Multimedia Tools and Applications (2022) 81:34633–34652
https://doi.org/10.1007/s11042-021-11313-0

1174: FUTURISTIC TRENDS AND INNOVATIONS IN MULTIMEDIA SYSTEMS USING BIG DATA, IOT AND CLOUD TECHNOLOGIES (FTIMS)

Human activity recognition by combining external features with accelerometer sensor data using deep learning network model

Neeraj Varshney[1] · Brijesh Bakariya[2] · Alok Kumar Singh Kushwaha[3] · Manish Khare[4]

Received: 9 September 2020 / Revised: 10 July 2021 / Accepted: 20 July 2021 /
Published online: 2 January 2022

Abstract

Various Human Activities are classified through time-series data generated by the sensors of wearable devices. Many real-time scenarios such as Healthcare Surveillance, Smart Cities and Intelligent surveillance etc. are based upon Human Activity Recognition. Despite the popularity of local features-based approaches and machine learning approaches, it fails to capture adequate temporal information. In this paper, the deep convolutional neural model has been proposed by combining external features, i.e. orientation invariant ($\|v\|$) and consecutive point trajectory information ($\|\Delta v\|$) with tri-axis data of the accelerometer. The proposed external features based approach experimented on three different deep learning architecture, namely Long-Short Term Memory (LSTM), Convolutional Neural Networks (CNN) and Convolution Long-Short Term Memory (ConvLSTM). Accuracy of the algorithms radically improve with the additional input feature $\|v\|$ and $\|\Delta v\|$ along with tri-axis data of accelerometer. The results show that the performance of all three LSTM, CNN and ConvLSTM models is better to compare with the state of art methods on WISDOM dataset and Activity dataset also the performance of ConvLSTM is 98.41% for WISDOM dataset and 98.04 for activity dataset, which is higher than that of CNN and LSTM model used in this paper.

Keywords Convolutional neural network · Wearable device Sensor · Human activity recognition · Deep learning · Time-series data

1 Introduction

Representation of human activity can be categories through body motion and gesture [3], and determine the predict states of action. Time series data created by different sensors of wearable gadgets are capable to give data about the movement as well as posture [2].

✉ Neeraj Varshney
neeraj.varshney@gla.ac.in

Extended author information available on the last page of the article

Majorly frequency-based features and Statistical features are used to identify the activities from time-series data. Statistical features offer less computational time and complexity as compare to the frequency-based feature [6]. Data received from the accelerometer is in the form of tri-axis (x, y, z) and effective for recognition of activity. The accelerometer can be easily integrated with wearable devices and mobile devices. Tri-axis data of accelerometer is capable to capture the human action in the time domain. Healthcare is one of the major sectors, utilizing sensor data. Many of the healthcare based applications are using the wearable sensors data for prescribing the health recommendation [11]. The present generation of handy mobile devices like a fitness band, smartphone, smartwatches includes a verity of sensors like accelerometer, gyroscope, magnetometer, etc. and capable of analyzing human activity and behavior analysis. The time series is segmented using a sliding window of a fixed length and split each time series into equal segments. The task of action/activity recognition involves the identification of activity from time-series data, which acts for a certain duration by a human. In early days' handcraft features and machine learning approaches were used for recognition of activity but deep learning-based approaches are in the great demand among the researchers. Deep learning-based approaches automatically learn the required feature representation directly from the data. Other than high accuracy and decent generalization, deep learning models trained in an end-to-end fashion [15].

Major challenges to identify activity from sensor data is to differentiate the similar activities like walking and jogging, up-stairs and down-stairs. The traditional approach extracted the features and fed into the classifier but accuracy compromise if these features missed the representation. This is a tedious and laborious task that employs a practical method that is not guaranteed to be optimal thus deep learning based approach for tri-axis data of the accelerometer is talk over in this paper. Using deep learning approach featuring engineering process can be avoided. Model itself correlate the features and combine them. Three different deep learning algorithms, namely CNN, LSTM and ConvLSTM are discussed in this paper, which combine orientation invariant and consecutive point trajectory information as additional input along with the tri-axis data of accelerometer. Accuracy of the algorithms radically improve with these additional features. CNN is very powerful without memory which is required while processing the sequential data like time series so LSTM is better option in this case where it keep past input [33].

