A CSI-based Human Activity Recognition using Canny Edge Detector

Hossein Shahverdi, Parisa Fard Moshiri, Mohammad Nabati, Reza Asvadi, and Seyed Ali Ghorashi*

Abstract Human Activity Recognition (HAR) is one of the hot topics in the field of human-computer interaction. It has a wide variety of applications in different tasks such as health rehabilitation, smart houses, smart grids, robotics, and human action prediction. HAR can be carried out through different approaches such as vision-based, sensor-based, radar-based, and Wi-Fi-based. Due to the ubiquitous and easy-to-deploy characteristic of Wi-Fi devices, Wi-Fi-based HAR has gained the interest of both academia and industry in recent years. WiFi-based HAR can be implemented by two channel metrics: Channel State Information (CSI) and Received Signal Strength Indicator (RSSI). Recently, converting the CSI data to images has led to increasing the accuracy level of activity prediction. However, none of the previous research has focused on extracting the features of converted images using the image-processing techniques. In this study, we investigate three available datasets, gathered using CSI property, and took the advantage of Deep Learning (DL), with convolutional layers and edge detection technique to increase overall system accuracy. The canny edge detector extracts the most important features of the image, and giving it to the DL model empowers the prediction of activities. In all three datasets, we witnessed an improvement of 5%, 27%, and 37% in terms of accuracy.

Hosein shahverdi e-mail: h.shahverdi@mail.sbu.ac.ir, Parisa Fard Moshiri e-mail: p.fardmoshiri@mail.sbu.ac.ir, Mohammad Nabati e-mail: mo.nabati@mail.sbu.ac.ir and Reza Asvadi e-mail: r_asvadi@sbu.ac.ir are with the Cognitive Telecommunication Research Group, Department of Electrical Engineering, Shahid Beheshti University G. C., Tehran, Iran.

- Seyed Ali Ghorashi is with the Department of Computer Science & Digital Technologies, School of Architecture, Computing, and Engineering University of East London, London, UK. e-mail: s.a.ghorashi@uel.ac.uk, Seyed Ali Ghorashi is also with the Cognitive Telecommunication Research Group, Department of Electrical Engineering, Shahid Beheshti University G. C., Tehran, Iran.
- *corresponding author

1 Introduction

Human Activity Recognition (HAR) is one of the state-of-the-art research topics with significant academic and industrial interests [1]. HAR is majorly used in applications such as health monitoring, smart houses and buildings [2], localization [3], and fall detection in elderly care [4]. As pictured in Fig.1, HAR can be conducted through four different approaches, including sensor-based, vision-based, radar-based, and Wi-Fi-based.

In sensor-based HAR, activities are detected by analyzing the data gathered by the sensors such as accelerometer, gyroscope, pressure, and temperature. Chernbumroong et. al. [5] use accelerometer and temperature sensors of smartwatches for detecting daily activities in an elderly care scenario by implementing of Machine Learning (ML) classifier called Support Vector Machine (SVM). Wang et. al. [6] built special sensor nodes, operating with a battery that has in-built memory for detecting postures in swimming. This research is not convenient due to its insufficient accuracy and battery charge leakage.

Another HAR method is the vision-based method in which data (images or videos) are captured by cameras. Weinland et. al. [7] categorize activities using the spatial-temporal structure of actions using a camera such as kicking, waving, and punching activities. Another method for data acquisition is using depth, 3D [8], and 3D skeleton-based cameras [9]. Ziaeefard et. al. [10] represent semantic-based HAR by taking advantage of depth images and videos by extracting spatial and semantic features like posing, pose let and scene context. Rauntaray et. al. [11] proposed a hand gesture recognition method by developing their exclusive software and platform. Tapus et. al. [12] in addition to detecting human activities, studied human interaction with each other called crowd behavior, through security cameras. Although vision-

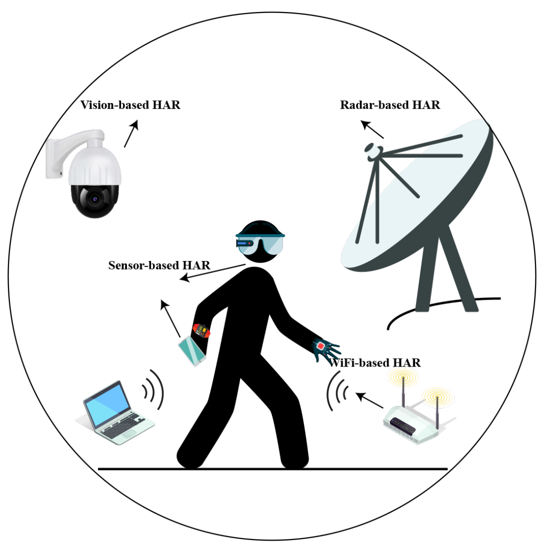


Fig. 1 Different types of data acquisition for Human Activity Recognition.

based HAR has received considerable attention with acceptable accuracies, it highly depends on the illumination and weather condition of the scene, the angle between device and targets, and the objects and hindrances between cameras and targets. In addition, due to the continuous monitoring, this may violate the privacy of users [32].

