



Human activity recognition from multiple sensors data using deep CNNs

Yasin Kaya¹ · Elif Kevser Topuz¹

Received: 8 November 2022 / Revised: 19 April 2023 / Accepted: 10 May 2023 /

Published online: 24 June 2023

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2023

Abstract

Smart devices with sensors now enable continuous measurement of activities of daily living. Accordingly, various human activity recognition (HAR) experiments have been carried out, aiming to convert the measures taken from smart devices into physical activity types. HAR can be applied in many research areas, such as health assessment, environmentally supported living systems, sports, exercise, and security systems. The HAR process can also detect activity-based anomalies in daily life for elderly people. Thus, this study focused on sensor-based activity recognition, and we developed a new 1D-CNN-based deep learning approach to detect human activities. We evaluated our model using raw accelerometer and gyroscope sensor data on three public datasets: UCI-HAPT, WISDM, and PAMAP2. Parameter optimization was employed to define the model's architecture and fine-tune the final design's hyper-parameters. We applied 6, 7, and 12 classes of activity recognition to the UCI-HAPT dataset and obtained accuracy rates of 98%, 96.9%, and 94.8%, respectively. We also achieved an accuracy rate of 97.8% and 90.27% on the WISDM and PAMAP2 datasets, respectively. Moreover, we investigated the impact of using each sensor data individually, and the results show that our model achieved better results using both sensor data concurrently.

Keywords Human activity recognition · 1D-CNN · Deep learning · Signal processing

1 Introduction

Human behavior has fascinated scientists for centuries. With the new era of technology, the approach to this subject has changed significantly, and research has been greatly intensified. The main reason for this is the realization that human behavior plays an essential role in human-computer interaction. Therefore, many research groups have studied topics such as how people behave, what they do, and how they perform activities. Recording, classifying, and

✉ Yasin Kaya
ykaya@atu.edu.tr
Elif Kevser Topuz
ektopuz@atu.edu.tr

¹ Department of Computer Engineering, Adana Alparslan Turkes Science and Technology University, Adana, Turkey

recognizing human activities are among the most remarkable research topics as smart devices, recognition methods, and data processing technologies have been developed. Nowadays, almost everyone has smart devices (smartphones, smartwatches, and music players) equipped with accelerometers, gyroscopes, and magnetometers. These devices make it easier to access and collect valuable data.

Human Activity Recognition (HAR) is used in many research areas, such as health assessment, environmentally supported living systems, smart buildings, sports, exercise, building energy management, and security systems [37, 49]. The HAR process can also detect activity-based anomalies in daily life. Vrigkas et al. [55] have classified human activities into six categories according to their complexity. Behaviors are physical activities related to the person's psychological state and personality. Actions are everything that happens in a given situation. Gesture stands for the primitive movements of a person's body parts, and atomic action stands for a single movement of a person. Group actions are described as actions performed by a group. Finally, human-to-object or human-to-human interactions are movements in which more than one person or object interacts.

HAR research involves recording and analyzing the daily activities of individuals to learn more about human behavior. Researchers have developed several methods to collect and classify activity data. Antar et al. [7] examined the methods used for the activity recognition problem in two groups; video-based activity recognition and sensor-based activity recognition [6, 21, 27, 43, 67]. Sensor-based data can be collected using almost any smart device, while specialized devices are required to collect video-based data.

In our research, we have mainly focused on sensor-based activity recognition. Sensors convert physical quantities into electrical signals, such as heat, distance, vibration, gravity, pressure, position, sound waves, and acceleration. Sensors are used in most everyday devices, such as smartphones, tablets, smart home systems, ships, cars, hospitals, smart-home appliances, aircraft radar systems, and navigation systems. In recent studies, sensor data provides more and more information about people. For example, sensors can detect diseases by monitoring heartbeats and brain signals. In addition, inferences can be made about people and their emotions [1, 24]. Activity classification can be done by recording and analyzing people's activities [17]. Researchers have classified sensor-based activity recognition into three categories: Wearable, environmental, and smartphone sensor-based activity recognition [7]. Wearable sensors are smart sensors that record body movements. The main wearable sensors are biological (ECG and EMG) and smartwatches. Environmental sensors, such as radar, camera, temperature, proximity, and motion sensors, are placed at a specific point in the environment. Smartphone-based sensors can detect changes in motion, changes in the environment, and magnetic fields in cell phones.

HAR processes involve complex steps, including data preprocessing, feature extraction, and classification. HAR datasets include signals from inertial measurement sensors such as accelerometers, gyroscopes, and magnetometers. Data obtained by different sensors and sensors' location on the body affects the HAR accuracy. In the data collection phase, outliers and noisy values can be recorded. Manual feature extraction methods such as size reduction and feature extraction are required for ML algorithms to perform better. These methods are used to find an informative and compressed feature set by generating new features from existing ones. This is the most important part of the HAR process because the performance drops drastically if the features are not suitable [8]. For example, the sitting-to-stand transition may be marked as sitting or standing. Due to the gradient between the labels being so small, such labels' class boundaries also overlap [34, 38].

Due to the extremely high variance in morphology, proper identification of human activities from sensor signals is difficult. To overcome the drawbacks of interpreting signals,

intensive work is being done to develop an automated system for HAR using machine learning (ML) and deep learning (DL) methods. ML has been applied wide range of applications such as malware detection [26], classification of IoT devices [15], emotion recognition from EEG [64], ECG beat classification [5, 18, 19], medical applications [5], and phishing website detection [2]. DL has also been popular in recent years and applied to topics such as cyber security [42], medical applications [16], HAR [11, 34, 38]. One of the most popular DL architectures in the literature is convolutional neural networks (CNN) which implement various fields. HAR [3, 12, 39, 41], image analysis [14], medical applications [32].

In the literature, human activity data have been recorded mainly with accelerometer, gyroscope, and magnetometer sensors, such as PAMAP2 [10], WISDM [35], UCI-HAR, UCI-HAPT [44] datasets. Recent studies show that it is possible to detect human activity using the sensors of smartphones or similar devices and be successful. Most researchers focused on mathematical or statistical feature extraction and feature selection methods of HAR process [13, 23, 50, 66]. Feature extraction became popular that some researchers used the features calculated by the author in the UCI-HAR dataset as an evaluation dataset. However, the computation of these features is very computationally intensive. This is because it requires the analysis of time and frequency domain features such as mean, median, standard deviation, signal magnitude range, interquartile range, correlation coefficient, signal entropy, skewness, kurtosis, band energy, angles between vectors, and weighted averages of frequency components. The UCI-HAR consists of 561 calculated features, and most researchers have used this dataset and applied these features instead of the raw signals. It is obvious that the model runs faster without the feature calculation step. In addition, recent studies have used basic activities such as running, walking, sitting, standing, and lying down. However, it is necessary to study complex postural transition activities to identify human actions or behaviors more accurately.

This study proposes a deep learning model for HAR and avoids the computational cost by using the raw signal as input. We used raw data containing postural transition activities in three widely used public datasets: UCI-HAPT, WISDM, and PAMAP2. We also investigated the effects of individual use of accelerometer and gyroscope sensor data, particularly postural transition activities.

The main contributions of this study can be summarized as follows:

1. We proposed a new Deep Learning model architecture whose parameters were optimized for HAR.
2. The study focused on avoiding computational overhead using the raw signal as input. Therefore, we used raw data that included postural transition activities in three widely used public datasets: UCI-HAPT, WISDM, and PAMAP2.
3. We investigated the effect of using accelerometer and gyroscope sensor data, especially for postural transition activities.
4. Optimization was performed to determine the model's architecture and the final design's hyperparameters.

The remainder of the article is organized as follows: Section 2 summarizes recent work on activity classification. The data description, pre-processing methods, and proposed HAR model are presented in Section 3. The experimental setup, numerical results, and discussion with previous works are described in Section 4 and 5. Finally, the conclusions and future work are presented in Section 6.

2 Related work

This research focuses on sensor-based human activity detection on smartphones and uses the UCI-HAPT [44], WISDM [35], and PAMAP2 [10] datasets to evaluate the proposed approach. In recent studies, traditional machine learning algorithms such as support vector machine (SVM) [4], multi-layer neural network and fuzzy logic [29], multi-layer perceptron (MLP), AdaBoost.M1, random forest (RF), J.48, naive bayes [61] and DL algorithms such as long-short term memory (LSTM) [51], CNN [12, 33, 39], LSTM-CNN [62], deep belief neural network (DBNN) [20], and hybrid methods [39, 54, 70] were used to classify human activities.

