International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Human Activity Detection using Profound Learning with Improved Convolutional Neural Networks

[1*]S. Anthonisamy,[2]P. Prabhu

Submitted: 06/02/2024 **Revised**: 14/03/2024 **Accepted**: 20/03/2024

Abstract: Human Activity recognition (HAR) is an interesting area of research mainly due to the availability of low cost sensors and accelerometers live streaming of data and advances in technology. HARs involve identifying various human activities such as walking, running, sitting, sleeping, standing, showering, cooking, driving, opening the door, abnormal activities, etc. are recognized. The data can be collected from wearable sensors or accelerometer. HARs can be extensively used in medical diagnostics for keeping track of elderly people, HARs approaches analyze data acquired from sensing devices, including vision and embedded sensors. HARs are assistive technologies mainly used for taking care of elders in healthcare. Approaches of HARs attempt to predict people's movements often indoors and based on sensor data like accelerometers of smart phones. In terms of classifications, HARs are challenging tasks as they involve time series data where Deep Learning Techniques (DLTs) like CNNs (Convolution Neural Networks) have the ability to correctly engineer features from these raw data while building their learning models. This paper proposes Human Activity Detections using Profound Learning (HADPL) based on CNNs which detects HARs from captured accelerometer data. HADPL was tested on WISDM_Act_v1.1 dataset and evaluated for its performances in terms of precisions, accuracies, recalls and F1-scores where it achieved a decent level of accuracy by scoring up to 95 percent. The proposed technique can be implemented for monitoring elderly people based on captured or stored HAR data.

Keywords*: *Human Activity Detections, Deep Leaning Techniques, Convolution Neural Networks, Machine learning Techniques, healthcare monitoring.

1. Introduction

Technological enhancements incomputing powers, lowered manufacturing costs [1] and sensors integrated into smart Hand Held Devices (HHDs) along with the evolution of video surveillances or CCTVs (closed-circuit televisions) [2] have all resulted in better video qualities and secure communications where CCTVs are used for security/ monitoring [3]. HARs refer computer-based detections; analyses and understanding of human activities for learning their behaviors with the use of Machine Learning Techniques(MLTs) where they help comprehend human activities from input data sources including multimedia and sensors [5]. HARs have wide prospects for applications in education, entertainments, detection of injuries in sports, elderly and smart home environment monitoring, specifically in identifying daily critical activities of elderly patients and managing their rehabilitations in addition to sensing symptoms of certain diseases. HARs use multi-modal data generated from various devices to detect human postures, physical activities behaviors where HAR researches can be classified based on assisted technologies, such as video, wearable and mobile phone sensors and wireless signals. Video-based methods primarily identify human activities by capturing images, video or camera surveillance whereas other wearable device like inertial sensors of mobile or wearable embedded sensors placed on different body parts to infer details of human activity and postural transition or signals from wireless devices have evolved. However, sensor data generated from accelerometers or wearable devices, play immeasurable roles on human's motion analyses, activity monitoring and detections. With the rapid advancement and popularization of smart phone technology, especially in the field of microelectronics and sensors, extracting knowledge from data acquired by ubiquitous sensors has become a very active research field. Figure 1 depicts some of the complex forms for human activities.

Fig. 1 - Human poses during complex activities

Though sensors are aimed at specific services, they generally collect raw data from targets [4], and need to be processed for analysis. History of HARs dates back to the 1990s when a study [6] processed data collected raw data

1 Research Scholar (Part-Time), Department of Computer Applications, Alagappa University, Karaikudi, India.
2 Associate Professor in Information Technology, Centre for Distance and Online Education, Alagappa University, Karaikudi.630003, Tamil Nadu, India;
Email:antony2006reegan@gmail.com;
prabhup@alagappauniversity.ac.in

and subsequent developments of HHDs, wearable devices, and CCTVs systems motivated researchers to work on practical applications of HARs. These systems have been applied in surveillances [7], analyzing human behaviors [8], recognizing gestures and patient monitoring [9], ambient assisted living [10] and a variety of other healthcare systems [11] that involve direct/indirect interactions with HHDs, For example, obese or diabetic or cardiovascular diseases need to strictly follow diets and exercises [12]. Tracking the activities of these patients is imperative to clinicians for monitoring their progress. People with reduced mental ability or disorders also need to be monitored continuously for assessing unusual activities which assist in preventing dire consequences [13]. In the army, assessing soldiers' actions or positions in tactical situations is fundamental for guaranteeing their safety while engaged in combats [14]. In spite of all these advantageous, HARs are considered challenging, due to the absence of standardizations in data collections and the volumes of data generated by devices. Approaches for HARs can be categorized into vision/sensor based on the type of data received [15]. Vision based approaches analyze images or videos while sensor based techniques examine raw data from sensors (wearable/environmental) where wearable devices worn by users help in automatic detections/tracking of their activities including sitting, jogging, running, and sleeping [16]. MLTs have been used for HARs, but under strictly constrained environments or limited data inputs. Moreover, MLTs require data to be clean resulting in pre-processing as the preliminary step. Manual interventions in these applications are time-consuming and lack efficacy [17]. MLTs, while using shallow features result in poor performances [18]. This has made researches look at DLTs as a solution as they have proven performances in object detections/classifications [19], and NLPs (natural language processes) [20]. Though a part of MLTs. DLTs select right features using their hidden layers [21]. Hence, many recent studies have proposed HAR frameworks using DLTs, the main motivation for this work. This paper proposes HADPL (Human Activity Detections using Profound Learning) based on CNNs which detect HARs from captured accelerometer data.

Profound learning techniques learns from the large volume of data and provides meaningful insights for better decision making. These techniques depend on different layer of information with changes in the input.Profound learning techniques have been in charge of a large number of the ongoing essential advances in machine learning [43]. Following this introductory section, the next section details on studies related to this work whole the proposed methodology is detailed in section three. Section four displays HADPL's results while its subsequent section concludes this paper with a summary.

