IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 5 October 2022, accepted 24 October 2022, date of publication 28 October 2022, date of current version 7 November 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3217910

RESEARCH ARTICLE

CSI-DeepNet: A Lightweight Deep Convolutional Neural Network Based Hand Gesture Recognition System Using Wi-Fi CSI Signal

M. HUMAYUN KABIR, (Member, IEEE), MD. ALI HASAN, AND WONJAE SHIN, (Senior Member, IEEE)

Department of Electrical and Computer Engineering, Ajou University, Suwon 16499, Republic of Korea

Corresponding author: Wonjae Shin (wjshin@ajou.ac.kr)

This work was supported in part by the National Research Foundation of Korea (NRF) under Grant 2021R1A4A1030775 and Grant 2022R1A2C4002065, in part by the Institute of Information and Communications Technology Planning and Evaluation (IITP) under Grant 2018-0-01424 and Grant 2021-0-00467, and in part by the BK21 FOUR Program under Grant 5199991514504.

ABSTRACT Hand gesture is a visual input of human-computer interaction for providing different applications in smart homes, healthcare, and eldercare. Most deep learning-based techniques adopt standard convolution neural networks (CNNs) which require a large number of model parameters with high computational complexity; thus, it is not suitable for application in devices with limited computational resources. However, fewer model parameters can reduce the system accuracy. To address this challenge, we propose a lightweight heterogeneous deep learning-based gesture recognition system, coined CSI-DeepNet. The CSI-DeepNet comprises four steps: i) data collection, ii) data processing, iii) feature extraction, and iv) classification. We utilize a low-power system-on-chip (SoC), ESP-32, for the first time to collect alphanumeric hand gesture datasets using channel state information (CSI) with 1,800 trials of 20 gestures, including the steady-state data of ten people. A Butterworth low-pass filter with Gaussian smoothing is applied to remove noise; subsequently, the data is split into windows with sufficient dimensions in the data processing step before feeding to the model. The feature extraction section utilizes a depthwise separable convolutional neural network (DS-Conv) with a feature attention (FA) block and residual block (RB) to extract fine-grained features while reducing the complexity using fewer model parameters. Finally, the extracted refined features are classified in the classification section. The proposed system achieves an average accuracy of 96.31% with much less computational complexity, which is better than the results obtained using state-of-the-art pre-trained CNNs and two deep learning models using CSI data.

INDEX TERMS Hand gesture recognition, channel state information (CSI), deep learning, depthwise separable convolutional neural network (DS-Conv), feature attention, residual block, system-on-chip (SoC).

I. INTRODUCTION

Gesture recognition is an artificial intelligence (AI) technique that aims at illustrating human gestures. Human behavior or activity thrives with the aid of facial expressions and proper gestures. Different types of gestures are performed daily during communication. Gestures act as a medium of interaction. In the case of people living with disability and aged persons, gestures have a significant impact. Gesture recognition is a fast-paced and demanding research topic. The integration of human gesture recognition provides more interactive applications in smart homes, healthcare, eldercare, resource utilization, security, and energy saving. Traditional gesture recognition systems have been devised using depth and infrared image sensors [1], ultrasonic sensors [2], wearable sensors [3], radio frequency identification (RFID) [4], radio detection and ranging (RADAR) [5] and other special-purpose devices. However, the various drawbacks of these

The associate editor coordinating the review of this manuscript and approving it for publication was Prakasam Periasamy.

systems limit their use. Image sensor-based systems lack privacy, and their recognition performance is significantly influenced by the line-of-sight (LOS), illumination condition, and view angle. Similarly, wearable sensors are obligated to be worn all the time for functioning, which can be uncomfortable for users. Moreover, the establishment cost for ultrasonic sensors and RADAR-based approaches is high, and they have a limited range of coverage.

Compared to previous approaches, Wi-Fi signal-based sensing has recently attracted considerable attention in indoor environments owing to its wide range of coverage, abundant availability, noninvasive nature, user's privacy and identity protection, non-LOS communication, and contactless sensing. However, Wi-Fi signals are affected by moving and stationary objects in the propagation path, thereby resulting in the reflection or refraction of the signal [6]. By analyzing the properties of the signals, we can model the change in the environment due to human body movement, which can help us passively recognize the movement of the human body. The reflection and refraction of the signal can be analyzed using various signal properties (received signal strength indicator (RSSI), and channel state information (CSI)). An RSSI-based technique has been widely used for indoor positioning [7] and tracking [6], which measures the variation in a single value from each packet. Furthermore, this method is unable to model complex scenarios because it cannot handle multi-path fading and time-varying properties. In contrast, CSI contains fine-grained information in each orthogonal frequency-division multiplexing (OFDM) symbol. The channel quality can be evaluated by calculating the amplitude and frequency at the receiver end of each channel using a complex number. The signal power is attenuated due to the multi-path effect, which can be characterized by the amplitude and frequency of the CSI signal. Recently, the CSI available in Wi-Fi 802.11n networks has been considered for fine-grained analysis. Therefore, the detection of human gestures and activities in both LOS and non-LOS scenarios within an indoor environment with CSI is very effective. The CSI captures the amplitude variations and phase information associated with different sub-carriers of the Wi-Fi channels. The amplitude and phase information of CSI signals are affected by multi-path effects and the existence of moving objects in the signal propagation path. The changes in the amplitude of the CSI signals are relatively more stable compared with the phase information. As such, in this study, we focus on the CSI amplitude to build the model.

Different types of hardware are used for the realization of Wi-Fi based gesture sensing applications. Commercial-off-the-shelf (COTS) network devices are used primarily as an access point (AP). A number of studies have addressed the commercial wireless network interface card (NIC) of Intel 5300 with CSI toolkit [8] or Atheros NIC [9]. However, there are some exceptions to using special hardware platforms such as the universal software radio peripheral (USRP) [10]. A laptop [11], [12] is used to collect the CSI; thus, it is a costly and mostly software-centric solution, which is difficult to deploy. Recently, smartphones [13] have been used for gesture recognition; the solution is primarily hardware-specific. Therefore, the aforementioned solutions are predominantly domain-specific and required computing devices, such as laptops, and are unsuitable for the limited computing power of edge devices. Recently, the low-cost and low-power system-on-chip (SoC), ESP-32, has provided a complete application programming interface (API) [14] for handling CSI information. Utilizing this SoC for RF-based sensing could be a viable alternative to modeling the change in environment due to human body movement.

Machine learning and deep learning methods have been used as classification techniques in RF-based gesture recognition systems. Machine learning-based CSI signal classification approaches rely on extracting handcrafted features from CSI signals using various signal processing methods. The extracted features are then classified using different machine learning classification methods. Among them, decision trees (DT), support vector machine (SVM), k-nearest neighbors (k-NN), random forest (RF), and hidden Markov model (HMM) [15] are the most commonly used. However, these methods do not consider optimizing feature extraction, and it is unlikely to obtain new features to manually characterize the information enclosed in the time, frequency, and spatial domains of the CSI signal. To avoid the process of manually designing features, a convolutional neural network (CNN) can be used to learn the features from the input signals. Most deep learning-based approaches focus on increasing the accuracy rate without sacrificing the model parameters, computational complexity, and energy consumption.

Gestures are influenced by individual diversity and inconsistencies. Instead of using the coarse-grained features by the handcraft-based method, a CNN with deep architecture can perform better compared with the other approaches. In this study, we focus on the computational deficiency of deep learning networks using CNN to achieve the desired accuracy considering the limited computing power of edge devices. A standard CNN combines a filter with an input in one step to obtain an output that is inefficient with respect to the model size and speed. In contrast, a factorized convolution operation using a depthwise separable convolutional neural network (DS-Conv) splits the convolution operation into two layers: a depthwise filtering operation and a linear combination to reduce the model computation and parameters. However, we also need to balance the trade-off between model complexity and performance. Hence, a lightweight heterogeneous deep learning architecture is adopted to obtain better accuracy and reduce computational complexity by minimizing the number of trainable parameters. A feature attention (FA) block with a residual block (RB) is utilized to enhance the feature extraction ability, which increases the recognition accuracy. Moreover, a low-power SoC is used as an AP and sensing device to provide a cost-effective and large-scale deployment solution with amplitudes of 52 CSI sub-carriers, including one transmitter (Tx) and receiver (Rx) antenna.

To summarize, the contributions of this study are as follows:

1) We utilize the low-cost, low-power SoC (ESP-32) with CSI API as a stand-alone device for the first time to collect the Wi-Fi CSI dataset for alphanumeric-based hand gesture recognition.
2) We develop a lightweight heterogeneous deep learning network with DS-Conv, FA block, and RB for alphanumeric-based hand gesture recognition.
3) We validate the effectiveness of the proposed model through the collected alphanumeric-based hand gesture datasets (20 alphanumeric characters) in terms of recognition accuracy, trainable parameters, train time, and recognition time.
4) We verify the performance of our proposed model with state-of-the-art models in terms of recognition accuracy, trainable parameters, training time, and recognition time.

The remainder of this study is summarized as follows: Section II, presents a detailed review of the existing technology based on CSI signals is discussed. Section III discusses the details of the system model, including alphanumeric-based gesture data processing, feature extraction, and gesture recognition methodology. Section IV presents the experimental results. Finally, Section V concludes the study and discusses future work.

II. RELATED WORKS

Wi-Fi based gesture recognition systems are widely used owing to their large range of benefits. In this study, we focus on device-free sensing and recognition systems and try to categorize the existing literature based on three factors: specification of signal sensing hardware, properties of Wi-Fi signals, and human gesture recognition techniques.