Radar-based HAR has some advantages over both Sensor and Vision-based approaches [14]. For instance, contrary to vision-based methods, the radar-based HAR method is independent of the light and weather conditions of the environment. Further, radar-based approaches can protect the privacy of users since recognition is conducted through velocity and range information gathered by returned modulated signals. Also, radar signals can pass through hindrances like walls, which results in applying this method in scenarios where vision-based approaches do not have direct access to users [14]. Kim et. al. [15] use SVM to extract features of 12 different activities gathered by micro-Doppler signatures. Zhou et. al. [16] proposed an enhanced Dynamic Time Wrapping (DTW) algorithm for hand gesture recognition using a radar in terahertz frequency. However, in the radar-based method, the magnitude of generated signals is weak, which results in missing or mixed noise values [17].

The last method, the Wi-Fi approach, is a subclass of the radar-based method except that it is more popular due to its access points, and has recently gained considerable attention due to its advantages such as being easy to deploy, light independence, ubiquitousness, and cost-efficiency [18]. Additionally, Wi-Fi-based HAR can have an acceptable accuracy in detecting activities if there is no Line-of-Sight condition between the Wi-Fi device and the targets [19]. Wi-Fi HAR is usually conducted using one of the two channel metrics of Received Signal Strength (RSS) and Channel State Information (CSI). RSS gives information about how the power of the signal has changed through its path from the transmitter to the receiver. The disadvantage of using RSS is that the power decay has a direct relation with distance. In other words, the accuracy of the model is mitigated dramatically by increasing the distance between the target and the Wi-Fi device. Also, RSS is unable to record small variations in signals, making it not appropriate in HAR [32],[19]. On the contrary, CSI can record even slight movements like chest ups and downs, and big movements like falling, sitting down, and standing [13]. Data acquisition for CSI purposes can be done by a few devices [32]. The most commonly used ones are Intel 5300 network interface card (NIC), raspberry Pi 3+ and 4 [33], Atheros 9580, and 9390 [32].

Although researchers have contributed to increasing the accuracy of activity recognition, the researches are underway to design more generic algorithms, which can predict the activities with high accuracy in various datasets and tasks. ML techniques are widely used for the classification of activities, such as SVM [26] and HMM [27]. These models need external and human assistants to extract features, which would affect the accuracy of the model based on the quality of extracted features [32]. To address this issue and extract features more accurately, researchers use Deep Learning (DL) algorithms [28], such as Recurrent Neural Network (RNN) [29], LSTM [32], and CNN [32]. Authors in [20] recognized activities by detecting the pace and speed of activities from CSI. Authors in [21] proposed a method to detect whether an activity has occurred in an indoor environment using an ML classifier called Decision Tree with two levels. The first level detects if an activity is performed while the second level recognizes which activity is performed. Chen et. al. [22] detects activities by extracting both forward and backward features. In the next step, based on the importance of features, a weight is assigned to each of the features for detecting activities. In [23], they applied Short-time Fourier Transform (STFT) for feature extraction and Random Forest along Hidden Markov Model (HMM) for the classification task.

In this research, we used datasets gathered by [32], [24], and [25], and converted them into RGB images as instructed in [13]. Then, because of the excellent results of Convolutional Neural Networks (CNNs) in image-based tasks, we used them as the DL classifier. For the next step, we applied the Canny filter as an edge detection technique for the generated images to highlight and extract more features to improve the accuracy of the model. Overall, we experienced considerable performance in terms of accuracy in all datasets, and also to detect Fall action with an acceptable accuracy, which is essential in elderly care applications.

The major contributions of this article are summarized as follows: We analyzed three available Wi-Fi-HAR datasets and based on their characteristics, we took advantage of preprocessing techniques such as Principal Component Analysis (PCA) and normalization. Then, we converted the data into RGB images and fed them into CNN. In another experiment, we applied the canny edge detector filter to our RGB images and repeated the training phase. Finally, we analyzed the performance of our method comparing it with the method proposed in [32].

The rest of this paper is organized as follows. Section II introduces the system model and how to extract the CSI data for the Activity Recognition task. In section III, the methodology used to convert CSI data into RGB images and apply the Canny edge detector technique to them to extract more details. In Section IV, experimental results are presented. The conclusion and future studies are discussed in Section V.

2 System Model

2.1 Preliminary

In Multiple-Input Multiple-Output (MIMO) and Orthogonal Frequency Division Multiplexing (OFDM) communication systems, CSI is considered as one of the metrices of wireless channels which describes how a signal deflects, reflects, fades and scatters during its transmission [30]. Knowing this information would lead us to adapt the transmission state with channel's properties in order to have the most favorable bit rate and achieve reliable communication with low bit error rate [13].

