given time t, $\mathcal{A}_t$ is the predicted activity, $\mathcal{P}_t$ is the predicted person and $(\mathcal{X}_t, \mathcal{Y}_t)$ are the Cartesian coordinates of the location.

Next, we focus on noise removal. In order to reduce burst noise, we carry out Butterworth filtering on each time series represented by the rows of $\hat{\boldsymbol{C}}_{m,n}$. Even though principal component analysis (PCA) based noise removal is proven to be more effective at burst noise removal in CSI signals than low-pass filtering [33], we again resort to the low complex method to minimize the implementation complexity. Butterworth low pass filtering provides sufficiently adequate noise removal, as shown later in our experimental results. We denote the noise filtered data matrix of $\hat{\boldsymbol{C}}_{m,n}$ by $\tilde{\boldsymbol{C}}_{m,n}$. Guidelines on selecting the cutoff frequency of the low pass filter are also provided with reference to the experiments in Section V.

We concatenate all filtered CSI for a particular activity such that the resultant $MN\bar{S}$-by-T_{jk} matrix is given by $\tilde{\boldsymbol{C}} = [\tilde{\boldsymbol{C}}_1 \cdots \tilde{\boldsymbol{C}}_M]^\top$, where $\tilde{\boldsymbol{C}}_m = [\tilde{\boldsymbol{C}}_{m,1} \cdots \tilde{\boldsymbol{C}}_{m,N}]^\top$ for all $m \in \{1, \ldots, M\}$. We segment $\tilde{\boldsymbol{C}}$ into time steps of $f_s T$ samples (corresponding to T seconds) with 90% overlap, to produce the labelled dataset D of dimensions $[R, f_s T, MN\bar{S}]$, where R is the number of training/testing samples. It is necessary to maintain that $T < T_{jk}, \ \forall j, k$, such that T is less than the lowest duration of any trial of any activity. The complete data processing stage is summarized in Algorithm 1.

Algorithm 1 CSI data pre-processing Algorithm

Input: $y_{m,n}(t) \ \forall m \in [1, ..., M]$, $\forall n \in [1, ..., N]$ and $\forall t$
Output: Processed dataset D to be fed in to the deep networks

1: $\forall m, n, \boldsymbol{h}_{m,n,t} \leftarrow$ Estimate Channel$(y_{m,n}(t))$
2: $\forall m, n, \hat{\boldsymbol{h}}_{m,n,t} \leftarrow$ Sparsity Reduce$(\boldsymbol{h}_{m,n,t})$
3: $\forall m, n, \hat{\boldsymbol{C}}_{m,n} \leftarrow [\hat{\boldsymbol{h}}_{m,n,1} \ \cdots \ \hat{\boldsymbol{h}}_{m,n,T_{jk}}]$
4: $\forall m, n, \tilde{\boldsymbol{C}}_{m,n} \leftarrow$ Butterworth$(\hat{\boldsymbol{C}}_{m,n})$
5: $\forall m, \tilde{\boldsymbol{C}}_{\boldsymbol{m}} \leftarrow [\tilde{\boldsymbol{C}}_{m,1} \cdots \tilde{\boldsymbol{C}}_{m,N}]^\top$
6: $\tilde{\boldsymbol{C}} \leftarrow [\tilde{\boldsymbol{C}}_1 \cdots \tilde{\boldsymbol{C}}_M]^\top$
7: $D \leftarrow$ Segment$(\tilde{\boldsymbol{C}})$

IV. Deep Networks for Activity Classification, Authentication and Tracking

We consider a threefold classification of activities as *dynamic*, *static* and *no activity* (NoAc). Firstly, we classify an activity as dynamic if the location coordinates of the performer

vary significantly during the action. This includes activities such as running and walking. Secondly, we classify an activity as static if the location coordinates of the performer do not vary significantly during the action. For example, activities such as sitting, jumping and falling can be categorized as static activities if minor location coordinate changes are disregarded. Finally, as a novel and very important contribution, we classify scenarios where there is no one in the sensing area, or the performer in the sensing area is not engaging in any activity, as NoAc. If NoAc is not captured in the classification process, the system will require the user to initiate the system before performing the activity, *e.g.*, through a push button input. Classification of NoAc makes this a system that requires zero user interaction.

