Citation:
F. Deng, E. Jovanov, H. Song, W. Shi, Y. Zhang and W. Xu, "WiLDAR: WiFi Signal-Based Lightweight Deep Learning Model for Human Activity Recognition," in IEEE Internet of Things Journal, doi: 10.1109/JIOT.2023.3294004.

DOI:
https://doi.org/10.1109/JIOT.2023.3294004

WiLDAR: WiFi Signal-Based Lightweight Deep Learning Model for Human Activity Recognition

Fuxiang Deng, Emil Jovanov, *Fellow, IEEE*, Houbing Song, *Fellow, IEEE*, Weisong Shi, *Fellow, IEEE*, Yuan Zhang*, *Senior Member, IEEE*, Wenyao Xu, *Senior Member, IEEE*

***Abstract*—In recent years, the WiFi channel state information (CSI) has been increasingly used for human activity recognition (HAR) during activities of daily living, because of non-intrusiveness and privacy preserving properties. However, most previous works require complex processing of CSI signals, and the large number of classification network parameters significantly increases the recognition time and deployment costs. Accordingly, a WiFi signal based lightweight deep learning (WiLDAR) network is developed in this study to ensure systematic operation on edge computing devices. We combine the random convolution kernel with deep separable convolution and residual structure, so that WiLDAR can easily extract CSI signal features without filtering and denoising. The parameter number and training time of WiLDAR are thus much less than those of previous neural networks. In addition, a tiny HAR system using only Raspberry Pi and router is implemented. Experiments verify that WiLDAR can achieve real-time HAR on IoT devices, which makes HAR deployment more convenient. We test WiLDAR on three different fine-grained action datasets to achieve 99%, 93.5% and 97.5% recognition accuracy, respectively. The demonstrated learning capability of WiLDAR makes it an excellent option for the remote HAR.**

***Index Terms*—human activity recognition, WiFi sensing, Channel state information (CSI), IoT, neural network, Edge Computing.**

I. Introduction

Human activity recognition (HAR) is traditionally used for monitoring of the elderly and chronic patients in their homes. Recently, activity monitoring is increasingly important as integral part of Internet of Things (IoT) in smart homes and offices [1]. HAR, can facilitate remote control of smart home appliances, security monitoring for special populations, and smart health monitoring [2].

The need for unobtrusive and privacy preserving monitoring of activity resulted in three types of HAR implementation: vision-based [3], [4], wearable device-based [5], [6] and wireless sensing-based methods [7], [8].

The vision-based approach usually relies on high-resolution camera or infrared camera, which allows for unobtrusive continuous monitoring [3], [4]. However, the camera is highly susceptible to the effect of ambient lighting and affected by obstructed view of the subjectys. Wearable devices-based method include inertial sensors with accelerometers, gyroscope, magnetometers, and physiological sensors (e.g. electromyogram for monitoring of muscle activity) [5]. However, these methods require the subjects to regularly wear the sensors, which exacerbates the physical burden and could be inconvenient for some users, such as the elderly and the children.

With respect to wireless sensing-based methods, several technologies have been developed, including radar technology [7], RFID technology [9], and WiFi technology [10], [11], among others. Radar sensors are known for their robustness, interference immunity, and wide detection range, but their high deployment costs, bulky devices, and high power consumption can be drawbacks. In the context of HAR, RFID can be employed to track individuals and their movements by attaching small RFID tags to clothing or accessories. However, this method may not be suitable for motion detection unless special reader designs are available. In contrast, pervasive deployment of WiFi wireless networks make possible, inexpensive use of existing devices and signals, without additional overhead. WiFi Channel State Information (CSI) signals have more propagation channels, and each channel works in a different frequency band, which makes it easy for us to use algorithms to reject the channels that are subject to more interference and ensure the overall reliability of the signal. Furthermore, WiFi based HAR is non-intrusive and protects privacy of users, as the most significant concerns of users.

The basic principle of passive WiFi monitoring is to monitor changes in the WiFi signal influenced by human movement in the propagation path of the signal. Therefore, real-time monitoring of signals in the physical network layer provides information of human activity. Moreover, development of new devices and tools [12]–[14] allow acquisition of the CSI using commercial WiFi devices. Gradually, the application of CSI in HAR has also evolved from coarse-grained actions [15], such as running, jumping, to fine-grained actions, such as identification [16], breathing [17] and even sleep monitoring [18].

Nevertheless, within the field of CSI, several challenges persist. These challenges encompass high signal dimensionality, intricate pre-processing procedures, and the absence

The authors acknowledge supports from National Natural Science Foundation of China (62172340). (Corresponding author: Yuan Zhang.)

Fuxiang Deng and Yuan Zhang are with the College of Electronic and Information Engineering, Southwest University, Chongqing, China (e-mail: yuanzhang@swu.edu.cn).

Emil Jovanov is with the Department of Electrical and Computer Engineering, University of Alabama in Huntsville, USA (e-mail:emil.jovanov@uah.edu).

Houbing Song is with the Department of Information Systems, University of Maryland, Baltimore County (UMBC), Baltimore, MD 21250 USA (email: h.song@ieee.org)

Weisong Shi is with the Department of Computer and Information Sciences, University of Delaware, Newark, DE 19716, USA (e-mail:weisong@udel.edu).

Wenyao Xu is with the Department of Computer Science and Engineering, University at Buffalo, The State University of New York, Buffalo, NY 14261 USA (e-mail: wenyaoxu@buffalo.edu).

of lightweight models. For dimensional processing of CSI, a commonly used approach involves performing principal component analysis (PCA) and subsequently removing the first component, which is typically associated with higher noise levels [19]. However, this method may lead to a loss of relevant action information in cases where the actions have small amplitudes. In addition, due to environmental interference, refraction of signal transmission process, and lack of synchronization between transceivers, the CSI signal often needs preprocessing such as filtering, downscaling, and outlier removing. Moreover, extracting meaningful features from the CSI signal is a challenging task which demands specialized expertise and algorithm development.

In order to solve the problems just mentioned, a WiFi signal based lightweight deep learning (WiLDAR) neural network combining a random convolution kernel, a residual block and a depthwise separable convolution is proposed. Accordingly, our contributions are summarized in three aspects.

1) We propose a neural network captioned WiLDAR consisting of two blocks: a) *the feature extraction block*, updated from the MiniRocket algorithm to achieve fast feature extraction of the original CSI signal without parameter learning and back propagation, and b) *the learning block* consists of a residual module combined with a depthwise separable convolution, which reduces the number of network parameters and decreases the risk of overfitting. Overall, WiLDAR is a lightweight network with no pre-processing, high learning capability, and simple structure.
2) By designing random convolution kernels with different sizes, we can achieve automatic extraction of features with different frequencies by using only a single layer of the random convolutional network, which well corresponds to the frequency differences of various activities and greatly improves the model recognition capability and interpretability. The diversity of extraction scales also allows us to perform simple fusion of subcarriers or multiple channels while preserving the amount of input information, and avoiding data redundancy and complex subcarrier selection algorithm design.
3) We tested WiLDAR on three different fine-grained action datasets, all of which showed a significant improvement in test accuracy. In addition, we also implemented our algorithm on Raspberry Pi. This greatly reduced the space and expense required for practical deployment, and also demonstrated the feasibility of integrating CSI collection and HAR recognition algorithms into IoT devices.