The rest of this paper is organized as follows: Sect. 2 present the motivation behind, in Sect. 3, we briefly review the literature based on time series data. Section 4, describe accelerometer data. Section 5, describe the proposed model. Experiment and result analysis discuss in Sect. 6 and finally, we conclude in Sect. 7.

2 Motivation

Activity recognition using sensor data is a high demanding research area because of its wide utility. We all are aware of the case of Asha Sahani in Mumbai whose Skelton was found in her flat. Many of such areas like healthcare system, monitoring the old people, monitoring the daily activities, sensor based mechanism is very useful however identification of activity would be a greater challenge for the automated system. An efficient mechanism with accurate result is the need of an hour.

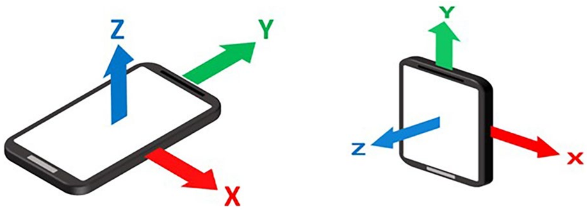
3 Previous work

The task of activity recognition involves the identification of activity from time-series data that act for a certain duration. Deep learning approaches attract the researchers and many deep learning architectures have been proposed to exploit the time series data. Lee et al. [14] proposed a one-dimensional convolutional neural network in which they calculate the magnitude of the tri-axis data of accelerometer captured by their research team. They recorded three activities walking, running, and still. Shoaib et al. [25] proposed a fourth-dimension i.e. magnitude of tri-axis (x, y, z) data. They proved this four-dimension data as an input and used various classifiers to classify the data. Robert et al. [29] proposed a Multi-layer perceptron solution implemented on a mobile phone. They extracted five features from the data and input to a multilayer perceptron. Cesar et al. [28] proposed a hierarchical neural classifier for activity classification. They extracted the time-domain feature from the tri-axis data. Transferring knowledge among the models having different probability distributions was proposed by Abdullah et al. [9]. Random forest, decision tree, and transfer boost algorithms were used for the evaluation. Alvina et al. [1] built an application to trace the physical activates of the user. They collect dataset by their own and implemented the classifier in mobile applications. A convolution neural network-based approach was proposed by Charissa et al. [22]. They experimented on various combinations of feature map and the number of convolution layers. Jinyong [18] proposed a transfer learning model based on the convolution network. He trained the model on the WISDM dataset and utilize this learning to classify the activities of the UCI HAR dataset. Wenchao et al. [8] proposed an idea of transform to time series data into an activity image of time–frequency-spectral. They apply a convolution neural network to learn features and classify the activities. Wanmin et al. [32] combine accelerometer and gyroscope data to extract features and classify activities. Charissa [23] used the accelerometer and gyroscope tri-axial sensor data to perform 6-axes, 1D convolution for constructing a convolution neural network. A hybrid feature selection method was proposed by Wang et al. [31]. This method combined the traditional feature selection method filter and wrapper. The experimental results showed balances between recognition efficiency and accuracy. Yuwen et al. [4] proposed LSTM based feature extraction from the tri-axis data of accelerometer.

4 Accelerometer data

For this research work, we are using the accelerometer dataset used in [13] and the WISDM dataset [12]. In both the dataset, data was collected through an accelerometer. The accelerometer is used to measure the acceleration of an object. It is measured in meter per second square and sampling frequency in Hz. Typical accelerometers are made up of two or three axis-vector components. Most smartphones typically make use of three-axis models. These devices are very sensitive intended to measure even tiny variations in acceleration. The accelerometer in the wearable device delivers the XYZ coordinate values, which is then used to identify the situation and the acceleration of the device. The XYZ coordinate indicates the direction and position of the device at which acceleration occurred (Fig. 1).

Fig. 1 Tri axis of accelerometer [5]



5 Proposed methodology

Three different convolutions deep learning network model are proposed. These models use five dimensions, x, y, z, orientation invariant feature and consecutive point trajectory information as input. The result shows all three models work well as compare to the state of art methods. Data received from the accelerometer is delivered to the XYZ coordinate values. Tri-axis data is preprocessed and extract the additional external feature vector. Detail discussion about feature extraction is given below is Sect. 5.1.