A. Activity classification

We propose the deep neural network illustrated in Fig. 4 for activity classification. The network consists of two recurrent layers followed by one fully connected layer. We add a dropout layer after the first recurrent layer to reduce overfitting and set the dropout rate to 0.3 [34]. We used three different types of recurrent layers in our study, Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU) and Bidirectional-Gated Recurrent Unit (B-GRU), in an attempt to find the best suited recurrent layer type for each task. Following the recommendations in [34], it is widely accepted that, in order to avoid overfitting, dropout should be used prior to the layers with the highest percentage of trainable parameters, which results in them being more prone to co-adapting themselves to the training data. For our networks, the highest percentage of the trainable parameters (more than 75%) are concentrated in the recurrent layers. Hence, we add the dropout layers between the recurrent layers. Literature suggests that in general, the dropout rate is set between 0.2 [35] and 0.5 [34] to achieve optimum results. Hence, in compliance, we empirically set the drop rate to 0.3 for our application. The two recurrent layers are set with tanh activation in order to maintain the layer outputs within $(1, -1)$, similar to the normalized CSI streams [36], whereas the final fully connected layer has softmax activation following the existing state-of-the-art classification approaches [37]. Each recurrent layer consists of $3 \times MN\bar{S}$ hidden units, and the fully connected layer consists of K units, where K is the number of activity classes, M is the number of receivers, N is the number of antennas per receiver and $\bar{S}$ is the number of subcarriers after sparsity reduction. Following the existing state-of-the-art classification approaches with more than two distinct classes, we use categorical cross-entropy as the loss function, where the total loss for activity recognition is given by

$$L_{\text{activity}} = -\frac{1}{R}\sum_{r=1}^{R}\sum_{k=1}^{K} a_r \ \log(b_{r,k}), \tag{2}$$

where a_r is the ground truth of the r-th sample, and $b_{r,k}$ is the predicted value for the k-th activity of the r-th training/testing sample. The network uses adam with a learning rate of 0.001 as the optimizer, following [38].

B. User Authentication

The network we propose for authentication consists of three recurrent layers, followed by one fully connected layer. We add a dropout layer after the first recurrent layer to reduce overfitting and set the dropout rate to 0.3. The fully connected layer consists of P units, where P is the number of participants. The rest of the network parameters are similar to the activity recognizer in IV-A. Using the same loss function as earlier, the total loss for activity recognition is given by

$$L_{\text{auth}} = -\frac{1}{R}\sum_{r=1}^{R}\sum_{p=1}^{P} c_r \log(d_{r,p}), \tag{3}$$

where c_r is the ground truth of the r-th sample, and $d_{r,p}$ is the predicted value for the p-th person of the r-th sample.

C. Tracking

The tracking network consists of two recurrent layers. First layer consists of two parallel recurrent layers with each having $3 \times MN\bar{S}$ hidden units, that extracts low level features from the input sequence. Second recurrent layer consists of $3 \times MN\bar{S}$ hidden units, which identifies the remaining patterns present in the data sequence. We add a dropout layer after the first recurrent layer and set the dropout rate to 0.2. Final regression layer is trained to regress on x and y distances using the features that are extracted in the second recurrent layer. We use the mean squared error (MSE) as the loss function such that the total loss for tracking is given by

$$L_{\text{tracking}} = \frac{1}{R}\sum_{r=1}^{R}(x_{p,r} - x_{t,r})^2 + \frac{1}{R}\sum_{r=1}^{R}(y_{p,r} - y_{t,r})^2, \tag{4}$$

where $(x_{p,r}, y_{p,r})$ are the predicted Cartesian coordinates and $(x_{t,r}, y_{t,r})$ are the ground truth Cartesian coordinates of the r-th training/testing sample.

Proper training is critical for the performance of the system. It is clear that the tracking network should only be trained on dynamic activities because effective learning necessitates significant changes in location. Similarly, the authentication network should only be trained on dynamic activities, as it will not be effective to authenticate the performer after learning through static activities. For example, consider authenticating a performer from the manner in which he jumps or falls. On the other hand, the activity recognition network is trained on all these three categories of activities. In this case, even not doing an activity should be classified, and hence, NoAc is of importance as well. Note that in a practical activity recognition system, NoAc will be the most common and frequent activity. Since the activity labels are available during the training phase, we can conveniently train the authentication and tracking networks with only the dynamic activities and the activity recognition network with all activity classes, including the balanced NoAc class.

Now, let us focus on the inference phase; that is, the operation of a functioning deployed system. In the inference phase, we do not have knowledge on the activity classification as a priori. Therefore, the incoming CSI data are fed into all three networks for prediction, irrespective of the activity. Obviously, the authenticator and the tracker will fail to predict

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSEN.2020.2987386, IEEE Sensors Journal

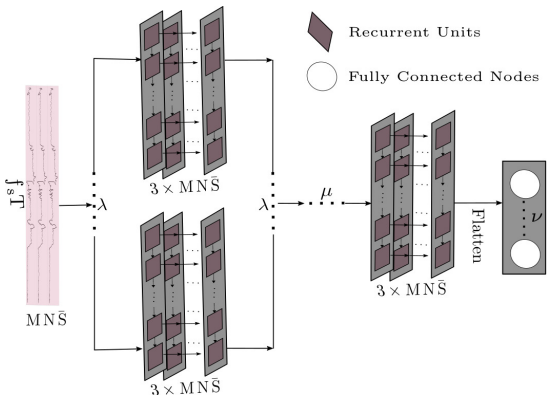

Fig. 4. General network architecture, where λ denotes the number of parallel hidden layers, μ denotes the number of sequential hidden layers after the block of parallel layers and ν denotes the dimension of the output vector for each task. For the activity recognizer, $\lambda = 1, \mu = 1, \nu = K$, for the authenticator $\lambda = 1, \mu = 2, \nu = P$, and for the tracker $\lambda = 2, \mu = 1, \nu = 2$. Here, M is the number of receivers, N is the number of antennas per receiver and $\bar{S}$ is the number of subcarriers after sparsity reduction.

