Int. j. inf. tecnol. (January 2025) 17(1):69–85
https://doi.org/10.1007/s41870-024-02305-y

ORIGINAL RESEARCH

Smartphone based human activity recognition irrespective of usage behavior using deep learning technique

Soumya Kundu[1] · Manjarini Mallik[1] · Jayita Saha[2] · Chandreyee Chowdhury[1]

Received: 14 June 2024 / Accepted: 7 November 2024 / Published online: 29 November 2024

Abstract Human activity recognition (HAR) from sensory data is a crucial task for a wide variety of applications. The in-built inertial sensor facilities of commercial smartphones have made the data collection process easier. However, different smartphone configurations exhibit variations in sensor readings for the same activities. Different smartphone holding positions, like in hand, shirt, or trouser pockets, also lead to variations in signal patterns for the same activity. Some recent works have shown that automated feature extraction using deep learning methods can significantly improve activity recognition, although there is a lack of experimentation considering device heterogeneity and different smartphone holding positions. The proposed work addresses this research gap with a two-fold contribution. First, a CNN-based HAR framework is proposed that forms 2-D frequency domain images to capture temporal patterns in the data along with inter-axis spatial features. Second, an ensemble of conditional classifiers has been designed based on CNN that exhibits generality in terms of device configurations and usage behavior. Real life data have been collected for different activities using different devices for experimentation. The proposed ensemble model is found to recognize activities with 94% accuracy even when the training and test devices are different for real datasets.

Keywords Human activity recognition · CNN · Deep learning · Smartphone · Activity image

✉ Jayita Saha
jayita.saha@manipal.edu

[1] Department of Computer Science and Engineering, Jadavpur University, Kolkata, India

[2] Department of Computer Science and Engineering, Manipal Institute of Technology Bengaluru, Manipal Academy of Higher Education, Manipal, India

1 Introduction

Due to expeditious technological growth and pervasiveness, wearable sensing have become an integral part of many research areas such as, human activity recognition (HAR) [1]. HAR is an essential component in various application domains such as, smart healthcare [2], surveillance [3], human–computer interaction [4], and many more. Initially, videos were used in identifying anomalies in human action, especially in the surveillance domain [5]. However, smartphone sensors provide a infrastructure-free and privacy preserving approach to HAR. The distinct static or moving posture can be uniquely identified using self-contained inertial sensors, namely, accelerometer or a combination of accelerometer, gyroscope, and/or magnetometer. These sensors are commercially available with smartphones, smart watches, and other wearable devices [6].

Since most citizens carry a smartphone nowadays, cost-effective ubiquitous systems could be designed for HAR based on smartphone sensing.

Individual activity generates unique time-series signal pattern, although the extent of distinguishing characteristics varies depending on the nature of activity [7]. Dynamic activities (walk, run, jump etc.) produce distinguishable patterns due to separate rhythm of acceleration. In case of different static activities (stand, sit, lie etc.), the difference in time series signal patterns is minor. Lack of sufficient movements causes the acceleration information along the time scale insufficient to identify the static activities. Better analysis of static activities is possible using two-dimensional data, that is inter axis patterns could be meaningful here.

Machine learning based HAR systems [8] require these features to be extracted that necessitate specific domain knowledge about the set of activities to be recognized and hence, makes such systems difficult to customize for various applications.

Deep learning techniques, such as Convolutional Neural Network (CNN) is being used widely in recent research works for its capability of automatic learning and feature extraction, particularly in the domain of computer vision. In the proposed work, we focus on the problem of designing CNN based HAR models for accelerometer signals. The problem of different sensor calibration of different smartphones and the effect of heterogeneous usage behavior are also taken into account.

Existing research works on HAR as in [9–11] are mainly focused on the recognition of a given bunch of activities, and feature extraction and fusion [12, 13] for different sensors. A few works could be found that study the challenging effect of different sensor calibration of smartphone configurations and the several usage behaviors as in [7, 14]. These works mostly apply an ensemble of supervised classifiers. Though a few recent works on smartphone-based HAR could be observed utilizing deep learning techniques, such as Long short-term memory (LSTM) as in [15, 16] we could not find any comprehensive deep learning framework for smartphone sensing that also handles the important challenge of different smartphone configurations and usage behavior. However, in real-life scenarios, sensors readings vary from one smartphone to another for the same activity and also it varies depending on how the device is carried by the subject (pant pocket or hand or shirt pocket). So, the classifier should retain sufficient generality to recognize activities from different test devices or smartphone usage patterns. Accordingly, the contributions of this paper are as follows.

- It provides a deep learning-based framework that first interprets accelerometer data as images so that CNN can effectively capture important features. Thus, inter-axes patterns and temporal patterns along the axes are analyzed for classification.
- It highlights the problem of different smartphone configurations and user behavior on the performance. Hence, an ensemble of conditional classifiers is designed based on 2D CNN in order to retain the generality of the system. The proposed model combines the concept of majority voting with average probability for the classification. Thus, even when the training and test data are collected from different devices, the system could still exhibit sustainable performance.
- It provides a detailed performance analysis of the proposed mechanisms subject to a real-life dataset collected from volunteers.

The rest of the paper is arranged as follows: Sect. 2 describes the related work; in Sect. 3, proposed work is explained; performance of the system is analyzed in Sect. 4, followed by conclusion in Sect. 5.

2 Related work

In this section, we have summarized some of the existing HAR works relevant for this paper based on machine learning or deep learning approaches, mostly using smart devices. Few works [17] could be found where activity continuously monitored with other factors like heart rate, temperature to identify several disease. The HAR works that focused on user, position, and/or device independence have also been discussed.

During the last decade, smartphones have become a part of our daily life, and, this fact was reflected in HAR research works too. An online activity recognition system for the Android platform was proposed in [18]. Data was collected with preferred window size and the training set for each of the activities was reduced into smaller subsets using clustered k-nearest neighbor (kNN). However, online activity recognition with kNN [19]is practically a time-consuming process as it requires high computational complexity for the lazy learning nature. Activities can be recognized in phases, as proposed in the two-layer approach [20]. In the first layer, similar activities are cl assified into separate groups like, static and dynamic. Then, different strategies and suitable classifiers were used according to the type of activities of each group. For dynamic activities, a position-assisted classifier was proposed as the position and orientation of smartphones highly affect the time-series signal pattern of dynamic activities. Static activities were identified with the help of transition recognition like sit-to-stand or vice-versa. Another similar approach was proposed in [21], the group-based context-aware HAR (GCHAR) approach, which outperforms single ML classifiers when evaluated on the UCI HAR dataset.

Recognizing transition and sequence of activities were the main objective in [22], using a smartphone-based Multi-Instance Multi-Label (MIML) ensemble model considering kNN distance metrics. Transition among two or three consecutive activities was successfully recognized in this work. Hand-gesture activities can also be recognized with higher accuracy, when smartphone is used along with wrist-worn sensors, as shown in [23]. The main challenge was to recognize less-repetitive activities like smoking, talking, eating, etc. with smaller window sizes. The system was evaluated with seven different window sizes considering thirteen activities to analyze this matter. Naive Bayes, decision tree, and kNN were used for classification.

Multiple features increase the time complexity of any HAR system. Identifying optimal feature set and combine with machine learning model is a crucial task for building any system. The author in [24] has selected a few meta-heuristic techniques for identifying relevant and optimal features from the overall feature space. The wolf search, elephant search and cuckoo search are combined with the correlation-based feature selection to perform as filter for identifying relevant feature set. The overall performance has increased with feature subsets compared to whole feature set. In [25],Fast Fourier Transform(FFT) and Discrete Cosine Transform (DCT) are used to calculate the frequency component of time domain signals. The Welch's power spectral density algorithm has applied to extract the detailed distribution of the power for different frequency related components of entire accelerometer signal. Breaking the chain of supervised learning, the unsupervised learning-based HAR method was proposed in [26]. The activities were recognized by applying the clustering method on smartphone data using the Jaccard distance measure. Applying C-index and FM-index before and after clustering respectively, researchers explained how this uncommon distance measurement can outperform popular Euclidean distance-based HAR approaches.

Like smartphones, smartwatches are also being popular day by day, making their usage obvious in HAR research works.

In [27], authors proposed a new HAR approach for detecting sitting positions. Office Workers Syndrome (OWS), that is having pain in the body due to sitting in a fixed position for a long time, is a common problem among numerous working people. The system identifies six different activities including sitting and computing approximate the time period of sitting using the accelerometer and gyroscope data of the smartwatch using an ensemble learning-based technique. Support Vector Machine(SVM), Decision Tree(DT) and Random Forest (RF) are used to identify abnormal activities in [28]. The RF algorithm, is more efficient for identifying the activity like climbing up and down as there are more spikes and changes of the features. The author in [29], explore the XGBoost algorithm for HAR. Examine the impact of the gyroscope on HAR results and compared results of existing models Decision Tree, SVM, Multilayer Perceptron, Naive Bayes, KNN, Random Forest. More training data improves the accuracy of each activity In [30], audio sensor of the Samsung Gear S3 smartwatch was included, along with an accelerometer and gyroscope for recognizing three basketball activities: handling, passing, and dribbling. Approximately 20% improvement in performance was observed after involving the audio sensor.