Let $T(f_n, t)$ and $R(f_n, t)$ represent the sequence transmitted from the transmitter and the received nth signal to the receiver at subcarrier frequency f_n at time t, respectively:

$$R(f_n, t) = T(f_n, t) \times CSI(f_n, t) + W; \tag{1}$$

where $CSI(f_n, t)$ is a vector consists of complex values. In other words, CSI has information about channel frequency response. This matrix with t transmitters and r receivers antennas can be represented as:

$$CSI_k = \begin{pmatrix} h_{1,1} & \cdots & h_{1,r} \\ \vdots & \ddots & \vdots \\ h_{t,1} & \cdots & h_{t,r} \end{pmatrix} \tag{2}$$

Here, $h_(t, r)$ is a complex number, containing both amplitude and phase of propagated signal from $t_t h$ transmitter and $r_t h$ receiver:

$$h_{t,r} = |h| \times e^{i\angle h} \tag{3}$$

where $|h|$ is amplitude and $\angle h$ is its phase response.

2.2 Canny Edge Detection

Edge detection is a technique in computer vision tasks whose main task is to find pixels in which brightness varies compared with adjacent pixels. This method discerns boundaries of an object from background through variation in local pixels intensities. By doing so, the boundaries of objects can be highlighted. In this methodology, filter finds the gradient of the image pixels in the direction of x and y [30]. Some well-studied filters, but not limited to, Sobel, Canny, Roberts, Prewitt and Laplacian [32]. The general rules of using edge detection techniques are as follow:

- Detecting edges as many as possible with low probability of error.
- The amount of noise must be mitigated to prevent detecting false edges.
- The edge point localized by filter, should be in the center of edges.

Canny edge detector is one of the well-known methods which satisfies the given rules in this era [30]. The properties of canny edge detector make it suitable for many applications. The process in detecting edges by canny filter can be summarized as follows:

- Apply gaussian filter to reduce the amount of noise in picture.
- Highlight pixels with intensity compared to others in the neighborhood by gradient method through x and y directions.
- Apply a threshold to omit false responses of founded edges.
- Apply another threshold to find the edges with high probability to be true ones.
- Remove the edges with weak connection to the other edges with higher connection.

as mentioned before, because of canny filter characteristics, we also took advantages of this filter in our experiment and applied it in our generated RGB images in order to extract more features out of images.

3 Proposed Method

3.1 Pre-processing

In this research, datasets from [13], [24] and [25] were scrutinized. In [13] the data was collected using Raspberry Pi 4 and Nexmon Tool [31] in 20MHz bandwidth on channel 36 in IEEE 802.11 ac standard. Authors in [13] collects data using Raspberry Pi 4 in IEEE 802.11 ac standard. The bandwidth was set to 20MHz in channel 36. This Data captured activities, namely: Bend, Fall, lie down, Sit Down, Standup, Run and Walk. Also, data has 52 columns which describes the number of subcarriers; however, the number of rows varies between 600 to 1100 depend on the duration of each activity has been done. Before converting data into RGB images, preprocessing methods like PCA, normalization have been applied and then fed into the proposed 2D-CNN. In [24] Authors used two Universal Software Radio Peripheral devices which one of them used as transmitter and the other one was used as receiver at 5GHz with 52 subcarriers. To generate data traffic, they used Ubuntu virtual machine and Gnu radio. This dataset, contains categories like: Empty, Sitting, Standing, and Walking.

Researchers in [25] similar to the [13] collected data by applying Raspberry Pi 4/3B+ at 5GHz frequency band in channel 36 with 40MHz bandwidth. Also, the number of subcarriers including Pilot and Zero Ones are 128. The collected data include categories such as: Empty, lie down, Standing, Sitting, Walking. Since data were saved as .pcap files, first we converted them into csv files by taking advantages of MATLAB software. Since CSI data are noisy, for removing futile values in data, we used PCA technique for detecting outliers with 3 components. Then, as instructed in [32] we converted CSI values into RGB images after normalization. In the next step, we applied canny filter as edge detection method to accentuate more details of the generated picture. In Fig.2 and Fig.3, sample pictures and the proposed pre-processing method are depicted, respectively.