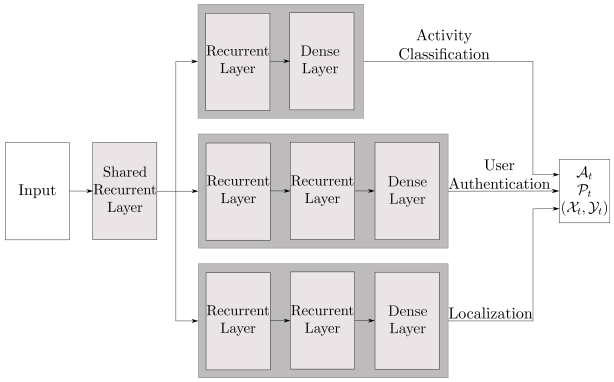

Fig. 5. Combined single multi-task network. For the dynamic activities, the pre-processed CSI data can be fed to the shared layer, in order to concurrently generate $[\mathcal{A}_t, \mathcal{P}_t, (\mathcal{X}_t, \mathcal{Y}_t)]$, where at a given time t, $\mathcal{A}_t$ is the predicted activity, $\mathcal{P}_t$ is the predicted person and $(\mathcal{X}_t, \mathcal{Y}_t)$ are the Cartesian coordinates of the location.

accurately for static and NoAc categories. However, since we have an activity predictor, we can observe the output of the activity predictor, and discard the predictions of the authenticator and the tracker for static and NoAc categories, as illustrated by the decision box in Fig. 3. We update the activity $\mathcal{A}_t$, authentication $\mathcal{P}_t$ and location $(\mathcal{X}_t, \mathcal{Y}_t)$ annotations if $\mathcal{A}_t$ is a dynamic activity. Else, we discard the predicted authentication and location annotations, and continue with the previously predicted annotations, *i.e.*, $\mathcal{P}_t = \mathcal{P}_{t-1}$ and $(\mathcal{X}_t, \mathcal{Y}_t) = (\mathcal{X}_{t-1}, \mathcal{Y}_{t-1})$, respectively.

D. Combined multi-task network for dynamic activities

It is evident that the authenticator and the tracker should only be trained on dynamic activities, whereas the activity recognizer can be trained on both static and dynamic activities. Due to this variability, it is challenging to design a single network for all three tasks. However, for applications with only dynamic activities, we propose a single multi-task network, as illustrated in Fig. 5. The network consists of an initial recurrent layer, which is shared by all the three tasks. Since the lower level features learnt for all three tasks are similar, the initial layer can be jointly learnt by all tasks. Subsequently, the network splits into three task heads, each corresponding to

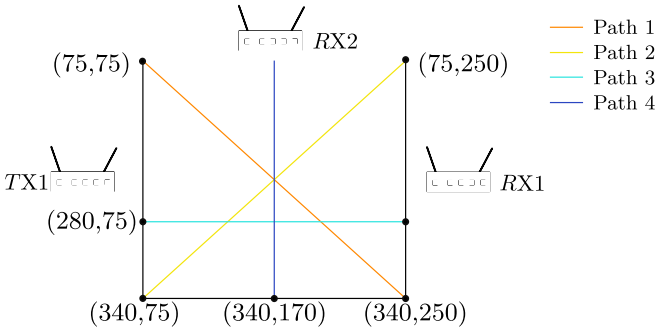

Fig. 6. Experiment setup, where different colored lines show the four walking paths used to experiment tracking algorithm, and TX1, RX1, RX2 refer to the transmitter and the two receivers. The distances are in cm.

activity recognition, classification and tracking. Each recurrent layer has tanh activation, and every fully connected layer in the classification heads has softmax activation, whereas every fully connected layer in the regression head has linear activation.

We define the objective function for the proposed multi-task network as a weighted sum of the three distinct loss functions defined in (2), (3) and (4), as follows:

$$L^* = \alpha\ L_{\text{activity}} + \beta\ L_{\text{auth}} + \gamma\ L_{\text{tracking}}, \tag{5}$$

where $0 < \alpha, \beta, \gamma < 1$ and $\alpha + \beta + \gamma = 1$.