The authors used classical machine learning methods such as k-nearest neighbor (k-NN), RF, and CNN-based deep learning methods for HAR. They used the accelerometer time series and computed statistical features as input. They also employed PCA for feature reduction. They tested their models on the UCI-HAR and WISDM datasets with different window sizes and reported that 128 as the window size gave better results on UCI-HAR [23]. Another commonly used dataset in the literature, WISDM, was first presented by [35]. They used classic machine learning methods, namely J48, logistic regression (LR), and MLP, to classify human activities and achieved accuracy rates of 85.1%, 78.1%, and 91.7%, respectively. More specifically, Bilal et al. [9] used time-domain and wavelet-based feature extraction methods to identify human activity. They used the dataset UCI-HAR to evaluate the performance of their model. They implemented IB3, J48, RF, Naive Bayes, and AdaBoost classifiers and achieved accuracy rates of 87.6%, 84.1%, 92.3%, 85.7%, and 95.1%, respectively. The researchers used the algorithms LR, gradient boosting, RF, Gaussian NB, and k-NN for the dataset UCI-HAR. They achieved the best accuracy rate of 96.20% using LR [60].

In addition to traditional ML algorithms, DL approaches are commonly used in the literature. Researchers achieved an accuracy rate of 97% with their divide and conquer based 1D-CNN model [12]. In another study, the authors proposed a model based on a DBNN, and they experimented with 12 activities from the UCI-HAR dataset [20]. They achieved accuracy and an error rate of 95.85% and 4.14%, respectively. Myo et al. [56] used ANN to classify UCI-HAR data and attained a 98.32% accuracy rate. Researchers [50] implemented CNN model-based transfer learning to classify six activities on the UCI-HAR dataset and achieved an accuracy rate of 94%.

Most authors used the same pre-processing steps in these studies with the work [44]. Arigbabu [39] conducted HAR experiments on UCI-HAR and WISDM datasets and proposed approaches testing CNN, RCN, SVM, and hybrid RCN+SVM models. They achieved accuracy rates of 91.9%, 93.8%, 96%, and 97.4% on the UCI-HAR dataset, respectively. Cho and Yoon [12] proposed the divide and conquer-based approach. They conducted tests by dividing the activities into static and dynamic groups. Static activities include sitting, standing, and lying down activities. Dynamic activities consist of walking, upstairs, and downstairs activities. They proposed a two-stage 1D-CNN model to recognize activities: In the first step, a binary 1D-CNN model to classify dynamic and static activities. The second step constructed two 3-class 1D-CNN models to classify individual activities. They achieved an accuracy rate of 97% using six activities from the UCI-HAR dataset.

LSTM, one of the researcher's most preferred deep learning methods, has been widely experienced in HAR studies. Researchers experimented with the hybrid LSTM-CNN model and achieved 95.78%, 95.85%, and 92.63% accuracies on UCI-HAR, WISDM, and opportunity datasets, respectively [62]. In another study, the authors proposed an LSTM model. Using the UCI-HAR dataset, they achieved an accuracy rate of 93.6% and 95.38%, respectively [69].

Using Baseline LSTM, Bidirectional-LSTM (Bidir-LSTM), Residual-LSTM (Res-LSTM), Residual-Bidirectional-LSTM (Res-Bidir-LSTM) models, the researchers attained success rates of 90.8%, 91.1%, 91.6%, and 93.6%, respectively [69]. In a study conducted in 2018 [48], the researchers performed tests with LSTM on PAMAP2, Daphnet Gait, and Skoda datasets and achieved accuracy rates of 89.96%, 83.73%, and 89.03%, respectively.

Kasnesis et al. [41] proposed a hybrid neural network model for the HAR problem, named PerceptionNet, consisting of a 2-layer 1D-CNN and a 1-layer 2D-CNN. They tested the model on UCI HAR and PAMAP2 datasets. In their tests on the UCI-HAR dataset, and obtained 97.25% accuracy. In the PAMAP2 data set, 9 subjects were used as test and validation data and reached 88.56% accuracy. Researchers used Opportunity, PAMAP2, and Order Picking datasets in another study for HAR classification. They proposed two CNN models, conducted tests with different learning rates and batch sizes, and obtained 92.22%, 90.7%, and 93.68% accuracies in Opportunity, PAMAP2, and Order Picking datasets, respectively [3].

Khatun et al. reached 99.93%, 98.76%, and 93.11% accuracies in H-Activity, MHEALTH, and UCI-HAR datasets using the hybrid CNN-LSTM model, respectively [28]. In another study [54], the researchers proposed a dual CNN model and LSTM hybrid model for recognizing the activities. This hybrid model achieved 97.89% accuracy on the UCI-HAR dataset. Kumar and Suresh [33] proposed a hybrid CNN-RNN model, which they named Deep-HAR. Their proposed model, 99.98%, 99.64%, and 99.98% accuracy rates on WISDM, PAMAP2, and KU-HAR datasets, respectively.

Researchers used various feature extraction methods before classification. Researchers [46] applied two different encoding algorithms on the UCI-HAR dataset: The gramian difference angular field (GADF) and the gramian summation angular field (GASF). Using classical machine learning algorithms such as k-NN, DT, SVM, Bayes network, AdaBoost, and RF, they achieved 57%, 47%, 89%, 39%, 54%, and 91% success rates, respectively. They applied an 8-layer deep CNN model to four sets of data created with the accelerometer and gyroscope sensors and achieved an accuracy rate of 94% using the GADF accelerometer data. In another study, the researchers used community empirical mode decomposition (EEMD)-based features. They also implemented feature selection methods (FS) based on game theory. They achieved a performance rate of 76.44% using the k-NN algorithm feeding calculated features [57].

Jain and Kanhangad [25] proposed a model using a feature extraction step that derives features in the time and frequency domains from the raw data of the UCI-HAR dataset. Using multi-class SVM and k-NN algorithms for classification, they achieved a 97.12% performance rate. In another study [20], the authors used median and low-pass Butterworth filters for pre-processing. Then, they applied statistical feature extraction methods such as mean, median, and correlation to the signals and obtained 561 features. Finally, they achieved a classification rate of 95.85% using the proposed Deep Belief Network.

Zhang et al. proposed a feature selection method based on an oppositional and chaos particle swarm optimization (OCPSO) algorithm. They used 1D-CNN, which uses signals in the time and frequency domains, and Deep Decision Fusion (DDF), which combines D-S evidence theory and entropy. The researchers achieved an accuracy rate of 97.81% for the dataset using the DDF method UCI HAR. In another study, the researchers proposed a genetic algorithm (GA) based clustering algorithm called NN-GAR. Their method reduced the size of the UCI-HAR dataset they used for the experiments by 40%, and they achieved 94.6%, 95.8%, and 97.43% accuracies using the k-NN, SVM, and RF algorithms, respectively [34].

In the 2016 study [36], the authors computed a spectrogram representation of the data and used these images to classify human activity in their Deep Learning model. They achieved accuracy rates of 89.37%, 91.5%, and 98.23% on the Skoda, Daphnet, and WISDM datasets,

respectively. In another study using these datasets, the data were preprocessed in the spectral domain before the Deep Learning phase and achieved success rates of 95.8%, 95.3%, and 98.6%, respectively. Pavliuk et al. [45] proposed a deep learning model pre-trained on scalograms. They performed tests with DenseNet configurations with varying numbers of frozen layers. With the pre-trained DenseNet121 model frozen in the first 308 layers, they obtained accuracies of 92.44% and 86.90%, and 97.48% with the UCI-HAPT, a subset of the UCI-HAPT dataset and the KU-HAR datasets, respectively .

Many signal processing methods have proven useful for extracting features from HAR data. For example, time domain features, frequency domain features, and principal component analysis are commonly used [44]. In addition to these works, Tufek et al. [53] did not use feature extraction and a selection step. The researchers divided the raw data into 75% train and 25% test. They experimented with traditional machine learning and deep learning methods such as 1D-CNN+LSTM and DTW+k-NN hybrid models, bidirectional, two- and three-layer LSTM models, and 1D and 2D-CNN. They used a hybrid model and achieved 88.5% accuracy on the UCI-HAPT raw data. Using the proposed 1D-CNN model with 5 layers, they achieved 76.3% accuracy without using a feature extraction and selection step. They achieved 69.1%, 76.3%, 85.2%, 93.7%, 97.4%, 90.3%, and 88.5% with k-NN, 1D-CNN, 2D-CNN, 2-layer LSTM, 3-layer LSTM, bidirectional LSTM, and hybrid 1D-CNN and LSTM methods for raw data.

In another study [40], researchers used a dataset consisting of accelerometer signals that included eight activities. They used a 6-layer CNN model without applying feature extraction to the dataset and achieved a 93.8% success rate. Chakraborty and Mukherjee [11] recorded data on walking and leg-swinging activities of 10 subjects with different body characteristics using accelerometer and gyroscope sensors. They rescaled the data using the z-score method to reduce variability. They used common features such as resultant vectors, statistical features, frequency domain features, and autocorrelation. Using the 1D-CNN model for performance evaluation, they achieved an average accuracy of 97%.

3 Methodology

In our study, the HAR process consisted of 3 stages: Data Pre-processing, Deep Learning, and Activity Classification. Fig. 1 shows these stages. The data obtained from the subjects were passed to the pre-processing step. The signal was passed through median filtering and windowing in this stage and prepared for the Deep Learning step. In the Deep Learning step, the data were first divided into training and testing subsets to compute experimental results. These subsets were used to train and test the proposed model and calculate the performance of the HAR system.