A. SPECIFICATION OF SIGNAL SENSING HARDWARE

Compared to other methods, Wi-Fi based sensing plays an important role because of its robust features, such as: device-free passive sensing, noninvasive nature to ensure user privacy, a wider range of coverage, and see-through-wall. Commercial wireless NIC (such as, Intel 5300 NIC [8], [16], [17], [18] and Atheros NIC [9], [19]), SoC (such as, Espressif SoC [14]), smartphones (such as Google Nexus 5 [13]), and USRP [10], [20], [21] devices are different types of hardware solutions that are used for Wi-Fi based gesture recognition systems. NICs are primarily designed to support the networking function; they are also widely used for Wi-Fi based gesture recognition. A NIC can be used as a receiver with other computing devices, such as a personal computer (PC), and a commodity Wi-Fi router is used as a transmitter. Although the NIC transceiver sends 56 sub-carriers, information from only 30 sub-carriers is accessible, resulting in a loss of 46% of information. The Intel 5300 NIC provides 30 out of 52 sub-carriers with 20 MHz bandwidth for each transmitter-receiver antenna pair. In contrast, the Atheros NIC and Espressif SoC support all 52 sub-carriers with a primary bandwidth of 20 MHz. NIC-based solutions require computing devices to process the signals. The Android Nexus 5 smartphone using Nexmon firmware is another CSI data collection device that offers 256 sub-carriers of the CSI signal with an 80 MHz bandwidth. However, USRP is a hardware that can be controlled using software, and the number of sub-carriers can be determined based on demands. Additionally, it allows modification of the operating frequency, transmission, and receiving power. Its reusability and programmable features make it a reliable device for the research community, irrespective of its high cost. In the case of low-cost, low-power, and large-scale deployment, SoCs provide an unbiddable solution. The SoC acts as a standalone device with the capability of processing CSI signals. Commercial NIC and SoC are operated at 2.4 GHz based on the 802.11b/g/n/e/i wireless local area network (WLAN) standards. However, CSI measurements using existing tools have several practical limitations when applied to a variety of domain-specific applications. The first constraint is the need for a laptop with a specific NIC to act as an Rx, with previous research studies requiring up to 10 laptops on the Rx side. In contrast, USRP can act as a stand-alone device with a configurable operating frequency and bandwidth; however, it is costly. Owing to their favorable features, SoC-based solutions are cost-effective for the deployment of large-scale applications.

B. PROPERTIES OF WI-FI SIGNAL

Wi-Fi signals are affected by reflection, refraction, and scattering caused by the presence or movement of an object between the transmitter and receiver while the signal is propagated. The environment can be easily modeled by analyzing the changes in the signal properties. Among the two signal properties, the RSSI is a widely used solution [9], [10], [20]. RSSI can be used to measure the variation in the signal value from each packet and can be extracted from any device. The signal provides only coarse-grained information which limits the recognition accuracy of this type of approach [22]. The limitations of handling multi-path fading and time-varying properties restrict its application to models in complex environments. An alternative solution is to use CSI, which provides information on OFDM in each packet and thus yields fine-grained information [12], [13], [17], [18], [19]. CSI can characterize the multi-path effect of the environment by amplitude and frequency. Hence, the research community is interested in the CSI of Wi-Fi signals.

C. HUMAN GESTURE RECOGNITION TECHNIQUES

Current human gesture recognition approaches can be categorized into two types: machine learning and deep learning method. The machine learning approach utilizes statistical features, principal component analysis (PCA), fast Fourier transform (FFT), and inverse fast Fourier transform (IFFT) for classification. Several researchers have focused on machine learning-based approaches, wherein subtle movements of finger gestures are identified through patterns

matching of CSI in WiFinger [12]. Researchers claimed that Wi-Fi CSI could be used to identify gestures even in non-LOS cases. WiFinger comprises two parts: noise filtering and pattern recognition. Wavelet-based denoising is used to mitigate multi-path interference in the environment. In addition, principal component identification is used to extract the gesture pattern. Finally, multi-dimensional dynamic time warping (MD-DTW) is exploited to calculate the similarity between the captured CSI pattern and the pre-constructed gesture profiles. In the study, the WiFinger approach is tested in two environments using two AP, and a laptop is used for CSI collection. Furthermore, the approach exhibited over 93% recognition accuracy for the identification of eight different hand gestures.

Another CSI-based approach is WiCatch [16], which utilizes an Intel 5300 NIC as a transmitter and receiver for CSI data collection. It utilizes one antenna at the transmitter end and two at the receiver end. The received data from the two antennas are fused to remove interference. A trained SVM is used as a classifier to recognize different gestures. WiCatch achieves an overall recognition accuracy of 95% in the case of nine different hand gestures. Dang et. al [17] proposed 10 air-gestures of handwritten numbers 0-9 using the S-DTW algorithm which is a combination of SVM with a dynamic time warping algorithm. An Intel 5300 NIC is used to obtain the Wi-Fi CSI amplitude and phase. Average accuracy of 93% is reported for two different indoor scenes.

An android smartphone-based gesture recognition system is presented by Li et al. [13], wherein specific hardware (Nexus 5 smartphone) using Nexmon firmware and commercial routers as AP are used to collect the amplitude of 256 CSI sub-carrier signals of six different gestures (push-pull, sweep, clap, slide, circle, zigzag). They adopted a new improved constraints multi-dimension dynamic time wrapping (CM-DTW) algorithm to classify and recognize gestures. The overall accuracy of 90% is achieved in the three environments.

Sigg et. al [10] used USRP to collect RSSI required to obtain the transmission channel information between devices owing to changes in the environment. The features are extracted using FFT. k-nearest neighbour (k-NN) classifier with $k = 10$ and a DT are adopted to classify five different activities. Another RSSI-based machine learning approach is presented in WiGest [11] which utilizes RSSI to sense in-air hand gestures with discrete wavelet transformation and pattern matching algorithm. Additionally, they reported that the classification accuracy is positively influenced by the number of AP. Ding et al. [20] presented a human gesture recognition system based on the RSSI signal using the XGBoost algorithm through the software-defined radio platform USRP N210. Their method achieves an accuracy of 94.55% when 10 features are used, whereas the accuracy decreases to 91.75% when two features are used.

Despite the effective results achieved by the previously mentioned machine learning methods, extracting features from CSI data that characterize the information related to the time, frequency, and spatial domains is challenging. A deep CNN can get proper features from an input signal without constructing them individually. Qirong et al. [18] introduced a gesture recognition approach based on CSI signals, using deep transfer learning. A TP-Link router and a laptop with an Intel 5300 NIC are utilized as the Wi-Fi transmitter and receiver, respectively. First, a segmented algorithm is applied to detect gestures; subsequently, the applied algorithm converted them into an image. They evaluated their dataset using a deep CNN and fine-tuned CNN, which achieved better gesture recognition compared with other state-of-the-art methods. A gesture recognition system, WiGR [19] is presented using CSI signals. They used depthwise separable convolution and an inverted residual layer to reduce the model computations and parameters. Accuracy of up to 91.4% with 50% fewer parameters than the other existing systems is reported. They used two Atheros NIC transceivers. Another dual-attention network based on a deep residual network (ResNet) gesture recognition technique coined WiGRUNT by Gu et al. [21]. They used the online available Widar3 dataset and achieved better results than state-of-the-art techniques.

From the above-mentioned research, most studies utilize the CNN-based deep learning model and NIC or USRP for human gesture recognition. They attempted to increase the recognition rate by sacrificing the model parameters and computational complexity. Studies on lightweight deep learning-based models for gesture recognition are limited. However, there is a trade-off between the model parameters and accuracy. In this study, we introduced a heterogeneous deep learning model using DS-Conv with an FA block and RB, which can increase the accuracy while reducing the parameters significantly. Moreover, we utilize a low-cost, low-power SoC (ESP-32) with CSI API as a stand-alone device for the first time to collect the Wi-Fi CSI dataset for alphanumeric hand gesture recognition. This study allows us to select the right-sized model for application based on the constraints of the problem.

III. SYSTEM MODEL

The proposed gesture recognition system is divided into three sections: dataset collection, data processing, and deep learning model (CSI-DeepNet). An alphanumerical hand gesture dataset is collected using a low-cost, low-power consuming specific SoC, called ESP-32 for the first time to collect gesture data, which can be helpful for large-scale deployment in the Internet of Things (IoT) environment. Fig. 1 shows the proposed system model. After collecting the dataset, a data processing technique is implemented for denoising and segmentation. Finally, a lightweight deep learning Network (CSI-DeepNet) with an FA block, RB and DS-Conv is adopted to capture the fine-grained automatic features that ensure a high recognition rate and reduced computational complexity.