2. Literature review

A. Hand Held Device Methods

HHDs have become a part of modern people's daily lives and integrating HARs in HHDs is growing in popularity. HHD based data are identified statistically for inferring HARs where they effectively reduce computational times and complexities [22]. On the other hand, HARs based on CCTVs and videos are considered as invasions of privacy [23]. These data sources also face illumination issues resulting performance degradations. These issues have made non-invasive sensors-baseddata, a popular source for processing HARs. HHDs and wearable devices are also location-independent, easy to deploy and cost-effective [24]. HARs data can also be stochastically predicted by considering them sequence of states which has been exploited by using HMMs (hidden Markov models) [25] and HCRFs (hidden conditional random fields) [26] which modeled human behavior as a sequence of actions stochastically.

The study in [27] proposed detection of complex actions from multiple subjects where subjects followed certain rules while performing actions. Based on the first-order logic and probabilistic approaches like Markov networks, the study inferred events when they occurred. The study in [28] recognized actions by discriminating attributes associated with certain characteristics displayed by subjects. The study treated these attributes as latent variables and SVMs (Support Vector machines) captured attribute importance. As complex human activities cannot be recognized directly using rules, they were decomposed into simpler atomic actions on which individual actions were employed for recognitions. The limitation of the study stemmed rule/attribute annotations, making training a time consuming process and sensitive to errors while processing based on user defined annotations. DLTs have been proposed to overcome this issue. The study in [29] used transfer learning[29] while semantic/discriminative attribute learning was proposed in [30] to automatically generate and handle most human activities. The growing popularity of data-driven approaches have resulted in studies using DLTs for recognizing activities from HAR datasets [31]. The study in [32] proposed a new approach for improving HARs using PCAs (principal component analyses), LDAs (linear Discriminant analyses) and enhanced SVMs. for handling non-informative sequence features and balancing imbalanced classes. Improved model using PCA was proposed for business intelligence recommender system[45-46].

B. Deep Leaning based methods

DLTs have the ability to discover required characteristics from raw data based on their representation learning. In recent years, DLTs have made major breakthroughs in recognizing images [33], voices [34] and healthcare [35]. DLTs have also been tested on wearable device sensors and they include restricted Boltzmann machines, auto-encoders, CNNs. The study in [36]. recognized human activities with SAEs (Stacked Autoencoders) and established relationships between HHDs and individual health. The proposal of [37] used deep SAEs for extracting high-level features,feature extractions and training classifiers on HARs. The proposal by [38] used DCNNs (deep CNNs) for efficient and effective activity recognition based on accelerometer and gyroscope data as these devices exploit inherently characterize activities into time-series signals. Multi-modal CNNs with 2D kernels was used in [39] to explore local and spatial dependencies of sensors. Relevant features for HARs were extracted in [40] which used LSTMs (long short-term memories) to recognize human activities from accelerometer data. DLTs can generate different models implement automatic features learning and hence have high accuracy performances, flexibility and robustness.

3. Methodology

HARs are important in human-human interactions and interpersonal relationships, though it is complex to extract them. One key area of research in computer vision is human ability to identify HARs where applications like video surveillances, HCIs (human-computer interactions), and robots have been used or mimicked human behaviours HARs need to understand kinetic states of humans in order to recognize them effectively like "walking" and "running". The complexities of these tasks can be attacked by breaking them into smaller activities for recognitions where identifying objects usually aid in better understanding of human behaviours by providing relevant information on occurrences [41].

C. Proposed HADPL model

Human's activities are influenced by their habits adding to complexities in determining them. This research work proposes HARs using HADPL based on time series data from datasets and aims to classify sensor-based HAR datasets. The HADPL framework is depicted as Figure 2.The basic aim of the suggested framework is analyzing raw sensor data for human activities where set of steps are followed sequentially. Since, the accuracy of any analysis is dependent on its selection of features.More efforts are spent on automations or tuning.

Pre-processing steps are preliminary before training and also demonstrates the use of CNNs on raw sensor data for predicting HARs. WISDM_Act_v1.1 dataset which includes HAR information from accelerometers UCI machine learning repository was used in this study. The raw accelerometer and gyroscope sensor data is collected from the smartphone and smartwatch at a rate of 20Hz and collected from 51 test subjects in performing 18 activities for 3 minutes apiece. The data used in this study, sampled signals, reflect human activities as implied by accelerometers and gyroscope sensors. The proposed framework clean normalizes and prepared raw sensor data for use by CNN's predictions. The preprocessing steps followed include Removing Unwanted Characters, Explicit conversions of data to float values and examining data Spread for assessments.Hyper parameter tuning was conducted for identifying best parameters for CNN.

Finally, the columns are labeled to identify features. It is better understanding how the recordings are spread across the different users. The data was encoded for each activity namely Downstairs, Jogging, Sitting, Standing, Upstairs, Walking. The labels were encoded for is the study for processing though they could be converted back to original label texts. HADPL follows its pre-processing with extensive preparation of data for training and testing. The data is split into training and testing. HADPL uses a better splitting approach by taking id number <= 20 as training sets and others as testing sets. After the split's, Training Data is normalized for enhanced predictions. This normalized data is then reshaped and labeled for training. As a part of feature selections, HADPL is interested in three main parameters namely, TIME_PERIODS, STEP_DISTANCES AND LABELS.

.

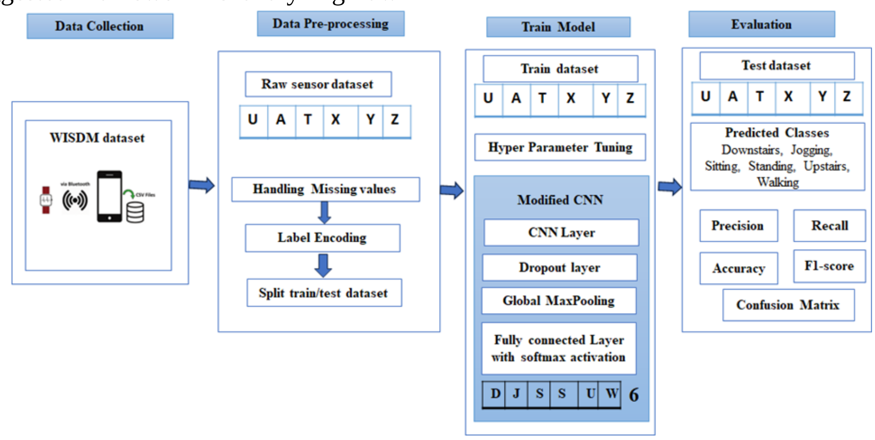

Fig.2 The HADPL framework

The data was examined for its spread by HADPL as depicted in Fig.3 where it can be seen that there is more data for walking and jogging activities.