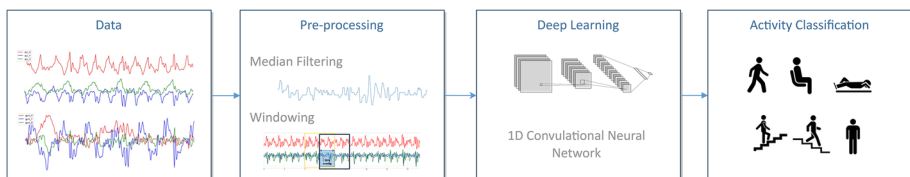


Fig. 1 Activity recognition steps: Data Pre-processing, Deep Learning, and Activity Classification

Sensor data may vary depending on various conditions, such as the characteristics and location of the devices and the physical characteristics of the subjects. In addition, the characteristics of each activity may differ. Therefore, each activity needs to be explicitly adjusted so that it can be better recognized and these differences can be eliminated. The proposed model uses RAW signals as input and reveals the advantages of CNN's feature extraction and classification abilities to identify different types of activities.

3.1 Datasets

In our study, we used the UCI-HAPT [44], WISDM [35], and Physical Activity Monitoring Dataset (PAMAP2) [10] datasets. The UCI-HAPT dataset, which contains accelerometer and gyroscope sensor data from 30 users, was first used in our experiments. The researchers recorded the data in a controlled laboratory environment at 50 Hz. While recording the activities, a waist bag was placed on the user's waist, and the mobile device was placed in the bag in the first phase. The waist bag was attached at chest level or the user's preferred location in the second phase. A video camera recorded the user's activity for data labeling. Six basic activities were recorded: Walking, going upstairs, going downstairs, sitting, standing, and lying down. They also labeled the postural transition data of the six activities: standing-sitting, sitting-standing, sitting-lying, lying-sitting, standing-lying, and lying-standing.

UCI-HAPT dataset was divided into groups according to the number of activities and sensor data to perform experimental tests. Table 1 shows the data groups: Type 1-3 contains a gyroscope and accelerometer data, Type 4-6 contains only accelerometer data, and Type 7-9 contains only gyroscope data. Types 1, 4, and 7 contain six classes; 3, 6, and 9 contain seven, and types 2, 5, and 8 contain 12. The six-activity groups include walking, upward walking, downward walking, sitting, lying, and standing activity data. The twelve activity group includes six transition activities and six basic activities. Finally, seven activity groups consist of 6 basic activities and one postural transition activity (the 7th label includes all transition activities).

We also used the WISDM dataset [35] to evaluate our model. The WISDM dataset was collected by the Wireless Sensor Data Mining Laboratory (WISDM) and consisted of accelerometer data. There are two versions: the WISDM Activity Prediction and the WISDM Actitrackers dataset. In our study, we used the WISDM Activity Prediction dataset. Data from 36 subjects were collected using a smartphone. The sampling frequency was 20 Hz, i.e., a sample was recorded every 50 ms. The dataset contains 1098203 samples, and the total acquisition time was approximately 915 minutes. The data set consists of six activities: walking, jogging/running, Upstairs, Downstairs, sitting, and standing. Their proportions are 38.6%, 31.2%, 11.2%, 9.1%, 5.5%, and 4.4%, respectively.

The last dataset we used for our test was PAMAP2 [10]. The dataset includes 18 physical activities: lying down, sitting, standing, walking, running, cycling, Nordic walking, watching TV, computer work, driving, upstairs, downstairs, vacuum cleaning, ironing, folding laundry,

Table 1 Table caption

	Acc+Gyro	Acc	Gyro
6 Activity	Type 1	Type 4	Type 7
12 Activity	Type 2	Type 5	Type 8
7 Activity	Type 3	Type 6	Type 9

house cleaning, playing soccer, rope jumping, and other transient activities for nine subjects. IMU Sensors, accelerometers, gyroscopes, magnetometers, and temperature sensors were attached to different body parts, such as the wrist, chest, and ankle. The signals were recorded at 100 Hz for at least 1 minute. We used only the chest data from the accelerometer and gyroscope sensors to evaluate our model.

3.2 Data pre-processing

This section explains the data preprocessing steps: median filtering and windowing. The UCI-HAPT and PAMAP2 datasets contained tri-axial angular velocity and linear acceleration signals received from the gyroscope and accelerometer sensors, and their sampling rates were 50 Hz and 100 Hz, respectively. The WISDM dataset contained only one tri-axial acceleration signal, and the sampling rate was 20 Hz. The signals were processed with 3rd order median filtering to reduce noise. Finally, the data were transposed after windowing with 128-size sliding windows, and one-dimensional data were created for the next steps [23].

3.2.1 Median filtering

When recording or acquiring data, noise is generated, negatively affecting signal processing. Therefore, noise in the data must be removed using filters to obtain more accurate results. De-noising is one of the pre-processing stages to improve the results of subsequent stages. Low-pass, median, and Kalman filters are commonly used noise removal methods in the literature [63]. The median filter is a nonlinear digital filtering method that removes noise from a signal or image. The main idea in median filtering is to remove noise from the data by replacing each input value with the median of the neighboring values [22]. A key advantage of median filtering is that this filter can eliminate the effects of input values with significant noise. The authors compared the performance of different filters for sensor-based activity detection [63]. They concluded that the Kalman, median, and low-pass filters were used most to de-noise the signal. Moreover, they found that median filtering can smooth the waveforms and facilitate activity detection.

The output of the median filter is calculated as the median value of the input data within the window centered on the selected point, as in (1).

$$y[t] = \text{median}\{x[i], (i) \in w\} \quad (1)$$

where w demonstrates a predefined neighborhood parameter, centered around location $[t]$ in the signal, $y[t]$ shows the output of the filter, and $x = (x[1], \dots, x[t-1], x[t], x[t+1], \dots, x[N])$ is a one dimensional signal vector. In this work, we used a median filter whose window size is 3.

3.2.2 Windowing operation

In this work, we used the sliding window method, the most widely used technique due to its stability and simplicity [58]. The method divides raw sensor signals into small usable parts before Deep Learning. Previous studies [23, 47] confirm that the window size of 128 provides better performance on average compared to the other window sizes. Therefore, in this study,

we extracted the segments of the same samples with a 50% overlap using a sliding window from the raw signal data of the three-axis gyroscope and accelerometer on the UCI-HAPT and PAMAP2 datasets. We tested our model on the WISDM dataset using 128 samples as the sliding window size, even though it contains only accelerometer signals.

The signal acquired from the inertial measurement unit (IMU) provides important information about human gait. We can use an accelerometer signal to predict the temporal characteristics of gait and a gyroscope signal to obtain information about angular velocity. An accelerometer signal works well to identify intense human activities such as walking, running, and jumping. In contrast, a gyroscope signal works well for sedentary activities such as sitting, standing, lying, walking upstairs and downstairs, or activities that depend on orientation [47]. Therefore, in this study, we tested both sensors simultaneously and separately in an experimental test. We constructed a 6×128 sliced signal representing an action and concatenated it to one dimension as 1×768 at UCI-HAPT. We extracted 12637 sample actions from the database and used two label columns to identify 6 and 12 classes of activities. As a result, we created a 12637×770 dataset for experimental testing by combining all windows. We constructed a 3×128 windowed signal representing an action and concatenated it to a dimension of 1×384 in the WISDM dataset. By combining all the windows, we created a dataset of size 10981×384 for experimental testing. In addition to these datasets, we constructed a 6×128 signal representing an action, concatenated it to a dimension as 1×768 on the PAMAP2 dataset, and constructed a 30067×770 dataset.

3.3 Proposed 1D-CNN model

A Convolutional Neural Network (CNN) is a deep learning architecture commonly used for image classification or analysis. In addition to image analysis problems, CNNs are also used for 1-dimensional signal classification problems and have shown promising results [12, 20, 56].

This network architecture includes convolutional layers, pooling operations, and fully connected layers to perform prediction and classification operations. The convolutional layers use various feature filters on the input data until all convolutional layers are formed and produce a vector output to generate feature maps of the input data and send the results to the network. In this layer, the non-linearity is achieved by an activation function such as ReLU, tanh, or sigmoid on the feature maps. The function of the pooling layer, which is used between the convolutional layers, is to reduce the number of parameters and computations in the network and to control over-fitting. Max-pooling and average-pooling are the most commonly used types. They adjust the data size by taking the maximum or average of each cluster in the feature map. The data obtained from the convolutional layers must be resized as one-dimensional in the flattening layer to feed the next layer with the data obtained from the convolutional layers. In the fully connected layer, as in conventional neural networks, each neuron produces an output by applying a function to the values of the previous layer.