A. DATASET COLLECTION

In this study, we utilize the SoC-based standalone device ESP-32 as a transmitter/AP and receiver for CSI data

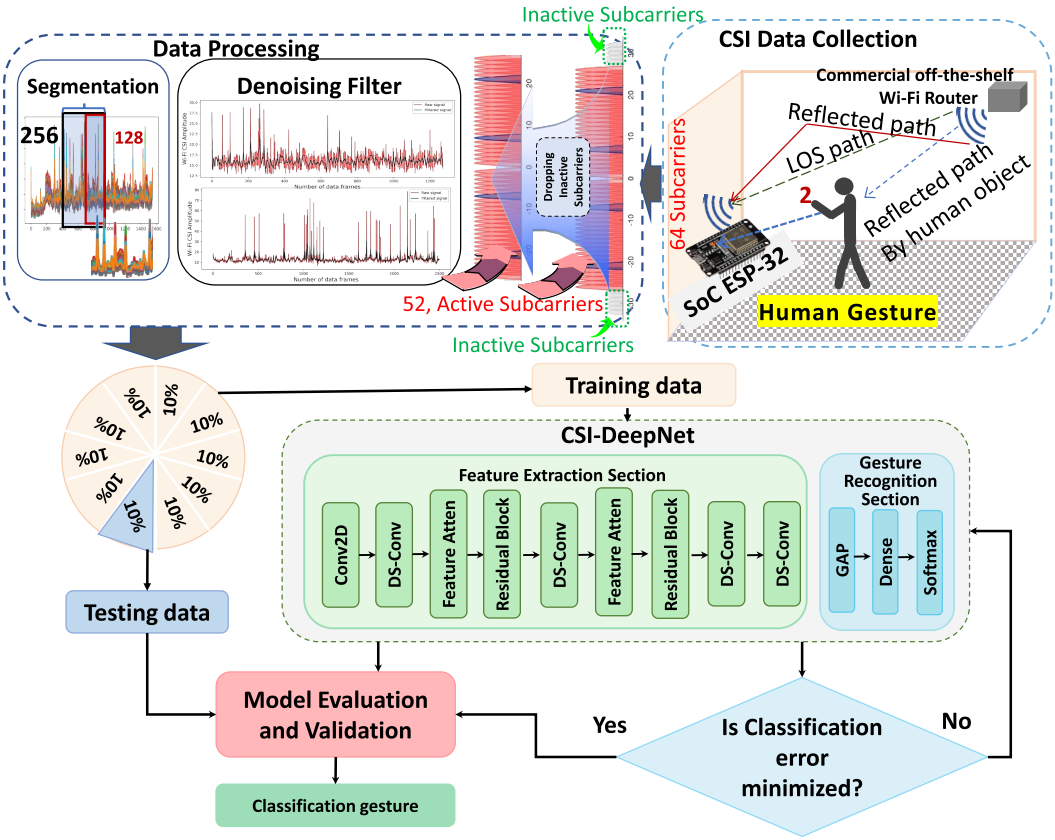

FIGURE 1. Block diagram of the CSI-DeepNet gesture classifier.

collection. ESP-32 [14] is a highly integrated low-cost SoC with a dual-core 32-bit processor developed by Espressif systems. ESP-32 is packed with peripherals, such as built-in antenna switches, RF baluns, power amplifiers, low-noise receiving amplifiers, filters, sensors, and power management modules. This SoC is suitable for different applications, such as mobile devices, IoT, and wearables owing to its ultra-low power consumption features. It can act as a standalone device with both Wi-Fi and Bluetooth connectivity. Furthermore, it can act as an AP mode with a full 802.11b/g/n/e/i WLAN MAC protocol and can communicate to most Wi-Fi routers in the station (client) mode. It can mimic both the Tx and Rx without the aid of any special Tx or Rx for CSI measurements. It favors the CSI API [23] to provide the most accurate CSI measurement from all 52 sub-carriers in the frequency domain and is thus suitable for use in device-free wireless sensing applications. Hence, we utilize for the first time ESP-32 with the CSI API to efficiently capture the CSI gesture dataset efficiently in our interest domain.

CSI in ESP-32 contains the channel frequency responses (CFRs) of the sub-carriers; it is calculated when packets travel from Tx to Rx. Each CFR of a sub-carrier registers as two bytes of signed characters; the first part is the imaginary, and the second part is the real value. The CSI API based on ESP-32 is set up during the installation process to measure the

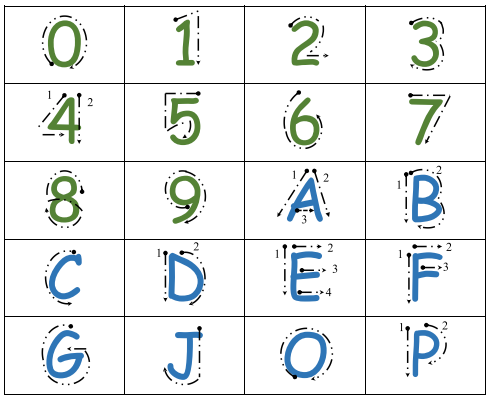

FIGURE 2. The stoke order of 20 alphanumeric characters.

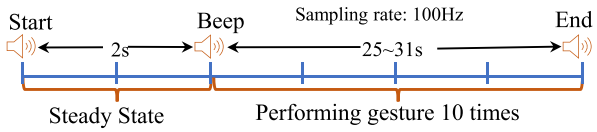

FIGURE 3. Timing diagram for gesture data collection.

CSI values efficiently. Twenty different alphanumerics based on hand gestures, including 10 capital letters (A, B, C, D, E, F, G, J, O, and P) and 10 numbers (0-9) are considered

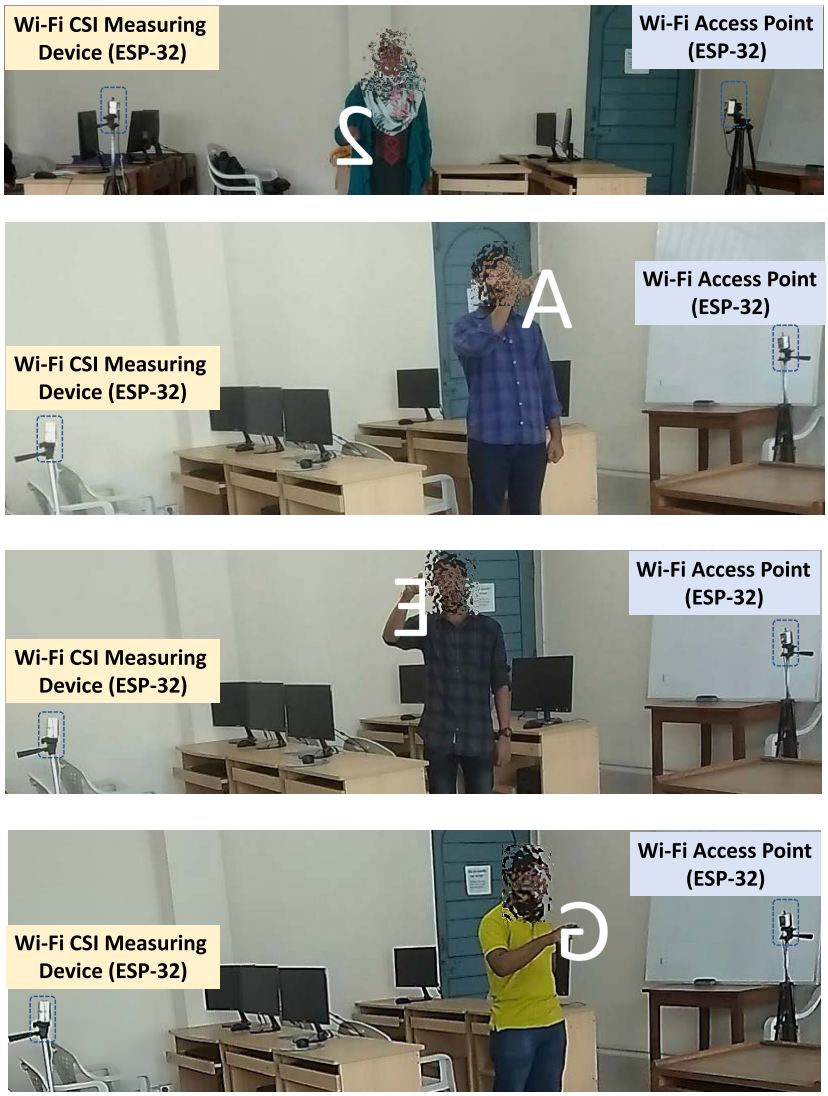

FIGURE 4. Test-bed scenario when the transmitter and receiver are 2 m apart.

for data collection. Ten participants are involved in recording the hand gestures of 20 alphanumeric characters. A total of 1,800 trials are performed; in each trial, each person performs ten gestures frequently with 2 sec of no activity (steady state) at the beginning. Fig. 2 shows the stokes order of 10 letters and 10 numbers used for CSI data collection. During the recording, one start beep is used as an indication to get ready, and after the second beep, the user starts making gestures on the air with the selected alphanumeric characters. This is repeated ten times. Fig. 3 Illustrates the timing diagram for recording. The testbed scenario is shown in Fig. 4. Ten participants are involved in the data collection, of which two are female, and eight are male; furthermore, three scenarios (distance between transmitter and receiver is 1 m, 1.2 m, and 2 m) are considered. The total number of trials is 1,800, with 10 gestures in each trial. The dataset statistics and number of samples for each gesture are presented in Fig. 5. Our dataset is imbalanced in that the numbers of samples for each gesture are different. The possibility of having a dataset with balanced classes in real-world data is low. Highly imbalanced data can hamper model accuracy a lot. Although our dataset is imbalanced in nature, this imbalance is negligible.

B. DATA PROCESSING

Scattering, diffraction, and reflectance events occur in the passage of the signal channel owing to the presence of moving and stationary objects. As the entire wireless channel is split into several narrowband sub-carriers in an OFDM, the communication system can be modeled as [25]

$$\mathbf{y}_i = \mathbf{H}_i \mathbf{x}_i + \mathbf{v}, \quad i = 1, 2, 3, \ldots, N, \tag{1}$$

where $\mathbf{H}_i \in C^{N_R \times N_T}$ denotes the CSI matrix of i^{th} sub-carrier, $\mathbf{v}$ denotes the noise term, N represents the number of OFDM sub-carrier frequencies, and $y_i \in \mathbb{R}^{N_{R_x}}$ and $x_i \in \mathbb{R}^{N_{T_x}}$ is the