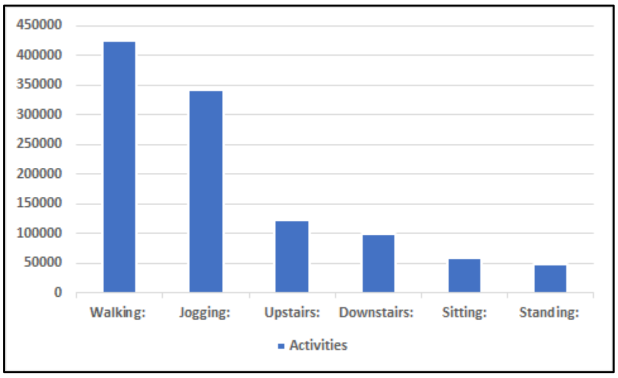

Fig.3 Class distribution of activities

The CNNs are fed with Number of time periods, Number of sensors and Number of classes. Since the data is two dimensional (80x3) the data is flattened for CNN's input layer (vector of length 240) which are fed and prior to being fed into CNNs, the labels are one-hot-encoded. CNN's first layer reshapes data into its old format while its last two layers flatten the data for running softmax activations and computing probabilities of classes (Downstairs, Jogging, Sitting, Standing, Upstairs, Walking). The outputs of HADPL are predicted values. Figure 4 depicts then data taken for training.

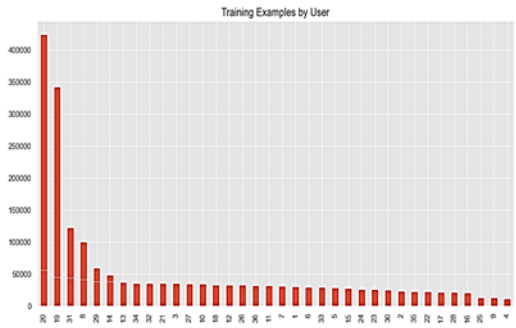

Fig. 4 – Training Data Snapshot

HADPL is based on modified CNNs (Convolution Neural Networks) are one of the most powerful DLTs (Deep Learning Techniques). CNNs have proved their ability in different domains of computer vision. CNNs used consists three layers namely convolution, sub-sampling and fully connected layers. The CNN architecture is depicted in Figure 5 and the explanation of each layer is subsequently explained.

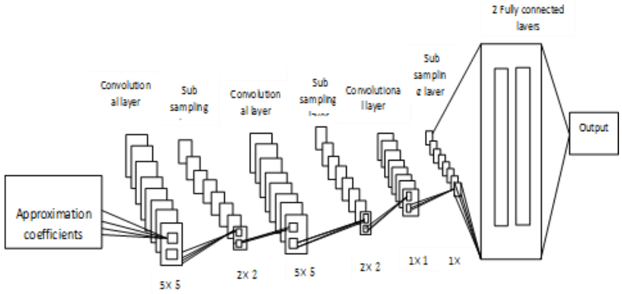

Fig.5: Convolution Neural Network

Convolution layer

In this work, obtained approximation coefficients are inputs. In the convolution layer, input coefficients are convolved with a kernel (filter) which is generically called a filter. These convolutions result in generation of *n* coefficient outputs with size i*i are referred to as feature maps. CNNs may use multiple convolution layers where inputs/outputs of these layers are feature vectors with n filters in the convolution layers which get convolved with inputs. The generated feature maps have a depth of (*n**) and is equal to the count of applied filters in convolutions. Each specific location of an input feature is a filter map. The resulting output feature maps of the *l*-th convolution layer ($C_i^{(l)}$) can be calculated using Equation (1)

$$C_i^{(l)} = B_i^{(l)} + \sum_{j=1}^{a_i^{(l-1)}} K_{i,j}^{(l-1)} * C_j^{(l)} (1)$$

Where, $B_i^{(l)}$ denotes a bias matrix while $K_{i,j}^{(l-1)}$ denotes *a*a sized kernel* that connects jth feature map of the layer $(l-1)$ with ith feature map of the same layer. In Equation (2), The first convolution layer $C_i^{(l-1)}$ is the input space or $X_i = C_i^{(0)}$. After feature maps are generated by the kernel an activation function transforms convolution layer outputs non-linearly

$$Y_i^{(l)} = Y(C_i^{(l)}) (2)$$

Where, $Y_i^{(l)}$ denotes activation function outputs while its inputs are denoted by $C_i^{(l)}$.