Up to a certain period, sensor-based activity recognition was dependent on the hand-crafted feature extraction process. With recent advancement in deep learning, automatic feature extraction have enhanced the field of HAR research to the next level. The advantage of self-contained sensors of smart devices and deep learning techniques both are justified in various recent HAR works.

Another smartwatch-based HAR system was proposed in [32] using a combination of CNN and LSTM, eliminating the necessity of manual feature extraction. The 2-layer hybrid system was evaluated using accelerometer and gyroscope data available in the WISDM dataset for 18 physical activities. A Deep Neural Network (DNN) based HAR framework was proposed in [36] for continuous monitoring of Parkinson's disease patients. The time and frequency of walking and transition of activities (sit-to-stand and vice versa) are different for such patients and healthy people. The gait and mobility of these patients were studied with smartphones and people were classified based on that information. In [34],'Long-term Recurrent Convolutional' or LRCN, or simply CNN-LSTM, where the CNN serves as the frontend for the LSTM. Results are compared with traditional models and other deep learning models. Machine learning model SVM and deep models CNN,BLSTM, MLP are applied in [33], with proper hyper parameter optimization technique. In [37], Deep Belief Network (DBN) was used to train the system for activity recognition using sensor data of smartphones. The network is initially pre-trained using Restricted Boltzmann Machine (RBM) [38], then weights are adjusted by a fine-tuning algorithm. This DBN-based recognition outperformed the Support Vector Machine (SVM) based and ANN-based recognition processes. In the deep learning model proposed in [39], feature extraction was done by using a temporal convolution on the spectrogram domain (representing with frequency and time) of the gyroscope and accelerometer data, considering fixed window size. A linear classifier and neural network performed the classification, with the best 23 features. The hybrid model combination of DS-CNN and LSTM achieves considerable performance in [35].

Researchers have followed semi-supervised learning methods to solve labeling issues of sensor data. In [16], authors tried to overcome the problem of weakly labeled sensor data with an intuitive auto-labeling scheme, based on Deep Q-Network (DQN) with a distance measurement rule. LSTM was used to identify fine-grained patterns from sequential motion data. Deep LSTM was used in [15], where the objective was to utilize both of the labeled and unlabeled data for activity recognition. A temporal ensemble method was used to compare the output of the neural network with predicted ensemble output while using unlabeled data.

Stepping out from the boundary of commonly recognized activities, the authors in [40, 41] considered some daily activities that are comparatively less considered HAR research works. In [40], activities like nordic walk, open and close door, etc along with common activities like walking, standing, etc. were recognized by the combination of Inception Neural Network and Recurrent Neural Network, by extracting multidimensional features. In [41], Simple Recurrent Units (SRUs) were combined with the Gated Recurrent Units (GRUs) for activity recognition, from multi-modal input data sequence using their internal memory states.

CNN has been used widely in recent deep learning-based HAR works, for learning the characteristics of time series signals [32, 33, 42]. CNN has also been used to convert tri-axial inertial data into images with tri-color channels to presume correlations between sensor signal values [43]. HAR performance has been improved with CNN due to fast computing and predicting final class labels faster compared to baseline models such as SVM or Multilayer perceptron networks [44]. In [45], the authors used deep Convolutional Neural Network (DCNN) to inherit temporal local dependency of time-series 1D signals and find out translation invariance and hierarchical characteristics of activities. DCNN also helps to extract low and mid-level features that in turn helps to identify detailed activities [46].

Although the availability of smartphones has spirited the HAR research works, different hardware configurations and holding positions resists the way of generalization. Researchers tried to overcome these issues in various ways.

In, [47] the authors proposed a framework to identify individual activities considering different device configurations and usage behaviors. Multiple sensors were used to remove the gravitational effect on the accelerometer values. To achieve complete orientation independence, accelerometer readings from the body coordinate system were converted to an earth coordinate system.

The authors in [48] proposed the PACP method i.e. Parameters Adjustment Corresponding to smartphone Position. After feature extraction from the raw inertial data, the position of the the smartphone was identified. Then, by adjusting the accelerometer data corresponding to the position, integrity was maintained before classifying activities with SVM. Extracting various features is a crucial part of recognizing robust activities [31]. An angle feature extraction model was designed in [49], to recognize activities irrespective of phone locations. Since the user movement direction is independent of changing phone locations, the addition of angle feature to other features helps to recognize activities more accurately. Fusion of sensors was used in [50], to build a user-independent HAR model. A large number of features including time domain and frequency domain were considered from multiple smartphone sensors. A Wi-Fi-based location-independent HAR system was proposed in [51], using CNN network architecture for feature extraction. Inspired from few-shot learning, the system was designed to learn the feature representation and the metric relation of the input activity samples, and location-independent classification was done based on this learning. Transfer learning was also used along with CNN to recognize various activities such as distinct hand gestures [52], or identifying activities irrespective of positions with very limited or null training samples [53]. The system achieved its goal by transferring features among different locations.

In [54], authors proposed a DCNN based model to make their HAR system position-independent. The idea was, all neurons of a single feature map will share the same weight, which acts as a parameter. This provides the ability to detect a specific pattern, irrespective of the position in the input data sample. The overall important key findings of literature survey are discussed in Table 1.

We can observe that the HAR research works were initially focused on analyzing the time-series signal pattern, using some supervised learning method. Recent works are trying to achieve the benefit of automatic feature extraction and overcome smart device-related challenges. Moreover, most of the works recognize individual activity from a bunch of different types of activities, without trying to figure out the best suitable features for different kinds of activities. In this paper, we have proposed a mechanism to apply CNN in order to extract Spatio-temporal pattern from signal images and generalize it by considering user and device independence.

The ensemble models are robust as it helps to reduce the overall variance components of prediction errors. The proposed HAR framework is explained in the next section.

3 Design of the proposed activity recognition framework

The proposed work aims to identify four human activities (sitting, standing, walking, jogging) based on inertial sensor reading of smartphones carried by users at Shirt Pocket (SPT), the right front pocket of the pant(RPT), or in the right hand (HAND). In this section, we present an overview of the proposed human activity recognition framework.

3.1 Data collection

The data are collected using the accelerometer of the smartphones. During dynamic activities (walking, jogging), users held smartphones according to their preference to avoid replication of smartphone positions. Each activity is defined by tri-axial accelerometer readings along A_x, A_y, and A_z axes of the certain device as shown in Fig. 1. Static and dynamic activity patterns are easily distinguishable, but it is difficult to identify two static activities from their patterns, using the same or different devices as shown in Fig. 2.

Even from the same device, different patterns are observed for keeping it in the same/different positions as shown in Fig. 3. In this work, we have used four devices and collected data for three positions SPT, RPT, HAND (stored in three different datasets) using each device individually.

Table 1 Summary of a few representative works on sensor-based HAR

Ref, Year	Dataset and sensor used	Objective	Data preparation	Learning technique	Remarks
2017, [31]	REALDISP, OPPORTUNITY. Wearable body sensor. Accelerometer, gyroscope, magnetometer and quaternion	The influence and importance of different types of sensors. Impact of normalization in data preprocessing based on different user variation	Time- domain signal features like mean, signal magnitude area, energy measures and weighted average, skewness and kurtosis are applied on frequency domain signal	Decision Tree and Random Forest algorithm	Vector normalization, mean removal, histogram equalization, zscore etc. are applied to reduce the degradation of performance when training data and test data from different users. The combination of all sensors provides better result other than individual sensor. Random Forest achieves better accuracy
2018, [27]	Real time data collected. Smartwatch Accelerometer and gyroscope	The office worker syndrome issues are considered specially for identifying how long user are sitting in one place	Mean, standard deviation, signal magnitude,energy, Interquartile range statistical features	Ensemble learning approach like stacking and majority voting	The combination of accelerometer and gyroscope performs well
2019, [25]	UCI-HAR Accelerometer	Mitigate the challenge of cyclo stationary nature of activity signal	FFT and DCT are used to calculate the frequency component of time domain signals.The Welch's power spectral density extracts the detailed distribution of the power for different frequency related components	Two channel-based CNN has used here	The power based extracted features and frequency based features are individually applied in two different channel of CNN. Lastly it has concatenated to identify activity
2020, [32]	WISDM. Smartwatch Accelerometer and gyroscope	Importance of Deep learning models for identifying smart watch based activity	Automated Spatial features by CNN and Temporal features by LSTM	Basic CNN and LSTM,Hybrid Deep Learning (CNN_LSTM)	Bayesian Optimization is used for parameter optimization
2020, [33]	UCI-HAR and Pamap2 Smartphone based Accelerometer and gyroscope	Comparison of machine learning and deep learning models	Denoising, normalization and segmentation are used for data preprocessing. The window size 25 with 50% overlap	SVM and CNN, BLSTM, MLP are applied with proper hyper parameter optimization technique	Extracting high efficiency features from original data with cross domain knowledge to segregate the differences from same categorical activity
2021, [28]	Real time data collected. Smartphone accelerometer and gyroscope	Detecting abnormal activity specially in application of healthcare	Raw dataset is used for further experiment	SVM, DT and RF are used to identify abnormal activities	The RF algorithm, is more efficient for identifying the activity like climbing up and down as there are more spikes and changes of the features. More training data improves the accuracy of each activity
2021, [34]	UCI-HAR. Accelerometer	Addressing the limitations of traditional machine learning techniques which cannot capture temporal correlations effectively. Overcoming the computational power issues of previous models that increased with more data	Data transformation based on Gaussian standardization	Long-term Recurrent Convolutional' or LRCN, or simply CNN-LSTM, where the CNN serves as the frontend for the LSTM	The combination of lightweight CNN-LSTM model is more robust and capable for identifying activity efficiently