Due to the difference in computational complexity between 1D and 2D CNNs, 1D-CNNs are often preferred because they are more advantageous when working with 1D signals. When 2D-CNN and 1D-CNN are applied to $n \times n$ dimensional data, the computational complexity is $O(n^2 \times k^2)$ and $O(n \times k)$, respectively [30]. Therefore, our study proposes the 1D-CNN model, whose layers and parameters are detailed in Table 2. The proposed model consists of four convolutional layers, one flattened layer, and three fully connected layers. Hyper-

Table 2 Proposed 1D-CNN architecture

Proposed 1D-CNN architecture design	
1:	CNN(X_train, Y_train, X_test, Y_test, learning rate=0.001, epoch=100, batch_size=64)
2:	inputs=shape(data points)
3:	model ← Conv1D(filters=256, kernels=1, activation=ReLU)(input)
4:	model ← MaxPooling(model)
5:	model ← BatchNormalization()(model)
6:	model ← Conv1D(filters=128, kernels=1, activation=ReLU)(input)
7:	model ← MaxPooling(model)
8:	model ← BatchNormalization()(model)
9:	model ← Conv1D(filters=128, kernels=1, activation=ReLU)(input)
10:	model ← MaxPooling(model)
11:	model ← BatchNormalization()(model)
12:	model ← Conv1D(filters=64, kernels=1, activation=ReLU)(input)
13:	model ← MaxPooling(model)
14:	model ← BatchNormalization()(model)
15:	model ← Flatten()(model)
16:	model ← Dense(neurons=1024, activation=ReLU)(model)
17:	model ← BatchNormalization(model)
18:	model ← Dropout(0.5)(model)
19:	model ← Dense(neurons=2048, activation=ReLU)(model)
20:	model ← BatchNormalization(model)
21:	model ← Dropout(0.5)(model)
22:	model ← Dense(neurons=activity_number, activation=softmax)(model)
23:	model.compile(loss=BinaryCrossEntropy, optimizer=Adam, learningrate)
24:	model.fit(X_train, Y_train, epoch, batchsize)
25:	Accuracy=model.evaluate(X_test, Y_test)
26:	return Accuracy

parameters of an artificial neural network express the topology of this network. For example, a neural network includes hyper-parameters such as batch- size, the optimization algorithm, the activation function, the epoch size, and design parameters such as the number of convolutional layers and the filter shape of these, and the number of neurons in the fully connected layers. We optimized hyper-parameters with different learning rates, epochs, optimizers, and batch sizes to build an effective CNN model and minimize the loss function. In addition, we determined the size and depth of the convolutional layers and the number of neurons in the fully connected layers using model optimization.

4 Results and discussion

This section focuses on the performance evaluation of the proposed 1D-CNN model. We implemented our model on a PC with a 3.7 GHz Intel Xeon processor and 64GB memory to compute the experimental results. We performed hyperparameter tuning and design parameter optimization using the UCI-HAPT dataset. Thus, we obtained an optimized model and evaluated our model with 10-fold cross-validation on all three datasets. After parameter optimization, we tested our fine-tuned model with all data sets defined in Table 2.

4.1 Evaluation metrics

We evaluated our model using the following evaluation metrics, namely F1-score, specificity, sensitivity, and accuracy, as in (2-5).

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (2)$$

$$\text{Sensitivity} = \frac{TP}{TP + FN} \quad (3)$$

$$\text{Specificity} = \frac{TN}{TN + FP} \quad (4)$$

$$\text{F1-Score} = \frac{TP}{TP + \frac{1}{2}(FP + FN)} \quad (5)$$

where False Positive (FP) represents the case where the algorithm calculates NO when the actual state is YES, False Negative (FN) depicts the case where the output of the algorithm is YES when the actual state is NO, True Positive (TP) shows the case where both the output of the algorithm and the actual state are YES, True Negative (TN) is the case where both the output of the algorithm and the actual state are NO.

4.2 Determining hyper-parameter and the architecture of the main design

To determine the hyper-parameters and design parameters of the proposed model, we performed the following hyper-parameter tuning and design optimization. These parameters are batch size, optimizer, learning rate, hidden unit size, epoch, convolutional layer depth, convolutional layer filter size, and fully connected layer output filter size. We used the accuracy metric as fitness to optimize these parameters for training the proposed model on the UCI-HAPT dataset. We evaluated the model's parameters using the same subjects as training and testing as in [41, 44].

Updating the weighting parameters in deep learning is done by deriving backward derivatives (back-propagation) and multiplying these derivatives by the learning rate parameter

Table 3 Accuracy and loss values for different learning rates

Learning Rate	Accuracy	Loss
0.1	0.9279	0.0659
0.01	0.9171	0.0776
0.001	0.9308	0.0465

Table 4 Model performance calculated using different batch sizes

Batch size	Accuracy	Loss
64	0.9308	0.0465
128	0.9131	0.0561
256	0.9221	0.0392
512	0.9308	0.0418

(LR). A high learning rate leads to over-fitting (excessive memorization). Conversely, training takes a long time because it proceeds in small steps when the LR is small. In this context, we used different LR values, such as 0.1, 0.01, and 0.001, to determine a better LR. The results showed that these LR values achieved accuracy rates of 92.79%, 91.71%, and 93.08%, respectively, and we decided to use 0.001 as LR. These results can be seen in Table 3.

The data are divided into small groups (batches) to avoid the hassle of training large datasets and to train our model more efficiently. The batch size parameter, set when the model is developed, specifies how much data is processed at once. We tested the batch size at 64, 128, 256, and 512 and achieved accuracy rates of 93.08%, 91.31%, 92.21%, and 93.08%, respectively. As you can see from Table 4, the results show that using 64 and 512 gives the same accuracy rates. Due to the small size of the transition activities, we have unbalanced data. Therefore, we determined the batch size to be 64.

Another hyper-parameter to be determined that best fits our model is the optimizer. We tested Adam, Adamax, Stochastic Gradient Descent (SGD), Adagrad, and Rmsprop for the optimizer. We achieved accuracy rates of 93.08%, 93%, 87.91%, 83.16%, and 92.92%, respectively. The results show that the Adam and Adamax algorithms achieve very similar results. Therefore, we decided to use the Adam optimizer. Table 5 shows the accuracy and loss values for these parameters.

One of the main problems of the training process is the number of epochs of the proposed model. The system cannot guarantee sufficient generalizability if the data are under-fitted or over-fitted. In general, the accuracy increases as the number of epochs increases. After a certain value of the training epoch, the model memorizes the data. We trained the proposed model with 250 epochs to observe the change in accuracy and loss values and determine the best epoch size using 20 percent of the training data as validation data. After the 100th epoch, the loss and accuracy values almost flattened out, and we chose epoch 100, so we ran all previous tests with 100 epochs.

As can be seen in Table 2, our model consists of convolutional layers followed by two fully connected layers. To investigate the depth of our model, we conducted experiments with different numbers of convolutional layers ranging from 2 to 8 and fixed the number of convolutional layers to 4. We also tested the size of the output filters of each convolutional

Table 5 Model performance achieved using different optimizers

Optimizer	Accuracy	Loss
Adam	0.9308	0.0465
Adamax	0.9300	0.0418
AdaGrad	0.8791	0.1119
SGD	0.8316	0.1329
RMSprop	0.9292	0.0544

Table 6 Accuracy and loss values for different output unit sizes

Hidden unit size	Accuracy	Loss
64-128	0.9342	0.0478
128-256	0.9340	0.0490
256-512	0.9342	0.0472
512-1024	0.9371	0.0479
1024-2048	0.9308	0.0465

layer between 32 and 4096 and determined the dimensions 256, 128, 128, and 64, respectively, for each convolutional layer. In addition, we tested different output sizes between 64 and 2048 to determine the output size of the fully connected layers. As a result, we obtained the best test accuracy and the lowest loss value by using 1024-2048 hidden unit sizes for fully concatenated layers. Tables 6 and 7 show these results in detail.

To summarize hyper-parameter optimization and determine model architecture processes, optimum parameters for batch size, optimizer, learning rate, hidden unit size, epoch, dept of convolution layers, filter sizes of convolution layer values were determined as 64, Adam, 0.001, 1024-2048, 100, 4, and 256-128-128-64, respectively.

4.3 Complexity and overhead analysis

1D-CNN provides advantages over 2D-CNN and other DL architectures, including a faster training procedure, lower computational cost, strong performance on small data sets, and the ability to extract key features sequentially. When an image with dimensions $n \times n$ is convolved with the $k \times k$ kernel, the computational complexity in 2D convolution is $O(n^2 \times k^2)$. Considering 1D signals, 1D convolution is applied, and its computational complexity is $O(n \times k)$. This means that the computational complexity of a 1D-CNN is much lower than that of a 2D-CNN under the same conditions [30].