CNN's activation functions can be one of sigmoid or tanh or ReLU(Rectified Linear Unit) functions. DLTs predominantly use activations functions as they reduce interactions and non-linear effects. Moreover, activation functions have the advantage of quicker training due to error derivatives which become marginal in saturated regions and thus resulting in nullifying weight updates. This concept is called vanishing gradients. This work uses ReLUs which output 0 for negative inputs while positive values are unchanged and depicted as Equation (3)

$$Y_i^{(l)} = \max(0, Y_i^{(l)}) \quad (3)$$

Sub sampling or pooling Layer

The primary objective of this layer is dimensionality reduction. Previously extracted convolution layer's features maps are sub-sampled. The sub-sampling operation executed between the feature maps and a mask can be executed in many ways including average/sum/max pooling. CNN's layers produce Y_q^c beyond the convex hull of inputs $\{X_p^c\}_{p\in R_q}$. for better recognition where channel superscript c and output subscript q are omitted for simplicity. Initially local neurons $\{X_p\}_{p\in R}$ are activated using Bernoulli distribution with mean μ_x and standard deviation σ_x.

$$\bar{X} \sim N(\mu_x, \sigma_x) \Leftrightarrow \bar{X} = \mu_x + \epsilon\sigma_x \quad (4)$$

Where, $\mu_x = X_S$, $\epsilon \sim N(0,1)$, $\epsilon \in (-\infty, +\infty)$, X_S − Input features successfully beyond the particular class and $\sigma_x = \sqrt{X_S(1 - X_S)}$

The local pooling knowledge obtained is used to modify equation (5) for limiting values below the mean μ_x in the output as shown in Equation (6)

$$Y = \mu_x + |\epsilon|\sigma_x, \qquad \epsilon \sim N(0,1) \Leftrightarrow Y = \mu_x + \eta\,\sigma_x, \quad \eta \sim N_h(1), \eta \in [0, +\infty) \quad (6)$$

Where the distribution $N_h(\sigma_0)$ with $\sigma_0 = 1$ is used as preceding probabilistic model as it results in a fixed pooling based on Equation (6) and produces Y stochastically without the use of any pooling parameter.

Fully Connected layer

The ultimate CNN layer is a traditional FFN (Feed Forward Network) and can consist of one or more hidden layers. This layer's Softmax activation function output is depicted in Equation (7) and (8):

$$Y_i^{(l)} = f(z_i^{(l)}) \quad (7)$$

$$\text{where } z_i^{(l)} = \sum_{i=1}^{m_i^{(l-1)}} w_{i,j}^{(l)} y_i^{(l-1)} \quad (8)$$

Where the weights $w_{i,j}^{(l)}$ are tuned by the fully connected layer for representations of each class and *f* denotes the non-linear transfer function. The non-linear outputs in the connected layer is built based on its neurons and not separate layers like convolution or pooling layers. The output of the Modified CNN is estimated approximation coefficients for the current image frames.

Algorithm : HADPL

Input: HAR sensor Dataset(Subject,Timestamp,activity,X,Y,Z)
Labels : Downstairs, Jogging, Sitting, Standing, Upstairs, Walking
Output: Classified Activities, Evaluation result

```
Begin
  //Data preprocessing:
1. Remove Special Characters
2.  Handling missing values
3.  For each Row in HD  // convert Axis into float values
      For Each Column in HD
          x= Float(x);     y= Float(y);     z= Float(z)
End for
End for
4. #Split Data into Training and Test Set
   For each Row in HD
      If Userid<=20 then Train Set
Else Test Set
      End if
End for
5. Transform the labels from String to Integer using Label Encoder
   For each Row in Train set
      For Each Column in train set
          x= avg(x);  y= avg(y);   z= avg(z)
End for
End for
```

6. Apply Modified CNNs on the basis of the design of the CNN structures
#Leaky ReLU activation function, and the over fitting prevention method is based on dropout and SGD

a) Pretrain the filter, and initialize the filter size for HADPL Data
b) Process the training set data into the same filter size, and read the data to form the dataset matrix X.
c) Initialize the weight $w^{(l)}_{i,j}$ and bias b_i and invoke the kernel function *def Kernel()* of CNNs to initialize parallel operations.
d) Conv2d is used for two-dimensional convolution operation to obtain first layer convolution feature matrix $X^{(1)}$.
e) The first layer convolution feature matrix $X^{(1)}$ is used as input data of pool layer and obtain feature matrix $X^{(2)}$ based on Equations stated above
f) Use CNN's optimizer function to derive learning rate of the tuning optimizer, and use e weights to update-weights w_i and the biases b_i, thus obtaining feature matrix $X^{(3)}$.
g) Generate the second convolution following Steps d, e, and f to derive the feature matrix $X^{(4)}$.
h) Merge the feature matrix $X^{(4)}$ into a column vector as the input of the neuron at the full-joint layer, multiply it with the weight matrix plus the bias, and then use the Leaky ReLU activation function to obtain the eigenvector b_1.
i) Use the eigenvector of the fully connected stratum as the input of the dropout layer, compute the output probability of the neuron in the dropout layers using equations stated above to obtain eigenvector b_2
j) Use the eigenvector b_2 as inputs and Softmax classification outputs to achieve the results.

#Note , in the convolution feature calculation for step 4 and step 5, the step length is 2 and the margin is set to ##0. The pooling operation uses a 3 × 3 matrix on the basis of a policy previously presented in literature to #ensure that the inputs and outputs after feature

extractions have the same size. ReLU activation function is #used in each step to activate the neuron.
7. Enable validations (Accuracy)
End

4. Results and discussion

This section displays stage wise experimental results of the proposed scheme executed using Python 3.9 on an AMD athelon processor with 4 GB memory. Python 3.7.5 was used for implementations. The experiments were coded for the WISDM_Act_v1.1 dataset which includes HAR information from accelerometers UCI machine learning repository. The raw accelerometer and gyroscope sensor data is collected from the smartphone and smartwatch at a rate of 20Hz. It is collected from 51 test subjects as they perform 18 activities for 3 minutes apiece. The sensor data for each device (phone, watch) and type of sensor (accelerometer, gyroscope) is stored in a different directory (so there are 4 data directories). In each directory there are 51 files corresponding to the 51 test subjects. Attribute Information includes subject-id values from 1600- 1650 that identifies one of the 51 test subjects. The activity-code: character between 'A' and 'S' (no 'N') identifies activities The Data Set includes conventional activities like general walking, walking upstairs/downstairs, sitting, standing, and lying down with the help of sensors of people between the ages of nineteen and forty eight. The x, y, and z accelerometer data (linear acceleration) and gyroscopic data (angular velocity) were recorded. Each participant executed the exercises two times with sensors attached to the left and right. Timestamp an integer where x: represents the sensor reading (accelerometer or gyroscope) for the x dimension, y: represents the sensor reading (accelerometer or gyroscope) for the y dimension and z: represents the sensor reading (accelerometer or gyroscope) for the z dimension. Table.I depicts a snapshot of the WISDM dataset.