The rest of the paper is organized as follows. Section II describes the related works. Section III presents the details of our proposed method. Section III-A introduces the signal preprocessing. Section III-B introduces the system architecture. Section III-C introduces the design of a tiny HAR system. Then the performance of the proposed neural network is evaluated in Section IV. Section V concludes the paper with an outlook.

II. Related Work

HAR research based on CSI signals can be broadly divided into two categories: *signal based* systems utilize feature engineering for signal feature extraction , and *deep learning based* systems use signal representation generated by the deep learning network. We summarize the comparison of related work in Table I

Signal based: In [20], the preprocessed CSI signals were divided into measurement matrices in the time and frequency domains. Coherence histograms representing the feature distribution with self-organizing feature map and softmax regressionbased are used for classification. In Wifinger [21], the authors designed a fine signal denoising module to combine CSI subcarriers into feature vectors for Dynamic Time Warping (DTW) and K-NearestNeighbor (KNN) classification. Wang in [22] extracted the duration and velocity of human motion using discrete wavelet transform. The features extracted from each movement were then modeled as a Hidden Markov Model (HMM), which ignored the differences in movements between individuals to focus only on the differentiation of movement categories.

All these methods design a preprocessing and feature engineering for CSI signals, and a simple classifier is used for the identification of activity. However, they all require the design of complicated feature extraction procedure, which increases the development effort and requires significant expertise, and decreases the scalability.

Deep Learning based: In WiSDAR [10], a comprehensive framework is proposed, leveraging the synergistic capabilities of Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM). This framework successfully integrates hidden features derived from both temporal and spatial dimensions and finally achieves the classification of actions. Zhang in [11] proposed a Deep Q-Network (DQN) network for data annotation, and designed a multi-sensor data fusion algorithm to generate sequential motion data, and finally achieved classification by LSTM. In [23], in order to solve the problem that segmentation of CSI action samples depends heavily on the threshold, Xiao et al. designed a Convolutional Neural Network (CNN) to transform the segmentation problem into a classification problem. Moreover, Dempster [24] proposed a MiniRocket algorithm for fast feature extraction of signals. Zou [25] verified that adjusting the activation function has an impact on the learning ability of the network. Bergstra [26] proposed a Tree-structured Parzen Estimator (TPE) for network parameter searching to simplify algorithm development.

Deep learning based methods enable automatic feature extraction and higher scalability. However, the correlation network is less interpretable, very complex, has large number of parameters, and requires large data sets for training.

III. Proposed Method

In this section we present the specific structure of WiLDAR. Firstly we introduce the data preprocessing and the general framework of the proposed network. Then the feature extraction module and the classification module in WiLDAR will be

TABLE I
COMPARISON OF RELATED WORK

Categories	Reference	Model	Task	Accuracy
Signal based	Wang et al. [20]	multi-domain features	Gesture recognition	0.89
	WiFinger [21]	DWT	Sign language recognition	0.90
	CARM [22]	HMM	Human activity recognition	0.96
Deep Learning based	WiSDAR [10]	CNN+LSTM	Human activity recognition	0.96
	Zhou et al. [11]	DNQ+LSTM	Human activity recognition	0.96
	DeepSeg [23]	CNN	Human activity recognition	0.95

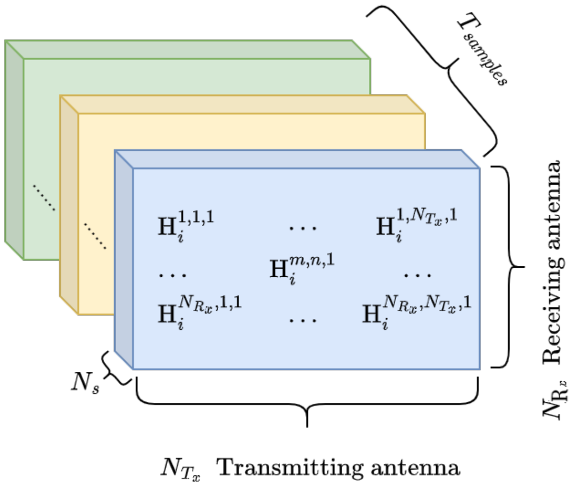

T samples
$H_i^{1,1,1}$
$H_i^{1,N_{T_x},1}$
$H_i^{m,n,1}$
$H_i^{N_{R_x},1,1}$
$H_i^{N_{R_x},N_{T_x},1}$
N_s
N_{R_x} Receiving antenna
N_{T_x} Transmitting antenna

Fig. 1. CSI dimensional diagram. 'N_S' represents the number of subcarriers, '$T_{samples}$' is the time dimension.

TABLE II
ACCURACY PERFORMANCE OF DIFFERENT NETWORKS BEFORE AND AFTER DATA FUSION

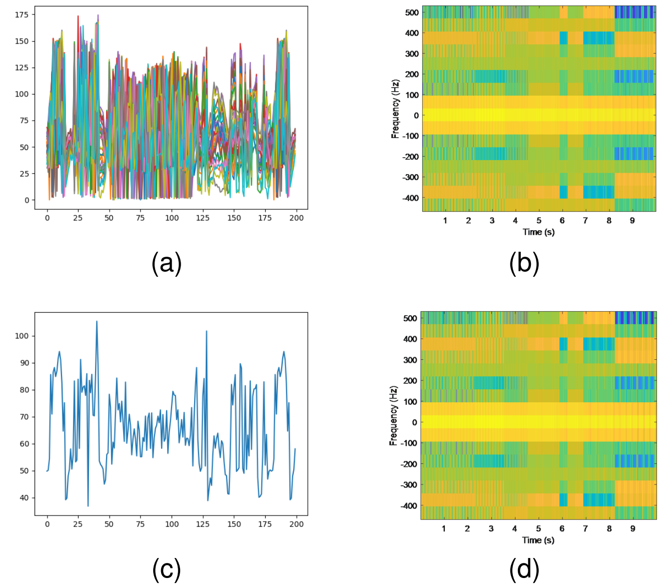
Method	Accuracy[†]	Accuracy[‡]
LSTM	0.550	0.410
GRU	0.805	0.669
WiLDAR	0.866	0.863

[†] Use of unfused data
[‡] Use of fused data.

analyzed. Finally, we present the implementation of the HAR system.