Table 1 (continued)

Ref, Year	Dataset and sensor used	Objective	Data preparation	Learning technique	Remarks
2023, [29]	Real time data collected. Accelerometer, gyroscope, magnetometer, GPS	Understand the impact of different sensors, particularly the gyroscope, on HAR performance. Improve the accuracy and applicability of HAR models by using realworld data	Data cleaning (removal of outliers, exclusion of first and last five seconds of each session, handling of gaps in data). Mean, standard deviation, minimum,maximum, interquartile range are applied except GPS. Total distance traveled computed for GPS	Explore the XGBoost algorithm. Compared results of existing models Decision Tree, SVM, Multilayer Perceptron, Naive Bayes, KNN, Random Forest	Continous grid search helps to identify set of hyper parameter. Find the optimal model within reasonable time for selected hyper parameter set
2023, [35]	Real time data collected. Accelerometer, gyroscope, magnetometer, GPS	Improve accuracy by considering real-life data variations and temporal aspects Data collected from smartphones, with free positioning to simulate real-life conditions	Outliers are removed. Data transformation using Linear interploation	NN, LSTM, RNN; Hybrid Models: (DS-CNN)- LSTM, (DSCNN) -(Bi-LSTM)	Combination of DS-CNN and LSTM achieves considerable performance. The intrinsic nature of LSTM capable to store past information of activity pattern
2024, [24]	Real time data collected. Accelerometer, gyroscope, magnetometer, Smartphone sensor	Identify aberrant and non-aberrant activity from smartphone based Multiple sensor data	Features like mean, interquartile range, difference are applied to all type of sensor data and around 96 statistical features are computed	Meta-heuristic techniques like wolf search, elephant search and cuckoo search are combined with the correlation-based feature selection for filtering relevant features. Various learning techniques SVM, Decision Tree, Random Forest are applied to identify individual activities from subset features	Random Forest algorithm achieves considerable accurate performance. The overall performance has increased with optimized feature subset

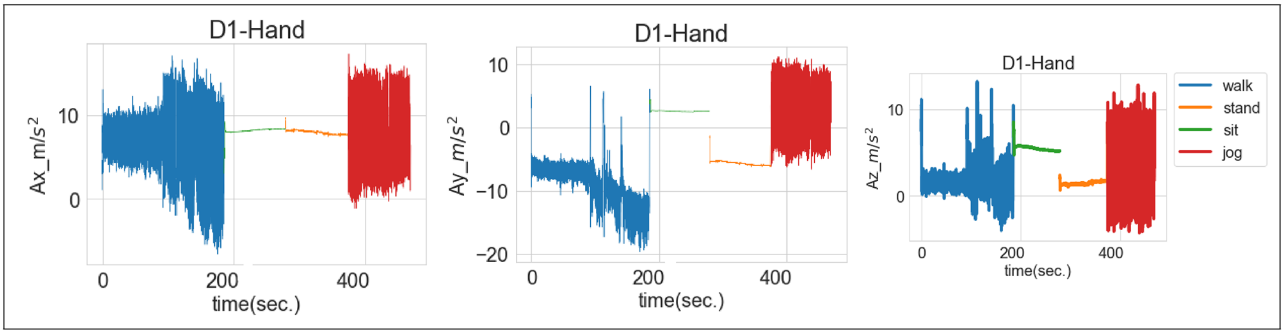

Fig. 1 Raw accelerometer readings along A_x, A_y and A_z axes of Device 1 for different activities performed for 400 s keeping the device at Hand

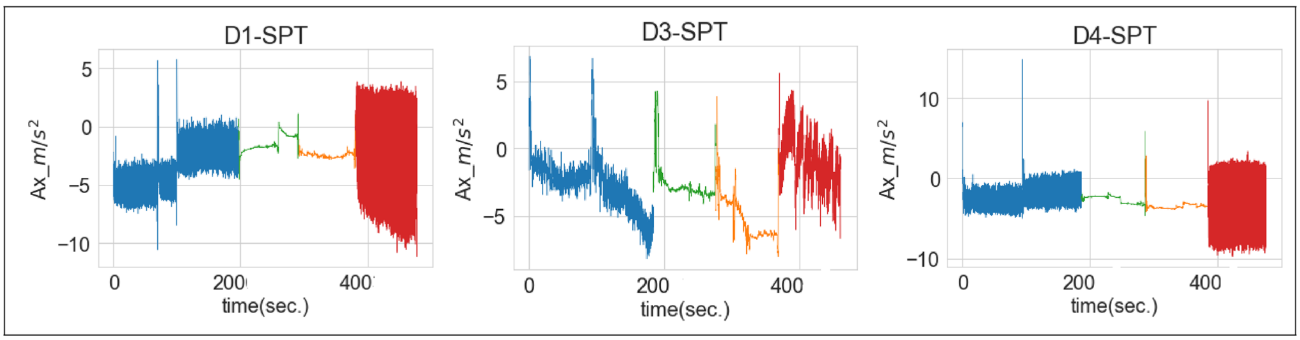

Fig. 2 A_x traces of Device 1, Device 3, and Device 4 for different activities performed for 400 s keeping the devices in Shirt Pocket(SPT)

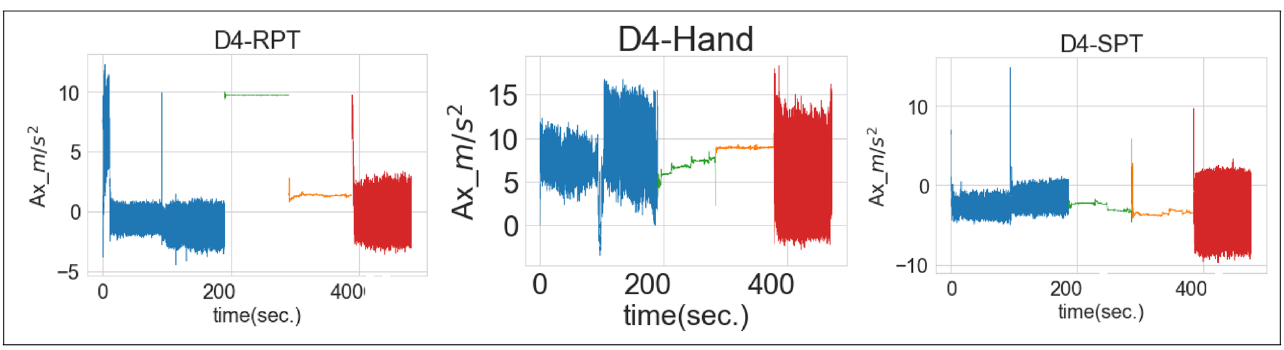

Fig. 3 A_x traces of Device 4 for different activities performed for 400 s keeping the device in Right Pant Pocket(RPT), Hand, and Shirt Pocket(SPT) respectively

3.2 Data preprocessing

The raw dataset contains several noises, it cannot be used directly in the learning model. Different preprocessing techniques are applied to the dataset to improve the overall performance that is detailed as follows.

Filtering The raw dataset may contain noise or abnormal spikes, due to various reasons, such as a positional change or unintentional change of sensor orientation. Filters are applied to remove accelerometer signal noise and outliers like low-frequency acceleration. We have used three common noise removal filter methods that detect and remove the noise generated, namely, butterworth filter, zero-phase filter and median filter. The filters remove the noise due to the dynamic motion of subjects considering the orientation of the sensors w.r.t. ground-level data and preserve the medium frequency signal components. Butterworth filter is used at first that passes the signal having a lower frequency than the pre-determined cut-off frequency. Then, a Zero-phase filter is applied to make the signal's phase slope zero. Finally, a Median filter is used to consider the median value of the signal within a certain time range.