The total number of parameters in a convolutional layer is calculated as $((m \times n \times p) + 1) \times f$, added 1 because of the bias term for each filter. Where m is the shape of the width

Table 7 Accuracy and loss values for different dept of hidden unit sizes

Depth of Hidden unit size	Depth	Accuracy	Loss
128-64	2	0.9255	0.0397
32-64-128	3	0.9276	0.0551
64-128-256	3	0.9255	0.0320
256-128-64	3	0.9271	0.0369
32-64-128-256	4	0.9290	0.0526
64-128-128-256	4	0.9345	0.0412
512-256-128-64	4	0.9245	0.0408
256-128-128-64	4	0.9308	0.0465
1024-512-256-128-64	5	0.7767	0.1594
2048-1024-512-256-128-64	6	0.9168	0.0513
32-64-128-256-512-1024-2048-4096	8	0.9221	0.0518

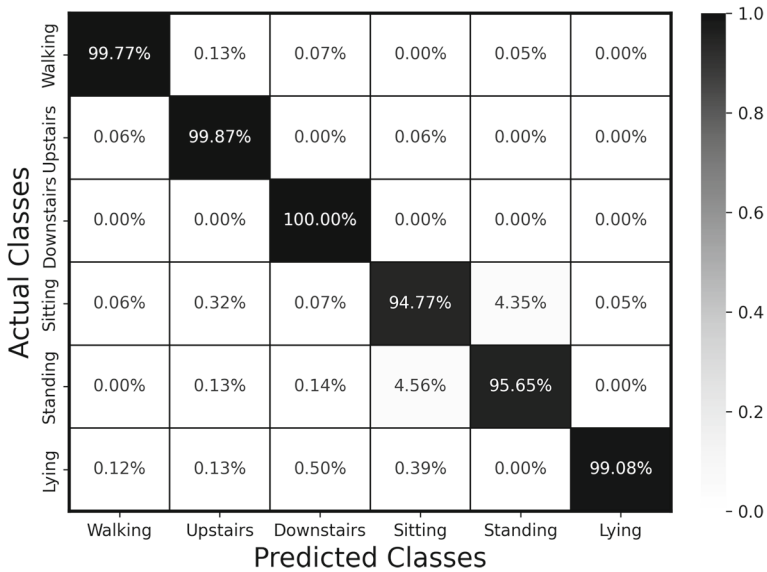


Fig. 2 Confusion Matrix for Type1 (6 activities)

of the filter, n is the shape of the height of the filter, p represents the number of filters in the previous layer, f means the number of filters, the filter refers to the number of filters in the current layer [52].

The trainable parameters of the proposed 1D-CNN model are 3225810, and the non-trainable parameters are 7308 after hyperparameter tuning. These parameters may differ depending on the dept and filter sizes of convolution layers and the number of convolution and dense layers.

4.4 Evaluation of sensors

Table 8 Classification results for different sensors

Dataset	F1-score	Sensitivity	Specificity	Accuracy
UCI-HAPT Type1	97%	97%	99.37%	96.95%
UCI-HAPT Type2	81%	80%	99.34%	93.08%
UCI-HAPT Type3	95%	95%	99.17%	95.03%
UCI-HAPT Type4	92%	92%	98.31%	91.65%
UCI-HAPT Type5	63%	70%	98.16%	79.73%
UCI-HAPT Type6	91%	91%	98.50%	91,10%
UCI-HAPT Type7	79%	80%	95.80%	79.01%
UCI-HAPT Type8	72%	70%	98.16%	81.18%
UCI-HAPT Type9	78%	79%	96.32%	78.41%
WISDM	98%	98%	98%	97.75%
PAMAP2	92%	92%	92%	91.92%

Table 9 Results for 10-fold cross-validation

Dataset	F1-score	Sensitivity	Specificity	Accuracy
UCI-HAPT Type 1	98%	98%	97.8%	98%
UCI-HAPT Type 2	96.9%	96.9%	96.8%	96.9%
UCI-HAPT Type 3	94.8%	94.8%	95%	94.8%
WISDM	97.8%	97.8%	98%	97.8%
PAMAP2	90.2%	90.2%	90%	90.2%

In our experimental tests, we first analyzed the effects of using the accelerometer and gyroscope sensors separately or simultaneously. In these analyzes, we used the classical splitting method, where 70% of the data is used as training and 30% of the data is used as testing to calculate the results. We used the same subjects for test and training as in [41, 44]. As seen in Table 8, using the accelerometer consistently achieved better performance in terms of F1-score, sensitivity, specificity, and accuracy than using the gyroscope. This result indicates that the accelerometer is more distinctive than the gyroscope sensors in HAR. The WISDM dataset contains only accelerometer data. Table 8 shows that a classification accuracy of 97.75% can be achieved with this sensor at WISDM.

Moreover, the combination of two sensor data provides higher recognition results. Specifically, we achieved a maximum accuracy of 91.65% for six activities and 81.18% for 12 activities when using only the accelerometer and gyroscope, respectively. Furthermore, we achieved a maximum recognition rate of 96.95% when we fused the fusion data from two sensors. Thus, we used two sensor data simultaneously to investigate the accuracy of our model in our experiments for the UCI-HAPT and PAMAP2 datasets. Table 8 shows that a classification accuracy of 91.92% can be achieved with these sensors for PAMAP2.

4.5 Classification results

To test the performance of our model on different datasets, we performed 10-fold cross-validation. Table 9 shows the accuracy, sensitivity, specificity, and F1-score for the experimental results in four different data groups. As shown in Tables 8 and 9, cross-validation increased the performance rates of the proposed model from 96.95% to 98%. For the WISDM dataset, we achieved an accuracy rate of 97.8%. Figures 2-5 show the average confusion matrix for the 10-fold cross-validation of UCI-HAPT types 1-3 and the WISDM dataset. We have unbalanced data for the UCI-HAPT dataset, especially for the transition activities numbered 7 to 12. From Fig. 3, it can be seen that the transition activities have lower correct classification rates. The model could not be trained sufficiently for these activities due to the small amount of data. In particular, the activities "sitting-lying" and "standing-lying" were the most confusing for the trained model in this group due to their similarity. As shown in Fig. 5, the static activities of standing and sitting had a higher misclassification rate on the WISDM dataset.

On the other hand, if we look at the results of the PAMAP2 dataset, which also contains homogeneously distributed activities of daily living, we can see from Fig. 6 that the activity recognition rates are quite successful.

Table 10 The results of the State-of-the-Art approaches and the proposed model for HAR

Study	Dataset	Method	Acc.	Raw	# Act.
Antar et al. [4]	UCI-HAR	MC-HF-SVM	89.0%	✓	6
Moya Rueda et al. [41]	UCI-HAR	PerceptionNet	97.25%	✓	6
Moya Rueda et al. [41]	PAMAP2	PerceptionNet	88.56%	✓	12
Liu et al. [39]	UCI-HAR	RCN+SVM	97.4%	X	6
Zeng et al. [66]	UCI-HAR	U-Net	98.4%	X	6
Ignatov [23]	UCI-HAR	CNN	97.25%	X	6
Ramachandran and Pang [50]	UCI-HAR	CNN+Transfer Learning	94%	X	6
Cho and Yoon [12]	UCI-HAR	Divide and Conquer-Based 1D-CNN	97%	X	6
Zhao et al. [69]	UCI-HAR	Res-Bidir-LSTM	93.6%	X	6
Khatun et al. [28]	UCI-HAR	CNN+LSTM	93.11%	X	6
Zhang et al. [68]	UCI-HAR	DDF	97.81%	X	6
Kumar and Suresh [34]	UCI-HAR	RF	97.43%	X	6
Xia et al. [62]	UCI-HAPT	LSTM+CNN	95.78%	✓	6
Zeng et al. [66]	UCI-HAPT	U-Net	93.1%	X	12
Tufek et al. [53]	UCI-HAPT	1D-CNN	76.3%	✓	6
Hassan et al. [20]	UCI-HAPT	Deep Belief Network	95.85%	X	12
Xia et al. [62]	WISDM	LSTM+CNN	95.85%	✓	6
Liu et al. [39]	WISDM	RCN+SVM	91.5%	✓	6
Zeng et al. [66]	WISDM	U-net	97%	X	6
Kiranyaz et al. [31]	WISDM	CNN	96.88%	✓	6
Ignatov [23]	WISDM	CNN	93.32%	✓	6
Ramachandran and Pang [50]	WISDM	CNN+Transfer Learning	92.3%	✓	6
Permatasari et al. [48]	PAMAP2	LSTM	89.96%	X	12
Anguita et al. [3]	PAMAP2	CNN	93.68%	X	12

Table 10 continued

Study	Dataset	Method	Acc.	Raw	# Act.
Proposed Model	UCI-HAPT Type 1	ID-CNN	98%	✓	6
Proposed Model	UCI-HAPT Type 2	ID-CNN	94.8%	✓	12
Proposed Model	UCI-HAPT Type 3	ID-CNN	96.9%	✓	7
Proposed Model	WISDM	ID-CNN	97.8%	✓	6
Proposed Model	PAMAP2	ID-CNN	90.24%	✓	12

Bold values are the results of the proposed models

5 Discussion

Table 10 shows the results of recent HAR studies. In the raw data column of Table 10, "✓" indicates the use of raw data, and "X" indicates the use of calculated features for HAR. Most researchers have used well-known datasets to detect human activities. In addition, they have applied some complex feature extraction methods using mathematical or statistical techniques [12, 23, 39, 50, 59, 62, 66, 69]. We used raw data in our experiments with both datasets to avoid computational overhead. Specifically for the WISDM dataset, researchers also used RAW data because the dataset does not contain hand-generated features [23, 31, 41, 53]. The results show that activity recognition models using a deep learning architecture significantly improve accuracy by extracting hierarchical features from triaxial sensor data. In [66], the authors achieved high accuracy rates of 93.1%, 98.4%, and 97% by using eigenvalues as features to feed the U-Net with the datasets UCI-HAPT, UCI-HAR, and WISDM, respectively. It can be clearly seen that the authors achieved good results of 98.4% when they used manually created features, but this dropped to 93.1% when they used raw data to calculate their features for the UCI-HAPT dataset. In another study, the authors used a hybrid model with LSTM and CNN using raw data for the UCI-HAR and WISDM datasets and achieved accuracy rates of 95.78% and 95.85%, respectively. It can be seen from Table 10 that the proposed model achieved the highest results using RAW data on all datasets.