A. Data Downscaling

Given the CSI data set has N samples $\{\mathcal{C}_n\}_{n=1}^{N}$, $C \in \mathbb{R}^{T_x \times R_x \times N_s \times T}$, where the dimensions represent the number of transmitting (T_x) , receiving antennas (R_x), subcarriers (N_s) and time (T) , as shown in the Fig. 1. To solve the excessive dimensionality problem, complex subcarrier selection or fusion algorithms need to be designed, because simply merging often results in information loss. However WiLDAR is able to reconstruct different feature patterns to the fused signal by designing multi-scale convolutional kernels. For example, high-frequency action features are restored using smaller-sized convolutional kernels, while, low-frequency actions are restored using larger-sized convolutional kernels. Therefore, WiLDAR requires only simple average of the input signal's subcarriers to reduce data redundancy while maintaining the information amount, as presented is



(a)
(b)
(c)
(d)

Fig. 2. WiLDAR feature extraction for unfused and fused signals. (a) and (c) represent the unfused and fused CSI signals. (b) and (d)represent the corresponding time-frequency features extracted by WiLDAR, respectively.

$$C_{In} = \frac{1}{n} \sum_{i=1}^{n} C^{T_x \times R_x \times N_i \times T} \tag{1}$$

where C_{In} is the input signal after dimensionality reduction and n represents the number of subcarriers. Fig. 2 demonstrates that the features extracted from the fused signal are almost indistinguishable from the unfused signal, and have no significant effect on the experimental performance. Furthermore, in Table II we also compare the accuracy of different

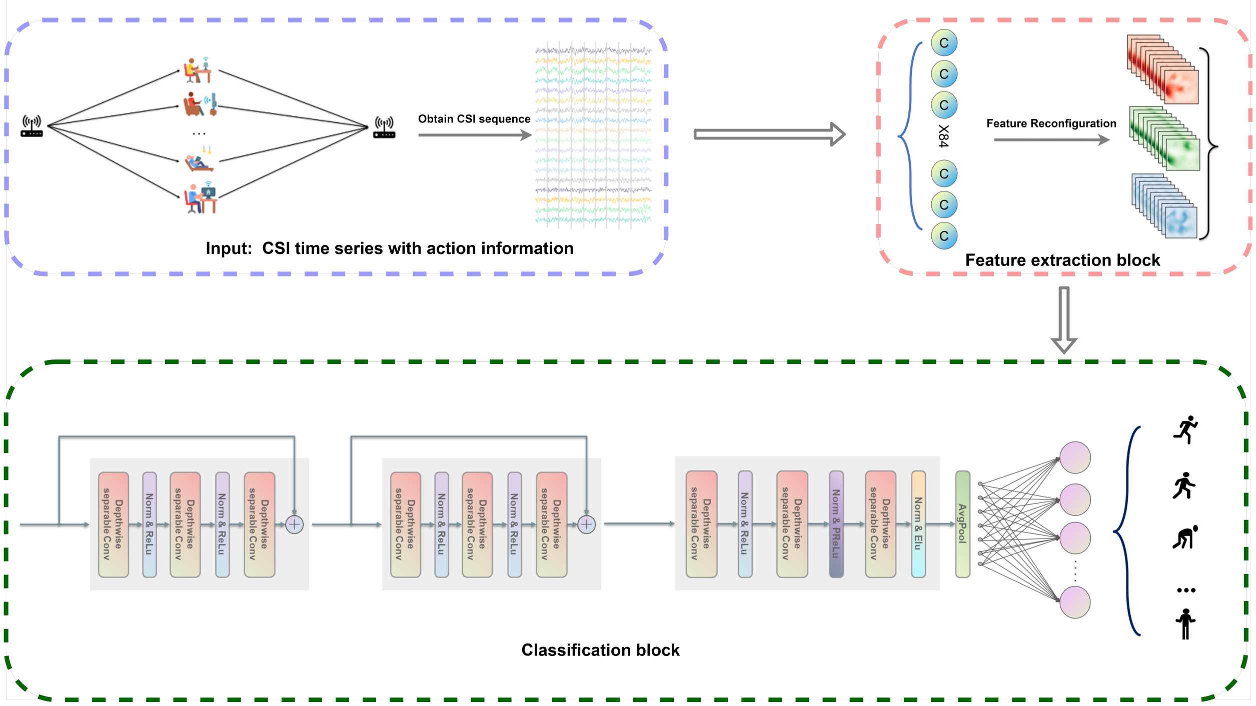


Fig. 3. Architecture of the WiLDAR. The convolution is followed by LReLU, PReLU, and ELU activation functions, respectively.

networks affected by the fused data. For convenience the experiment epoch was set to 100. The results indicate that the signals were simply fused, which causes information loss and thus makes the recognition accuracy decrease. In contrast, WiLDAR undergoes multi-scale feature reconstruction, and there is no significant variation in terms of accuracy before and after the fusion.

In addition, please be noted that all processing in this place is only for adjusting the dimensionality of the signal input, not for PCA dimensionality reduction, Hampel outlier removal, and filtering and denoising operations as recommended in other works [19]. This greatly reduces the pre-processing effort on the CSI signal and avoids the need to design the corresponding signal-denoising works for different application areas.

B. System Architecture

The overall pipeline of WiLDAR consists of a feature extraction module and a classification module is shown in Fig. 3. The feature extraction module further consists of a random convolution kernel that performs multi-scale feature extraction on the input CSI sequence. Then, the extracted features are recalibrated and relearned using a classification module which combines residual structure and depthwise separable convolution. Ultimately, the fast recognition of CSI action sequences is achieved by the combination of the two modules.

Feature Extraction: Our feature extraction module WiRocket, updated from the MiniRocket [24], consists of 84 random convolution kernels of the same size, different weights, and different dilation. In summary, the WiRocket algorithm flow is shown in Algorithm 1.

Algorithm 1 WiRocket algorithm flow.

```
Input: $S$: Time series $\alpha$: Series scaling factor
Output: $F$ : Feature
Fit:
  1. generate 84 convolution kernels with different weights.
     // set the indices of $\beta$
     $I \leftarrow [[0,1,2],[0,1,3],\ldots,[6,7,8]]$
  2. derive the dilation group
     $\max \leftarrow \log_2(\alpha * \text{ length } (S)-1)/8$
     $\hat{D} \leftarrow \left[\left\lfloor 2^0\right\rfloor,\left\lfloor 2^{\max/32}\right\rfloor,\ldots,\left\lfloor 2^{32\cdot\max/32}\right\rfloor\right]$
  3. randomly selected a sample to calculate the bias
  4. Combine dilation/bias/padding to form a set $kernel\_set$
Transform:
  $F = [\ ]$
  for $k$ in $kernel\_set$ do
      for $d$ in $\hat{D}$ do
          $feature = PPV(S \otimes k)$
          $F = F \cup feature$
      end
  end
```

We know that different types of actions not only differ in action amplitude, but also correspond to different frequencies. Therefore, in order to better achieve action recognition, we need to extract different frequency features. Unlike the convolution kernel with deterministic parameters, the random convolution kernel can extract signal feature at different scales and in different modes with various feature extraction patterns for different signals. This allows extraction of features at different frequencies, with more comprehensive and targeted features. Previous work using random convolutional kernels for feature extraction such as the classical U-net [27], utilized convolutional kernels of different sizes to extract features from the input. However, the size and depth of this network need to be adjusted for different inputs, and as a deep learning network, the number of parameters is large and requires more samples for training. The feature extraction module of WiLDAR consists of convolutional kernels of different sizes with defined parameters, avoiding the need for model training, which makes the model feature extraction easier and reduces the deployment overhead of the model.