TABLE.I A SNAPSHOT OF THE WISDM DATASET.

User	activity	timestamp	X-axis	Y-axis	Z-axis
33	Jog	4.911E+13	-0.6946	12.68	0.5039
33	Walk	4.941E+13	-0.8036	13.4432	0.6129
35	Sit	1.896E+11	5.05	6.74	5.75
21	Up	1.192E+14	1.69	14.86	-0.38
34	Down	2.409E+12	-4.37	16.59	-5.01
20	Stand	5.961E+13	-1.04	9.66	2.6

D. Preprocessing

As a part of pre-processing following constants were created: LABELS for a list of labels to be used multiple times (Downstairs, Jogging, Sitting, Standing, Upstairs, Walking), TIME_PERIODS to store lengths of time segments (80) and STEP_DISTANCE for determining overlaps between consecutive time segments (50). Special characters after the last columns were removed and z-axis column was explicitly transformed to float for the learning as non-conversions to floats will result in NAN error. The samples of six activities were examined for their spread. Higher accelerations for jogging/walking were found compared to sitting. The activities of users were also encoded as a part of pre-processing.

E. Train the model

It is important to separate the whole data set into training and test sets as it enhances overall performance of model during training and validations against test sets. But model may not generalize data unseen data effectively. Data splits ensure that networks learn from a few samples and then use it for predicting human activities in unseen samples. Improper data splits also result in reduced performances of networks and hence HADPL used ID's for splits where IDs <= 20 were used for training the CNN model while other formed the test set. As a part of training the values of features were normalized (values between 0 and 1) and the actual accelerometer axis data were used by rounding it off to the three features. The feature space was then reshaped into segments for CNN's learning. Reshaping involved data and the defined labels constant along with record lengths where 80 steps for 20 Hz sampling rates amount to 4 seconds (0.05 * 80 = 4) were used to separate features (x/y/z acceleration values with labels or associated activities. Figure 6 depicts the wave forms of accelerometer data.

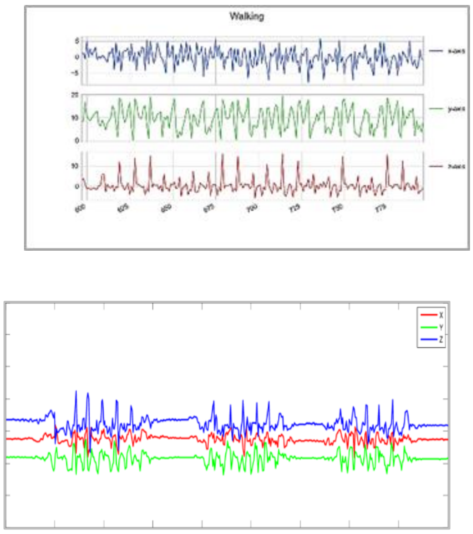

Fig. 6 - Accelerometer waveform of walking activity

The samples split for training resulted in a two dimensional matrix of the shape 80x3. The number of time periods within

Model: "sequential"

Layer (type)	Output Shape	Param #
reshape (Reshape)	(None, 80, 3)	0
dense (Dense)	(None, 80, 100)	400
dense_1 (Dense)	(None, 80, 100)	10100
dense_2 (Dense)	(None, 80, 100)	10100
flatten (Flatten)	(None, 8000)	0
dense_3 (Dense)	(None, 6)	48006

Total params: 68,606
Trainable params: 68,606
Non-trainable params: 0

one sample record was 80 while the number of sensors three. The CNN model was constructed using classes count which are the output layer nodes of the CNN. As activities needed to be predicted, classes count is the count received from encoding of the previous steps. The data fed into CNN (80x3) directly cannot be processed and hence, the inputs were flattened (240 values). HADPL also converts any data that is not float to float as they are easily acceptable to CNNs. HADPL also uses one-hot-encoding of labels. The CNNs proposed in HADPL framework used 3 hidden layers of 100 fully connected nodes each. Since, the data inputs were reshaped inputs with vector of length 240, this was reversed in CNN's first layer to its "old" format. The final two layers flatten this data again before running activations (softmax) which compute probabilities of classes in the data. Figure 7 depicts the model's summary.

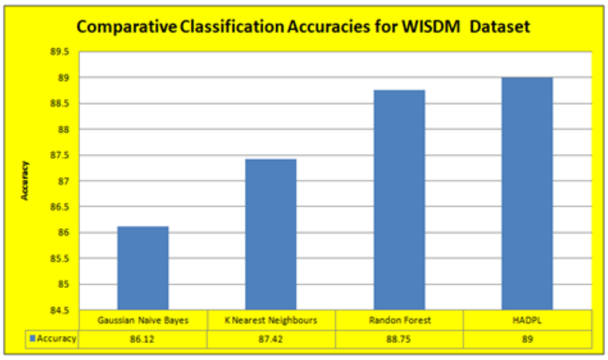

Fig. 7 HADPL Model Summary

HADPL monitors training accuracies and when raining results fail to improve for two consecutive epochs, training is stopped with the best model. The hyper-parameters used in training were simple and easily configurable parameters: batch size (400 samples) and epochs (50) with a 70:30 split for training and validations. Gradient descent uses a parameter called learning rate. Initially, the steps are bigger with greater learning rates and later reduce by shorter size of steps. The Loss Function also decreases reducing losses/costs. Epochs, batch size, iterations only when the data is voluminous and complete data cannot be passed all at once to the computer, hence they are divided into smaller chunks and fed one after the other and network's weights are updated at the end of every step to fit it to the data given. Epochs pass data forward and backward through network only once where it in sent as several smaller batches. The complete data is passed multiple times to optimize learning and Gradient Descent graphs iteratively. Hence, weight updates on increasing epochs transforms under fits of data to optimal to over fits for the gradient descent curve. Batch size and batches are two different things where batches counts are dependent on the batch size. Iterations are number of batches needed to complete one epoch.