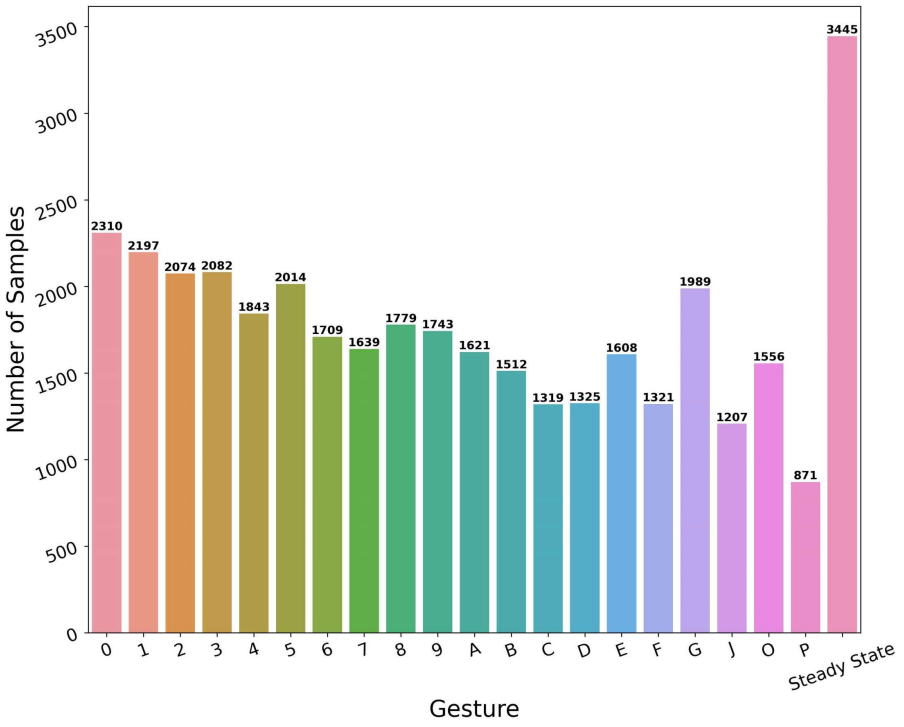

FIGURE 5. Summary of the collected data sample in CSI-based gesture dataset.

i^{th} received and transmitted signal.

$$\mathbf{H}_i = \begin{bmatrix} h_i^{11} & h_i^{12} & \cdots & h_i^{1N_R} \\ h_i^{21} & h_i^{22} & \cdots & h_i^{2N_R} \\ \vdots & \vdots & \ddots & \vdots \\ h_i^{N_T 1} & h_i^{N_T 2} & \cdots & h_i^{N_T N_R} \end{bmatrix}, \tag{2}$$

where h_i^{jk} is the CSI of the i^{th} sub-carrier for the link between the j^{th} transmitted antenna and the k^{th} receiving antenna. The h_i^{jk} is a complex value represented as

$$h_i^{jk} = |h_i^{jk}| e^{j\angle h_i^{jk}}, \tag{3}$$

where $|h_i^{jk}|$ and $\angle h_i^{jk}$ denote amplitude and phase respectively. In this study, we use one transmitting antenna and one receiving antenna. Hence, the CSI measurement matrix contains the frequency response of all the 52 sub-carriers. In the data packets, the real and imaginary values of CSI data are subsequently stored. Subsequently, amplitude and phase components are calculated from the captured CSI data as follows:

$$|h_i^{jk}| = \sqrt{d_{\text{imag}} + d_{\text{real}}}, \tag{4}$$

$$\angle h_i^{jk} = \tan^{-1}\left(\frac{d_{\text{imag}}}{d_{\text{real}}}\right), \tag{5}$$

where d_{imag} and d_{real} denote the imaginary and real parts of the collected CSI data.

CSI in ESP-32 contains the channel frequency responses of 52 active subcarriers. In OFDM, Inverse Fast Fourier Transform (IFFT) and Fast Fourier Transform (FFT) are used on transmitter and receiver sides respectively. The number of inputs in IFFT and FFT should be 2^n. To satisfy this, 12 inactive subcarriers are replaced with 0, which makes a total of 64 subcarriers. We have excluded all the inactive subcarriers in the data processing section, to make the input dimension 256×52.

Noise may be induced during propagation owing to high-frequency environmental noise and multi-path effect. Hence, a Butterworth filter is used to remove the noise. Subsequently, Gaussian smoothing is employed to suppress the small peaks. The samples of the recorded CSI, filtered, and smooth signals are shown in Fig. 6.

The CSI signal is split into smaller dimensions or windows through segmentation to ensure effective use of the hardware resources. Gesture performance time is not unique; hence, proper segmentation is helpful in extracting the feature that will eventually affect the recognition accuracy. It took average 2.5s to 3.1s to perform each gesture and the sampling rate was 100Hz. We could record 255 to 310 data frames for each gesture. We have set different window lengths empirically and found maximum recognition accuracy in the case of window length 256. Therefore, in this study, we use a sliding window size of 256×52, and it is shifted by 128 to ensure a 50% overlap of the signal.

C. DEEP LEARNING MODEL (CSI-DeepNet)

The deep learning model (CSI-DeepNet) consists of two main sections: feature extraction and recognition section as shown

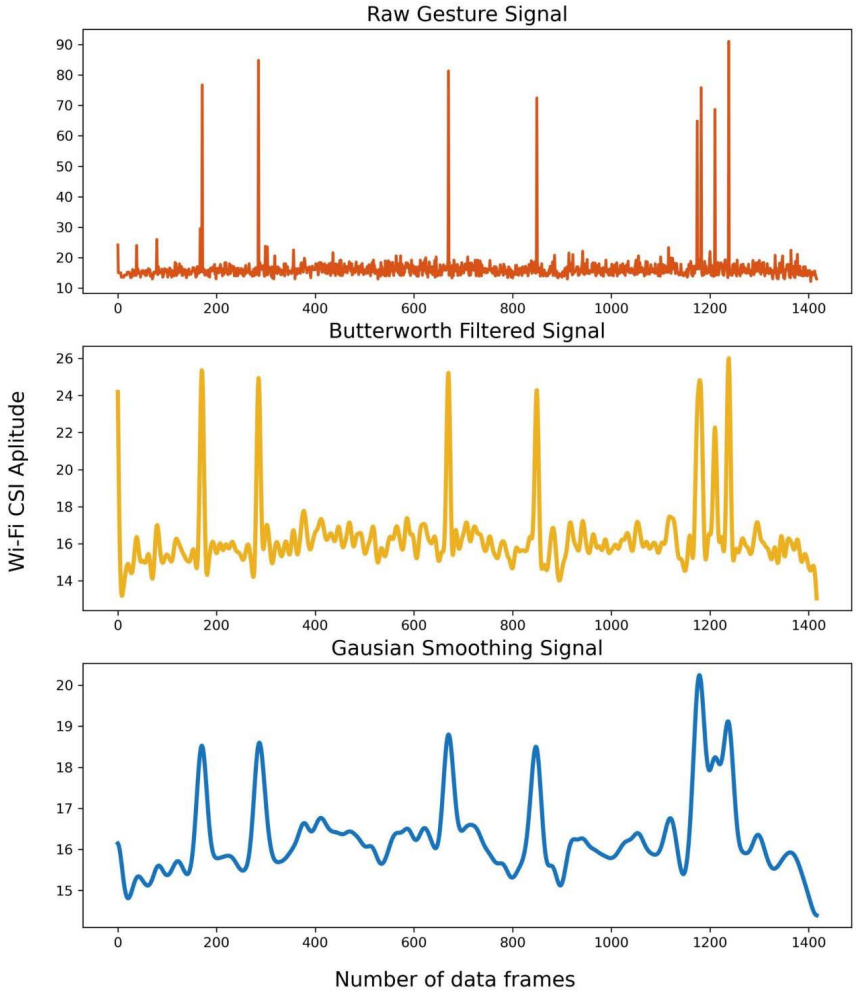

FIGURE 6. Raw, filtered, and smoothing signals of the gesture CSI data.

in Fig. 7. The feature extraction section involves obtaining the fine-grained feature that is used by the subsequent recognition section to detect the exact gestures. In addition, the features extraction section use one 2D CNN followed by batch normalization (BN), ReLU activation, and four layers of dense convolutional (DS-Conv) blocks. Among the layers, two DS-Conv blocks are followed by an FA-block and RB. DS-Conv blocks significantly reduce the number of parameters compared to the conventional CNN-based approach, which is advantageous for low-resource device implementation. However, FA blocks assist in obtaining fine-grained features that are more resilient. Moreover, RB provides features that are able to remove the vanishing gradient removal, increase strength propagation, and support feature reuse. In contrast, the recognition section takes advantage of the global average pooling, dropout, dense, and softmax for recognition. As such, the proposed model uses fewer parameters compared with the existing state-of-the-art approach; thus, its training time is reduced in addition to the testing time with low memory. By doing so, the proposed model is advantageous for implementation in low-resource devices in IoT devices. The features from the feature extraction section are introduced to classifiers in the recognition section to ensure that they are classified into 21 different classes. The model summary of the feature extraction and recognition sections is tabulated in Table 1 and Table 2, respectively.

1) DEPTHWISE SEPARABLE CONVOLUTION (DS-CONV)

To build a lightweight deep learning model, we utilize the power of DS-Conv [24] instead of state-of-the-art CNN architecture. DS-Conv splits the convolution processes into depthwise convolution (D_{conv}) and pointwise convolution (P_{conv}). The D_{conv} utilizes a single convolution kernel in each input channel, whereas a 1×1 convolution operation is conducted on the outcomes from the D_{conv} to combine the outputs using P_{conv}. This approach reduces the computational complexity as well as model size, which is advantageous for low hardware resource device applications. The DS-Conv architecture is illustrated in Fig. 8.