The length of each convolution kernel $S \in \{S_1, S_2, ..., S_{84}\}$ is fixed to 9, and the weights are random combinations of α and β under the condition that the sum of weights is 0. (We set six of $\alpha = -1$ and three of $\beta = 2$). When the weight sum equals to 0, the convolution for the input X with weights W will not be affected by panning, i.e., $X * W = (X \pm c) * W$. That is, adding this constraint ensures that the kernels are only sensitive to the relative magnitude of the input values, making the output of the convolution translation invariant and reducing the operation burden.

Spectral power of human activity mostly concentrated from 0 to 20 Hz, and the common CSI acquisition frequency is 1,000Hz. The sampling points are calculated as follows.

$$N = f * \frac{1}{F} \tag{2}$$

where N represents the number of sampling points, f represents the signal sending rate, and F represents the frequency corresponding to the action. Therefore, 20 Hz is equivalent to 50 samples in the time domain. Accordingly, we design the dilation to ensure the receptive field is within this range. The range of the dilation group is determined by the input length within the range $D = \left[2^0, 2^{\max}\right]$, where $\max = \log_2\left[(\alpha * L_{\text{input}} - 1) / (L_{\text{kernel}} - 1)\right]$. L_{input} is the input length, L_{kernel} is kernel size, and α is an artificial parameter based on the input length to control the receptive field. Dilation group will produce a geometric progression of 32 values from the range, and finally the dilation group is combined with each kernel to extract features.

Bias is taken from the convolutional output by randomly selecting a single training sample, with quartiles of its convolutional output computed as bias. The convolution layer automatically calculates the proportion of positive values (PPV) metrics to enrich the extracted spatio-temporal features, in addition to the convolutional output. PPV is calculated by the following equation.

$$\text{PPV}(X * W - b) = \frac{1}{n} \sum [X * W - b > 0] \tag{3}$$

where X is the input sequence, W is the convolution kernel weight, and b is bias. Calculating PPV is essentially equivalent to calculating the empirical cumulative distribution function, which allows the classifier to determine the prevalence of a given pattern in a time series.

The random convolution layer automatically extracts multi-channel features for each sample considering temporal coherence, thus retaining more detailed information. The features are automatically split according to the convolutional combinations of different frequency. Finally we get a [84,119] multi-channel message, where different channels are composed by random convolution kernels at the same point of the sequence, to maximize the spatio-temporal information of the extracted feature.

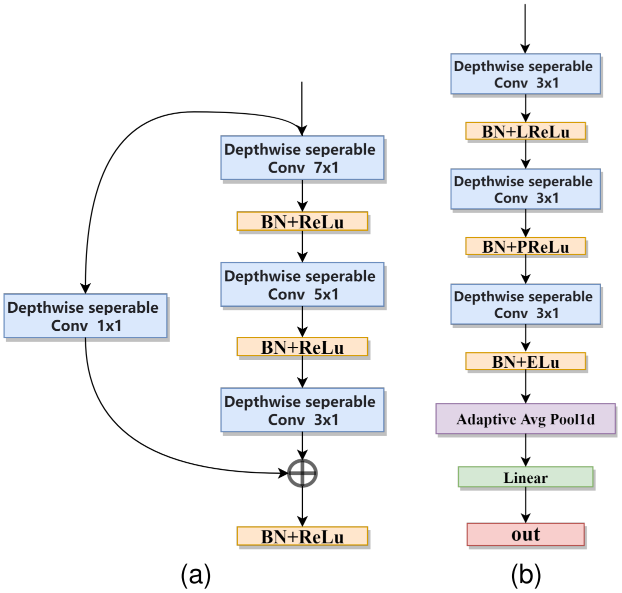

Depthwise seperable Conv 7x1
BN+ReLu
Depthwise seperable Conv 5x1
BN+ReLu
Depthwise seperable Conv 3x1
BN+ReLu
Depthwise seperable Conv 1x1
(a)
Depthwise seperable Conv 3x1
BN+LReLu
Depthwise seperable Conv 3x1
BN+PReLu
Depthwise seperable Conv 3x1
BN+ELu
Adaptive Avg Pool1d
Linear
out
(b)

Fig. 4. The classification module of WiLDAR. (a) Residual block. (b) the posterior network.

Classification: To classify the features, a CNN network with residual structure is designed, as shown in Fig. 4. We combine the residual structure with depthwise separable convolution to relearn and classify the features while reducing the parameters. The TPE algorithm is applied for parameter search and relevant updates are made to the activation function to improve the network classification ability.

The residual structure can transform network learning objectives by introducing shortcut connections and identity mappings. In turn, the gradient disappearance and gradient explosion problems could be avoided. Given the input $F = \{f_1, f_2, ..., f_N\} \in \mathbb{R}^{N \times 84 \times 119}$ where N is the number of inputs and f is the feature extracted by WiRocket, the transforming function is represented as follows.

$$\mathbf{y} = H(f) + \mathcal{F}(f, \mathcal{W}) \tag{4}$$

where $\mathbf{y}$ represents the output, H represents the identity mappings, and $\mathcal{F}$ represents the residual function which is often a series of convolution operations. After the residual structure, we use a posterior network consisting of multiple convolution layers, as shown in Fig. 4 for classification. Three different activation functions, Leaky Rectified Linear Unit (LReLu), Parametric Rectified Linear Unit (PReLU) and Exponential Linear Unit (ELu) [28], are implemented after the convolutional layers to improve the network mapping ability and avoid the gradient problem caused by a single activation function. Specifically, LReLU can be used to alleviate the problem of the activation function encountering zero gradients by slightly tilting it in the negative range. The PReLU, on the other hand, uses the parameters of the adaptive learning

This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2023.3294004

rectifier to avoid parameter settings for the activation function. The last used ELU can produce negative outputs, which helps to speed up the learning process and increase the robustness to noise. This function does not produce smaller derivatives and can avoid the problem of gradient disappearance due to the mismatch between the input and output space sizes.