F. *Evaluation*

Smartphone users access personal information through a variety of sensors to enhance user experiences. Motion sensors (accelerometer and gyroscope) provide important information to facilitate the identification and monitoring of user movements. This work's suggested HADPL was tested on WISDM_Act_v1.1 dataset which includes HAR information from accelerometers. The framework was evaluated for its performances in terms of precisions, accuracies, recalls and F1-scores where it achieved a decent level of accuracy by scoring 89%. Figure.8 depicts HADPL framework's Accuracy with Loss.

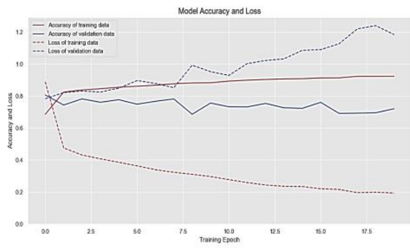

Fig. 8 HADPL framework's Accuracy with Losses

Performance metrics

The performance metrics used are accuracy, precision, recall,F-measure, Micro and Macro average precision using TP- Truly Predicted; TN – truly Rejected; FP = True but Negated and FN – Negative and truly negated. The metrics are shown in equations (9) to (14).

\

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad \ldots\ldots\ldots\ldots\ldots\ldots(9)$$

$$Precision = \frac{TP}{TP + FP} \quad \ldots\ldots\ldots\ldots\ldots\ldots(10)$$

$$Recall = \frac{TP}{TP + FN} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(11)$$

$$F1 - score = \frac{2}{\frac{1}{Recall} + \frac{1}{Precision}} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(12)$$

$$\text{Micro avg Precision} = \frac{TP1 + TP2}{TP1 + TP2 + FP1 + FP2} \quad \text{………..(13)}$$

$$\text{Macro avg Precision} = \frac{P1 + P2}{2} \quad \text{………………….(14)}$$

Weighted average is the weighted average of precision/recall/f1-score. Figure.9 depicts comparative accuracies obtained from MLAs along with MLAs. Fig.Comparative Classification AccuracyIt is evident from figure 9 that the proposed HADPL performs better when compared to other techniques on the WISDM HAR dataset. Confusion Matrix is summarization of prediction where the performance metrics of accuracy, precision, recalls and F1-scores are used. Dataset may be imbalanced in terms of samples and these measures point out the efficiency of proposed models. Precisions imply reliability of predictions and are useful when False Positive are higher than False Negatives. Recall values are when False Negatives are more than False Positives. Recall values are important in medical cases where false alarms may not be an issue, but actual positive cases should be detected. F1-Score captures both precision and recall trends in a single value and the harmonic mean of the two.

Table II shows the HADPL classification accuracy for WISDM dataset (2).

TABLE II. HADPL classification Conv2D accuracy for WISDM dataset(1)

Classes	Precision	Recall	F1	Support
Downstairs	0.71	0.61	0.66	1033
Jogging	0.98	0.99	0.98	3837
Sitting	1.00	0.99	1.00	750
Standing	0.99	0.94	0.96	542
Upstairs	0.74	0.54	0.63	1315
Walking	0.86	0.95	0.90	4987
Accuracy			0.89	12464
Macro avg	0.88	0.84	0.86	12464
Weighted avg	0.89	0.89	0.89	12464

The precision of the model is good for predicting jogging (1), sitting (2), standing (3), and walking (5).

Table III shows the confusion matrix obtained for WISDM dataset(1) using HADPL method.

TABLE III Confusion matrix obtained for WISDM dataset(1) using HADPL method.

Confusion Matrix

Actual Label	Class Labels	W.Downstairs	Jogging	Sitting	Standing	W.Upstairs	Walking
	W.Downstairs	673	9	0	6	102	32
	Jogging	12	2777	0	1	28	14
	Sitting	3	0	482	12	2	0
	Standing	2	1	5	400	2	0
	W.Upstairs	57	62	1	8	878	24
	Walking	27	4	0	0	20	3416
		Predicted Label					

Table IV shows the results of HADPL classification Conv1D accuracy for WISDM dataset(2).

Table IV – HADPL classification Conv1D accuracy for WISDM dataset (2)

Classes	Precision	Recall	F1 score	Support
Downstairs	0.87	0.82	0.84	822
Jogging	0,97	0.98	0.98	2832
Sitting	0.99	0.97	0.98	499
Standing	0.94	0.98	0.96	410
Upstairs	0.85	0.85	0.85	1030
Walking	0.98	0.99	0.98	3467
Accuracy			0.95	9060
Macro avg	0.93	0.93	0.93	9060
Weighted avg	0.95	0.95	0.95	9060

The focus of this article was on data-driven methods to HAR categorization. Although data-driven methods rely on high-quality data for training, there is a scarcity of high-quality, large-scale, and correctly annotated HAR datasets This work makes two contributions: it improved the quality of a freely accessible HAR datasets for data-driven HARs and proposed a new classification approach with improved classification accuracy up to 95 percentage.