Consider that $\mathbf{K}$ and $\mathbf{x}$ are kernel and feature map, respectively; Subscripts i, j, and m represent the height, width, and depth of the feature map, respectively; and k and l are

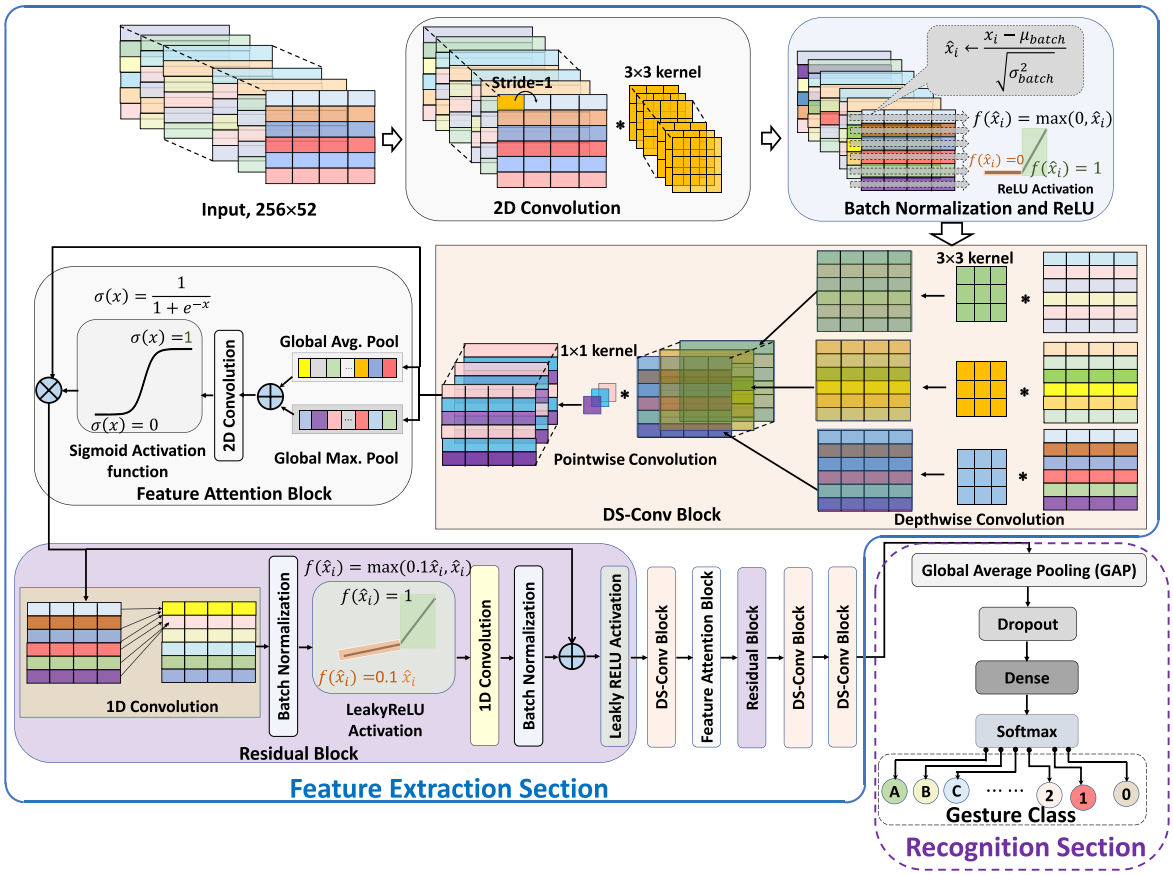

FIGURE 7. Proposed CSI-DeepNet model architecture.

TABLE 1. Summary of the feature extraction block.

Layer Type	Output Shape	No. of Parameters
Conv2d	$256 \times 32 \times 12$	120
BN and ReLU	$256 \times 32 \times 12$	48
DS-Conv	$128 \times 16 \times 32$	712
Feature Attention	$128 \times 16 \times 32$	18
Residual Block	$128 \times 16 \times 32$	10560
DS-Conv	$64 \times 8 \times 64$	2816
Feature Attention	$64 \times 8 \times 64$	18
Residual Block	$64 \times 8 \times 64$	41600
DS-Conv	$32 \times 4 \times 128$	9728
DS-Conv	$16 \times 2 \times 256$	35840

TABLE 2. Summary of the recognition block.

Layer Type	Output Shape	No. of Parameters
GAP	1×256	0
Dropout 0.20	1×256	0
Dense	1×64	16448
Softmax	1×21	1365

the height and width of the kernel, respectively. Depthwise convolution with one filter per input channel (depth) yields

$$D_{\text{conv}}(\mathbf{K}, \mathbf{x})_{i,j,m} = \Sigma_{k,l}\mathbf{K}_{k,l} \cdot \mathbf{x}_{i+k,j+l,m}. \quad (6)$$

However, pointwise convolution can be expressed as

$$P_{\text{conv}}(\mathbf{K}, \mathbf{x})_{i,j} = \Sigma_{m}\mathbf{K}_{m} \cdot \mathbf{x}_{i,j,m}. \quad (7)$$

2) FEATURE ATTENTION (FA) BLOCK

The attention concept is used to enhance the performance of CNNs. Neural network architecture focuses on the local features that fail to show the relationships among local features. The feature attention mechanism helps to reveal the relationship between the different descriptive local features among the two neighbors. To motivate our study based on the previous studies, we have utilized a version of the feature attention block with the average pooling and max pooling. First, the features map obtained from the previous layer underwent average pooling and max pooling operations to obtain the spatial features; subsequently, they are summed up element-wise to obtain the concatenated features. Next, a 2D

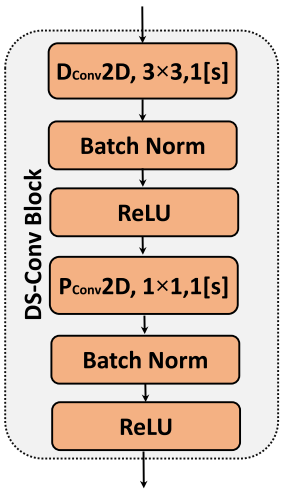

FIGURE 8. DS-Conv model architecture.

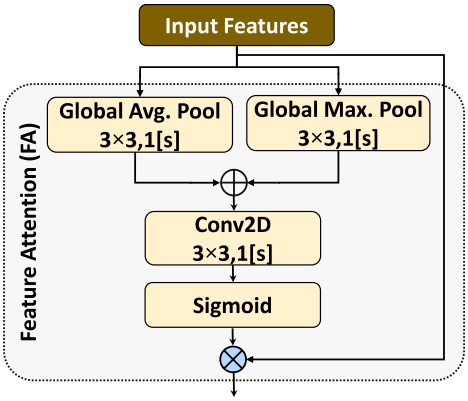

FIGURE 9. Feature attention block architecture.

convolution layer with a sigmoid activation function is used to build the spatial attention feature map. Fig. 9 shows the feature attention block.

Considering, the features $\mathbf{x} \in \mathbb{R}^{F\times S}$ where F is the number of frames, and S is the number of sub-carriers. Variable $\mathbf{x}$ undergoes two different pooling operations concurrently: $\mathbf{x}_{\text{max-pool}} \in \mathbb{R}^{F\times S}$ and $\mathbf{x}_{\text{avg-pool}} \in \mathbb{R}^{F\times S}$, where $\mathbf{x}_{\text{max-pool}}$ denotes max pooling and $\mathbf{x}_{\text{avg-pool}}$ is average pooling. After an element-wise concatenation, a 2D convolution operation is used with a 3×3 single kernel and stride size of 1. However, sigmoid activation function linearizes the output feature map $\mathsf{SA}(\mathbf{x})$. The output features map is matched with input features ($\mathbf{x}$) using element-wise multiplication to obtain finer features $\widetilde{\mathbf{x}}$,

$$\mathbf{x}_{\text{avg-pool}} = \mathsf{AvgPool}(\mathbf{x}), \tag{8}$$

$$\mathbf{x}_{\text{max-pool}} = \mathsf{MaxPool}(\mathbf{x}), \tag{9}$$

$$\mathsf{SA}(\mathbf{x}) = \sigma\left(f^{3\times 3}[\mathbf{x}_{\text{avg-pool}}] \oplus [\mathbf{x}_{\text{max-pool}}]\right), \tag{10}$$

$$\widetilde{\mathbf{x}} = \mathbf{x} \cdot \mathsf{SA}(\mathbf{x}), \tag{11}$$

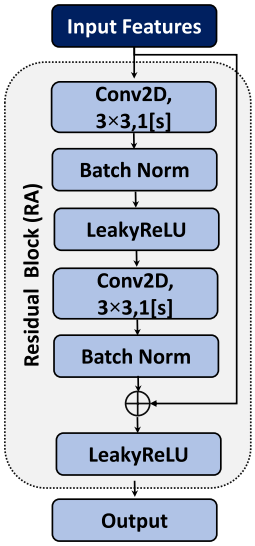

FIGURE 10. Residual block architecture.

where σ represents the sigmoid activation function, $f^{3\times 3}$ denotes a 3×3 single kernel, and $\oplus$ denotes a concatenation operator.

3) RESIDUAL BLOCK (RB)

The conventional deep learning networks utilize a number of convolution layers, followed by the fully connected layers for classification without evaluating the features transaction in a block. Each layer passes its processed data to the next layer, similar to a sequential network. As layer size increased, the vanishing or exploding gradient in the network progressed. RB is used to enhance the gradient propagation, and it enabled the training of the deeper block without gradient vanishing problems. The RB consists of two convolutional layers, BN, and the LeakyReLU function activation layer as shown in Fig. 10. The LeakyReLU activation function layer is performed on the concatenated features from the previous layers with the fresh input features before feeding into the convolution layer.

The functions of the residual layer can be defined as:

$$x' = x \oplus f(x), \tag{12}$$

$$\mathsf{LeakyRelU}(x') = \begin{cases} 1, & \textbf{if } x' > 0 \\ \alpha, & \text{otherwise.}, \end{cases} \tag{13}$$

where x denotes the input feature, $f(x)$ the output of any layer, $\oplus$ the concatenation operator, x' denotes the concatenated inputs, and α denotes the leakage factor.