To reduce the computational effort, we replace all the convolutions in the network with one-dimensional depthwise separable convolutions. It uses depthwise convolution to reduce the depth and pointwise convolution to feature fusion and depth expansion. After that, the number of convolutional parameters can be reduced to about one-ninth, which greatly reduces the overhead of convolutional operations.

Algorithm 2 TPE algorithm flow

```
Input: Search Target $T$  Search Scope $S$
    Maximum number of iterations $N$
Output: Specific results for each search $OUT$
1: Create an objective function applied in $T$ and output a score that we want to minimize.
2: Get couple of observations (score) using randomly selected set of $S$.
3: Sort the collected observations by score and divide them into two groups $x_1$, $x_2$ based on some quantile.
4: Two densities $\ell(x_1)$ and $g(x_2)$ are modeled using Parzen Estimators.
5: Draw sample hyperparameters from $\ell(x_1)$, evaluating them in terms of $\frac{\ell(x_1)}{g(x_2)}$, and returning the set that yields the minimum value under $\frac{\ell(x_1)}{g(x_1)}$ corresponding to the largest expected improvement. These hyperparameters are then evaluated by the objective function.
6: Update the observation list from step 3.
7: Repeat step 3-6 with a fixed number of trials or until time limit is reached
```

To avoid manually tuning the hyperparameters of the network, the TPE algorithm is used to automatically search the hyperparameters. The algorithm flow is shown in Algorithm 2. It fuzzily slices the sample points into two categories of superiority $g(x)$ and inferiority $\ell(x)$. The optimal parameters are obtained by iterating to update the two sets and finally maximize the Expected Improvement (EI) function. The EI after the Bayes' rule transformation is shown below.

$$\begin{aligned}\mathrm{EI}_{y^*}(x) &= \frac{\gamma y^* \ell(x) - \ell(x)\int_{-\infty}^{y^*} p(y)dy}{\gamma \ell(x) + (1-\gamma) g(x)} \\ &\propto \left(\gamma + \frac{g(x)}{\ell(x)}(1-\gamma)\right)^{-1}\end{aligned} \tag{5}$$

Equation 5 indicates that to maximize EI we need to make $\frac{g(x)}{\ell(x)}$ minimum, so the set of x which makes $g(x)$ smaller and $\ell(x)$ larger is returned in each iteration. The hyperparameters in the set are evaluated on the objective function. Eventually the process is repeated to achieve the hyperparameter search.

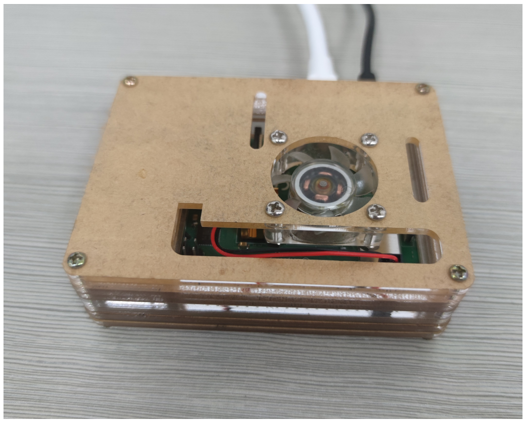


Fig. 5. Raspberry Pi 4B Platform.

TABLE III
Training and testing time on Raspberry Pi

Method	Training(sec)	Testing per sample(sec)	Parameters
T-Unet [29]	1152.56	0.154	5.18M
WiLDAR	60.01	0.031	0.14M

C. Tiny HAR System

In order to apply CSI signals to remote monitoring of special population and controlling of smart home in application scenarios of the IoT, we designed a tiny HAR system using Raspberry Pi and existing WiFI router. By modifying the Raspberry Pi's network card configuration [14], CSI signal can be acquired through Raspberry Pi and WiFI router. The specific version is the Raspberry Pi 4B with 8GB RAM and 64GB ROM, as shown in Fig. 5. The deployment scenario is a typical office scenario, with furniture such as desks and chairs, and the presence of more electronic devices such as computers, cell phones, etc. The placement of the devices ensures the existence of the Line of Sight (LoS). The actual performance of the system is shown in Table III.

We use the ARIL dataset for testing on the Raspberry Pi platform, and the training time for a single epoch across all samples is presented in Table III. From the results, it can be seen that compared to other CSI based HAR networks, WiLDAR has a substantially lower training and testing time. This is because of the feature extraction module in WiLDAR has no parameters to learn and therefore does not need back propagation, which greatly reduces the training time of the network. The depthwise separable convolution reduces the network parameters and time consumption. Furthermore, the test time of WiLDAR is only 0.03 seconds, which can fully achieve the purpose of real-time monitoring of the action. This shows that it is feasible to migrate WiLDAR to IoT devices and the simultaneous acquisition and real-time classification of CSI signals on IoT devices in the future. The system can also significantly reduce the actual cost and facilitate the deployment of HAR.

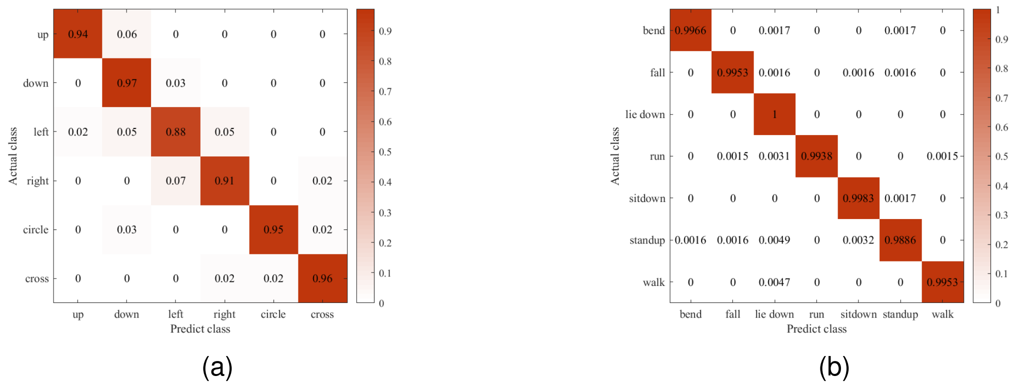


Fig. 6. Confusion matrix of WiLDAR on publicly available datasets at different fine grains of granularity. (a) ARIL. (b) CSI-HAR.