5. conclusion

HAR based on powerful sensors embedded in smart phones has received widespread attention in recent years, and the demand for applications in the research area of pervasive computing and mobile computing, surveillance-based security, environment-aware computing, and environment-assisted living has grown rapidly because of its efficient ability to recognize human activities [42]. A smartphone is almost a must-have item for everyone, usually equipped with various sensors such as accelerometer, gyroscope, thermometer, hygrometer, barometer, magnetometer, heart rate sensor, sound sensor, image sensor, and so on. The basic aim of the suggested framework is analyzing raw sensor data for human activities were set of steps are followed sequentially. Since, the accuracy of any analysis is dependent on its selection of features, more efforts are spent on automations or tuning. The proposed framework clean normalizes and prepared raw sensor data for use by CNN's predictions. Finally the columns are labeled to identify features. It is better to understand how the recordings are spread across the different users. The data was encoded for each activity namely Downstairs, Jogging, Sitting, Standing, Upstairs, Walking. As a part of feature selections, HADPL is interested in three main parameters namely, TIME_PERIODS, STEP_DISTANCES AND LABELS. The CNNs are fed with Number of time periods, Number of sensors and Number of classes. In this paper, recognition accuracy of up to 89% on various everyday activities using a accelerometer data was obtained. A new set of features was taken into account and CNNs used in as a classifier with evaluations of the performances. The proposed HADPL is an approach for predicting HARs from dataset values. Thus, this work's suggested HADPL was implemented and demonstrated with results and can be implemented on HAR dataset. The model has problems for clearly identifying upstairs and downstairs activities. This could definitely improve, maybe with further hyper parameter tuning and especially with a modified neural network design. Although this data can most naturally be used for activity recognition, it can also be used to build behavioral biometric models since each sensor reading is associated with specific subjects.

Data Availability Statement

The experiments have been carried out using sensor-based HAR datasets such as WISDM[44] which are open for use in the research work.

References

[1] Y. Liu, L. Nie, L. Liu, D.S. Rosenblum, From action to activity: sensor-based activity recognition, Neurocomputing 181 (2016) 108–115.

[2] A. Bux, P. Angelov, Z. Habib, Vision based human activity recognition: a review, in: Advances in Computational Intelligence Systems, Springer, 2017, pp. 341–371.

[3] B. Jagadeesh, C.M. Patil, Video based human activity detection, recognition and classification of actions using SVM, Trans. Mach. Learn. Artif.Intell. 6 (6) (2019) 22.

[4] J.A. Muñoz-Cristóbal, M.J. Rodríguez-Triana, V. Gallego-Lema, H.F. Arribas-Cubero, J.I. Asensio-Pérez, A. Martínez-Monés, Monitoring for awareness and reflection in ubiquitous learning environments, Int. J. Hum.–Comput.Interact. 34 (2) (2018) 146–165.

[5] L.M. Dang, S.I. Hassan, S. Im, H. Moon, Face image manipulation detection based on a convolutional neural network, Expert Syst. Appl. 129 (2019) 156–168.

[6] F. Foerster, M. Smeja, Joint amplitude and frequency analysis of tremor activity., Electromyogr. Clin. Neurophysiol. 39 (1) (1999) 11–19.

[7] X. Ji, J. Cheng, W. Feng, D. Tao, Skeleton embedded motion body partition for human action recognition using depth sequences, Signal Process. 143 (2018) 56–68.

[8] G. Batchuluun, J.H. Kim, H.G. Hong, J.K. Kang, K.R. Park, Fuzzy system based human behavior recognition by combining behavior prediction and recognition, Expert Syst. Appl. 81 (2017) 108–133.

[9] A. Prati, C. Shan, K.I.-K. Wang, Sensors, vision and networks: from video surveillance to activity recognition and health monitoring, J. Ambient Intell. Smart Environ. 11 (1) (2019) 5–22.

[10] S. Sankar, P. Srinivasan, R. Saravanakumar, Internet of things based ambient assisted living for elderly people health monitoring, Res. J. Pharm. Technol. 11 (9) (2018) 3900–3904.

[11] C. Aviles-Cruz, E. Rodriguez-Martinez, J. Villegas-Cortez, A. Ferreyra-Ramirez, Granger-causality: an efficient single user movement recognition using a smartphone accelerometer sensor, Pattern Recognition. Lett. 125 (2019) 576–583.

[12] G. Plasqui, Smart approaches for assessing free-living energy expenditure following identification of types of physical activity, Obes. Rev. 18 (2017) 50– 55.

[13] R. Varatharajan, G. Manogaran, M. Priyan, R. Sundarasekar, Wearable sensor devices for early detection of alzheimer disease using dynamic time warping algorithm, Cluster Comput. 21 (1) (2018) 681–690.

[14] E.E. Cust, A.J. Sweeting, K. Ball, S. Robertson, Machine and deep learning for sport-specific movement recognition: a systematic review of model development and performance, J. Sports Sci. 37 (5) (2019) 568–600.

[15] Z.S. Abdallah, M.M. Gaber, B. Srinivasan, S. Krishnaswamy, Activity recognition with evolving data

streams: areview, ACM Comput. Surv. (CSUR) 51 (4) (2018) 71.

[16] B. Alsinglawi, Q.V. Nguyen, U. Gunawardana, A. Maeder, S.J. Simoff, Rfid systems in healthcare settings and activity of daily living in smart homes: a re-view, E-Health Telecommun. Syst. Networks. 6 (2017) 1–17.

[17] I. Portugal, P. Alencar, D. Cowan, The use of machine learning algorithms in recommender systems: a systematic review, Expert Syst. Appl. 97 (2018) 205–227.

[18] T.N. Nguyen, H. Nguyen-Xuan, J. Lee, A novel data-driven nonlinear solver for solid mechanics using time series forecasting, Finite Elem. Anal. Des. 171 (2020) 103377.

[19] T.-H. Tan, M. Gochoo, S.-C. Huang, Y.-H. Liu, S.-H. Liu, Y.-F. Huang, Multi--resident activity recognition in a smart home using RGB activity image and DCNN, IEEE Sens. J. 18 (23) (2018) 9718–9727.

[20] T. Young, D. Hazarika, S. Poria, E. Cambria, Recent trends in deep learning based natural language processing, IEEE Comput Intell Mag 13 (3) (2018) 55–75.

[21] T.N. Nguyen, S. Lee, H. Nguyen-Xuan, J. Lee, A novel analysis-prediction ap-proach for geometrically nonlinear problems using group method of data handling, Comput. Methods Appl. Mech. Eng. 354 (2019) 506–526.

[22] Figo, D.; Diniz, P.C.; Ferreira, D.R.; Cardoso, J.M. Preprocessing Techniques for Context Recognition from Accelerometer Data. Pers. Ubiquit Comput. 2010, 14, 645–662.