5.1 Feature extraction

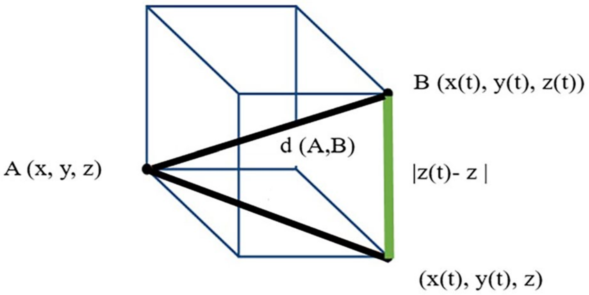
The Accelerometer data provides the XYZ coordinate values, which are used to measure the position and the acceleration of the device [5]. We add two more dimensions in this three-dimension data i.e. orientation invariant feature of the sensor and the consecutive point trajectory information between the two sensor positions. Orientation invariant feature minimizes the effect of change in orientation [25]. The motion between two consecutive points trajectory can be identify through displacement vector (Fig. 2), it maps the translation between two consecutive movements and boosts the learning process.

The orientation invariant feature can be achieved through the magnitude of the vector. The magnitude of a vector (‖v‖) is the length of the line segment that defines it. Magnitude can be obtained by:

$$||v|| = \sqrt{x^2 + y^2 + z^2} \tag{1}$$

consecutive point trajectory information can be determined by the displacement vector. The displacement vector describes the motion in the space. To establish a coordinate system and a convention for the axes, the coordinates x, y, and z to locate a particle at point P

Fig. 2 Displacement vector



(x, y, z) in three dimensions. If the particle is moving, the variables x, y, and z are functions of time (t), therefore.

$$x = x(t),\ y = y(t),\ z = z(t)$$

The position vector from the origin to the point B is $\overrightarrow{v(t)}$. In the unit vector notation $\overrightarrow{v(t)}$ will be-

$$\overrightarrow{v(t)} = x(t)\tilde{i} + y(t)\hat{j} + z(t) \tag{2}$$

With definition of the position of a particle in three-dimensional space, we the three-dimensional displacement can formulate. Figure shows a particle at time t_1 located at position A with position vector $\vec{v}\ 1 \rightarrow (t_1)$. At a later time t_2, the particle is located at B with position vector $\vec{v}\ 2 \rightarrow (t_2)$.

The displacement vector Δv is found by

$$\overrightarrow{\Delta v} = \vec{v}\left(t_1\right) - \vec{v}\left(t_2\right) \tag{3}$$

The magnitude of the displacement can be calculated as

$$||\Delta v|| = \sqrt{(\mathrm{x} - \mathrm{x(t)})^2 + (\mathrm{y} - \mathrm{y(t)})^2 + (\mathrm{z} - \mathrm{z(t)})^2} \tag{4}$$

So now the input data is of five-dimension x, y, z, magnitude, and the magnitude of the displacement vector between two consecutive sensor values (Figs. 3 and 4).

Window size (frame size) plays an important role sometimes not choosing the correct window size may give a penalty and reduce the accuracy of the data. Dataset [13] is captured with the frequency rate 10 Hz and WISDM dataset captured with a frequency