6 Conclusion

This study proposed a novel 1D-CNN-based deep neural network that uses smartphone-based sensor data to recognize human activities. The proposed model was evaluated using the UCI-HAPT, WISDM, and PAMAP2 datasets. Raw accelerometer and gyroscope sensor data were used to evaluate the model. We also investigated the effects of hyper-parameters and design

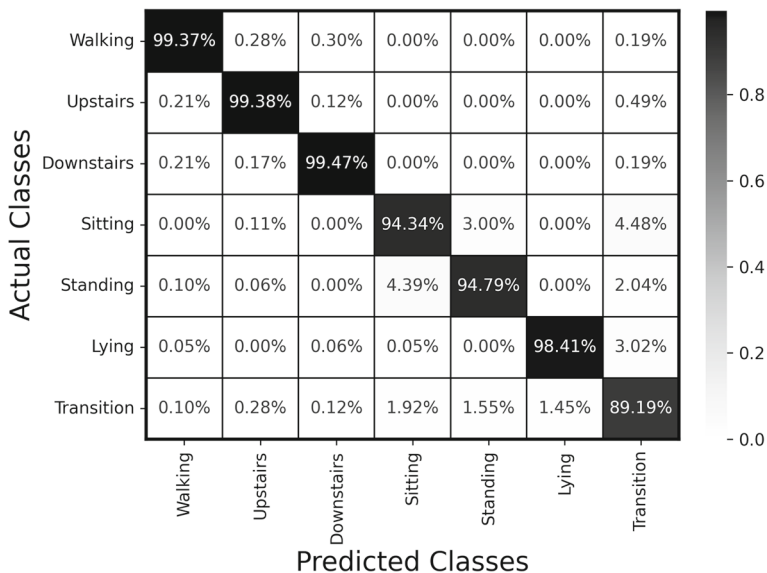


Fig. 3 Confusion Matrix for Type3 (7 activities)

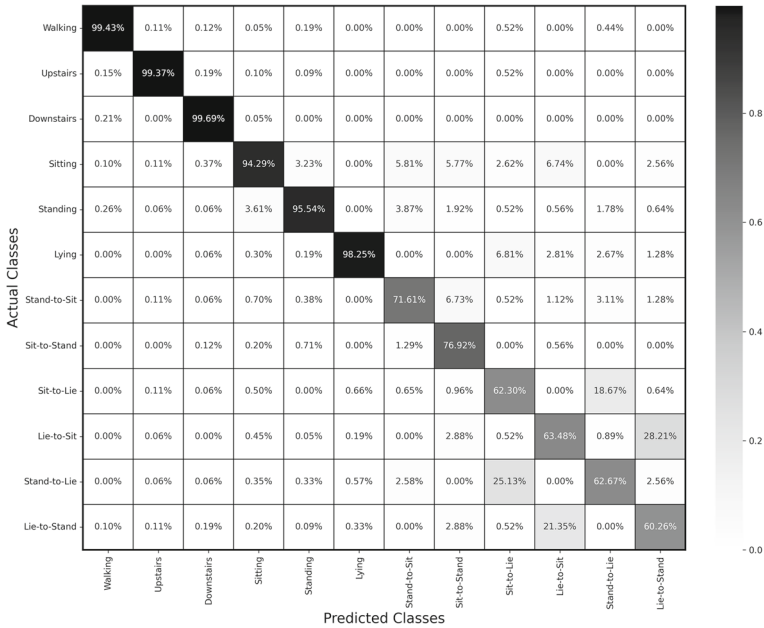


Fig. 4 Confusion Matrix for Type2 (12 activities)

parameters of the final model. In addition, we investigated the effects of using the accelerometer and gyroscope sensors simultaneously or alone for HAR. The results showed that using the sensors together achieved acceptable performance. The proposed model obtained promis-

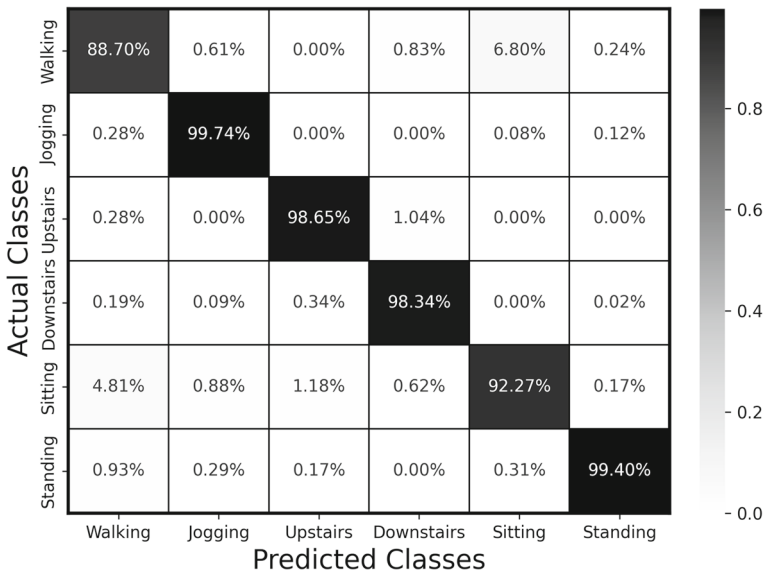


Fig. 5 Confusion Matrix for WISDM dataset

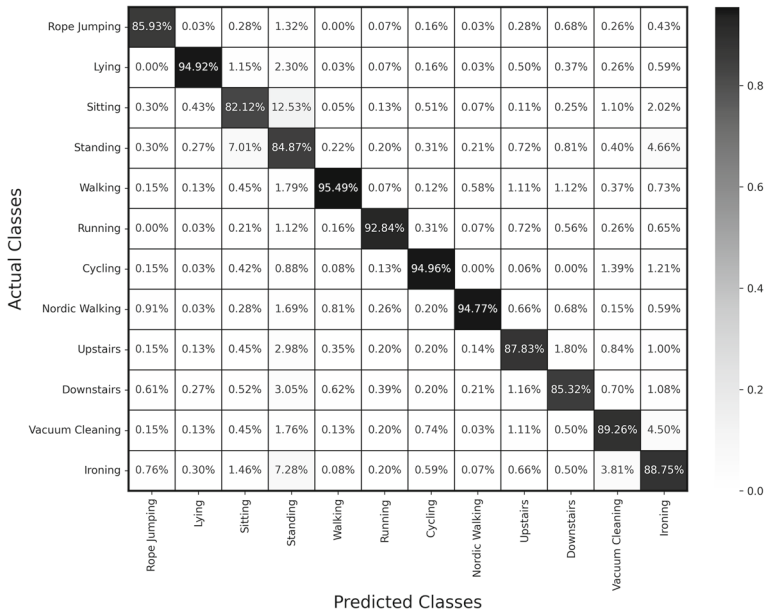


Fig. 6 Confusion Matrix for PAMAP2 dataset

ing results compared to other studies using raw signals. However, the unbalanced data makes the results slightly worse for the transition activities. This issue can be solved in the future by recording more transition data or expanding the data using data augmentation techniques. In the future, the proposed method can also be explored to classify transition activities accurately.

Due to the low computational cost, the proposed model can be applied to various signal-processing applications such as medical applications, ECG [5] and EEG analysis [64], speech emotion recognition [65], anomaly detection, and structural health monitoring [30].

Author Contributions The authors contributed equally to this work

Funding No funds, grants, or other support was received.

Data Availability The data used in this study was cited in the text.

Declarations

Conflicts of interest The authors have no conflicts of interest to declare relevant to this article’s content.