V. Experimental Validation

A. Experimental Setup

We deploy our system using three Universal Software Radio Peripheral (USRP) N210 software defined radios (SDRs), each configured to have one omni-directional antenna, such that we have one transmitter and two receivers. The acquisition sampling rate, f_s, for each antenna is 100Hz. Optimum placement of the transmitter and the receivers is not studied in this paper, and we resort to the simplest form of placing the two receivers. To this end, one receiver (RX1) is placed directly opposite to the transmitter, and the other (RX2) is placed perpendicular to the line that joins the transmitter and RX1, as illustrated in Fig. 6. The perpendicular placement allows us to achieve perspective invariance. We show later that the results obtained from this simple setup, which may or may not be optimal, can be used to get insights on the optimal placement of receivers. The SDRs are programmed to transmit and receive Wi-Fi packets with 64 subcarriers ($S = 64$), and in the 2.45GHz frequency band. The SDRs that act as receivers are utilized for the channel estimation. The experiment is done inside an indoor environment (lecture room), where there is rich scattering, and high interference from other Wi-Fi networks and radio frequency (RF) sources.

B. Data Collection

We collected data using $P = 13$ voluntary participants, where each participant is requested to perform a set of pre-designated activities. For the experiments, we use three static activities *sit*, *fall* and *jump*, two dynamic activities *walk* and *run*, and the *NoAc* category. The data collection duration for one activity trial is 10 seconds, during which the participant is instructed to remain stationary for a brief period, prior to

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSEN.2020.2987386, IEEE Sensors Journal

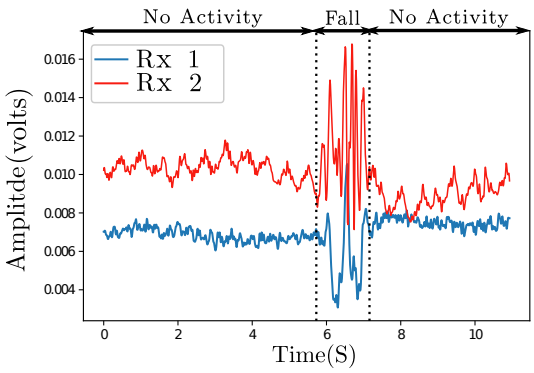

Fig. 7. Activity and no-activity labelling of the CSI streams, in order to obtain the training data for each activity class, where RX1 and RX2 refer to the receivers.

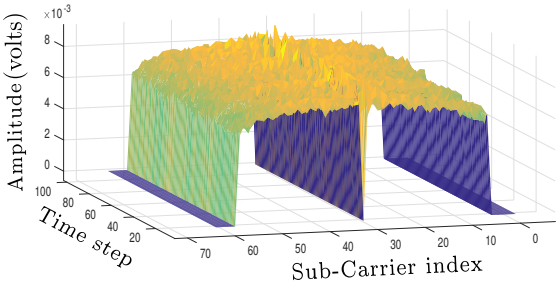

Fig. 8. A sample $\hat{\boldsymbol{h}}_{1,1,t}$, which is the estimated channel state vector, as a function of the subcarrier index. It can be clearly observed that amplitudes of certain subcarriers are approximately zero.

commencing the activity, and after completing the activity until instructed to stop. The *walk* activity consists of four different paths (resulting in $K = 9$: *sit, jump, fall, run, NoAc, walk1, walk2, walk3, walk4*), which are marked on the floor, as illustrated by Fig 6. We obtained camera video recordings of each trial in order to annotate the collected CSI streams. To this end, the stationary periods are annotated as NoAc and the activity period is annotated with the corresponding activity, as illustrated by Fig. 7. Fig. 8 illustrates an example for an estimated channel state vector, obtained from a commercial Wi-Fi device with 64 sub-carriers. Note that sub-carriers with indices $[1, 2, 3, 4, 5, 6, 33, 60, 61, 62, 63, 64]$ have approximately zero amplitude in this case, and hence, highlighting the importance of the sparsity reduction proposed in Section III-B. The data preprocessing technique proposed in Section III-B is applied on the extracted CSI results in frames with dimensions $[1, 80, 104]$, for $T = 0.8$ s. The duration of the *jump* activity generally registers the lowest duration, between $0.9 - 1.1$ s. Hence, the value of T is chosen to be 0.8 s, which is less than 0.9 s. We assume that human movements typically do not exceed 10 m/s, which is consistent with the 7.7 m/s limit set in [33], where running is not considered. Thus, typical human movements introduce frequencies less than $\frac{10}{0.1223/2} = 163.53$ Hz [33] at a wavelength of 12.23 cm since our system operates at 2.45 GHz. Hence, the cut-off frequency of the 10^{th} order Butterworth low pass filter is set to 200 Hz.

For the classification tasks, which are activity recognition and authentication, we train two different models, that consist of LSTM layers, and GRU layers, respectively. For the regression task, which is tracking, we train three different models, that consist of the state-of-the-art recurrent layers: LSTM layers, GRU layers and B-GRU layers [36], respectively.