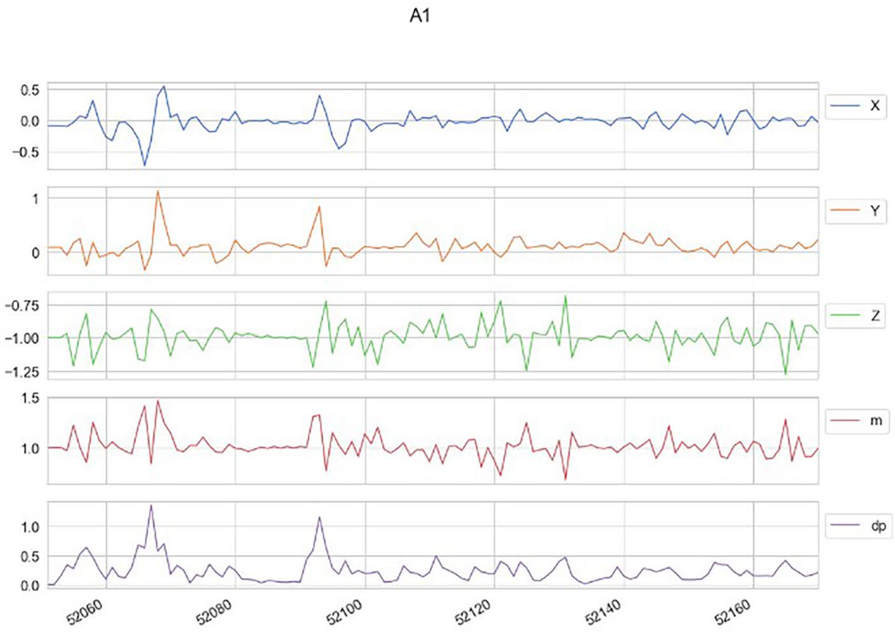


Fig. 3 Signal representation of dataset [13] with addition features magnitude and displacement vector

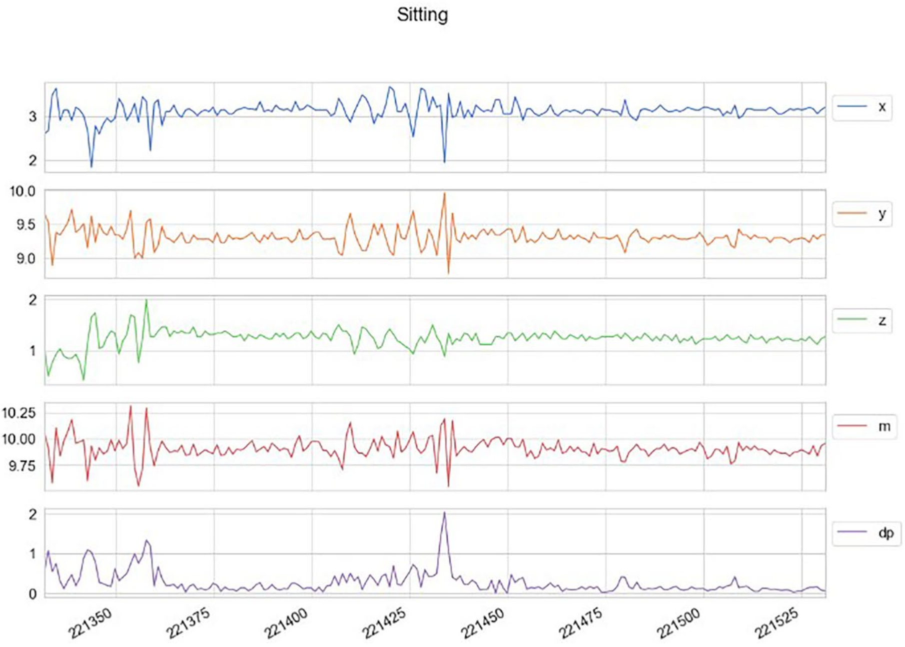


Fig. 4 Signal representation of WISDOM dataset with addition features magnitude and displacement vector

rate of 20 Hz, therefore we choose window size of 4 s during the experiment and same for the WISDM dataset [12] as well, also we choose overlapping of one fourth the size of the window.

5.2 CNN model

In this Convolution deep learning model, the data received from the accelerometer are first divided into time-series segments which are the same size as the window frame. The size of the window frame we selected was 4 s. A hop size with 25% overlapping was considered for the experiment. Initially, the tri-axis data of accelerometer is preprocessed and finds the value of magnitude $\|v\|$ and the Consecutive point trajectory information $\|\Delta v\|$. Now the whole five-dimensional data provide as an input to the convolution neural network. A batch of segments, each segment sized 1 × 20, was stored in a 5D tensor. These segments are divided into 80% training data and 20% testing data. With these training data, the deep model is trained with a learning rate of 0.001. The trained model was tested with the 20% test data in each epoch. A checkpoint was created in which the model was saved if the performance improved with each epoch in the validation loss. The CNN model contains a total of five convolutional layers with dropout layers for regularization with a probability constant of 0.2 and a dense layer.