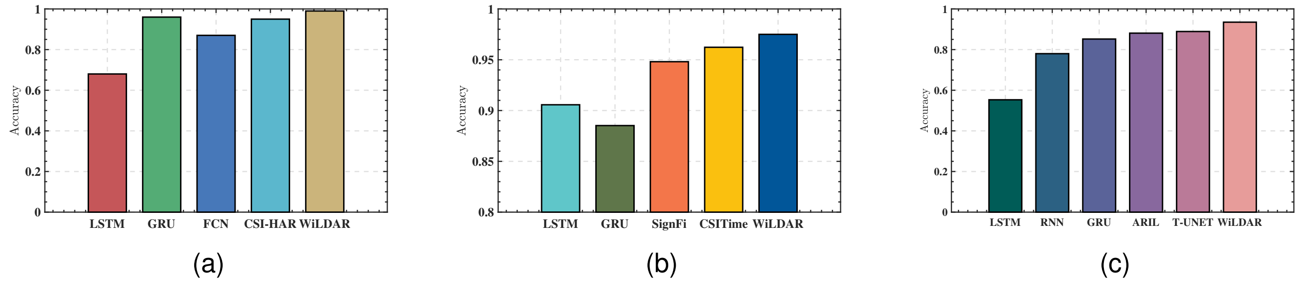


Fig. 7. Accuracy comparison of different networks on three publicly available datasets. (a) CSI-HAR. (b) SignFi. (c) ARIL.

IV. Experimental Evaluation

In this section, we present the effectiveness of WiLDAR by comparing the test accuracy, number of parameters and time complexity with other networks. Furthermore, we analyze the impact of individual diversity and ablation experiments.

A. *Experimental Setup*

We tested WiLDAR on three different fine-grained public datasets. The key information of these datasets are shown in Table IV.

1) **SignFi**: In [30] CSI data for 276 sign language poses were collected from 5 males, either at home or in the laboratory, which is more complex. In our experiments, a total of 8280 samples from laboratory and home were used.
2) **ARIL**: The dataset [31] was originally captured to enable remote control of smart homes. Six hand gestures such as hand up, hand down, hand left, etc. were collected from 16 different location to form a total of 1394 samples.
3) **CSI-HAR**: In [32] the CSI data of seven actions were collected from three subjects. The collected actions are common sitting, standing, running, etc. Each action, with unfixed duration, was performed 20 times per person, and therefore the number of sample packets acquired was not fixed. In this paper, we downsample irregular time series to the same length.

In the actual experiment, we use Adam as the optimizer, Cross Entropy as the loss function, batch size set to 32, learning rate set to 0.0001, and a weight decay of 0.01 in the training. All of our training and testing processes are conducted on a Lenovo r9000p laptop with AMD Ryzen 7 5800H CPU and NVIDIA GeForce RTX 3070 Laptop GPU.

TABLE IV
Information about CSI datasets

Name	Acquisition Platform	Action Type	Data Dimension
SingFi	Intel WiFi Link 5300	sign language	1x3x30x200
ARIL	USRPs	hand gestures	1x52x192
CSI-HAR	Nexmon	daily activities	1x52

TABLE V
Comparison with classification macro indicators

Method	Precision	Recall	Specificity	F1-Score
LSTM	0.564	0.554	0.911	0.560
RNN	0.784	0.778	0.956	0.780
GRU	0.857	0.852	0.970	0.850
ARIL [31]	0.884	0.883	0.977	0.880
WiLDAR	0.936	0.934	0.987	0.934

TABLE VI
COMPARISON OF CLASSIFICATION INDICATORS AT DIFFERENT MOVEMENTS

Model	Metrics	ARIL						CSI-HAR						
		up	down	left	right	circle	cross	bend	fall	lie-down	run	sit-down	stand-up	walk
LSTM	Precision	0.654	0.510	0.431	0.538	0.688	0.565	0.574	0.740	0.609	0.816	0.759	0.565	0.748
	Recall	0.723	0.553	0.532	0.438	0.512	0.565	0.690	0.791	0.585	0.802	0.518	0.600	0.777
	Specificity	0.922	0.892	0.857	0.922	0.957	0.914	0.919	0.952	0.934	0.969	0.974	0.925	0.956
	F1-Score	**0.681**	0.530	0.476	0.482	0.587	0.565	0.626	0.764	0.596	**0.808**	0.615	0.581	0.780
GRU	Precision	0.815	0.977	0.816	0.820	0.800	0.919	0.972	0.965	0.975	0.983	0.955	0.948	0.981
	Recall	0.936	0.894	0.851	0.854	0.837	0.739	0.981	0.986	0.960	0.994	0.936	0.931	0.990
	Specificity	0.957	0.996	0.961	0.961	0.962	0.987	0.995	0.994	0.996	0.997	0.993	0.992	0.997
	F1-Score	0.871	**0.933**	0.833	0.836	0.818	0.819	0.976	0.975	0.967	0.988	0.945	0.939	**0.985**
WiLDAR	Precision	0.936	0.974	0.881	0.912	0.947	0.964	0.997	0.995	0.996	0.994	0.994	0.998	0.995
	Recall	0.978	0.864	0.881	0.945	0.973	0.964	0.998	0.997	0.985	0.995	0.995	0.994	0.998
	Specificity	0.987	0.996	0.979	0.978	0.992	0.991	0.998	0.997	0.994	0.994	0.994	0.996	0.998
	F1-Score	0.956	0.915	0.881	0.928	0.959	**0.964**	**0.997**	0.995	0.990	0.994	0.994	0.995	0.996

B. Performance Evaluation

We will show the specific performance of WiLDAR on the relevant datasets. The metrics we calculate include accuracy, precision, recall, specificity, and F1-Score [33], all of which are calculated on a macro-average. The confusion matrix and classification performance of the experiment are shown in Fig. 6, Fig. 7 and Table V.

Confusion matrices of ARIL and CSI-HAR are presented in Fig. 6, where SignFi is not included due to the large number of classification categories. It can been seen that in the ARIL dataset, the gesture with the highest recognition accuracy is "down" and the lowest is "left". All the actions except the left and right gestures achieve an accuracy higher than 95%. Considering that the left and right gestures are consistent in terms of movement amplitude and frequency, the network will cause confusion in classification. In the CSI-HAR dataset, characterized by notable action amplitudes and clear distinctions among individual actions, all actions demonstrate classification accuracies exceeding 99%. This can be attributed to the inherent robustness and efficacy of WiLDAR's feature extraction capabilities, which enable it to leverage the dataset's characteristics effectively.

In Fig. 7 and Table V, we compared WiLDAR with other methods, including some classical networks and some recent networks using the same dataset [29]–[32], [34]. By analyzing the charts, we achieved the highest accuracy of 97.5%, 93.5%, and 99.5% on the three datasets, respectively. It can be seen that the classical network shows the worst learning ability for CSI signals, indicating that the high dimensionality and multi-channel characteristics of the CSI signal make the feature extraction difficult. Although previous work tried to improve the accuracy by refining the network structure, the single feature extraction mode cannot adapt to multiple actions. However, with the combination of random convolution and residual structure, WiLDAR is able to extract action features on different frequencies in multi-dimensional CSI signals. Multi-scale random convolution makes it easier to capture the expression patterns of different action information in different dimensions. Furthermore, by designing different activation functions, a more comprehensive feature map is achieved. Through multi-channel reduction of the extracted features according to time nodes, the spatio-temporal characteristics could be maximized. All this ensures WiLDAR's feature extraction ability and action recognition performance.