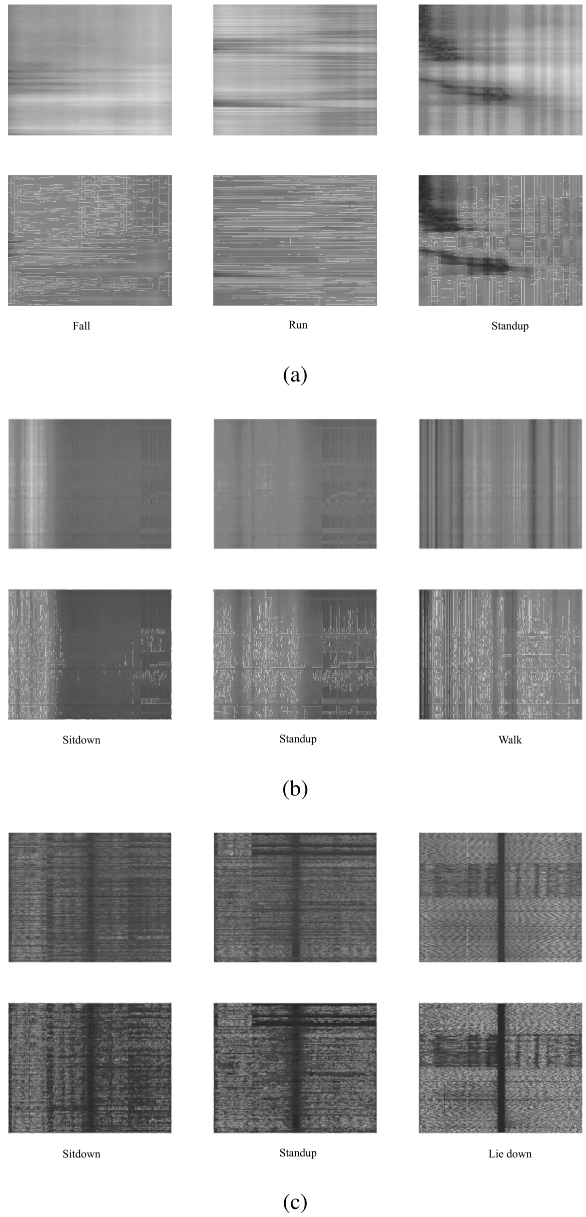


Fig. 2: Sample generated images from data: (a) [32], (b) [24], (c) [25].

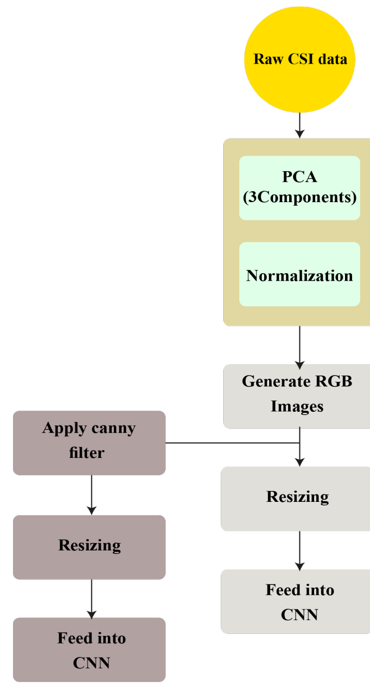


Fig. 3 Flowchart of preparing data for CNN.

3.2 CNN

CNN is one of the neural networks which generally used in vision-based tasks. Unlike the conventional classification algorithms, which need human assistance in preprocessing steps, CNN can take those actions desolately. Moreover, CNN can automatically extract the features of images by applying filters on pixels. A CNN usually consists of some main layers like: Convolution, Max or Average Pooling and Dense layers. Convolution layers have the task of extracting features from pixels of a given images. First Conv layer captures features such as color, edges and angles and by adding Conv layer, it can pinpoint more complex details out of images. Once features have been extracted, the exact locations of those are not crucial, so each Conv layer is normally followed by a Max or Average Pooling layer. The main advantage of applying this layer is that it diminishes the number of trainable parameters and provide the ability called translation invariance to the network which means that regardless of how the input has been shifted or translated, the output and response would be unique [13]. The last main layer is called Dense or fully connected layer which is a similar network used in conventional models where each one of the neurons are connected to other ones in the next layer. The output of the first stage (output of Conv and Pooling layer) is fed into fully connected layer, and dot product of weight vector and input vector are calculated to obtain final output. Apart from these layers, there are some parameters which are used for tuning and prevent from overfitting like Activation Function (A.F), Dropout layer, Batch Normalization (B.N). Dropout layer is one of the methods which can be used to prevent network from overfitting by dropping and inactivating some of the neurons randomly in every epoch. B.N. is used to make the network more stable during training process. Depending on the complexity of classification task, the number of used layers varies.

The proposed CNN in [32] was implemented in this research. This network is comprised of two Conv layer followed by ReLU activation function which then fed into Average Pooling layer. After the second Average Pooling layer, the matrix of features must be converted into one-column matrix, therefor Flatten Layer was used. This one-column feature matrix is passed through a Dense Layer which finally followed by the output layer that comprises the number of neurons based on the number of classes. Fig.4 depicts the proposed CNN. The main reason why ReLU A.F is used is that it avoids from vanishing gradient problem, computationally efficient and provide better convergence performance [13].