Each convolution layer of model X is trained in a similar way and the individual unit is shown as Fig. 5. In the lth layer's training there is an input $x^{[l-1]}$ with channel c_{l-1}, i.e. c_{l-1} feature maps in 2-D arrays of x_1^{l-1} …………. x_{cl-1}^{l-1}. The representation of m^th^ feature map is [10]

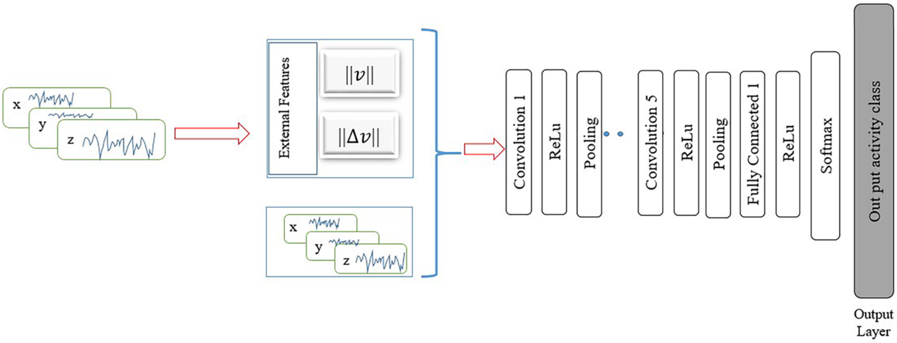


Fig. 5 Architecture CNN model

$$Z_m^{[l]} = \sum_{i=1}^{c_{l-1}} X_i^{l-1} * w_{im}^{[l]} + b_m^i \tag{5}$$

where m = 1, 2,….cl, $w_{im}^{[l]}$ denote the mth kernel, b_m^i represent the bias of lth layer and * represent 2-D convolution operation.

Among the nonlinear activation function ReLU activation function is used in the model to transform output data [17] and softmax for the classification of the activity class. At the end of the training, we had the latest optimized model that shows high performance in the dataset, which is then used for testing against the testing data.

5.3 LSTM model

The initial phase of data preprocessing is same as we did in the CNN model. A batch of segments, each segment was stored in a 5D tensor. These segments are divided into 80% training data and 20% testing data. With these training data, the deep model is trained with a learning rate of 0.001. The trained model was tested with the 20% test data in each epoch.

The LSTM memory is also known as “gated” cell, where the word gate means the ability to make the decision of avoiding or maintaining the memory information. Important features from the input are preserved by an LSTM model over a long period of time. An LSTM has three of these gates, to protect and control the cell state [7].

A deep network was built by stacking multiple layers of LSTM memory units with 2 fully connected layers (Fig. 6). Sparse categorical crossentropy is applied to measure the loss. The major formulation used in LSTM obtained from [7] are mention in Eq. 6.