Windowing The dataset is sorted according to timestamp values. For each activity, we split the dataset based on time. We select approximately first 80% data for training purpose and last 20% data for validation purpose. In order to extract temporal patterns from the data along the data dimensions, the data is split into small overlapping segments, called windows. To reduce the data loss at the window edge and capture sequence of activities, the overlapping is considered here. In general, a human being requires 2.56 s on average to complete a walk cycle. Data are collected as 50 samples (data instances) for every unit second. Thus, for 2.56 s, we have total (2.56×50)=128 data instances. For this reason, window sizes are considered to be factors or multiples of 128. Using individual devices, holding them in specified positions, we collect data for each of the four activities. The total number of instances for each activity should be multiple of 128, rest are discarded. For example, we have taken (128×126)=16128 instances for the walking activity collected using the Google Pixel device holding it in the HAND position for a user. As commonly used in accelerometer data [55], 50% overlapping windows are considered. The structure is represented in Fig. 4. In the figure, it can be observed that for 256 data instances, $\frac{256 \times 2}{128} - 1$=3 windows are formed. The general equation is as follows.

$$w = \frac{2n}{s} - 1 \tag{1}$$

In Eq. (1), w represents a number of windows, n represents a total number of data instances and s represents window size.

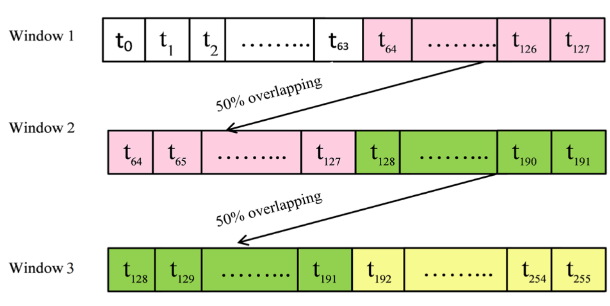

Fig. 4 Overlapping of data in each activity dataset

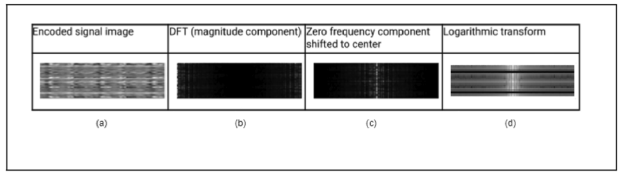

Fig. 5 All are walk activity data collected from a Google Pixel smartphone held at hand after applying the encoding technique; **a** encoded signal image resulted, **b** Fourier transform' on magnitude component encoded image, **c** frequency component shifted at center in magnitude component encoded image, **d** logarithmic transformation is applied on the image shown in **c**

The overall step-by-step procedure for data preprocessing is summarized in Algorithm 1.

Algorithm 1 Create Feature

Input: Activity Dataset D
Output: $\{F\}$ ▷ Activity Feature Set
procedure CREATEFEATURE(D)
$D = Ac_x, Ac_y, Ac_z : T_c$ ▷ Extract accelerometer data and time stamp
$D' = filter\,(Ac_x, Ac_y, Ac_z)$
$Group\{f_j : t_c\} \in F$ into F_i : $f_j = (ac_x, ac_y, ac_z)_j$' s are m consecutive samples/sec and $(F_i \cap F_k) \neq \phi$
Data converted in 2D format
$F_i \rightarrow A_M[i][j]$
▷ i total number of instance and j is eqivalent to accelerometer axis
$A_M[i][j] \xrightarrow[Conversion]{2D-3D} A'_M[k][m][j]$
▷ k is number of rows m is window size and j is number of axis present in acclerometer
for i in k:
$A'_M[m][j] \xrightarrow[Transpose]{} A^T_M[j][m]$
Converting into image
a. $\binom{j}{r} = \frac{j!}{(j-r)!}$ stacked the [m] one after another all possible way new matrix is[r][m]
▷ r is number of permutation
b. $F_i = DFT([m][r])$
c.$F_i = FFTSHIFT(F_i)$
d.$F_i = taketherealpart(F_i)$
e.$F_i = log(1 + F_i)$
f. Normalize value
$F_i = [\frac{F_i - min(F_i)}{max(F_i) - min(F_i)}] * 255$
▷ end of for loop
return $\{F\}$
end procedure

3.3 Formation of image by signal encoding

Each window of the accelerometer instances is converted to a 2D image to be fed to a 2D CNN for classification. Each pixel value of an image must be computed from the collected sensor data belonging to that window. A *signal image* is generated from a permutation algorithm for stacking of the signals [10]. The process is done in a way to achieve correlation between the pairs of different signal dimensions. For example, from a sequence of 3 given signals, there are ${}^2P_3 = 6$ ways to choose 2 signals.

Hence, for every segment of length p, an image of $6 \times p$ is generated. In the *signal image*, every signal sequence has the chance to be adjacent to every other sequence, which enables CNN to extract hidden correlations among the neighboring signal dimensions. The value of p depends on the sampling rate of the sensor readings and the walking cycle. Figure 5 represents the sample 2D images obtained for walk data collected from a Google Pixel smartphone held in hand. It is evident from the figure that the image captures the different signal patterns of the accelerometer signals for the activity. We explored frequency domain representation in this work to better capture the signal patterns.

In the frequency domain, signals are decomposed into a combination of waves of fixed frequencies. Each point of the Discrete Fourier Transform (DFT) [56] is a complex number representing the magnitude and phase of a set of particular "horizontal frequency" and "vertical frequency". The number of frequencies is same as the number of pixels in the spatial domain image, and it is sufficient to fully describe the image. 2D Fast Fourier Transform (FFT) [57] algorithm is used to compute the DFT.

2D DFT is applied to the signal image. From the complex result at each point, the magnitude is taken. The phase represents information about the onset of each frequency and it is ignored in this case. The zero-frequency component is shifted to the center of the image by swapping the first quadrant of the image with the third, and the second quadrant with the fourth. Finally, a logarithmic transformation is applied to the image, to compress the large dynamic range obtained after applying DFT. Thus, a signal image is generated from a set of permutations using the axes of the raw signal. Figure 6 represents sample images

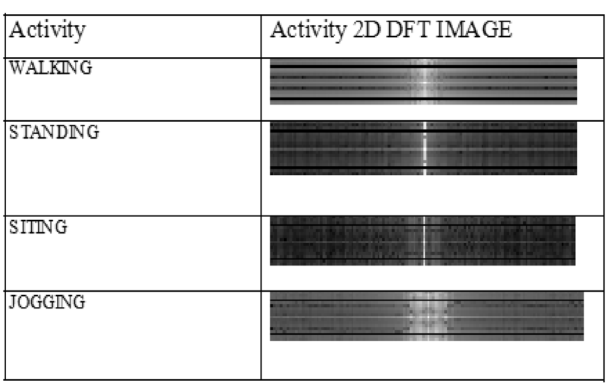

Fig. 6 Sample activity images generated for the 4 basic activity types considered in the work

of different activities. The images clearly show distinguishing patterns for the different activities that would be captured by CNN [58].

3.4 Design of the proposed ensemble classification model

After signal preprocessing, the entire dataset has been converted to signal images for classification. However, in order to address the issue of differing signal values due to different smartphone configurations and usage behavior, it is important to retain the generality of the classification model. Thus, an ensemble of condition-based classifiers has been designed here. It combines majority voting with average probability mechanism as summarized in Algorithm 2.

Algorithm 2 Identify Human Activity

Input:Training Dataset D_{i_t}, L_t ▷ L_t Training dataset label
Output:DAc_i ▷ Identify activity device and position independent manner
F_t=CreateFeature(D_{i_t})
L_t = LabelCreate(F_t)
if user is in experimental region **then**
  collect D_{i_s} ▷ D_{i_s} collected from other device
  $\{F_s\}$=CreateFeature(D_{i_s})
  L = LabelCreate(F_s)
end if
for all $f_j \in F_t$ **do**
  for $q \leftarrow 1$ to n **do**
    PL_q=Classify($F_t : l$) ▷ Based on different parameter tuning and condition
  end for
end for
$< PL_1, PL_2, \cdots PL_n >$, Set of predicted labels are generated $PL_i \subseteq L$
Calculate average probability $< Ap_1, Ap_2, \ldots Ap_n >$ ▷ Refer equation no 2;
for all $sf_j \in F_s$ **do**
  for all $q \leftarrow 1 to n$ **do**
    $TPL_q = Classify(F_t, F_s)$
  end for
end for
$< TPL_1, TPL_2, \cdots TPL_n >$ set of test labels are generated from base classifier
Feed maximum of average probability
New Ensemble labels are generated $< E_1, E_2, \cdots E_n >$
$E_i \subseteq L$ is a set of classified labels
$DAc_i = \{\}$
for e **do**achcorrectly predicted label
  if $TPL_k is '1'$ **then**
    $DAc_i = DAc_i \cup TPL_k$
  end if
  $Ic++$
end for
$Acr = \frac{I_c}{I_c^{total}} * 100\%$ ▷ Calculate Accuracy **return** (Acr, DAc_i)