A major challenge in generating a dataset for activity recognition is the class imbalance caused by the variable durations of the activities. For example, the *sit* activity has an average duration of 1 s, whereas the *run* activity has an average duration of 4 s, resulting in more data samples. On average, NoAc accounts for 60%-90% of the duration of each trial, largely contributing to the data imbalance. In an attempt to reduce this inherent class imbalance, we vary the number of trials (estimated based on the average duration per activity and the comfort of the participant in performing repeated trials) for each activity inversely proportional to the duration of the activity. Hence, for *sit* and *jump* activities, we conduct 10 trials for each participant, whereas we conduct 8 trials for the *fall* activity and 6 trials for the *run* and *walk* activities, for each participant, respectively. The reduced number of training samples due to the low activity duration can be compensated by obtaining samples from a higher number of trials. Further, we randomly sample the *no activity* duration in order to obtain training samples matching the average number of training samples obtained for other activity classes, thus resolving the class imbalance.

C. Robustness analysis of the proposed networks

We study the robustness of the proposed networks through several experiments. We use the collected data from an in-the-wild participant, whose data is not used for training, to investigate the robustness of the activity recognizer and the authenticator. We hypothesize that if the two networks are sufficiently robust, the activity recognizer should successfully classify the activities with high confidences, whereas the authenticator should fail to provide a prediction with high confidence.

VI. Results and Discussion

In this section, we present and discuss the experimental results. We use ensembling techniques [39] to improve the performance of our networks by reducing the model variance, and take a simple weighted voting of the predictions of the set of classifiers to produce the ensemble prediction. Each model is trained on a GTX 1080 graphics processing unit (GPU) for 60 epochs. The results obtained for the proposed networks for the three tasks, with different types of recurrent layers, are summarized in Table I. Each result depicts the average of the maximum accuracy obtained in 3 independent trials. The average inference time per training sample in milliseconds for each model is summarized in Table II.

TABLE I
Results for different types of recurrent layers.

Task	LSTM	GRU	B-GRU	Ensemble Model	Precision	Recall
Activity recognition	96.36%	98.11%	-	**98.74%**	0.9883	0.9826
Authentication	94.41%	95.53%	-	**97.41%**	0.9754	0.9745
Tracking	$\pm$21 cm	$\pm$19 cm	$\pm$10 cm	$\pm$**10 cm**	-	-

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JSEN.2020.2987386, IEEE Sensors Journal

TABLE II
INFERENCE TIME PER 800 MS LONG SAMPLE.

Task	LSTM	GRU	B-GRU	Ensemble Model
Activity recognition	4.28 ms	3.75 ms	-	**5.82 ms**
Authentication	6.32 ms	5.63 ms	-	**8.64 ms**
Tracking	3.23 ms	3.10 ms	5.60 ms	**8.72 ms**

TABLE III
COMPARISON ON OUR TRACKING METHOD, WITH THE STATE-OF-THE-ART DEVICE FREE TRACKING RESULTS. P TRANSMITTERS WITH Q ANTENNAS EACH AND M RECEIVERS WITH N ANTENNAS EACH

Method	Sensing area(m)	$M \times N$	$P \times Q$	Error
Youssef *et al* [25]	2.74m long strip	2×2	2×2	$\pm$15.7cm
IndoTrack [26]	7m $\times$ 7.4m 68.5m^2	2×3 4×3	2×3 2×3	$\pm$62cm $\pm$62cm
This paper	3.4m $\times$ 2.5m	2×1	1×1	$\pm$10cm

A. Performance of the Activity Recognition Network

It is evident from Table I that the proposed model with GRU layers marginally outperforms the model with LSTM layers (by 1.45%). Nevertheless, ensembling over the said two models provides the best performance of 98.74%, outperforming the GRU model by a margin of 0.63%. According to Table II, the time taken by the ensemble model to perform inference on a sample 800 ms long is 5.82 ms, which is higher than that for the LSTM and GRU models, yet, acceptable for a real time system since inference is completed in 0.72% of the total sample duration. Table IV tabulates error rates for each activity used for activity recognition. The highest error rate of 2.49% is recorded by the 3-rd variation of the *walk* activity, and yet, it is within acceptable margins. The *NoAc* class is evidently distinct from every other activity class and the *jump* activity contains upwards motion not present in any other activity class, resulting in 0% error rate. Furthermore, *sit* and *fall* have similar downwards motions, and *run* and *walk* have similar forwards motions causing them to have higher error rates.

B. Performance of the Authentication Network

Similar to the activity recognizer, it can be observed from Table I that the proposed model with GRU layers marginally outperforms the model with LSTM layers (by 1.12%) and ensembling over the said two models provides the best performance, outperforming the GRU model by a margin of 1.88%. According to Table II, the time taken by the ensemble model to perform inference on a sample 800 ms long is 8.64 ms, which is higher than that for the LSTM and GRU models, yet, acceptable for a real time system since inference is completed in 1.08% of the total sample duration.