$$f_t = \sigma\left(w_f * \left[h_{t-1,}X_t\right] + b_f\right)$$

$$I_t = \sigma(w_I * \left[h_{t-1,}X_t\right] + b_I)$$

$$C_t = tanh(W_C * \left[h_{t-1,}X_t\right] + b_C)$$

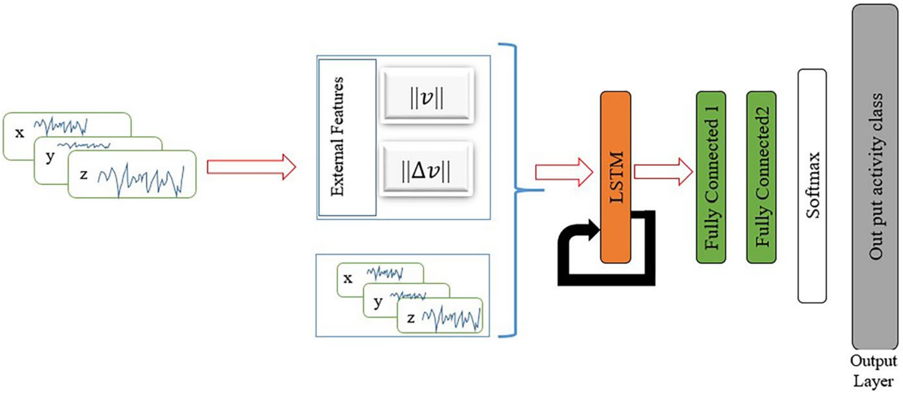


Fig. 6 Architecture LSTM model

$$C_t = f_t * C_{t-1} + I_t * C_t$$

$$O_t = \sigma(w_O * [h_{t-1,} X_t] + b_O)$$

$$h_t = O_t * \tanh(C_t)$$

where f_t, I_t, O_t are forget, new cell and output gates. X_t is input and h_t is hidden unit.

5.4 ConvLSTM model

The tri-axis data of accelerometer is preprocessed and finds the value of magnitude ||*v*|| and the magnitude of displacement vector ||Δ*v*||. Now the whole five-dimensional data provide as an input to the convolution neural network. A batch of segments, each segment was stored in a 5D tensor. ConvLSTM model was first introduced by Satya et al. [26]. The above picture shown in Fig. 7 describes how a general ConvLSTM model work. The CovnLSTM has nice properties for activity recognition as convolution operator may handle local spatial features the LSTM part manage temporal correlation in the sensor data. As discuss in the Sect. 5.3, An LSTM has three of these gates, to protect and control the cell state. The model consists

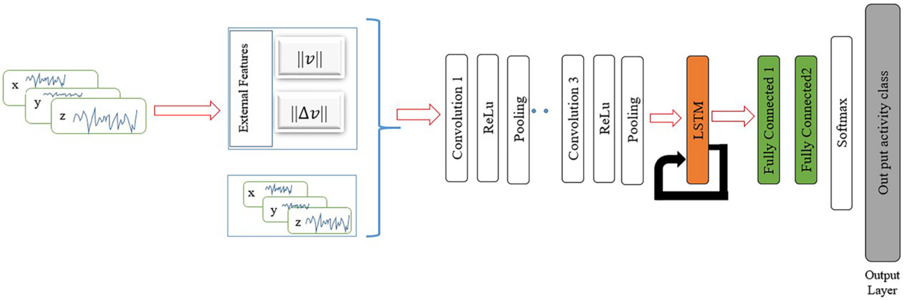


Fig. 7 Architecture ConvLSTM model

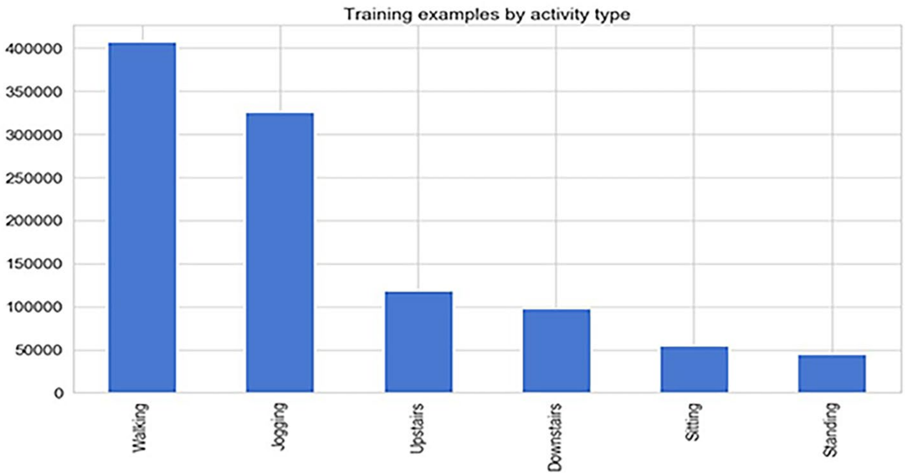


Fig. 8 Activity Dataset [13] Activity type and location information

three convolution layer and the output of third convolution layer flatten and input to the LSTM (Fig. 7). Timedistribution function is used to manage the dimensions of the input layer.