We also tested the recognition performance of different movements as shown in Table VI. The bolded data are the actions with the highest F1-score for each network, from which it can be seen that different networks fit different actions, however WiLDAR achieves a very high recognition accuracy for each action. Notably, WiLDAR showcased reduced fluctuations in Fi scores across different actions, indicating a significant enhancement in its capacity to extract multi-scale features. This improvement can be attributed to the integration of the WiRocket algorithm, facilitating automatic feature extraction within the network.

C. Discussion

We test the effect of WiLDAR's structure on accuracy, and verification associated with its time complexity and independence.

Hyperparameter search: Results of hyperparametric search are shown in Fig. 8. We use the TPE algorithm described in Section III to automatically search for the structural parameters of WiLDAR. Three search parameters are batch size, convolutional kernel size, and the number of kernels.

The yellow line represents the combination with higher accuracy, so the more concentrated the yellow line crosses, the better choice of the parameters. From the Fig. 8, we can see that the best choice of batch size is 64, convolutional kernel size is [7,5,3], and the number of kernels is 256. It

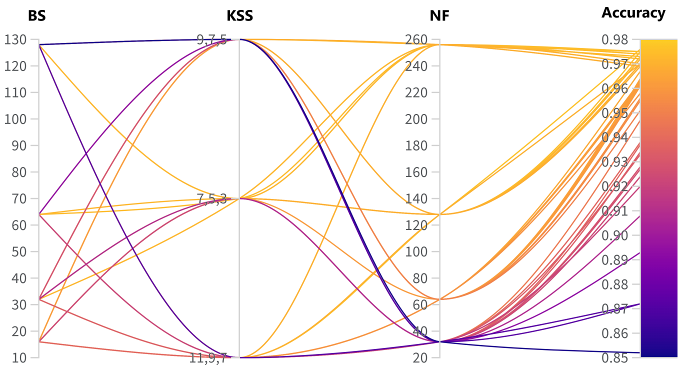


Fig. 8. Hyperparametric search results. In the figure, 'BS' is the batch size, 'KSS' is the convolutional kernel size of the three convolutional layers, and NF is the number of filters.

TABLE VII
THE ABLATION TEST OF WILDAR

Method	Accuracy
Baseline	0.916
Residual block	0.917
Residual block + DS-Conv	0.957
Baseline + Residual block + DS-Conv	**0.975**

TABLE VIII
IMPACT OF INDIVIDUAL DIVERSITY OF WILDAR

Method	Precision	Recall	Accuracy
Without user 3	0.584	0.566	0.566
Add 7% of user 3 sample	0.891	0.941	0.890
All data	0.995	0.996	0.995

TABLE IX
COMPARISON OF TIME COMPLEXITY OF DIFFERENT NETWORKS

Time Complexity	T-Unet [29]	ARIL [31]	WiLDAR
Training(sec)	104.03	65.77	**24.45**
Testing(sec)	4.21	3.17	**0.04**
Parameters	5.18M	3.49M	**0.14M**

can be seen that a moderate batch size and convolutional kernel size are more conducive to the extraction of detailed features by the network, and more convolutional kernels bring more feature extraction patterns. However, considering the number of parameters in the actual deployment, the number of convolution kernels was chosen to be 64, which can reduce a lot of operation overhead without notable accuracy degradation.

Ablation test: Accuracy performance of ablation test is shown in Table VII. We performed ablation tests on the individual blocks of WiLDAR, where Baseline refers to the MiniRocket [24] network, Residual block refers to the residual block structure proposed in Section III, and DS- Conv refers to the depthwise separable convolution. The experiments use the SigFi dataset with the training epoch set to 250.

From the results, the accuracy performance of MiniRocket and residual convolution is similar. With the combination of depth-separable convolution, the feature extraction and fusion is separated, bringing some degree of accuracy improvement. The combination of the three shows the highest accuracy and maximum accuracy improvement, which indicates that the combination of MiniRocket and the residual structure greatly improves the learning ability of the network, and the improvement is more pronounced than that of depthwise separable convolution.

Impact of Individual Diversity: In Table VIII, we performed the impact of individual diversity on WiLDAR. The network was trained using the data of the first two users in CSI-HAR, and tested by the third one.

The recognition accuracy decreases substantially when there is a difference between the source and target domains. This is because the network extracts the environment and background features during feature extraction, and when these conditions change, it affects the specific performance of the model. We also add 7% of the target domain samples to the training set, and the performance improved dramatically. This indicates that only a small number of target domain samples are needed to significantly improve the recognition capability of WiLDAR. In tests, WiLDAR recognized key actions such as falls with up to 95% accuracy, even with a small number of target domain samples. This verifies that WiLDAR is able to perform well in remote health monitoring even when the subject changes.

Time complexity: We compared the number of parameters and the time complexity of WiLDAR with ARIL and T-Unet, and the results are shown in Table IX. The data are taken from the ARIL dataset, the batch size is taken as 128, and the training epoch is 200. The testing time is the time required to test all 278 samples.

It can be seen from the Table IX that both the training and testing time of WiLDAR are much smaller than the other two networks. This is precisely due to the random convolution kernel, which does not use backpropagation, reducing the gradient calculation in training. In addition, WiLDAR has less than one-tenth of the parameters compared with the other two networks, due to the adaption of depthwise separable convolution. These results demonstrate the lightweight features of WiLDAR.

V. CONCLUSION

In this paper, we propose the WiLDAR, a lightweight network that can easily perform feature extraction on the original CSI signal for HAR. We design multi-scale convolution to extract different action features and eliminate the tedious signal preprocessing and manual feature extraction. A block combining residual networks with depthwise separable convolution is proposed to reduce the number of parameters and the training time. We tested WiLDAR on three different finegrained public datasets, and achieved the highest classification accuracy with less than one-tenth of the parameters comparing to other networks, resulting in shorter training. Finally, we implemented a tiny HAR system with only Raspberry Pi and WiFi router, which can greatly reduce the space requirements and cost of the deployment. The experimental results show that WiLDAR is fully capable of real time human activity monitoring. In the future, we can build an all-round monitoring system with multi-terminal interconnection using embedded terminal, cell phone terminal, and PC terminal around the home WiFi LAN, which aligns with the IoT development trend of the Internet of everything. We believe that WiLDAR can be well applied to the Internet of Things, human-computer interaction, remote medical monitoring, and other applications that require the lightweight implementation and learning ability.

REFERENCES

[1] E. Jovanov, "Wearables meet iot: Synergistic personal area networks (spans)," *Sensors*, vol. 19, no. 19, 2019.