C. Performance of the Tracking Network

By referring to Table I, the proposed model with GRU layers slightly outperforms the model with LSTM layers by an error margin of $\pm$ 2 cm, which is an improvement of approximately 9.52%, whereas the model with B-GRU layers significantly outperforms the model with GRU layers by an error margin of $\pm$ 9 cm, which is a significant improvement

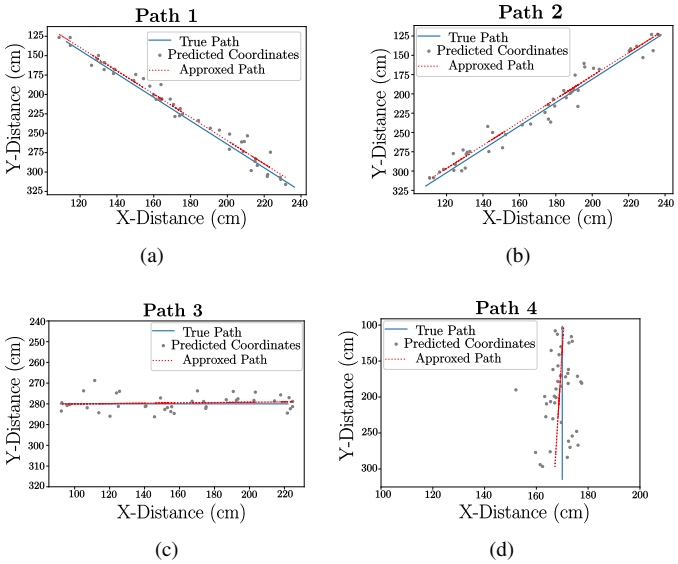

Fig. 9. Trajectories predicted by the tracking network, for the four different walking paths defined in Fig. 6. It is evident from the observation that the approximated trajectory (red) is close to the true trajectory (blue) in each instance.

of approximately 47.36%. It achieves a mean squared error of $\pm$ 10 cm on both x and y coordinates, which is highly acceptable for an indoor tracking system. Table III compares our system with the existing state-of-the-art, in terms of the size of the sensing area, number of transmitters/receivers used, mean squared error achieved, etc.

Fig. 9 illustrates the estimated trajectories for the four different paths of the *walk* activity. The ground truth is represented by the blue lines, whereas the Cartesian coordinates predicted by the network are represented by the small gray circles. The red dotted line illustrates the predicted trajectory, which is the regression line estimated from the predicted Cartesian coordinates using the least squares regression technique. It is evident that the proposed network is able to successfully track the trajectory for the first three paths of the user, including the starting and the ending coordinates of the trajectory. We can observe the predicted path diverging from the ground truth in the fourth path, as the user approaches the side of the sensing area where a receiver has not been placed. It is not hard to see that placing a third receiver at $(340, 170)$ (please refer Fig. 6) will lead to superior performance in all predictions (we were unable to experiment with a three receiver set up due to hardware limitations).

When comparing the inference times between B-GRU and other uni-directional models, B-GRU inference time is 1.75 times higher than that of the other two models. However, B-GRU completes the inferring within 0.7% of the sample duration, which qualifies for a real time system.

TABLE IV
ERROR RATES FOR ACTIVITY RECOGNITION.

sit	*jump*	*fall*	*run*	*NoAc*	*walk1*	*walk2*	*walk3*	*walk4*
1.65%	0%	0.68%	1.58%	0%	1.38%	1.01%	2.49%	1.71%

TABLE V
RESULTS OF THE COMBINED MULTI-TASK NETWORK.

Task	Model performance	Inference time
Activity recognition	97.36 %	14.35 ms
Authentication	95.90 %	14.35 ms
Tracking	$\pm$ 12 cm	14.35 ms

TABLE VI
CONFIDENCE SCORES FOR THE IN-THE-WILD DATASET.

Task	Confidence score margin	Main test set	In-the-wild dataset
Activity recognition	75%	98.85%	**86.86%**
Authentication	99.9%	79.12%	**25.57%**

D. Performance of the Combined Multi-task Network

We empirically set values for α, β and γ as $0.15, 0.15$ and 0.70, respectively. Using these values, we obtain the results tabulated in Table V. The performance of the combined model for the three tasks marginally falls below the performance of the respective individual models, yet, achieves a significant speedup. The ensemble models collectively require $5.82 + 8.64 + 8.72 = 23.18$ ms for the entire prediction, whereas the combined model requires only 14.35 ms, achieving a speedup of 38%. Operating all the ensemble models in parallel is not feasible in practice due to memory constraints of GPUs. Thus, the system operates sequentially.

E. Results of the robustness analysis

We base our robustness analysis of the activity recognizer and the authenticator on the highest confidence score of its prediction, as summarized in Table VI. For the authenticator, we argue that the confidence of a successful authentication should be near perfect and we raise the confidence score margin to a high value of 99.9%. This is due to the fact that, if an authenticator network recognizes an individual, it should almost be fully confident about its prediction, due to the sensitive nature of authentication. Thus, after thresholding, the accuracy for the in-the-wild dataset should be as low as possible, to avoid unauthorized persons being authenticated. Raising the confidence margin in this manner reduces the performance of the main test set to a 79.12%, yet, we can tolerate false negatives over false positives from an authentication system. The authenticator only predicts 25.57% of the in-the-wild dataset, establishing that it often fails to recognize a person that it has never seen before, demonstrating its robustness.