References

1. Acharya UR, Oh SL, Hagiwara Y, Tan JH, Adeli H (2018) Deep convolutional neural network for the automated detection and diagnosis of seizure using eeg signals. *Computers in biology and medicine* 100:270–278. <https://doi.org/10.1016/j.combiomed.2017.09.017>
2. Almomani A, Alauthman M, Shatnawi MT, Alweshah M, Alrosan A, Alomoush W, Gupta BB (2022) Phishing website detection with semantic features based on machine learning classifiers: A comparative

- study. *International Journal on Semantic Web and Information Systems (IJSWIS)* 18(1):1–24. <https://doi.org/10.4018/IJSWIS.297032>
3. Anguita D, Ghio A, Oneto L, Parra X, Reyes-Ortiz JL (2012) Human activity recognition on smartphones using a multiclass hardware-friendly support vector machine. In: *International Workshop on Ambient Assisted Living*, pp 216–223 https://doi.org/10.1007/978-3-642-35395-6_30. Springer
 4. Antar AD, Ahmed M, Ahad MAR (2019) Challenges in sensor-based human activity recognition and a comparative analysis of benchmark datasets: a review, In: 2019 Joint 8th International Conference on Informatics, Electronics & Vision (ICIEV) and 2019 3rd International Conference on Imaging, Vision & Pattern Recognition (icIVPR), pp 134–139, IEEE <https://doi.org/10.1109/ICIEV.2019.8858508>
 5. Arigbabu OA (2020) Entropy decision fusion for smartphone sensor based human activity recognition. arXiv preprint [arXiv:2006.00367](https://arxiv.org/abs/2006.00367). <https://doi.org/10.48550/arXiv.2006.00367>
 6. Asghari P, Soleimani E, Nazerfard E (2020) Online human activity recognition employing hierarchical hidden markov models. *Journal of Ambient Intelligence and Humanized Computing* 11(3):1141–1152. <https://doi.org/10.1007/s12652-019-01380-5>
 7. Asghari P, Soleimani E, Nazerfard E (2020) Online human activity recognition employing hierarchical hidden markov models, *Journal of Ambient Intelligence and Humanized Computing* 11(3):1141–1152 <https://doi.org/10.1007/s12652-019-01380-5>
 8. Balaha HM, Hassan AES (2023) Comprehensive machine and deep learning analysis of sensor-based human activity recognition. *Neural Computing and Applications*. <https://doi.org/10.1007/s00521-023-08374-7>
 9. Bilal M, Shaikh FK, Arif M, Wyne MF (2019) A revised framework of machine learning application for optimal activity recognition. *Cluster Computing* 22(3):7257–7273. <https://doi.org/10.1007/s10586-017-1212-x>
 10. Bilal M, Shaikh FK, Arif M, Wyne MF (2019) A revised framework of machine learning application for optimal activity recognition. *Cluster Computing* 22(3):7257–7273 <https://doi.org/10.1007/s10586-017-1212-x>
 11. Chakraborty A, Mukherjee N (2023) A deep-cnn based low-cost, multi-modal sensing system for efficient walking activity identification. *Multimedia Tools and Applications* 82(11):16741–16766. <https://doi.org/10.1007/s11042-022-13990-x>
 12. Cho H, Yoon SM (2018) Divide and conquer-based 1d cnn human activity recognition using test data sharpening. *Sensors* 18(4):1055. <https://doi.org/10.3390/s18041055>
 13. Cho H, Yoon SM (2018) Divide and conquer-based 1d cnn human activity recognition using test data sharpening. *Sensors* 18(4):1055 <https://doi.org/10.3390/s18041055>
 14. Clarkson B, Mase K, Pentland A (2000) Recognizing user context via wearable sensors. In: *Digest of Papers. Fourth International Symposium on Wearable Computers*, pp 69–75 <https://doi.org/10.1109/ISWC.2000.888467>, IEEE
 15. Cvitić I, Peraković D, Periša M, Gupta B (2021) Ensemble machine learning approach for classification of iot devices in smart home. *International Journal of Machine Learning and Cybernetics* 12(11):3179–3202. <https://doi.org/10.1007/s13042-020-01241-0>
 16. Dinç B, Kaya Y (2023) A novel hybrid optic disc detection and fovea localization method integrating region-based convnet and mathematical approach. *Wireless Personal Communications* 129(4):2727–2748. <https://doi.org/10.1007/s11277-023-10255-0>
 17. Gaurav A, Gupta BB, Panigrahi PK (2023) A comprehensive survey on machine learning approaches for malware detection in iot-based enterprise information system. *Enterprise Information Systems* 17(3):2023764 <https://doi.org/10.1080/17517575.2021.2023764>
 18. Gupta V (2023) Application of chaos theory for arrhythmia detection in pathological databases. *International Journal of Medical Engineering and Informatics* 15(2):191–202. <https://doi.org/10.1504/IJMEI.2023.10051949>
 19. Gupta V, Mittal M, Mittal V (2022) A simplistic and novel technique for ecg signal pre-processing. *IETE Journal of Research* 1–12. <https://doi.org/10.1080/03772063.2022.2135622>
 20. Hassan MM, Uddin MZ, Mohamed A, Almogren A (2018) A robust human activity recognition system using smartphone sensors and deep learning. *Future Generation Computer Systems* 81:307–313. <https://doi.org/10.1016/j.future.2017.11.029>
 21. He X, Zhu J, Su W, Tentzeris MM (2020) Rfid based non-contact human activity detection exploiting cross polarization. *IEEE Access* 8:46585–46595. <https://doi.org/10.1109/ACCESS.2020.2979080>
 22. He X, Zhu J, Su W, Tentzeris MM (2020) Rfid based non-contact human activity detection exploiting cross polarization. *IEEE Access* vol 8, pp 46585–46595 <https://doi.org/10.1109/ACCESS.2020.2979080>
 23. Ignatov A (2018) Real-time human activity recognition from accelerometer data using convolutional neural networks. *Applied Soft Computing* 62:915–922. <https://doi.org/10.1016/j.asoc.2017.09.027>

24. Irvine N, Nugent C, Zhang S, Wang H, Ng WW (2020) Neural network ensembles for sensor-based human activity recognition within smart environments. *Sensors* 20(1):216. <https://doi.org/10.3390/s20010216>
25. Jain A, Kanhangad V (2017) Human activity classification in smartphones using accelerometer and gyroscope sensors. *IEEE Sensors Journal* 18(3):1169–1177. <https://doi.org/10.1109/JSEN.2017.2782492>
26. Kaya Y (2021) Detection of bundle branch block using higher order statistics and temporal features. *Int. Arab J. Inf. Technol.* 18(3):279–285 <https://doi.org/10.34028/iajit/18/3/3>
27. Khan NS, Ghani MS (2021) A survey of deep learning based models for human activity recognition. *Wireless Personal Communications*, pp 1–43 <https://doi.org/10.1007/s11277-021-08525-w>
28. Khatun MA, Yousuf MA, Ahmed S, Uddin MZ, Alyami SA, Al-Ashhab S, Akhdar HF, Khan A, Azad A, Moni MA (2022) Deep cnn-lstm with self-attention model for human activity recognition using wearable sensor. *IEEE Journal of Translational Engineering in Health and Medicine* 10:1–16. <https://doi.org/10.1109/JTEHM.2022.3177710>
29. Kim E, Helal S (2014) Training-free fuzzy logic based human activity recognition. *JIPS* 10(3):335–354. <https://doi.org/10.3745/JIPS.04.0005>
30. Kiranyaz S, Avci O, Abdeljaber O, Ince T, Gabbouj M, Inman DJ (2021) 1d convolutional neural networks and applications: A survey. *Mechanical systems and signal processing* 151:107398. <https://doi.org/10.1016/j.ymsp.2020.107398>
31. Kiranyaz S, Avci O, Abdeljaber O, Ince T, Gabbouj M, Inman DJ (2021) 1d convolutional neural networks and applications: A survey. *Mechanical systems and signal processing vol 151*, p 107398 <https://doi.org/10.1016/j.ymsp.2020.107398>
32. Kıymaç E, Kaya Y (2023) A novel automated cnn arrhythmia classifier with memory-enhanced artificial hummingbird algorithm. *Expert Systems with Applications* 213:119162. <https://doi.org/10.1016/j.eswa.2022.119162>
33. Kumar P, Suresh S (2023) Deep-har: an ensemble deep learning model for recognizing the simple, complex, and heterogeneous human activities. *Multimedia Tools and Applications*. <https://doi.org/10.1007/s11042-023-14492-0>
34. Kumar P, Suresh S (2023) Deep-har: an ensemble deep learning model for recognizing the simple, complex, and heterogeneous human activities. *Multimedia Tools and Applications* <https://doi.org/10.1007/s11042-023-14492-0>
35. Kwapisz JR, Weiss GM, Moore SA (2011) Activity recognition using cell phone accelerometers. *ACM SIGKDD Explorations Newsletter* 12(2):74–82. <https://doi.org/10.1145/1964897.1964918>
36. Lara OD, Labrador MA (2012) A survey on human activity recognition using wearable sensors. *IEEE communications surveys & tutorials* 15(3):1192–1209 <https://doi.org/10.1109/SURV.2012.110112.00192>
37. Lara OD, Labrador MA (2012) A survey on human activity recognition using wearable sensors. *IEEE communications surveys & tutorials* 15(3):1192–1209. <https://doi.org/10.1109/SURV.2012.110112.00192>
38. Li Y, Yang G, Su Z, Li S, Wang Y (2023) Human activity recognition based on multi-environment sensor data. *Information Fusion* 91:47–63. <https://doi.org/10.1016/j.inffus.2022.10.015>
39. Liu J, Spakowicz DJ, Ash GI, Hoyd R, Zhang A, Lou S, Lee D, Zhang J, Presley C, Greene A (2020) Bayesian structural time series for biomedical sensor data: A flexible modeling framework for evaluating interventions. *bioRxiv* <https://doi.org/10.1371/journal.pcbi.1009303>
40. Li Y, Yang G, Su Z, Li S, Wang Y (2023) Human activity recognition based on multi-environment sensor data. *Information Fusion vol 91*, pp 47–63 <https://doi.org/10.1016/j.inffus.2022.10.015>
41. Moya Rueda F, Grzeszick R, Fink GA, Feldhorst S, Ten Hompel M (2018) Convolutional neural networks for human activity recognition using body-worn sensors. In: *Informatics*, vol 5, p 26 <https://doi.org/10.3390/informatics5020026>. Multidisciplinary Digital Publishing Institute
42. Mughaid A, AlZu'bi S, Alnajjar A, AbuElsoud E, El Salhi S, Igried B, Abugaligah L (2023) Correction to: Improved dropping attacks detecting system in 5g networks using machine learning and deep learning approaches. *Multimedia Tools and Applications* 82(9):13997–13998. <https://doi.org/10.1007/s11042-022-14059-5>
43. Nafea O, Abdul W, Muhammad G (2022) Multi-sensor human activity recognition using cnn and gru. *International Journal of Multimedia Information Retrieval* 11(2):135–147. <https://doi.org/10.1007/s13735-022-00234-9>
44. Nafea O, Abdul W, Muhammad G (2022) Multi-sensor human activity recognition using cnn and gru. *International Journal of Multimedia Information Retrieval* 11(2):135–147 <https://doi.org/10.1007/s13735-022-00234-9>
45. Pavliuk O, Mishchuk M, Strauss C (2023) Transfer learning approach for human activity recognition based on continuous wavelet transform. *Algorithms* 16(2):77. <https://doi.org/10.3390/a16020077>
46. Pavliuk O, Mishchuk M, Strauss C (2023) Transfer learning approach for human activity recognition based on continuous wavelet transform. *Algorithms* 16(2):77 <https://doi.org/10.3390/a16020077>