2D CNN is used here as a base classifier for automated feature extraction from the signal images. The traditional neural network extracts feature with a high dimension which makes the system computationally expensive. The special features of CNN reduce these expenses and make system more efficient. The property of the convolution layer helps to capture the translational invariance from the data and it is also very helpful to maintain the small size of the neural network with small number of inputs without affecting the overall accuracy. The padding is helpful to increase the moving space of the kernel filter and provide a more accurate analysis. CNN extracts the features in low dimensional space. Several parameters of CNN are tuned and considered to build the individual base classifiers for the different conditional data sets. The ensemble model is considered as the individual classifiers fail in this work to address the heterogeneity of devices and usage behavior. The k number of base classifiers $(Cl_1, Cl_2, \dots Cl_k)$ are developed based on k number of conditions. Each condition indicates a (device, usage behavior) combination. 2D CNN is applied to each of the conditional training data sets to build respective classification models, Cl_i. The parameters are tuned for each of the training conditions based on the respective validation datasets. When an unknown test set is fed to the ensemble model, each of the base classification models is applied to get the prediction outcomes. Based on the majority voting of the prediction outcomes, the ensemble decides the final predicted activity class for each test instance. However, simple majority voting may not always resolve the issue. There may be a tie. So, the softmax layer at the last layer of CNN is utilized that calculates the relative probability value of each of the predicted classes for every instance of test data based on Eq. (2).

$$Pr(y_i) = \frac{exp^{y_i}}{\sum_{j=1}^{n} exp^{y_j}} \tag{2}$$

Now, if in the case of a tie, half the number of classifiers predict one activity and the other half predicts another activity then, we consider the activity which has the highest average probability. There can be another situation where the predicted outcomes of all the base classifiers are different. In this case, the prediction outcome with the highest probability value will be chosen to be the predicted class of the ensemble. Thus, the proposed ensemble utilizes a combination of majority voting and the notion of average probability to handle data pertaining to different device configurations and usage behavior. So, the ensemble is not biased to a specific (device, usage behavior) combination, but appropriately prioritizes some base classifiers if majority voting does not resolve the issue. The overall phases from data preprocessing to the ensemble model have been summarized in Fig. 7. The experimentation on the collected dataset and the performance results of the implementation of the proposed framework are detailed in the next section.

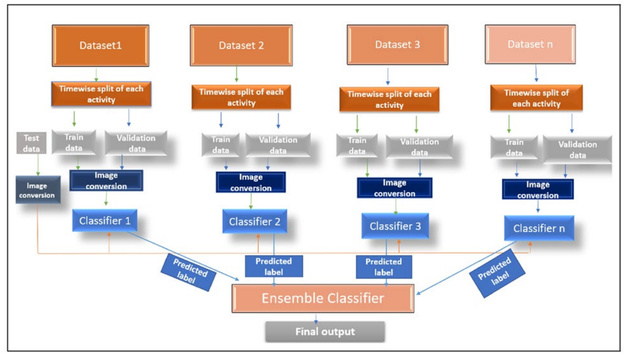

Fig. 7 Block diagram of the proposed CNN based HAR framework

Table 2 Experimental setup of the proposed work

Devices	Pixel device1 (D1)
Pixel device 2 (D2)	Samsung note3 (D3), RedmiY1 (D4)
Position	Right Pant Pocket (RPT)
	Hand, Shirt Pocket (SPT)
Sampling rate	50 samples/s
Filtering method	Butterworth and median filter [59]
Dataset size	540,00 samples approximately
Number of users	8

4 Performance analysis

The experimental setup is discussed first followed by the set of experiments carried out to explore the performance of both CNN and the ensemble model subject to device independence.

4.1 Experimental setup

Real time data are collected from four smart handheld devices ($D1 - D4$) for 8 users. The devices are carried in three positions on/around the body, namely, SPT, RPT and Hand. An android application G-Sensor Logger collects the embedded tri-axial accelerometer sensor data for four daily activities. The phone is held upright in each of these three positions while collecting the dataset. Each dataset contains around 54,000 accelerometer records. $D1 - hand$, $D1 - SPT$, $D2 - SPT$, and $D3 - RPT$ are considered as training dataset and others as test data like $D1 - RPT$, $D4 - SPT$, $D4 - HAND$, $D4 - RPT$, and $D3 - SPT$. The experimental setup is detailed in Table 2.

The butterworth filter and median filter are used to remove the unwanted noises. After removing the low-frequency acceleration (gravity) and noise, the preprocessed data are generated.

The conditional datasets are divided into two partitions (temporal split) first 80% is considered as set of training data and remaining 20% is validation set. The data are segmented and each overlapping segment contains 128 instances by default. The segments are converted to signal images. The train test split method is used to train the datasets and parameters are tuned for each of the conditional training sets. These tuned parameter values are saved as part of the base classification models of the proposed ensemble. The experimental results are detailed in the following subsections.

A series of experiments are conducted to validate and explore the working of the proposed deep learning-based HAR framework. The experiments conducted have been grouped into three categories—(1) the motivation for 2D CNN has been investigated; (2) the effect of data segmentation has been explored on the overall performance; (3) finally, the effect of heterogeneous devices and usage behaviors on the performance are explored. These are detailed in the respective subsections.

4.2 Performance of 2D CNN on HAR data

Existing machine learning and deep learning classifiers mostly perform well when the training and test conditions are the same. An experiment is conducted to investigate the effect of using different training and test device when both the devices were held in hand. The results are shown in Fig. 8. A lazy learning classifier (kNN), a regression-based classifier (Logistic Regression), a tree-based classifier (Decision Tree), and a strong classifier (SVM) have been applied for classification. Two deep learning models, LSTM and CNN are also applied to the data. Both validation accuracy and test accuracy are reported. The validation accuracy is more as the parameters are tuned according to the validation dataset. Then, the test data is classified based on the saved tuned classification models. It has been found that due to automated feature extraction, CNN is found to perform really well for the accelerometer dataset. This validates the usage of CNN for this work.

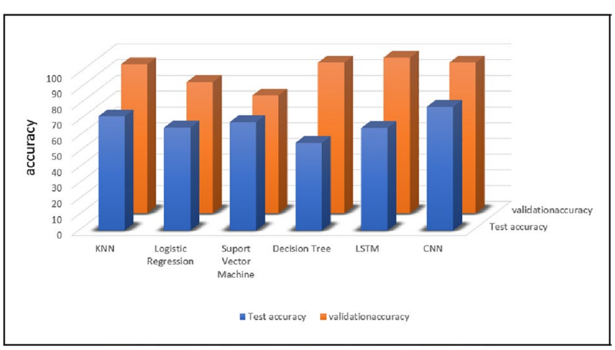

Fig. 8 Performance of ML and DL algorithms when train device is Google Pixel held at hand and the test data is collected from Redmi phone held at hand

One cycle of training the CNN with all training data is known as an epoch. The accuracy and the loss value are generated in each epoch. The comparison of validation loss and training loss of different classifiers for increasing no of epochs are summarized in Fig. 9. It is observed that the loss value is decreased with the increasing value of the epoch and is getting somewhat stabilized. Here, the validation loss and training loss of the four base classifiers are shown. It is found that the difference between two-loss curves is minimal indicating that the model is not overfitting. The internal architecture of the four conditional base classifiers is shown in Fig. 10. The parameters are in correspondence to the loss curve shown in Fig. 9.

4.3 Studying the effect of data segmentation

This subsection studies the effect of segmentation on the performance of the ensemble model. The entire dataset is split into overlapping windows in order to extract temporal patterns from the data. An experiment is conducted to find the suitable window size for effectively extracting patterns of the accelerometer data for daily activities. The window sizes are taken to be either a

factor or a multiple of the default window size 128 that corresponds to the average walking cycle of 2.56 s. Thus, four window sizes are considered in this work—32, 64, 128, and 256. The results are shown in Fig. 11.

D1-RPT was taken to be the test device for this experiment. The results indicate that window size 64 is found to be the best window size for all the devices. The default window size of 128 also gives good results. However, a window size of 32 is found to be too small to capture any pattern. On the contrary, a window size of 256 is found to be too large as minute temporal patterns would be interleaved in each window that is hard to generalize into an activity-specific signal pattern.

The next experiment is conducted to detail the effect of segmentation on static and dynamic activities subject to different training and test conditions. The results are reported in Fig. 12. It can be observed that both static and dynamic activities exhibit similar effects of windowing. For both cases, window sizes of 128 and 64 seem to work better than the others reinforcing the claim of the previous experiment. Interestingly, for dynamic activities, the system performs better as there is a temporal rhythm for the dynamic activities that are effectively captured by CNN.