[2] H. Zhao, S. Wang, G. Zhou, and D. Zhang, "Ultigesture: A wristband-based platform for continuous gesture control in healthcare," *Smart Health*, vol. 11, pp. 45–65, 2019.

[3] Y. Zhang, F. Zhang, Y. Jin, Y. Cen, V. Voronin, and S. Wan, "Local correlation ensemble with gcn based on attention features for cross-domain person re-id," *ACM Transactions on Multimedia Computing, Communications and Applications*, vol. 19, no. 2, pp. 1–22, 2023.

[4] Z. Gao, H.-Z. Xuan, H. Zhang, S. Wan, and K.-K. R. Choo, "Adaptive fusion and category-level dictionary learning model for multiview human action recognition," *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 9280–9293, 2019.

[5] O. D. Lara and M. A. Labrador, "A survey on human activity recognition using wearable sensors," *IEEE communications surveys & tutorials*, vol. 15, no. 3, pp. 1192–1209, 2012.

[6] B. Fang, F. Sun, H. Liu, and C. Liu, "3d human gesture capturing and recognition by the immu-based data glove," *Neurocomputing*, vol. 277, pp. 198–207, 2018.

[7] Z. Wang, Z. Yu, X. Lou, B. Guo, and L. Chen, "Gesture-radar: A dual doppler radar based system for robust recognition and quantitative profiling of human gestures," *IEEE transactions on human-machine systems*, vol. 51, no. 1, pp. 32–43, 2020.

[8] 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 Transactions on Mobile Computing*, vol. 13, no. 4, pp. 907–920, 2013.

[9] J. R. Smith, K. P. Fishkin, B. Jiang, A. Mamishev, M. Philipose, A. D. Rea, S. Roy, and K. Sundara-Rajan, "Rfid-based techniques for human-activity detection," *Communications of the ACM*, vol. 48, no. 9, pp. 39–44, 2005.

[10] F. Wang, W. Gong, and J. Liu, "On spatial diversity in wifi-based human activity recognition: A deep learning-based approach," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 2035–2047, 2018.

[11] X. Zhou, W. Liang, I. Kevin, K. Wang, H. Wang, L. T. Yang, and Q. Jin, "Deep-learning-enhanced human activity recognition for internet of healthcare things," *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 6429–6438, 2020.

[12] D. Halperin, W. Hu, A. Sheth, and D. Wetherall, "Tool release: Gathering 802.11 n traces with channel state information," *ACM SIGCOMM computer communication review*, vol. 41, no. 1, pp. 53–53, 2011.

[13] Y. Xie, Z. Li, and M. Li, "Precise power delay profiling with commodity wi-fi," *IEEE Transactions on Mobile Computing*, vol. 18, no. 6, pp. 1342–1355, 2018.

[14] M. Schulz, D. Wegemer, and M. Hollick. (2017) Nexmon: The c-based firmware patching framework. [Online]. Available: https://nexmon.org

[15] B. Wei, W. Hu, M. Yang, and C. T. Chou, "From real to complex: Enhancing radio-based activity recognition using complex-valued csi," *ACM Transactions on Sensor Networks (TOSN)*, vol. 15, no. 3, pp. 1–32, 2019.

[16] H. Kong, L. Lu, J. Yu, Y. Chen, and F. Tang, "Continuous authentication through finger gesture interaction for smart homes using wifi," *IEEE Transactions on Mobile Computing*, vol. 20, no. 11, pp. 3148–3162, 2020.

[17] K. Ali, M. Alloulah, F. Kawsar, and A. X. Liu, "On goodness of wifi based monitoring of sleep vital signs in the wild," *IEEE Transactions on Mobile Computing*, 2021.

[18] Y. Gu, Y. Zhang, J. Li, Y. Ji, X. An, and F. Ren, "Sleepy: Wireless channel data driven sleep monitoring via commodity wifi devices," *IEEE Transactions on Big Data*, vol. 6, no. 2, pp. 258–268, 2018.

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

[20] J. Wang, L. Zhang, Q. Gao, M. Pan, and H. Wang, "Device-free wireless sensing in complex scenarios using spatial structural information," *IEEE Transactions on Wireless Communications*, vol. 17, no. 4, pp. 2432–2442, 2018.

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

[22] W. Wang, A. X. Liu, M. Shahzad, K. Ling, and S. Lu, "Device-free human activity recognition using commercial wifi devices," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 5, pp. 1118–1131, 2017.

[23] C. Xiao, Y. Lei, Y. Ma, F. Zhou, and Z. Qin, "Deepseg: Deep-learning-based activity segmentation framework for activity recognition using wifi," *IEEE Internet of Things Journal*, vol. 8, no. 7, pp. 5669–5681, 2020.

[24] A. Dempster, D. F. Schmidt, and G. I. Webb, "Minirocket: A very fast (almost) deterministic transform for time series classification," in *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, 2021, pp. 248–257.

[25] X. Zou, Z. Wang, Q. Li, and W. Sheng, "Integration of residual network and convolutional neural network along with various activation functions and global pooling for time series classification," *Neurocomputing*, vol. 367, pp. 39–45, 2019.

[26] J. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl, "Algorithms for hyper-parameter optimization," *Advances in neural information processing systems*, vol. 24, 2011.

[27] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*. Springer, 2015, pp. 234–241.

[28] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Icml*, 2010.

[29] F. Wang, Y. Song, J. Zhang, J. Han, and D. Huang, "Temporal unet: Sample level human action recognition using wifi," *arXiv preprint arXiv:1904.11953*, 2019.

[30] Y. Ma, Z. Gang, S. Wang, H. Zhao, and W. Jung, "Signfi: Sign language recognition using wifi," *Proceedings of the ACM on Interactive Mobile Wearable and Ubiquitous Technologies*, vol. 2, no. 1, pp. 1–21, 2018.

[31] F. Wang, J. Feng, Y. Zhao, X. Zhang, and J. Han, "Joint activity recognition and indoor localization with wifi fingerprints," *IEEE Access*, vol. 7, pp. 1–1, 2019.

[32] P. F. Moshiri, R. Shahbazian, M. Nabati, and S. A. Ghorashi, "A csi-based human activity recognition using deep learning," *Sensors*, vol. 21, no. 21, p. 7225, 2021.

[33] T. A. Sorensen, "A method of establishing groups of equal amplitude in plant sociology based on similarity of species content and its application to analyses of the vegetation on danish commons," *Biol. Skar.*, vol. 5, pp. 1–34, 1948.

[34] S. K. Yadav, S. Sai, A. Gundewar, H. Rathore, K. Tiwari, H. M. Pandey, and M. Mathur, "Csitime: Privacy-preserving human activity recognition using wifi channel state information," *Neural Networks*, vol. 146, pp. 11–21, 2022.