47. Permatasari J, Connie T, Ong TS (2020) Inertial sensor fusion for gait recognition with symmetric positive definite gaussian kernels analysis. *Multimedia Tools and Applications* 79(43):32665–32692. <https://doi.org/10.1007/s11042-020-09438-9>
48. Permatasari J, Connie T, Ong TS (2020) Inertial sensor fusion for gait recognition with symmetric positive definite gaussian kernels analysis. *Multimedia Tools and Applications*, 79(43):32665–32692 <https://doi.org/10.1007/s11042-020-09438-9>
49. Qiu S, Zhao H, Jiang N, Wang Z, Liu L, An Y, Zhao H, Miao X, Liu R, Fortino G (2022) Multi-sensor information fusion based on machine learning for real applications in human activity recognition: State-of-the-art and research challenges. *Information Fusion* 80:241–265. <https://doi.org/10.1016/j.inffus.2021.11.006>
50. Ramachandran K, Pang J (2020) Transfer learning technique for human activity recognition based on smartphone data. *International Journal of Civil Engineering Research* 11(1):1–17
51. Steven Eyobu O, Han DS (2018) Feature representation and data augmentation for human activity classification based on wearable imu sensor data using a deep lstm neural network. *Sensors* 18(9):2892. <https://doi.org/10.3390/s18092892>
52. Sun M, Song Z, Jiang X, Pan J, Pang Y (2017) Learning pooling for convolutional neural network. *Neurocomputing* 224:96–104. <https://doi.org/10.1016/j.neucom.2016.10.049>
53. Tufek N, Yalcin M, Altıntas M, Kalaoglu F, Li Y, Bahadir SK (2019) Human action recognition using deep learning methods on limited sensory data. *IEEE Sensors Journal* 20(6):3101–3112. <https://doi.org/10.1109/JSEN.2019.2956901>
54. Venkatachalam K, Yang Z, Trojovský P, Bacanin N, Deveci M, Ding W (2023) Bimodal har—an efficient approach to human activity analysis and recognition using bimodal hybrid classifiers. *Information Sciences* 628:542–557. <https://doi.org/10.1016/j.ins.2023.01.121>
55. Vrigkas M, Nikou C, Kakadiaris IA (2015) A review of human activity recognition methods. *Frontiers in Robotics and AI* 2:28. <https://doi.org/10.3389/frobt.2015.00028>
56. Walse RV, Kishor H, & Dharaskar, Thakare VM (2017) A study on the effect of adaptive boosting on performance of classifiers for human activity recognition. In: *Proceedings of the International Conference on Data Engineering and Communication Technology*, pp 419–429, Springer, Singapore https://doi.org/10.1007/978-981-10-1678-3_41
57. Wang Z, Wu D, Chen J, Ghoneim A, Hossain MA (2016) A triaxial accelerometer-based human activity recognition via emd-based features and game-theory-based feature selection. *IEEE Sensors Journal* 16(9):3198–3207. <https://doi.org/10.1109/JSEN.2016.2519679>
58. Wang G, Li Q, Wang L, Wang W, Wu M, Liu T (2018) Impact of sliding window length in indoor human motion modes and pose pattern recognition based on smartphone sensors. *Sensors* 18(6):1965. <https://doi.org/10.3390/s18061965>
59. Wang J, Chen Y, Hao S, Peng X, Hu L (2019) Deep learning for sensor-based activity recognition: A survey. *Pattern Recognition Letters* 119:3–11. <https://doi.org/10.1016/j.patrec.2018.02.010>
60. Wang G, Li Q, Wang L, Wang W, Wu M, Liu T (2018) Impact of sliding window length in indoor human motion modes and pose pattern recognition based on smartphone sensors. *Sensors* 18(6):1965 <https://doi.org/10.3390/s18061965>
61. Wu W, Zhang Y (2019) Activity recognition from mobile phone using deep cnn. In: *2019 Chinese Control Conference (CCC)*, pp 7786–7790 <https://doi.org/10.23919/ChiCC.2019.8865142>. IEEE
62. Xia K, Huang J, Wang H (2020) Lstm-cnn architecture for human activity recognition. *IEEE Access* 8:56855–56866. <https://doi.org/10.1109/ACCESS.2020.2982225>
63. Xiao F, Chen J, Xie X, Gui L, Sun L, Wang R (2018) Seare: A system for exercise activity recognition and quality evaluation based on green sensing. *IEEE Transactions on Emerging Topics in Computing* 8(3):752–761. <https://doi.org/10.1109/ETEC.2018.2790080>
64. Yildirim E, Kaya Y, Kılıç F (2021) A channel selection method for emotion recognition from eeg based on swarm-intelligence algorithms. *IEEE Access* 9:109889–109902. <https://doi.org/10.1109/ACCESS.2021.3100638>
65. Yildirim S, Kaya Y, Kılıç F (2021) A modified feature selection method based on metaheuristic algorithms for speech emotion recognition. *Applied Acoustics* 173:107721. <https://doi.org/10.1016/j.apacoust.2020.107721>
66. Zeng M, Nguyen LT, Yu B, Mengshoel OJ, Zhu J, Wu P, Zhang J (2014) Convolutional neural networks for human activity recognition using mobile sensors. In: *6th International Conference on Mobile Computing, Applications and Services*, pp 197–205 <https://doi.org/10.4108/icst.mobicase.2014.257786>. IEEE
67. Zhang L, Hua Y, Cotton SL, Yoo SK, Da Silva CR, Scanlon WG (2020) An rss-based classification of user equipment usage in indoor millimeter wave wireless networks using machine learning. *IEEE Access* 8:14928–14943. <https://doi.org/10.1109/ACCESS.2020.2966123>

68. Zhang Y, Yao X, Fei Q, Chen Z (2023) Smartphone sensors-based human activity recognition using feature selection and deep decision fusion. *Theory & Applications, IET Cyber-Physical Systems*. <https://doi.org/10.1049/cps2.12045>
69. Zhao Y, Yang R, Chevalier G, Xu X, Zhang Z (2018) Deep residual bidir-lstm for human activity recognition using wearable sensors. *Mathematical Problems in Engineering*. <https://doi.org/10.1155/2018/7316954>
70. Zhu J, Chen H, Ye W (2020) A hybrid cnn-lstm network for the classification of human activities based on micro-doppler radar. *IEEE Access* 8:24713–24720. <https://doi.org/10.1109/ACCESS.2020.2971064>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.