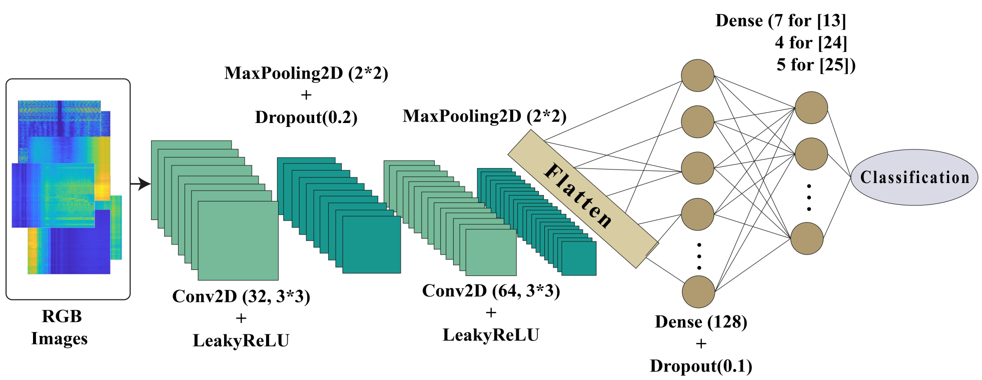


Fig. 4: Proposed CNN for classification in both before and after applying canny filter.

4 Experimental Setups and Results

For the software of simulation, Python 3.8 with TensorFlow 2.8 accelerated by GeForce RTX 3060 graphic card was used. We used previously collected datasets were used in [32], [24], and [25], which include 7, 5, and 4 activities (classes), respectively. Data in [25] are consist of a considerable amount of noise compared with [32] and [24], therefore it requires more preprocessing such as PCA in order to mitigate the undesirable information. Also, data were collected as Wireshark pcap file, therefore, in the first step data were converted into csv files and in the next step, they were transformed into RGB images. For the other datasets, except normalization no extra preprocessing were applied. Finally, before feeding images to CNN, they are resized to 32×32 to be suitable for CNN's input layer.

For the first part, we fed RGB-images to the proposed network and achieved 89%, 70% and 61% for [32], [24] and [25], respectively in 150 epochs. For the next step, instead of RGB images, we fed images which canny filter had been applied into them, and experienced a significant increase in accuracies compared to previous experiment results. The proposed preprocessing scored 93%, 98%, and 92% for the datasets, respectively. Fig.5 depicts the accuracies before applying canny filter, while Fig.6 is corresponding to post applying canny edge detector. It can clearly be observed that in all of the experiments, there is overfitting, which in another word, the gap between the accuracy of training and validation sets can not be ignored, while after applying the canny edge detector on the generated images, the accuracies has increased and there is no overfitting in our models. Also, the activity "Fall" with a F1-Score of 90% in dataset [32] is an acceptable accuracy for elderly health monitoring. As figure 5.b and 5.c show, the model faced with overfitting issue, while after applying canny filter, this problem has been solved and scored around 98% and 92%, respectively.

Confusion matrix is a technique for observing how well the model was capable of classifying the data and. Its performance is evaluated through four different metrices: Recall, Precision, Accuracy and F-measure. Before explaining the above metrices, one should be familiar with terms such as True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN). we can define aforementioned metrices as follow:

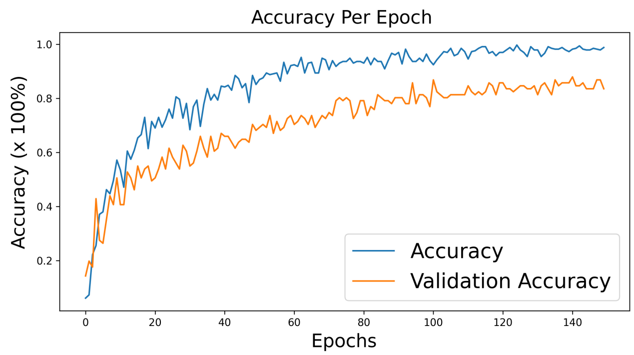
$$Recall = \frac{TP}{TP + FN} \tag{4}$$

$$Precision = \frac{TP}{TP + FP} \tag{5}$$

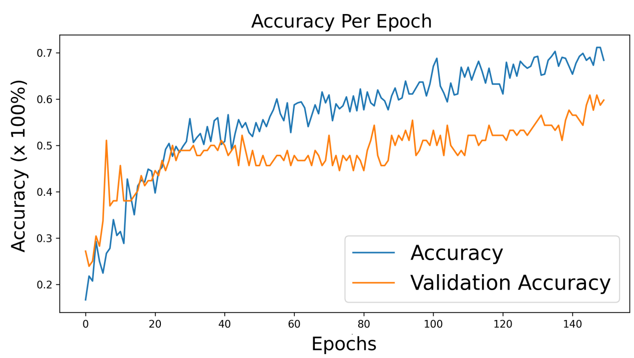
$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{6}$$