4) METHODOLOGY

A statistical model is developed in three steps: building, training and validation, and evaluation of the model. A sufficient amount of data diversity is necessary for proper training of a model. However, the inability to set up a proper model hyper-parameter leads to misconceptions. The proposed CSI-DeepNet is verified using the collected alphanumeric

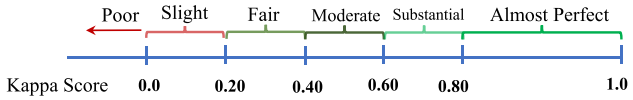

FIGURE 11. Cohen's kappa score interpretation.

gesture dataset. This model is trained for up to 100 epochs with a batch size of 64. An early stop callback for validation loss with 10 epochs of patience is utilized to end the training in case of improvement failure. A desktop computer with an AMD Ryzen 5 5600X 6-core processor of 3.70 GHz and NVIDIA GeForce RTX 3060 12 GB GPU, 500 GB SSD, and 32 GB RAM are utilized to perform the experiment. The network is run using the Keras API supported with TensorFlow as a backend running in a Python environment. First, the network starts with a small learning rate. When the validation accuracy failed to improve in six consecutive epochs, the learning rate is updated by 0.75 times with respect to the previous value. The Adam optimizer [25] is adopted to minimize the error. For the evaluation of the proposed model, three evaluation metrics (accuracy, F_1-score, and k-score) are reported. Accuracy is defined as the total number of correctly identified predictions divided by a total number of predictions produced using the dataset. It is adequate when the target class is well balanced, but inaccurate when the target class is unbalanced. True positive (TP) is a result where the model accurately identifies the positive class, whereas true negative (TN) is a result where the model accurately identifies the negative class, false positive (FP) is a result in which the model incorrectly identifies the positive class, and false negative (FN) is a result in which the model incorrectly identifies the negative class. The mathematical representations of the performance metrics are as follows:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \tag{14}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \tag{15}$$

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}, \tag{16}$$

$$\text{F}_1\text{-score} = 2 \times \frac{\text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}}. \tag{17}$$

The F_1-score represents the harmonic mean of two measures (recall and precision). Its numerical value ranges from 0 to 1, where 0 denotes the worst value, and 1 denotes the best value. Another important metric is Cohen's kappa score, (k-score) which indicates the recognition performance produced by random guessing based on the number of samples in each class [26]. The interpretation of the k-score is shown in Fig. 11.

IV. RESULT AND DISCUSSION

In this section, we present a performance evaluation of our proposed method. The collected alphanumeric hand-gesture datasets are used for the evaluation. A 10-fold cross-validation (CV) technique is followed to train and test the model. The entire dataset is randomly partitioned into ten non-overlapping data of equal size. Subsequently, the data are fitted with the model in an iterative manner. The overall performance is the average of the outcomes achieved in each iteration. The number of epochs is set to 100, and the Adam optimizer [27] is used to update the weights by considering the cross-entropy [28] loss function. The performance of the proposed model is evaluated based on three well-known metrics: accuracy, F_1-score and Cohen's kappa (k-score). The performance evaluation results are summarized in Table 3. The average $\pm$ standard deviation values of accuracy (%), F_1-score, and Cohen's kappa (k-score) are 96.31$\pm$0.28%, 0.97$\pm$0.0042, and 0.96$\pm$0.0067, respectively. The maximum values of accuracy (%), F_1-score, and Cohen's kappa (k-score) are 96.76%, 0.97, and 0.96, respectively. The values are achieved for 3$^{\text{rd}}$ fold. The minimum case of accuracy (%), F_1-score and Cohen's kappa (k-score) are 95.88%, 0.96, and 0.95, respectively. The values are achieved in 1$^{\text{st}}$ fold. Twenty gesture classes, including one steady state (no gesture) in total twenty-one (21) gesture classes are considered for classification. A confusion matrix with a heatmap is presented in Fig. 12 for insight into the accuracy of each class. The diagonal elements represent the average recognition accuracy for each of the 21 classes. Misclassifications occurred because the similarity between gestures and the beginning of certain gestures is identical to that of steady-state gestures. The non-diagonal elements of the confusion matrix indicate the rate of misclassification.

The t-SNE plot helps to evaluate the generalization capabilities of a model. This shows how the model represents data in a high-dimensional feature space. Fig. 13 (a) shows a sample of the data before processing using the classifier. The samples are congested and more challenging to identify. However, 21 well-separated distributions of the data are shown in Fig. 13 (b). The clear and well margin among the 21 classes demonstrated the capability of the classifier to separate the feature space.

The proposed model is trained and evaluated using a 10-fold CV technique. The fold-wise performance results in Table 3 assumes that the highest accuracy is achieved for the 3$^{\text{rd}}$ fold. To observe the training and validation losses, the accuracy and loss curves are shown in Fig. 14. The figures reveal that the proposed model converged within 60 epochs.

To make a performance comparison with other deep learning based approaches, we use four different models: two pre-trained CNNs (ResNet-50 [29], DenseNet-121 [30]), an end-to-end deep learning framework (E2EDLF) [31], and CSI-IANet [32]. Two pre-trained CNNs are tuned via the transfer learning concept using the collected CSI gesture dataset. Zero-padding is applied to adjust the size of each input, and the number of neurons in the last dense layer is set to 21, which is equal to the gesture classes, followed by global average pooling (GAP) and a dense layer. The E2EDLF consists of three blocks of the CNN architecture, with two blocks for the feature extraction phase and one

TABLE 3. Results obtained from 10-fold CV of the proposed CSI-DeepNet model.

Metrics	Fold										Average
	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th	
Accuracy (%)	95.88	96.14	96.76	96.42	96.21	96.34	96.39	96.71	96.04	96.20	96.31±0.28
F_1-score	0.96	0.97	0.97	0.96	0.97	0.97	0.97	0.97	0.97	0.97	0.97±0.0042
Cohen's kappa (k-score)	0.95	0.95	0.96	0.96	0.96	0.96	0.96	0.96	0.95	0.95	0.96±0.0067

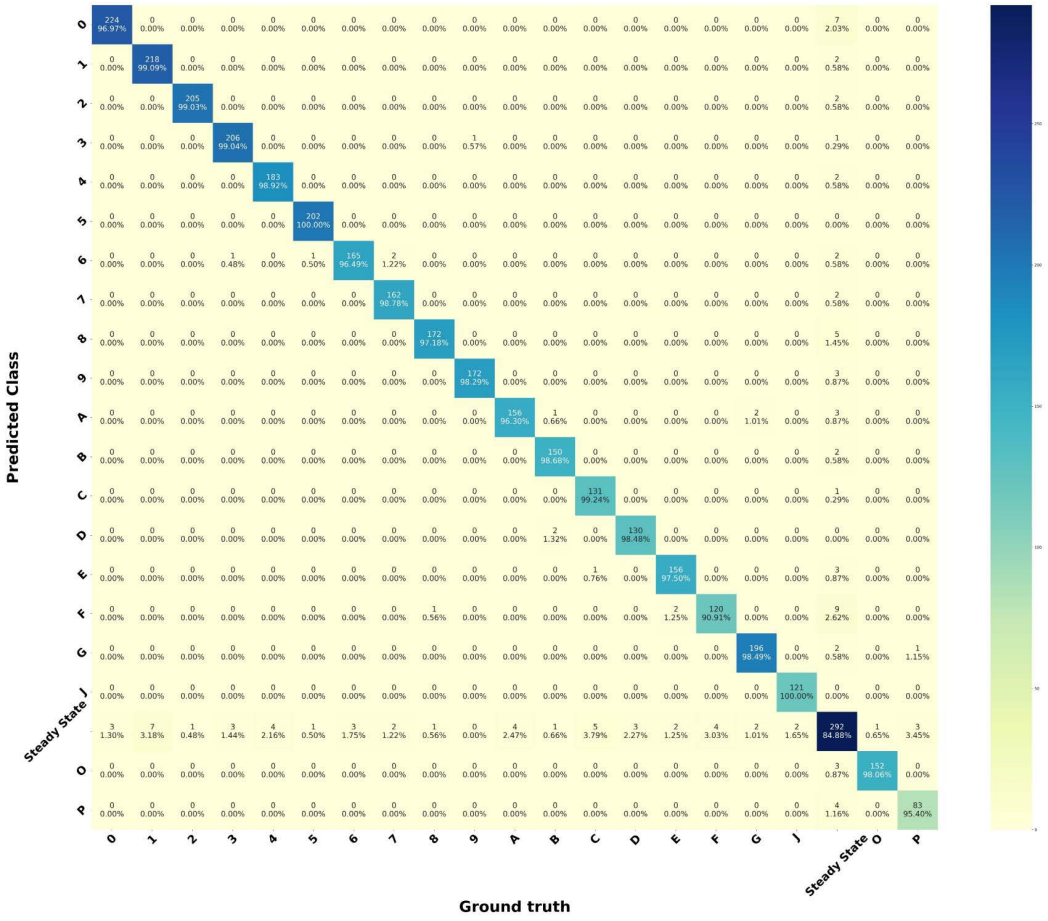

FIGURE 12. Confusion matrix of the proposed CSI-DeepNet model for gesture recognition.

TABLE 4. Performance comparison of CSI-DeepNet model.