For the activity recognizer, we accept the prediction of the network if the confidence of prediction is above 75%. The confidence of a successful prediction should be high, yet intuitively, the threshold need not be enforced as strictly as for the authenticator. Hence, we choose a generic value of 75% as the confidence threshold. 98.85% of the activity predictions on the main test set by the activity recognizer were made with a confidence score in excess of 75%, whereas a significant portion of 86.86% of the activity predictions on the in-the-wild dataset were also made with a confidence score in excess of 75%. Hence, it can be concluded that the activity recognizers can successfully recognize the activities of a person that the network has never seen before, which ensures its robustness.

F. Potential enhancements to the proposed system

Even though the simple setup proposed in Section V-A provides adequate performance, a superior performance can be obtained via several enhancements. Our system discarded the phase information of the estimated CSI for implementational simplicity, yet integrating phase information for a system where the required additional processing can be dispensed can lead to better performance. Similarly, increasing the number of receivers and the number of antennas per receiver can provide finer predictions in more complex scenarios. Optimal antenna placement for the transmitters and receivers can also enhance the performance of the system.

VII. CONCLUSIONS

This paper proposes a novel system capable of completely characterizing an event, using processed channel state information gathered from commercial Wi-Fi devices. The system performs activity recognition, authentication and tracking simultaneously, by utilizing deep neural networks. It is fully autonomous, non-invasive, requires zero user intervention, device free and passive. Experimental results have been presented to demonstrate the feasibility and the achievable performance of the proposed system. The results have shown that the proposed system achieves promising prediction scores for all three tasks on a collected dataset, by only using two single-antenna Wi-Fi receivers. The robustness of the system has been established through an in-the-wild study. Possible improvements of the proposed system have also been highlighted. Future work includes performance improvements for the single user case and extending the framework to accommodate multiple simultaneous users.

REFERENCES

[1] J. Metzler, "Appearance-based re-identification of humans in low-resolution videos using means of covariance descriptors," in *Proc. IEEE International Conference on Advanced Video and Signal-Based Surveillance*, Sep. 2012.

[2] H. Han, M. Zhou, and Y. Zhang, "Can virtual samples solve small sample size problem of kissme in pedestrian re-identification of smart transportation?" *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–11, 2019.

[3] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Proc. Advances in neural information processing systems*, pp. 568–576, Dec. 2014.

[4] J. Kang, I. Cohen, and G. Medioni, "Continuous tracking within and across camera streams," in *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2003.

[5] X. Yuan, L. Kong, D. Feng, and Z. Wei, "Automatic feature point detection and tracking of human actions in time-of-flight videos," *IEEE/CAA Journal of Automatica Sinica*, vol. 4, no. 4, pp. 677–685, Sep. 2017.

[6] D. Konings, N. Faulkner, F. Alam, E. M. . Lai, and S. Demidenko, "Fieldlight: Device-free indoor human localization using passive visible light positioning and artificial potential fields," *IEEE Sensors Journal*, vol. 20, no. 2, pp. 1054–1066, Jan. 2020.

[7] T. Grosse-Puppendahl, X. Dellangnol, C. Hatzfeld, B. Fu, M. Kupnik, A. Kuijper, M. R. Hastall, J. Scott, and M. Gruteser, "Platypus: Indoor localization and identification through sensing of electric potential changes in human bodies," in *Proc. Annual International Conference on Mobile Systems, Applications, and Services*, pp. 17–30, Jun. 2016.

[8] J. Iqbal, M. T. Lazarescu, O. B. Tariq, A. Arif, and L. Lavagno, "Capacitive sensor for tagless remote human identification using body frequency absorption signatures," *IEEE Transactions on Instrumentation and Measurement*, vol. 67, no. 4, pp. 789–797, Jan. 2018.

[9] J. Wang, X. Zhang, Q. Gao, H. Yue, and H. Wang, "Device-free wireless localization and activity recognition: A deep learning approach," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 7, pp. 6258–6267, Dec. 2016.

[10] J. Preis, M. Kessel, M. Werner, and C. Linnhoff-Popien, "Gait recognition with Kinect," in *Proc. International Workshop on Kinect in Pervasive Computing*, Jun. 2012.

[11] F. Sun, C. Mao, X. Fan, and Y. Li, "Accelerometer-based speed-adaptive gait authentication method for wearable IoT devices," *IEEE Internet of Things Journal*, vol. 6, no. 1, pp. 820–830, Jul. 2018.

[12] R. Chellappa, A. K. Roy-Chowdhury, and A. Kale, "Human identification using gait and face," in *Proc. IEEE International Conference on Computer Vision*, Jun. 2007.