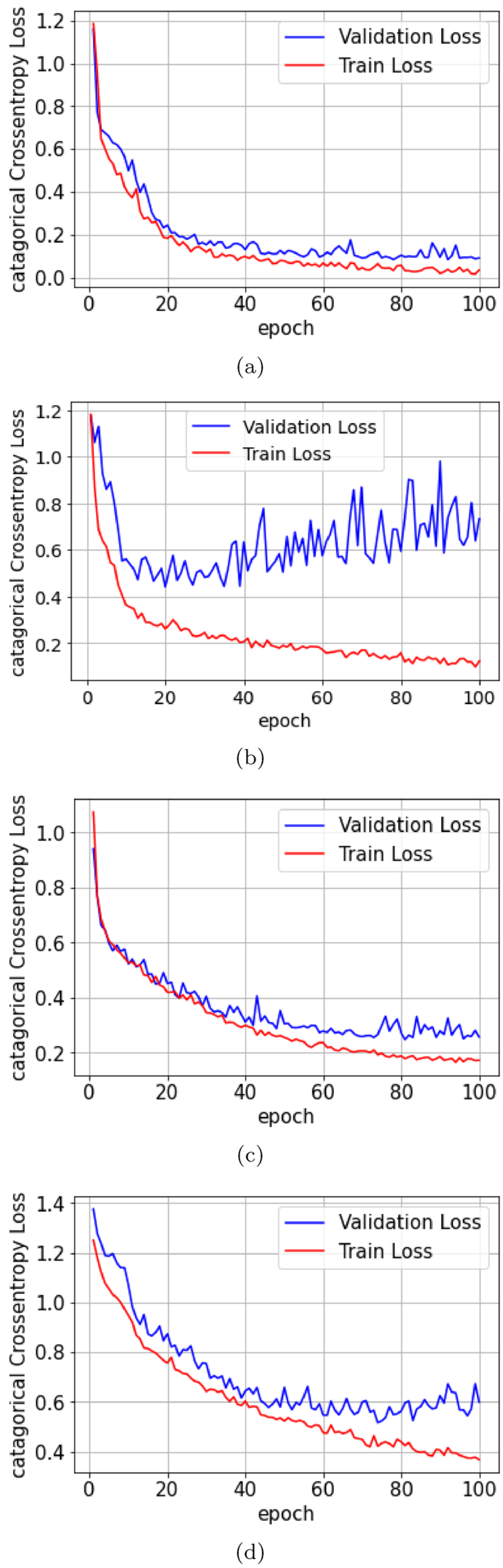

Fig. 9 Epoch-wise validation and training loss comparison for **a** Classifier 1 (Training data D1-hand), **b** Classifier 2 (Training data D1-SPT), **c** Classifier 3 (Training data D2-SPT), **d** Classifier 4 (Training data D3-RPT)

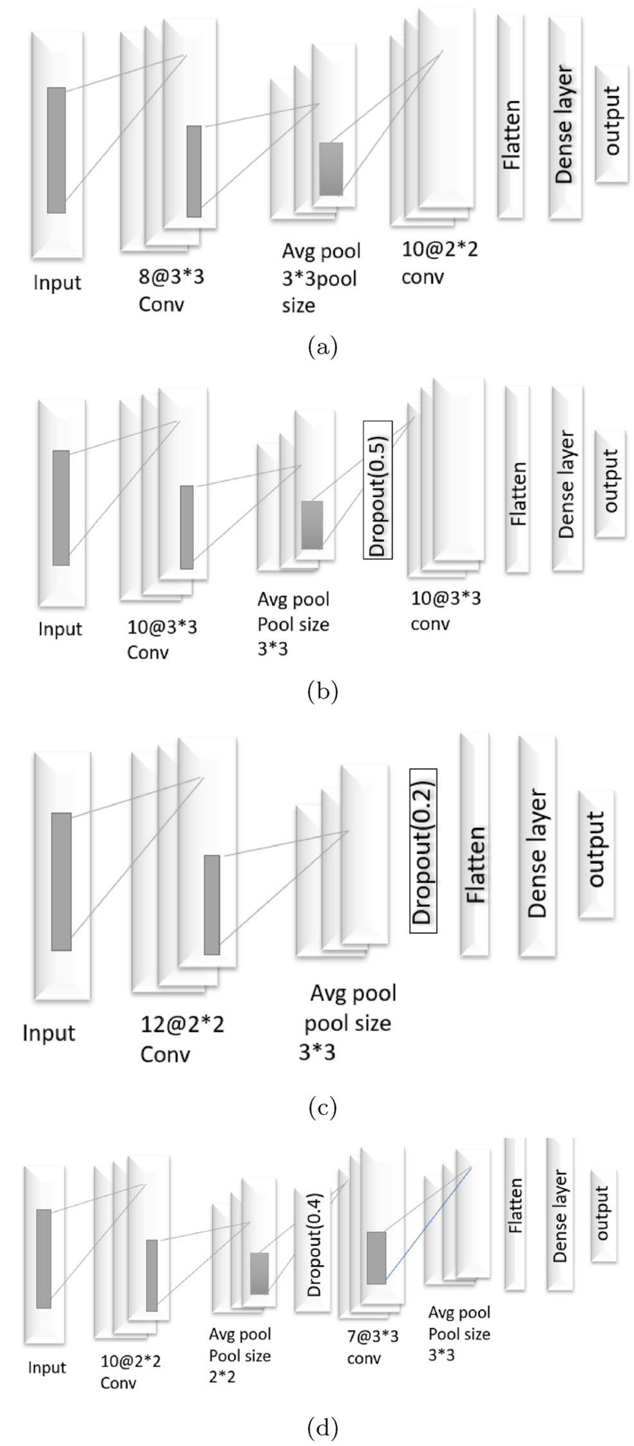

Fig. 10 Structure of the four base classifier CNN corresponding to the training datasets D1-HAND, D1-SPT, D2-SPT, and D3-RPT where all the parameters are tuned for window size 64

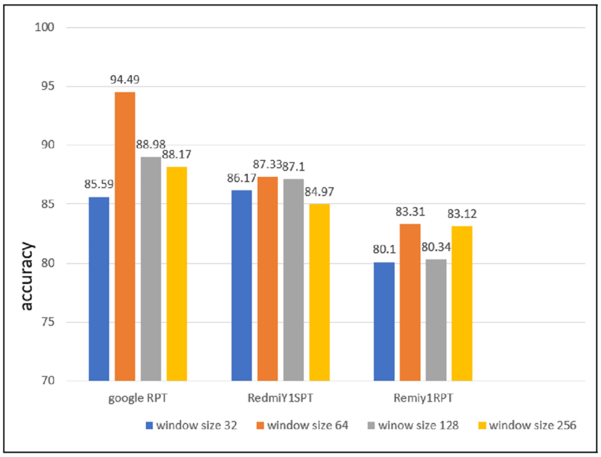

Fig. 11 Ensemble accuracy for different window sizes for different devices

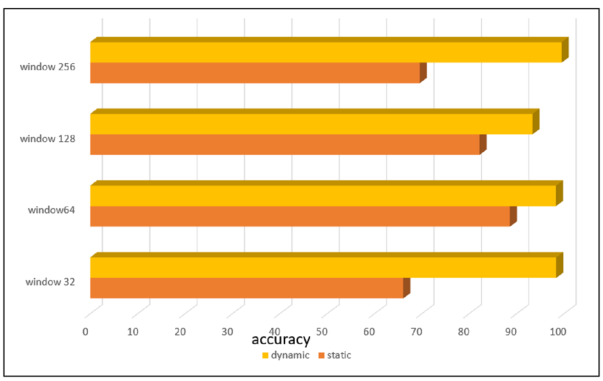

Fig. 12 Ensemble accuracy in static and dynmaic activity on different window sizes on googleRPT as test device

4.4 System performance subject to device independence

This subsection explores the effectiveness of the ensemble model. Once the parameters are tuned and saved for the base classifiers, the test datasets are fed to the system for classification. Accordingly, the next experiment is conducted to show the performance subject to different test datasets. For CNN, in each of the base classifiers, the batch size is 10, number of epoch is 100, loss function considered is "categorical crossentropy" and optimizer is Adam. Performance of the proposed ensemble model is shown for both window size of 64 in Table 3 and window size 128 in Table 4.

The condition-based deep ensemble model is found to perform better than the individual base classifiers. When the test set is $D1-RPT$, considerable accuracy is achieved by the ensemble model. In the training set, $D1-SPT$ dataset is present, hence, the device of training and test is the same but the position is different. When $D4-SPT$ is considered as the test set, the overall accuracy of the proposed ensemble model is increased by 2–9% from the individual base classifiers for window size 64. This use case indicates that the system can recognize human activities when test and training devices are different. Thus, for both window sizes, activities are found to be recognized with appreciable accuracy even when the training and test conditions in terms of device configuration and usage behavior differ.

Activity wise classification accuracy values are detailed in Fig. 13 when the test set is $D1-RPT$ for window size 64. It can be seen that static, as well as dynamic activities, are better recognized using the proposed model. Few jogging instances are not recognized properly, those are recognized as walk instead. This is because during jog few walk instances may happen automatically due to a change of force.

The window or segmentation size makes a different impact on the overall result. For the dynamic activities, every movement, their variation, and transitions of movements are better captured in long window sizes (128). However, for both the window sizes, the ensemble model is found to perform well.