$$F1 - Score = \frac{Precision \times Recall}{Precision + Recall} \tag{7}$$

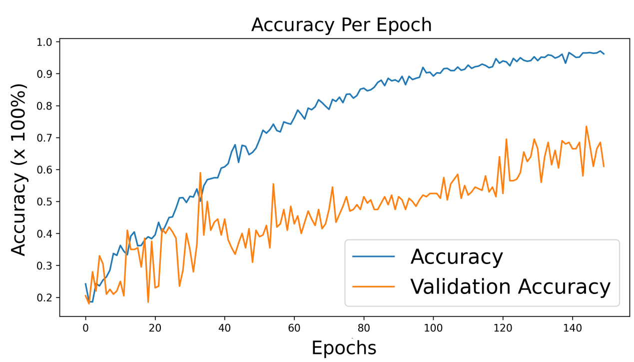
As can be seen in Fig.7, the model has achieved an acceptable and state-of-the-art accuracy in all of the studied datasets. Fig. 7. a shows that the model with a probability of 91% can detect fall accident, which is a vital situation needs to be attended immediately. To show clearly the accuracy of the proposed method on classification, we used confusion matrix (in Fig. 7) in which rows represent predicted classes while columns show predicted labels. As can be seen in Fig .6-a, the network has discerned similar activities like lying with sitting down.



(a)

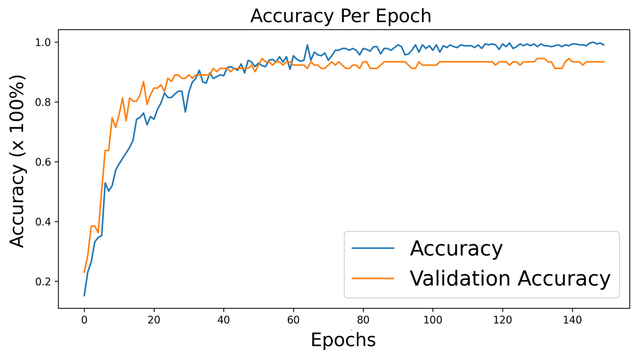


(b)

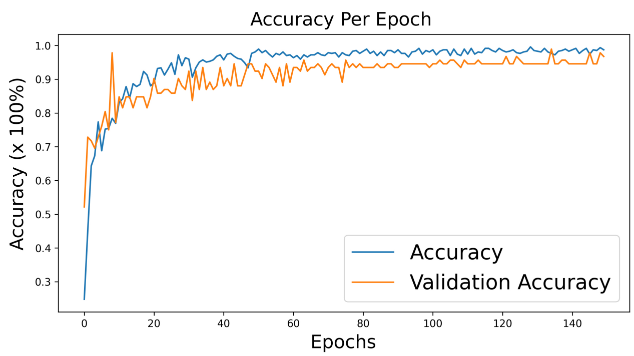


(c)

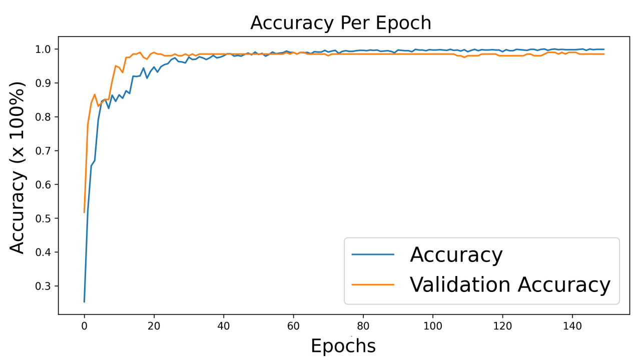
Fig. 5: Accuracy metric before applying canny filter for: (a) [32], (b) [24], (c) [25].



(a)



(b)



(c)

Fig. 6: Accuracy metric after applying canny filter for: (a) [32], (b) [24], (c) [25].