[13] Y. Zeng, P. H. Pathak, and P. Mohapatra, "WiWho: Wifi-based person identification in smart spaces," in *Proc. IEEE International Conference on Information Processing in Sensor Networks*, p. 4, Apr. 2016.

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

[15] A. Pokkunuru, K. Jakkala, A. Bhuyan, P. Wang, and Z. Sun, "NeuralWave: Gait-based user identification through commodity WiFi and deep learning," in *Proc. Annual Conference of the IEEE Industrial Electronics Society*, pp. 758–765, Oct. 2018.

[16] C. Lin, J. Hu, Y. Sun, F. Ma, L. Wang, and G. Wu, "WiAU: An accurate device-free authentication system with ResNet," in *Proc. IEEE International Conference on Sensing, Communication, and Networking*, pp. 1–9, Jun. 2018.

[17] Q. Pu, S. Gupta, S. Gollakota, and S. Patel, "Whole-home gesture recognition using wireless signals," in *Proc. ACM International Conference on Mobile Computing & networking*, Sep. 2013.

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

[19] S. Yun, Y.-C. Chen, H. Zheng, L. Qiu, and W. Mao, "Strata: Fine-grained acoustic-based device-free tracking," in *Proc. ACM Annual International Conference on Mobile Systems, Applications, and Services*, pp. 15–28, Jun. 2017.

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

[21] J. Zhang, S. Zhu, D. Zang, and M. Zhou, "A sliding window method for online tracking of spatiotemporal event patterns," in *Proc. International Conference on Internet and Distributed Computing Systems*, pp. 513–524, Sep. 2016.

[22] X. Wang, L. Gao, S. Mao, and S. Pandey, "CSI-based fingerprinting for indoor localization: A deep learning approach," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 1, pp. 763–776, Mar. 2016.

[23] K. S. Kim, S. Lee, and K. Huang, "A scalable deep neural network architecture for multi-building and multi-floor indoor localization based on Wi-Fi fingerprinting," *Big Data Analytics*, vol. 3, no. 1, p. 4, 2018.

[32] S. Sen, B. Radunovic, R. R. Choudhury, and T. Minka, "You are facing the Mona Lisa: Spot localization using PHY layer information," in *Proc. ACM International Conference on Mobile Systems, Applications, and Services*, Jun. 2012.

[24] J.-W. Jang and S.-N. Hong, "Indoor localization with WiFi fingerprinting using convolutional neural network," in *Proc. IEEE International Conference on Ubiquitous and Future Networks*, pp. 753–758, Jul. 2018.

[25] M. Youssef, M. Mah, and A. Agrawala, "Challenges: device-free passive localization for wireless environments," in *Proc. ACM International Conference on Mobile Computing and Networking*, Oct. 2007.

[26] X. Li, D. Zhang, Q. Lv, J. Xiong, S. Li, Y. Zhang, and H. Mei, "Indotrack: Device-free indoor human tracking with commodity Wi-Fi," *Proc. ACM Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 1, no. 3, p. 72, Sep. 2017.

[27] F. Adib, Z. Kabelac, D. Katabi, and R. C. Miller, "3D tracking via body radio reflections," in *Proc. USENIX Symposium on Networked Systems Design and Implementation*, Apr. 2014.

[28] F. Li, M. Al-qaness, Y. Zhang, B. Zhao, and X. Luan, "A robust and device-free system for the recognition and classification of elderly activities," *IEEE Sensors*, vol. 16, no. 12, p. 2043, Dec. 2016.

[29] W. Zhang, S. Zhou, L. Yang, L. Ou, and Z. Xiao, "WiFiMap+: High-level indoor semantic inference with WiFi human activity and environment," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 8, pp. 7890–7903, Jul. 2019.

[30] Y. G. Li and G. L. Stuber, *Orthogonal frequency division multiplexing for wireless communications*, 1st ed. Springer Science & Business Media, 2006.

[31] Z. Yang, Z. Zhou, and Y. Liu, "From RSSI to CSI: Indoor localization via channel response," *ACM Computing Surveys*, vol. 46, no. 2, p. 25, Nov. 2013.

[33] W. Wang, A. X. Liu, M. Shahzad, K. Ling, and S. Lu, "Understanding and modeling of Wi-Fi signal based human activity recognition," in *Proc. ACM International Conference on Mobile Computing and Networking*, Sep. 2015.

[34] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, Jan. 2014.

[35] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE conference on computer vision and pattern recognition*, pp. 4700–4708, Jan. 2017.

[36] A. Graves, A.-r. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Proc. IEEE international conference on acoustics, speech and signal processing*, pp. 6645–6649, May. 2013.

[37] C. Nwankpa, W. Ijomah, A. Gachagan, and S. Marshall, "Activation functions: Comparison of trends in practice and research for deep learning," *arXiv preprint arXiv:1811.03378*, 2018.

[38] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[39] T. G. Dietterich, "Ensemble methods in machine learning," in *Proc. Springer International workshop on multiple classifier systems*, Jun. 2000.