5 Conclusion

In this paper, we have proposed a framework for deep learning based human activity recognition, that classifies both static and dynamic activities irrespective of usage behaviour and different hardware configurations of smartphones. Static activities are found to be recognized more accurately with 2D CNN as it takes care of intra and inter axes patterns from the activity images. Time-series signal patterns for different activities are sometimes found to be similar, however, the relation between adjoining axes can indicate distinguishing patterns. Thus, we converted time-series signals to 2D activity images and fed as input to CNN. Ensemble of condition-based classifiers has been designed for a robust system. Proposed work has been evaluated on real datasets collected from subjects using different smartphones and the subjects carried the smartphones differently as is done in daily life. Experimentation is conducted on the datasets considering different window sizes, to find out the variation of

Table 3 Summary of accuracy (%) of condition based classifier and the proposed ensemble

Trset	Testset	Classifier	Tuning pr	val-acc	Testacc
D1-hand	D1-RPT	Classifier1	2 conv 1 avg pooling	95.97	77.05
D1-SPT		Classifier2	2conv 1 avg pooling 1 dropout	97.56	93.36
D2-SPT		Classifier3	1 conv 1 avg pooling 1 dropout	95.10	87.59
D3-RPT		Classifier4	2conv 2 avg pooling 1 dropout	83.16	79.48
		Ensemble			94.49
D1-hand	D4-SPT	Classifier1	2 conv 1 avg pooling	95.97	78.86
D1-SPT		Classifier2	2conv 1avg pooling 1 dropout	97.56	82.14
D2-SPT		Classifier3	1 conv 1 avg pooling 1 dropout	95.10	87.15
D3-RPT		Classifier4	2conv 1 avg pooling 1 dropout	83.16	70.48
		Ensemble			87.33
D1-hand	D4-RPT	Classifier1	2 conv 1 avg pooling	95.97	72.06
D1-SPT		Classifier2	2conv 1 avg pooling1 dropout	97.56	82.48
D2-SPT		Classifier3	1 conv 1 avg pooling 1 dropout	95.10	82.75
D3-RPT		Classifier4	2conv 2 avg pooling 1 dropout	83.16	66.84
		Ensemble			83.15

The training datasets from D1, D2, D3 and test datasets from D1, D4. The training positions are SPT, RPT, Hand. Window size is 64

Table 4 Summary of accuracy (%) of condition based classifier and the proposed ensemble

Trset	Testset	Classifier	Tuning pr	Val-acc	Testacc
D1-hand	D1-RPT	Classifier1	2 conv 1 avg polling 1 dropout	96.39	80.05
D1-SPT		Classifier2	2 conv 1 avg polling 1 dropout	83.16	87.89
D2-SPT		Classifier3	1 conv 1 avg polling 1 dropout	94.38	82.44
D3-RPT		Classifier4	2 conv 2 avg polling 1 dropout	81.10	79.19
		Ensemble			88.92
D1-hand	D4-SPT	Classifier1	2 conv 1 avg polling 1 dropout	96.39	76.21
D1-SPT		Classifier2	2 conv 1 avg polling 1 dropout	83.16	84.32
D2-SPT		Classifier3	1 conv 1 avg polling 1 dropout	94.38	86.48
D3-RPT		Classifier4	2 conv 2 avg polling 1 dropout	81.10	69.05
		Ensemble			87.16
D1-hand	D4-RPT	Classifier1	2 conv 1 avg polling 1 dropout	96.39	77.13
D1-SPT		Classifier2	2 conv 1 avg polling 1 dropout	83.16	73.37
D2-SPT		Classifie3	1 conv 1 avg polling 1 dropout	94.38	76.48
D3-RPT		Classifier4	2 conv 2 avg polling 1 dropout	81.10	74.60
		Ensemble			80.34

The training datasets from D1, D2, D3 and test datasets are from D1, D4. The training positions are SPT, RPT, Hand. The Window size is 128. The headings are Trset: Training set, Tuning pr: Tuning Parameter, Val-acc: Validation accuracy, Testacc: Test accuracy

performance when the training devices and the test device are different. The system achieves 94% recognition accuracy even when the training and test conditions differ.

In future, we plan to collect more data from more number of users and publish the dataset. Other sensing dimensions would also be explored along with accelerometer readings, and fusion mechanisms are planned to be designed.

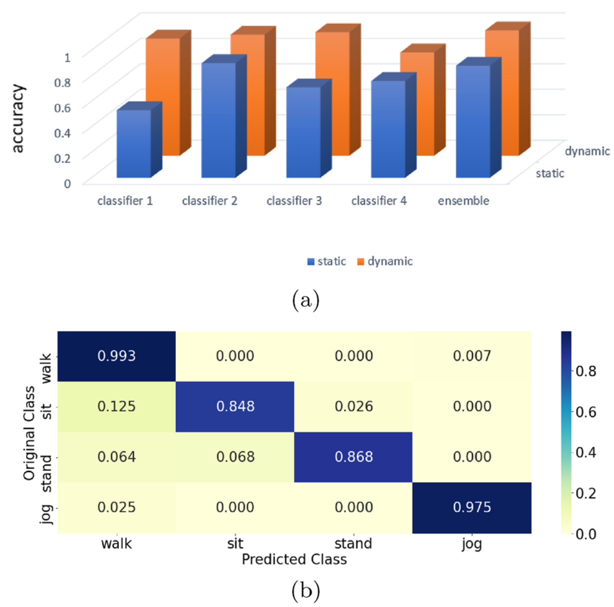

Fig. 13 **a** Classification accuracy (%) of four base classifiers and proposed ensemble model for static and dynamic activities. **b** Confusion matrix for all four activities when test dataset D1-RPT. The window size is 64

Funding Open access funding provided by Manipal Academy of Higher Education, Manipal.

Data availability Not applicable.

Declarations

Conflict of interest The authors declare that there is no Conflict of interest.

References

1. Wannenburg J, Malekian R (2017) Physical activity recognition from smartphone accelerometer data for user context awareness sensing. IEEE Trans Syst Man Cybern Syst 47:3142–3149
2. Subasi A, Khateeb K, Brahimi T, Sarirete A (2020) Human activity recognition using machine learning methods in a smart healthcare environment. In: Innovation in health informatics, Elsevier, pp 123–144
3. Babiker M, Khalifa OO, Htike KK, Hassan A, Zaharadeen M (2017) Automated daily human activity recognition for video surveillance using neural network. In: 2017 IEEE 4th international conference on smart instrumentation, measurement and application (ICSIMA). IEEE, pp 1–5
4. Sachdeva A, Mohan A (2021) A novel approach to human–computer interaction using hand gesture recognition. In: Data Science and Security: Proceedings of IDSCS 2020, Springer Singapore, pp 7–15
5. Kashika PH, Venkatapur RB (2022) Automatic tracking of objects using improvised Yolov3 algorithm and alarm human activities in case of anomalies. Int J Inf Technol 14:2885–2891
6. Nandy A, Saha J, Chowdhury C, Singh KP (2019) Detailed human activity recognition using wearable sensor and smartphones. In: 2019 International conference on opto-electronics and applied optics (Optronix). IEEE, pp 1–6
7. Saha J, Chowdhury C, Biswas S (2018) Two phase ensemble classifier for smartphone based human activity recognition independent of hardware configuration and usage behaviour. Microsyst Technol 24(6):2737–2752
8. Mannini A, Sabatini AM (2010) Machine learning methods for classifying human physical activity from on-body accelerometers. Sensors 10(2):1154–1175
9. Deng G, Cahill L (1993) An adaptive gaussian filter for noise reduction and edge detection. In: 1993 IEEE conference record nuclear science symposium and medical imaging conference. IEEE, pp 1615–1619
10. Jiang W, Yin Z (2015) Human activity recognition using wearable sensors by deep convolutional neural networks. In: Proceedings of MM '15: the 23rd ACM international conference on Multimedia, Australia, pp 1307–1310
11. Ragab MG, Abdulkadir SJ, Aziz N (2020) Random search one dimensional cnn for human activity recognition. In: 2020 International conference on computational intelligence (ICCI). IEEE, pp 86–91
12. Ronao CA, Cho SB (2016) Human activity recognition with smartphone sensors using deep learning neural networks. Expert Syst Appl 59:235–244
13. Nandy A, Saha J, Chowdhury C (2020) Novel features for intensive human activity recognition based on wearable and smartphone sensors. Microsyst Technol 26:1889–1903
14. Ahmad Z, Khan NM (2019) Multidomain multimodal fusion for human action recognition using inertial sensors. In: 2019 IEEE fifth international conference on multimedia big data (BigMM). IEEE, pp 429–434
15. Zhu Q, Chen Z, Soh YC (2018) A novel semisupervised deep learning method for human activity recognition. IEEE Trans Ind Inform 15(7):3821–3830
16. Zhou X, Liang W, Kevin I, Wang K, Wang H, Yang LT, Jin Q (2020) Deep-learning-enhanced human activity recognition for internet of healthcare things. IEEE Internet Things J 7(7):6429–6438
17. Jayanthi S, Priyadharshini V, Kirithiga V, Premalatha S (2024) Mental health status monitoring for people with autism spectrum disorder using machine learning. Int J Inf Technol 16:43–51
18. Paul P, George T (2015) An effective approach for human activity recognition on smartphone. In: 2015 IEEE international conference on engineering and technology (ICETECH), Coimbatore, India, pp 1–3
19. Jameel R, Shobitha M, Mourya AK (2022) Predictive modeling and cognition to cardio-vascular reactivity through machine learning in Indian adults with sedentary and physically active lifestyle. Int J Inf Technol 14:2129–2140
20. Guo Q, Liu B, Chen CW (2016) A two-layer and multi-strategy framework for human activity recognition using smartphone. In:

2016 IEEE international conference on communications (Icc). IEEE, pp 1–6
21. Cao L, Wang Y, Zhang B, Jin Q, Vasilakos AV (2018) Gchar: an efficient group-based context—aware human activity recognition on smartphone. J Parallel Distrib Comput 118:67–80
22. Saha J, Chowdhury C, Ghosh D, Bandyopadhyay S (2021) A detailed human activity transition recognition framework for grossly labeled data from smartphone accelerometer. Multimed Tools Appl 80:9895–9916
23. Shoaib M, Bosch S, Incel OD, Scholten H, Havinga PJM (2016) Complex human activity recognition using smartphone and wrist-worn motion sensors. Sensors 16:2–24
24. Sakshi BMPS, Chakraborty P (2024) Meta-heuristic based feature selection for aberration detection in human activity using smartphone inertial sensors. Int J Inf Technol 16:559–568
25. Sikder N, Chowdhury MS, Arif ASM, Nahid AA (2019) Human activity recognition using multichannel convolutional neural network. In: 2019 5th international conference on advances in electrical engineering (ICAEE), Dhaka, Bangladesh, pp 560–565
26. Wang X, Lu Y, Wang D, Liu L, Zhou H (2017) Using jaccard distance measure for unsupervised activity recognition with smartphone accelerometers. In: Web and Big Data: APWeb-WAIM 2017 International Workshops: MWDA, HotSpatial, GDMA, DDC, SDMA, MASS, Beijing, China, July 7–9, 2017, Revised Selected Papers 1, Springer International Publishing, pp. 74–83
27. Mekruksavanich S, Hnoohom N, Jitpattanakul A (2018) Smartwatch-based sitting detection with human activity recognition for office workers syndrome. In: 2018 International ECTI Northern section conference on electrical electronics, computer and telecommunications engineering (ECTI-NCON), Chiang Rai, Thailand, pp 160–164
28. Sri Harsha NC, Anudeep YGVS, Vikash K et al (2021) Performance analysis of machine learning algorithms for smartphone-based human activity recognition. Wirel Pers Commun 121:381–398
29. Garcia-Gonzalez D, Rivero D, Fernandez-Blanco E, Luaces MR (2023) New machine learning approaches for real-life human activity recognition using smartphone sensor-based data. Knowl Based Syst 262(110):260
30. Asmara RA, Siradjuddin I, Hendrawan N (2020) Improving basketball recognition accuracy in samsung gear s3 smartwatch using three combination sensors. In: 2020 4th International conference on vocational education and training (ICOVET). IEEE, pp 386–390
31. Zhu J, San-Segundo R, Pardo JM (2017) Feature extraction for robust physical activity recognition. Hum Cent Comput Inf Sci 7(1):16
32. Mekruksavanich S, Jitpattanakul A (2020) Smartwatch-based human activity recognition using hybrid lstm network. In: 2020 IEEE sensors, pp 1–4
33. Wan S, Qi L, Xu X, Tong C, Gu Z (2020) Deep learning models for real-time human activity recognition with smartphones. Mob Netw Appl 25:743–755
34. Ankita SR, Babbar H, Coleman S, Singh A, Aljahdali HM (2021) An efficient and lightweight deep learning model for human activity recognition using smartphones. Sensors 21:3845
35. Garcia-Gonzalez D, Rivero D, Fernandez-Blanco E, Luaces MR (2023) Deep learning models for real-life human activity recognition from smartphone sensor data. Internet Things 24(100):925
36. Cheng WY, Scotland A, Lipsmeier F, Kilchenmann T, Jin L, Schjodt-Eriksen J, Wolf D, Zhang-Schaerer YP, Garcia IF, Siebourg-Polster J et al (2017) Human activity recognition from sensor-based large-scale continuous monitoring of Parkinson's disease patients. In: CHASE '17:2017 IEEE/ACM international conference on connected health: applications, systems and engineering technologies (CHASE), Philadelphia Pennsylvania, pp 249–250
37. Hassan MM, Uddin MZ, Mohamed A, Almogren A (2018) A robust human activity recognition system using smartphone sensors and deep learning. Future Gener Comput Syst 81:307–313
38. Hinton GE, Osindero S, Teh YW (2006) A fast learning algorithm for deep belief nets. Neural Comput 18(7):1527–1554
39. Ravi D, Wong C, Lo B, Yang GZ (2016) Deep learning for human activity recognition: a resource efficient implementation on low-power devices. In: 2016 IEEE 13th international conference on wearable and implantable body sensor networks (BSN). IEEE, pp 71–76
40. Xu C, Chai D, He J, Zhang X, Duan S (2019) Innohar: a deep neural network for complex human activity recognition. IEEE Access 7:9893–9902
41. Gumaei A, Hassan MM, Alelaiwi A, Alsalman H (2019) A hybrid deep learning model for human activity recognition using multimodal body sensing data. IEEE Access 7:99152–99160
42. Chen Y, Xue Y (2015) A deep learning approach to human activity recognition based on single accelerometer. In: 2015 IEEE international conference on systems, man, and cybernetics. IEEE, pp 1488–1492
43. Hur T, Bang J, Lee J, Kim JI, Lee S et al (2018) Iss2image: a novel signal-encoding technique for cnn-based human activity recognition. Sensors 18(11):3910
44. Zebin T, Scully PJ, Ozanyan KB (2016) Human activity recognition with inertial sensors using a deep learning approach. In: 2016 IEEE sensors. IEEE, pp 1–3
45. Ronao CA, Cho SB (2016) Human activity recognition with smartphone sensors using deep learning neural networks. Expert Syst Appl 59:235–244
46. Sadouk L, Gadi T (2017) Convolutional neural networks for human activity recognition in time and frequency-domain. In: First international conference on real time intelligent systems. Springer, pp 485–496
47. Ustev Y, Incel O, Ersoy C (2013) User, device and orientation independent human activity recognition on mobile phones: challenges and a proposal. In: UbiComp 2013 adjunct-adjunct publication of the 2013 ACM conference on ubiquitous computing, Zurich Switzerland. pp 1427–1436
48. Yang R, Wang B (2016) Pacp: a position-independent activity recognition method using smartphone sensors. Information 7:72
49. Wang C, Zhang J, Li M, Yuan Y, Xu Y (2014) A smartphone location independent activity recognition method based on the angle feature. In: International conference on algorithms and architectures for parallel processing. Springer, pp 179–191
50. Siirtola P, Koskimäki H, Röning J (2019) From user-independent to personal human activity recognition models exploiting the sensors of a smartphone. arXiv:1905.12285
51. Ding X, Jiang T, Zhong Y, Yang J, Huang Y, Li Z (2021) Device-free location-independent human activity recognition via few-shot learning. In: 2021 IEEE/CIC international conference on communications in China (ICCC Workshops). IEEE, pp 106–111
52. Ozcan T, Basturk A (2019) Transfer learning-based convolutional neural networks with heuristic optimization for hand gesture recognition. Neural Comput Appl 31(12):8955–8970
53. Ding X, Jiang T, Li Y, Xue W, Zhong Y (2020) Device-free location-independent human activity recognition using transfer learning based on cnn. In: 2020 IEEE international conference on communications workshops (ICC Workshops). IEEE, pp 1–6
54. Almaslukh B, Artoli AM, Al-Muhtadi J (2018) A robust deep learning approach for position-independent smartphone-based human activity recognition. Sensors 18(11):3726
55. Saha J, Ghosh D, Chowdhury C, Bandyopadhyay S (2021) Smart handheld based human activity recognition using multiple instance multiple label learning. Wirel Pers Commun 117:923–943
56. Winograd S (1978) On computing the discrete Fourier transform. Math Comput 32(141):175–199

57. Nussbaumer HJ (1981) The fast fourier transform. In: Fast Fourier transform and convolution algorithms. Springer, pp 80–111
58. Shukla AK, Shukla A, Singh R (2024) Automatic attendance system based on CNN–LSTM and face recognition. Int J Inf Technol 16(3):8955–8970
59. Wang W, GuoY HB, Zhao G, Liu B, Wang L (2011) Analysis of filtering methods for 3d acceleration signals in body sensor network. Int Symp Bioelectron Bioinform 2011:263–266