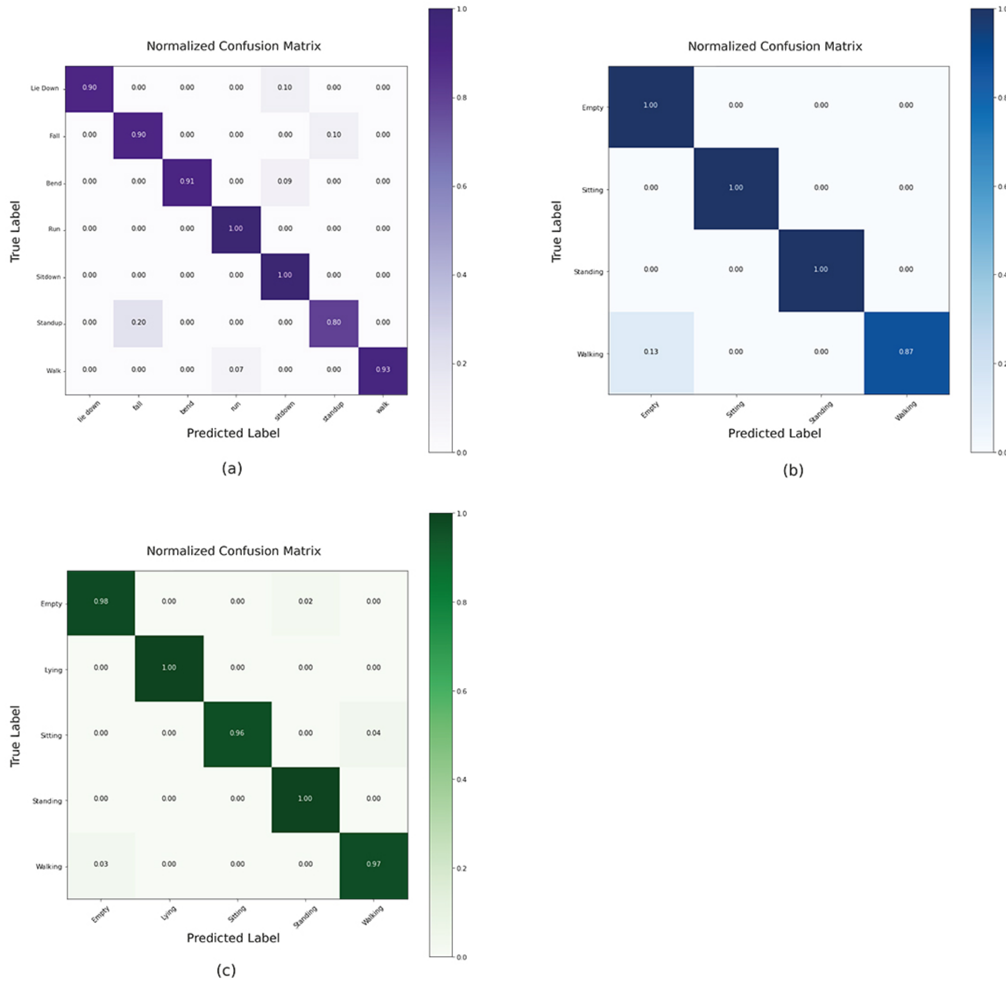


Fig. 7: Confusion Matrix for datasets: (a) [32], (b) [24] and (c) [25].

5 Conclusion and Future Work

Human activity recognition (HAR) is one of the hot topics in the field of human-computer interaction, which has gained both academic and industrial interest. HAR can be carried out through different approaches, such as vision-based, radar-based, sensor-based, and Wi-Fi-based. In this paper, we analyzed three public CSI datasets extracted from Wi-Fi signals. After converting data to RGB images as instructed in [13], we fed them into the CNN model, proposed in [32]. In the next step, we applied the canny filter to highlight more important details. Then, we applied filtered images to the same network and witnessed a significant improvement of around 5%, 28%, and 32% for [32], [24], [25], respectively. In future studies, we will focus on multimodal HAR datasets and apply the different preprocessing methods to achieve and score higher accuracy.

References

1. Y. Ma, G. Zhou, and S. Wang, “WiFi sensing with Channel State Information: A survey,” ACM Comput. Surv., vol. 52, no. 3, pp. 1–36, 2020.
2. H. Jiang, C. Cai, X. Ma, Y. Yang, and J. Liu, “Smart home based on WiFi sensing: A survey,” IEEE Access, vol. 6, pp. 13317–13325, 2018.
3. Y. Wang, K. Wu, and L. M. Ni, “WiFall: Device-free fall detection by wireless networks,” IEEE Trans. Mob. Comput., vol. 16, no. 2, pp. 581–594, 2017.
4. T. Wang et al., “Recognizing parkinsonian gait pattern by exploiting fine-grained movement function features,” ACM Trans. Intell. Syst. Technol., vol. 8, no. 1, pp. 1–22, 2016.
5. S. Chernbumroong, S. Cang, A. Atkins, and H. Yu, “Elderly activities recognition and classification for applications in assisted living,” Expert Syst. Appl., vol. 40, no. 5, pp. 1662–1674, 2013.
6. Kolkar, Ranjit, and V. Geetha, "Issues and challenges in various sensor-based modalities in human activity recognition system," In Applications of Advanced Computing in Systems, pp. 171-179, 2021.
7. D. Weinland, R. Ronfard, and E. Boyer, “A survey of vision-based methods for action representation, segmentation and recognition,” Comput. Vis. Image Underst., vol. 115, no. 2, pp. 224–241, 2011.
8. J. K. Aggarwal and L. Xia, “Human activity recognition from 3D data: A review,” Pattern Recognit. Lett., vol. 48, pp. 70–80, 2014.
9. F. Han, B. Reily, W. Hoff, and H. Zhang, “Space-time representation of people based on 3D skeletal data: A review,” Comput. Vis. Image Underst., vol. 158, pp. 85–105, 2017.
10. M. Ziaeefard and R. Bergevin, “Semantic human activity recognition: A literature review,” Pattern Recognit., vol. 48, no. 8, pp. 2329–2345, 2015.
11. S. S. Rautaray and A. Agrawal, “Vision based hand gesture recognition for human computer interaction: a survey,” Artif. Intell. Rev., vol. 43, no. 1, pp. 1–54, 2015.
12. A. Tapus, A. Bandera, R. Vazquez-Martin, and L. V. Calderita, “Perceiving the person and their interactions with the others for social robotics – A review,” Pattern Recognit. Lett., vol. 118, pp. 3–13, 2019.
13. P. F. Moshiri, M. Nabati, R. Shahbazian, and S. A. Ghorashi, “CSI-based human activity recognition using convolutional neural networks,” in 2021 11th International Conference on Computer Engineering and Knowledge (ICCKE), 2021.
14. X. Li, Y. He, and X. Jing, “A survey of deep learning-based human activity recognition in radar,” Remote Sens. (Basel), vol. 11, no. 9, p. 1068, 2019.