Classifier	Accuracy (%)	F_1-score	k-score	Number of trainable Parameter
Pre-trained ResNet-50 [29]	70.38	0.70	0.69	264,987
Pre-trained DenseNet-121 [30]	69.13	0.69	0.68	133,915
E2EDLF [31]	84.30	0.84	0.83	972,321
CSI-IANet [32]	90.5	0.91	0.89	516,321
Proposed CSI-DeepNet	96.31	0.97	0.96	119,273

for the recognition phase. Each CNN block in the featured extraction phase is followed by a BN layer and rectified linear unit (ReLU) activation layer. The third block consists of a flattened layer, a fully connected layer, and a softmax layer. Another CNN-based approach is CSI-IANet, which utilizes a modified inception CNN with a feature-attention mechanism to classify the CSI signal. All four models are evaluated using the same training and testing dataset using a 10-fold CV

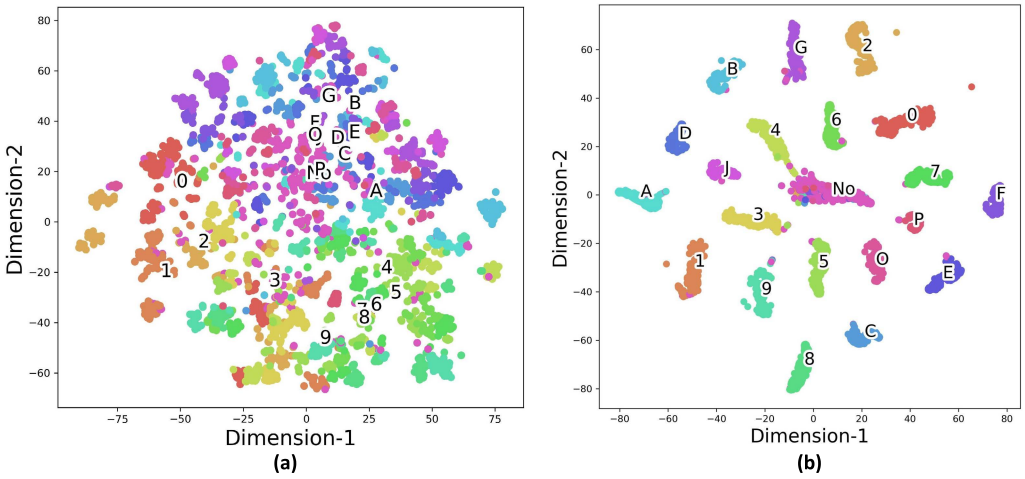

FIGURE 13. Two-dimensional t-SNE visualization of the testing data a) before and b) after the prediction.

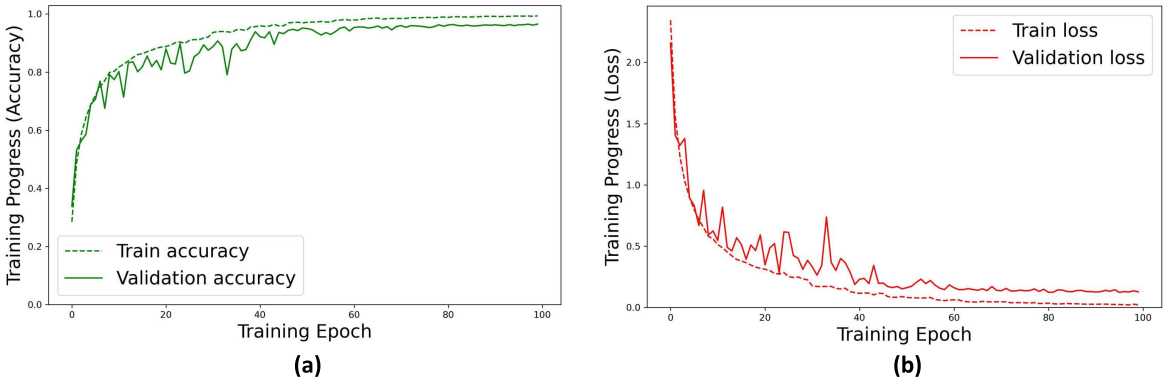

FIGURE 14. Training and validation a) accuracy and b) loss curve of proposed CSI-DeepNet model.

procedure. The proposed model, along with four state-of-the-art techniques, is compared based on the percentage of recognition accuracy, number of trainable parameters, training time, and recognition time.

A performance comparison between the proposed CSI-DeepNet and other state-of-the-art techniques is presented in Table 4. The average recognition accuracies computes across all 21 gesture classes for the pre-trained ResNet-50 and DenseNet-121, E2EDLF, and CSI-IANet are 70.38%, 69.13%, 84.30%, and 90.50% respectively. However, the average recognition F_1-score computes across all gesture classes for the pre-trained ResNet-50 and DenseNet-121, E2EDLF, and CSI-IANet are 0.70, 0.69, 0.84, and 0.91, respectively. Furthermore, the average k-scores computes across all gesture classes for the pre-trained ResNet-50 and DenseNet-121, E2EDLF, and CSI-IANet are 0.69, 0.68, 0.83, and 0.89, respectively. The number of trainable parameters for the pre-trained ResNet-50 and DenseNet-121, E2EDLF, and CSI-IANet are 264987, 133915, 972321, and 516321, respectively. The proposed CSI-DeepNet obtaines a recognition accuracy, F_1-score, k-score, and number of trainable parameters of 96.31%, 0.97, 0.96, and 119273 respectively. Compared with existing studies discusses in the literature, our proposed model exhibits superior performance to any existing work in terms of gesture recognition from CSI data. The performance analysis of the proposed CSI-DeepNet model demonstrates that it outperforms the existing best model, CSI-IANet, by 6% in terms of accuracy, F_1-score, and k-score. In the case of computational complexity, the proposed model utilizes four times less than that of the existing best model. This improvement may be due to the new architecture of the proposed model and optimal hyper-parameter selection. Therefore, our proposed model can use for gesture recognition and is suitable for low-resource device applications.

The number of trainable parameters in the model relates with the computational complexity. The fewer trainable parameters, the lesser computational complexity. We have evaluated the complexity of a model based on the training and recognition time. Therefore, a runtime comparison of the

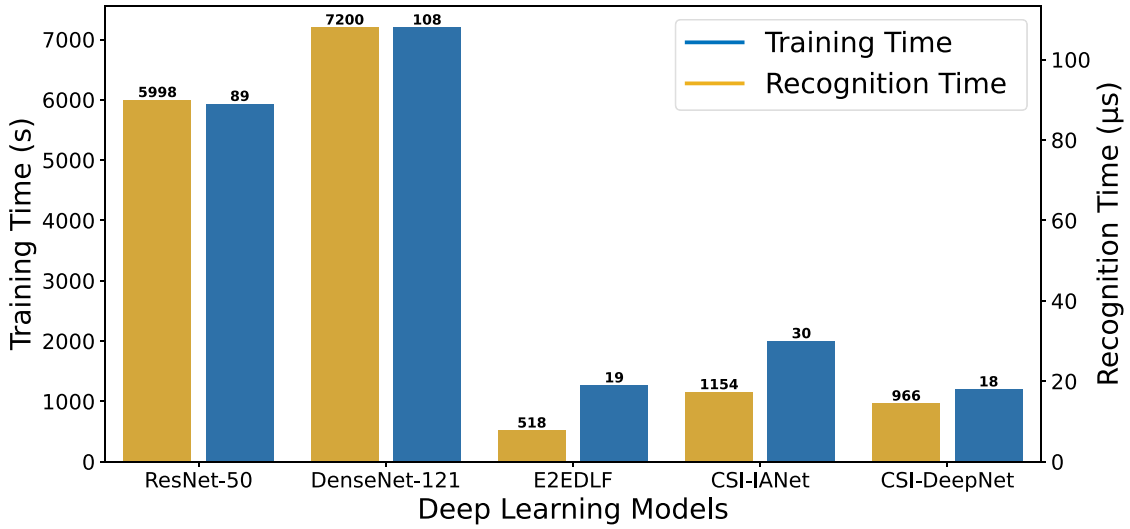

FIGURE 15. Runtime comparison between different deep learning model.

proposed CSI-DeepNet with other state-of-the-art techniques is performed. The training and recognition times of the proposed CSI-DeepNet along with four other state-of-the-art techniques (pre-trained ResNet-50 and DenseNet-121, E2EDLF, and CSI-IANet) are shown in Fig 15. Two pre-trained models ResNet-50 and DenseNet-121 requires additional training time (5998 s and 7200 s, respectively) and recognition time (89μs and 108μs, respectively). Although E2EDLF can train (518 s) and recognize (19μs) in less time owing to the utilization of the less depth model, it suffers from low accuracy (84.30%). Previous studies on CSI-IANet reports moderate training time (1154 s) and recognition time (30μs) with moderate accuracy (90.5%). The proposed CSI-DeepNet has a shorter training time (966 s) and recognition time (18μs), and better accuracy (96.31%) than the CSI-IANet.

V. CONCLUSION

A device-free gesture recognition system plays an important role in satisfying user privacy and comfort. In this study, a low-power SoC has been used for the first time to capture 20 alphanumeric hand gestures among which ten are numbers and ten letters. Ten people are involved in recording 1,800 trails of 20 alphanumeric hand gestures. Concurrently, a lightweight heterogeneous deep learning model, CSI-DeepNet using DS-Conv, FA block, and RB has utilized for the extraction and classification of features. DS-Conv with the FA block and RB help to learn fine-grained features while significantly utilizing fewer model parameters without sacrificing the recognition accuracy. Moreover, RB increases the propagation of gradients and allows the training of deeper CNNs, mitigating gradient vanishing problems. The average accuracy of 96.31% is achieved for the classification of 21 gestures, which outperformed the two pre-trained CNNs and two state-of-the-art deep learning CSI-based classifiers. Overall, the proposed CSI-DeepNet has utilized fewer parameters, as well as training and recognition time. Owing to the use of low-power SoCs in data collection and lightweight deep learning models, this system can be applied to low-resource devices that ensure large-scale deployment. Annotating the data for large systems is a tedious and complicated task. In the future, we will expand our system to include a greater number of different gestures in multiple antennas scenarios and adopt a semi-supervised learning-based solution to handle data annotation challenges.