[23] Yang, J.; Nguyen, M.N.; San, P.P.; Li, X.L.; Krishnaswamy, S. Deep Convolutional Neural Networks on Multichannel Time Series for Human Activity Recognition. In Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI-15), Buenos Aires, Argentina, 25–31 July 2015; pp. 3995–4001

[24] Alsheikh, M.A.; Selim, A.; Niyato, D.; Doyle, L.; Lin, S.; Tan, H.P. Deep Activity Recognition Models with Triaxial Accelerometers. In Proceedings of the Workshops at the 30th AAAI Conference on Artificial Intelligence (AAAI-16), Phoenix, AZ, USA, 12–17 February 2016; pp. 8–13.].

[25] Bishop, C. M. (2006). Pattern Recognition and Machine Learning. Secaucus, NJ: Springer.

[26] Quattoni, A., Wang, S., Morency, L. P., Collins, M., and Darrell, T. (2007). Hidden conditional random fields. IEEE Trans. Pattern Anal. Mach. Intell. 29, 1848–1852. doi:10.1109/TPAMI.2007.1124.

[27] Morariu, V. I., and Davis, L. S. (2011). “Multi-agent event recognition in structured scenarios,” in Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Colorado Springs, CO), 3289–3296.

[28] Liu, J., Kuipers, B., and Savarese, S. (2011). “Recognizing human actions by attributes,” in Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Colorado Springs, CO), 3337–3344.

[29] Kulkarni, P., Sharma, G., Zepeda, J., and Chevallier, L. (2014). “Transfer learning via attributes for improved on-the-fly classification,” in Proc. IEEE Winter Conference on Applications of Computer Vision (Steamboat Springs, CO), 220–226.

[30] Farhadi, A., Endres, I., Hoiem, D., and Forsyth, D. A. (2009). “Describing objects by their attributes,” in Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Miami Beach, FL), 1778–1785.

[31] Kwapisz, J.R.; Weiss, G.M.; Moore, S.A. Activity Recognition Using Cell Phone Accelerometers. ACM SigKDD Explor. Newsl. 2011, 12, 74–82.

[32] Abidine, B.M.H.; Fergani, L.; Fergani, B.; Oussalah, M. The Joint Use of Sequence Features Combination and Modified Weighted SVM for Improving Daily Activity Recognition. Pattern Anal. Appl. 2018, 21, 119–138.

[33] Yeh, R.A.; Chen, C.; Yian Lim, T.; Schwing, A.G.; Hasegawa-Johnson, M.; Do, M.N. Semantic Image Inpainting with Deep Generative Models. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2017), Honolulu, HI, USA, 21–26 July 2017; pp. 5485–5493..

[34] Bukhari, D.; Wang, Y.; Wang, H. Multilingual Convolutional, Long Short-term Memory, Deep Neural Networks for Low Resource Speech Recognition. Proc. Comput. Sci. 2017, 107, 842–847.

[35] Sannino, G.; De Pietro, G. A Deep Learning Approach for ECG-based Heartbeat Classification for Arrhythmia Detection. Future Gener. Comput. Syst. 2018, 86, 446-455.

[36] Li, H.; Trocan, M. Deep Learning of Smartphone Sensor Data for Personal Health Assistance. Microelectron. J. 2018.

[37] Wang, A.; Chen, G.; Shang, C.; Zhang, M.; Liu, L. Human Activity Recognition in a Smart Home Environment with Stacked Denoising Autoencoders. In Proceedings of the International Conference on Web-Age Information Management (WAIM 2016), Nanchang, China, 3–5 June 2016; pp. 29–40.

[38] Ronao, C.A.; Cho, S.B. Human Activity Recognition with Smartphone Sensors Using Deep Learning Neural Networks. Expert Syst. Appl. 2016, 59, 235–244.

[39] Ha, S.; Yun, J.M.; Choi, S. Multi-modal Convolutional Neural Networks for Activity Recognition. In Proceedings of the 2015 IEEE International Conference on Systems, Man, and Cybernetics (SMC 2015), Kowloon, China, 9–12 October 2015; pp. 3017–3022.

[40] Chen, Y.; Zhong, K.; Zhang, J.; Sun, Q.; Zhao, X. Lstm Networks for Mobile Human Activity Recognition. In Proceedings of the 2016 International Conference on Artificial Intelligence: Technologies and Applications (ICAITA 2016), Bangkok, Thailand, 24–25 January 2016; pp. 50–53.

[41] Gupta, A., and Davis, L. S. (2007). "Objects in action: an approach for combining action understanding and object perception," in Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Minneapolis, MN), 1–8.

[42] Chen, L.; Hoey, J.; Nugent, C.D.; Cook, D.J.; Yu, Z. Sensor-based Activity Recognition. IEEE Trans. Syst. Man Cybern. C Appl. Rev. 2012, 42, 790–808.

[43] Kiruthiga Devi M, Dr. K.P. Yadav, Artificial Intelligence, Machine Learning And Cognitive Computing With Profound Learning Technique, Cosmos Journal of Engineering & Technology,A Refereed Research Journal, Vol 8 / No 2 / Jul-Dec 2018.

[44] Kwapisz, J.R.; Weiss, G.M.; Moore, S.A. Activity recognition using cell phone accelerometers. ACM SigKDD Explor. Newsl. 2011, 12, 74–82.

[45] Prabhu P., Anbazhagan N. (2013) FI-FCM Algorithm for Business Intelligence. In: Prasath R., Kathirvalavakumar T. (eds) Mining Intelligence and Knowledge Exploration. Lecture Notes in Computer Science, vol 8284. Springer, Cham, pp 518-528.

[46] Paulraj P., Neelamegam A. (2014) Improving Business Intelligence Based on Frequent Itemsets Using k-Means Clustering Algorithm. In: Meghanathan N., Nagamalai D., Rajasekaran S. (eds) Networks and Communications (NetCom2013). Lecture Notes in Electrical Engineering, vol 284. Springer, Cham, DOI : 10.1007/978-3-319-03692-2_19, pp 243-254.