15. Y. Kim and H. Ling, “Human activity classification based on micro-Doppler signatures using a support vector machine,” IEEE Trans. Geosci. Remote Sens., vol. 47, no. 5, pp. 1328–1337, 2009.
16. K. A. Smith, C. Csech, D. Murdoch, and G. Shaker, “Gesture recognition using mm-wave sensor for human-car interface,” IEEE Sens. Lett., vol. 2, no. 2, pp. 1–4, 2018.
17. G. Lee and J. Kim, “Improving human activity recognition for sparse radar point clouds: A graph neural network model with pre-trained 3D human-joint coordinates,” Appl. Sci. (Basel), vol. 12, no. 4, p. 2168, 2022.
18. Y. Mei, T. Jiang, X. Ding, Y. Zhong, S. Zhang, and Y. Liu, “WiWave: WiFi-based human activity recognition using the wavelet integrated CNN,” in 2021 IEEE/CIC International Conference on Communications in China (ICCC Workshops), 2021.
19. B. A. Alsaify, M. M. Almazari, R. Alazrai, S. Alouneh, and M. I. Daoud, “A CSI-based multi-environment human activity recognition framework,” Appl. Sci. (Basel), vol. 12, no. 2, p. 930, 2022.
20. W. Wang, A. X. Liu, M. Shahzad, K. Ling, and S. Lu, “Device-free human activity recognition using commercial WiFi devices,” IEEE j. sel. areas commun., vol. 35, no. 5, pp. 1118–1131, 2017.
21. J. Ding and Y. Wang, “WiFi CSI-based human activity recognition using deep recurrent neural network,” IEEE Access, vol. 7, pp. 174257–174269, 2019.
22. Z. Chen, L. Zhang, C. Jiang, Z. Cao, and W. Cui, “WiFi CSI based passive human activity recognition using attention based BLSTM,” IEEE Trans. Mob. Comput., vol. 18, no. 11, pp. 2714–2724, 2019.
23. F. Wang, J. Feng, Y. Zhao, X. Zhang, S. Zhang, and J. Han, “Joint activity recognition and indoor localization with WiFi fingerprints,” arXiv [cs.HC], 2019.
24. A. M. Ashleibta et al., “5G-enabled contactless multi-user presence and activity detection for independent assisted living,” Sci. Rep., vol. 11, no. 1, p. 17590, 2021.
25. D. Trabelsi, S. Mohammed, F. Chamroukhi, L. Oukhellou, and Y. Amirat, “An unsupervised approach for automatic activity recognition based on hidden Markov model regression,” IEEE Trans. Autom. Sci. Eng., vol. 10, no. 3, pp. 829–835, 2013.
26. J. Wang, Y. Chen, S. Hao, X. Peng, and L. Hu, “Deep learning for sensor-based activity recognition: A survey,” arXiv [cs.CV], 2017.
27. Z. Shi, J. A. Zhang, R. Xu, and G. Fang, “Human activity recognition using deep learning networks with enhanced channel state information,” in 2018 IEEE Globecom Workshops (GC Wkshps), 2018.
28. Y. Wang, J. Liu, Y. Chen, M. Gruteser, J. Yang, and H. Liu, “E-eyes: Device-free location-oriented activity identification using fine-grained WiFi signatures,” in Proceedings of the 20th annual international conference on Mobile computing and networking, 2014.
29. X. Ding, T. Jiang, Y. Zhong, Y. Huang, and Z. Li, “Wi-Fi-based location-independent human activity recognition via meta learning,” Sensors (Basel), vol. 21, no. 8, p. 2654, 2021.
30. Singh Kiran Jot. Novel Technique for Edge Detection Using Logical Operations. S.L., Lap Lambert Academic Publ, 2013.
31. nexmon: The C-based Firmware Patching Framework for Broadcom/Cypress WiFi Chips that enables Monitor Mode, Frame Injection and much more.
32. P. Fard Moshiri, R. Shahbazian, M. Nabati, and S. A. Ghorashi, “A CSI-based Human Activity Recognition using deep learning,” Sensors (Basel), vol. 21, no. 21, p. 7225, 2021.
33. Ltd, R., 2022. Buy a Raspberry Pi 4 Model B – Raspberry Pi. [online] Raspberry Pi. Available at: <https://www.raspberrypi.com/products/raspberry-pi-4-model-b/> [Accessed 29 July 2022].