REFERENCES

[1] J. K. Aggarwal and M. S. Ryoo, "Human activity analysis: A review," *ACM Comput. Surv.*, vol. 43, no. 3, pp. 1–43, 2011.
[2] D. Caicedo, A. Pandharipande, and F. M. J. Willems, "Detection performance analysis of an ultrasonic presence sensor," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, May 2013, pp. 2780–2784.
[3] O. D. Lara and M. A. Labrador, "A survey on human activity recognition using wearable sensors," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 3, pp. 1192–1209, 3rd Quart., 2013.
[4] J. L. Martinez Flores, S. Sai Srikant, B. Sareen, and A. Vagga, "Performance of RFID tags in near and far field," in *Proc. IEEE Int. Conf. Pers. Wireless Commun. (ICPWC)*, Jan. 2005, pp. 353–357.
[5] X. Li, Y. He, and X. Jing, "A survey of deep learning-based human activity recognition in radar," *Remote Sens.*, vol. 11, no. 9, p. 1068, Jan. 2019.
[6] F. Adib and D. Katabi, "See through walls with WiFi!" in *Proc. ACM SIGCOMM Conf. SIGCOMM*, Aug. 2013, pp. 75–86.
[7] Y. Moustafa, M. Mah, and A. Agrawala, "Challenges: Device-free passive localization for wireless environments," in *Proc. 13th Annu. ACM Int. Conf. Mobile Comput. Netw.*, 2007, pp. 222–229.
[8] D. Halperin, W. Hu, A. Sheth, and D. Wetherall, "Tool release: Gathering 802.11n traces with channel state information," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 41, no. 1, p. 53, Jan. 2011.
[9] Y. Xie, Z. Li, and M. Li, "Precise power delay profiling with commodity Wi-Fi," *IEEE Trans. Mobile Comput.*, vol. 18, no. 6, pp. 1342–1355, Jun. 2019.
[10] S. Sigg, M. Scholz, S. Shi, Y. Ji, and M. Beigl, "RF-sensing of activities from non-cooperative subjects in device-free recognition systems using ambient and local signals," *IEEE Trans. Mobile Comput.*, vol. 13, no. 4, pp. 907–920, Apr. 2014.

[11] H. Abdelnasser, M. Youssef, and K. A. Harras, "WiGest: A ubiquitous WiFi-based gesture recognition system," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, Apr. 2015, pp. 1472–1480.
[12] T. Sheng and J. Yang, "WiFinger: Leveraging commodity WiFi for fine-grained finger gesture recognition," in *Proc. 17th ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, 2016, pp. 201–210.
[13] L. Tao, C. Shi, P. Li, and P. Chen, "A novel gesture recognition system based on CSI extracted from a smartphone with nexmon firmware," *Sensors*, vol. 21, no. 1, p. 222, 2020.
[14] *Espressif*. Accessed: Aug. 28, 2022. [Online]. Available: https://www.espressif.com/en/ and hardware/esp32/overview/
[15] M. H. Kabir, M. R. Hoque, K. Thapa, and S.-H. Yang, "Two-layer hidden Markov model for human activity recognition in home environments," *Int. J. Distrib. Sensor Netw.*, vol. 12, no. 1, Jan. 2016, Art. no. 4560365.
[16] Z. Tian, J. Wang, X. Yang, and M. Zhou, "WiCatch: A Wi-Fi based hand gesture recognition system," *IEEE Access*, vol. 6, pp. 16911–16923, 2018.
[17] X. Dang, Y. Liu, Z. Hao, X. Tang, and C. Shao, "Air gesture recognition using WLAN physical layer information," *Wireless Commun. Mobile Comput.*, vol. 2020, pp. 1–14, Aug. 2020.
[18] Q. Bu, G. Yang, X. Ming, T. Zhang, J. Feng, and J. Zhang, "Deep transfer learning for gesture recognition with WiFi signals," *Pers. Ubiquitous Comput.*, vol. 26, no. 3, pp. 543–554, Jun. 2022.
[19] P. Hu, C. Tang, K. Yin, and X. Zhang, "WiGR: A practical Wi-Fi-based gesture recognition system with a lightweight few-shot network," *Appl. Sci.*, vol. 11, no. 8, p. 3329, Apr. 2021.
[20] X. Ding, T. Jiang, W. Xue, Z. Li, and Y. Zhong, "A new method of human gesture recognition using Wi-Fi signals based on XGBoost," in *Proc. IEEE/CIC Int. Conf. Commun. China (ICCC Workshops)*, Aug. 2020, pp. 237–241.
[21] Y. Gu, X. Zhang, Y. Wang, M. Wang, H. Yan, Y. Ji, Z. Liu, J. Li, and M. Dong, "WiGRUNT: WiFi-enabled gesture recognition using dual-attention network," *IEEE Trans. Human-Mach. Syst.*, vol. 52, no. 4, pp. 736–746, Aug. 2022.
[22] H. F. Thariq Ahmed, H. Ahmad, and C. V. Aravind, "Device free human gesture recognition using Wi-Fi CSI: A survey," *Eng. Appl. Artif. Intell.*, vol. 87, Jan. 2020, Art. no. 103281.
[23] *ESP-IDF Programming Guide*. Accessed: Aug. 28, 2020. [Online]. Available: https://docs.espressif.com/projects/esp-idf/en/latest/api-guides/ and https://wifi.html?highlight
[24] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*.
[25] Y. Wang, K. Wu, and L. M. Ni, "WiFall: Device-free fall detection by wireless networks," *IEEE Trans. Mobile Comput.*, vol. 16, no. 2, pp. 581–594, Feb. 2016.
[26] Y.-M. Fang, H.-L. Feng, J. Li, and G.-H. Li, "Stress wave signal denoising using ensemble empirical mode decomposition and an instantaneous half period model," *Sensors*, vol. 11, no. 8, pp. 7554–7567, Aug. 2011.
[27] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.
[28] K. Janocha and W. Marian Czarnecki, "On loss functions for deep neural networks in classification," 2017, *arXiv:1702.05659*.
[29] F. Wang, M. Jiang, C. Qian, S. Yang, C. Li, H. Zhang, X. Wang, and X. Tang, "Residual attention network for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3156–3164.
[30] M. M. Yapici, A. Tekerek, and N. Topaloglu, "Performance comparison of convolutional neural network models on GPU," in *Proc. IEEE 13th Int. Conf. Appl. Inf. Commun. Technol. (AICT)*, Oct. 2019, pp. 1–4.
[31] R. Alazrai, M. Hababeh, B. A. Alsaify, M. Z. Ali, and M. I. Daoud, "An end-to-end deep learning framework for recognizing human-to-human interactions using Wi-Fi signals," *IEEE Access*, vol. 8, pp. 197695–197710, 2020.
[32] M. H. Kabir, M. H. Rahman, and W. Shin, "CSI-IANet: An inception attention network for human-human interaction recognition based on CSI signal," *IEEE Access*, vol. 9, pp. 166624–166638, 2021.

M. HUMAYUN KABIR (Member, IEEE) received the B.Sc. and M.Sc. degrees from Islamic University, Kushtia, Bangladesh, in 2001 and 2003, respectively, and the Ph.D. degree from the Department of Electronic Engineering, Kwangwoon University, Seoul, Republic of Korea, in 2016. He is currently a Postdoctoral Researcher with Ajou University in Suwon, South Korea. Prior to joining Ajou University, he was a Faculty Member at the Department of Electrical and Electronic Engineering, Islamic University, Kushtia. He is acting as a Reviewer for several reputed journals specially IEEE TRANSACTIONS ON EMERGING TOPICS IN COMPUTATIONAL INTELLIGENCE, IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, and *IET Networks*. His research interests include the IoT, satellite communication, machine learning, and deep-learning-based signal processing. He was awarded for the Best Paper in the 32nd Joint Conference on Communication and Information (JCCI) and the Best Workshop Paper in *International Conference on ICT Convergence* (ICTC), in 2022.

MD. ALI HASAN received the B.Sc. degree in electrical and electronic engineering from Islamic University, Kushtia, Bangladesh, in 2020. He is currently pursuing the M.S. degree at Ajou University, Suwon, South Korea. His main research interests include machine learning, deep learning-based signal processing, and the IoT.

WONJAE SHIN (Senior Member, IEEE) received the B.S. and M.S. degrees from the Korea Advanced Institute of Science and Technology, in 2005 and 2007, respectively, and the Ph.D. degree from the Department of Electrical and Computer Engineering, Seoul National University (SNU), South Korea, in 2017. From 2007 to 2014, he was a member of Technical Staff with the Samsung Advanced Institute of Technology and Samsung Electronics Company Ltd., South Korea, where he contributed to next-generation wireless communication networks, especially for 3GPP LTE/LTE-advanced standardizations. From 2016 to 2018, he was a Visiting Scholar and a Postdoctoral Research Fellow at Princeton University, Princeton, NJ, USA. He is currently an Assistant Professor with the Department of Electrical and Computer Engineering, Ajou University, Suwon, South Korea. Prior to joining Ajou University, he was a Faculty Member at Pusan National University, Busan, South Korea, from 2018 to 2021. His research interests include the design and analysis of future wireless communications, such as interference-limited networks, and machine learning for wireless networks. He was awarded the Fred W. Ellersick Prize and the Asia–Pacific Outstanding Young Researcher Award from the IEEE Communications Society, in 2020; the ICTC Best Workshop Paper Award, in 2022; the Best Ph.D. Dissertation Award from SNU, in 2017; the Gold Prize from the IEEE Seoul Section Student Paper Contest, in 2014; and the Award of the Ministry of Science and ICT of Korea in IDIS-Electronic News ICT Paper Contest, in 2017. He was a co-recipient of the SAIT Patent Award, in 2010, Samsung Journal of Innovative Technology, in 2010, the Samsung Human Tech Paper Contest, in 2010, and the Samsung CEO Award, in 2013. He was recognized as an Exemplary Reviewer by the IEEE WIRELESS COMMUNICATIONS LETTERS, in 2014, and the IEEE TRANSACTIONS ON COMMUNICATIONS, in 2019. He was also awarded several fellowships, including the Samsung Fellowship Program, in 2014, and the SNU Long Term Overseas Study Scholarship, in 2016. He is currently an Editor for the IEEE OPEN JOURNAL OF COMMUNICATIONS SOCIETY.