5.5 Dataset

We test our model in two datasets. The first dataset is used in [13] and the author makes it publically available (Fig. 8). The dataset was collected from two volunteers who performed activities using a smartwatch attached to the wrist of their dominant hand for four weeks. This data was captured with a sampling rate of 10 Hz. The dataset contains activities majorly at three locations office, kitchen, and outdoor. It consists of 11 different activates like office work, reading, writing, cooking, walking, running, etc. Another dataset used for the experiment is the WISDOM [12] dataset, which is publically available and can be download from http://www.cis.fordham.edu/wisdm/. WISDM dataset contains the tri-axis accelerometer data along with the information of time and user performed (Fig. 9). The dataset contains a total of 1,098,207 examples with six classes named walking, jogging, upstairs, downstairs, sitting, and standing. WISDM dataset is highly imbalance therefore

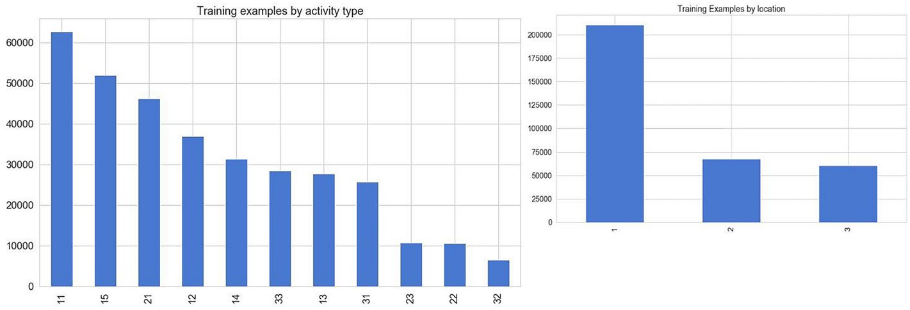


Fig. 9 WISDOM dataset Activity type

Table 1 Activity recognition accuracy over the dataset-1

Activity	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11
Description	Office Work	Reading	Writing	Taking a rest	Playing game	Eating	Cooking	Washing dishes	Walking	Running	Taking a transport
CNN	93%	93%	95%	95%	97%	84%	95%	99%	82%	78%	93%
LSTM	93%	97%	97%	98%	99%	81%	95%	100%	87%	89%	100%
ConvLSTM (without location)	96%	99%	96%	99%	99%	88%	98%	100%	80%	95%	100%
ConvLSTM (with location)	97%	99%	99%	99%	99%	100%	99%	100%	92%	84%	100%

Table 2 Result of activity dataset

Method	Activity Dataset [13] (Accuracy %)
CNN	93.04
LSTM	94.63
CNN LSTM	96.38
CNN LSTM (With location)	98.04

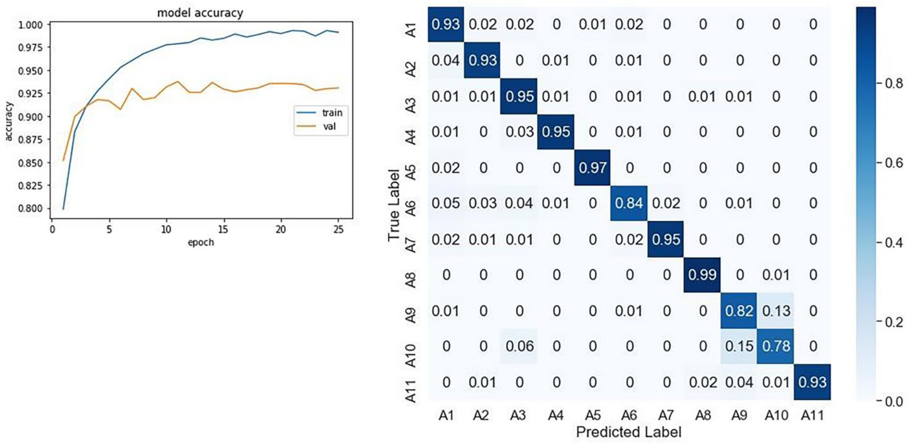
during the preprocessing we balanced the dataset. WISDM dataset was captured with a sample rate of 20 Hz.

6 Experiment and result analysis

This section discussed the result of the proposed model. We use two datasets to compared with the-state-of-art methods for human activity recognition: activity dataset [13] and WISDM dataset [12]. We also compare our method with state of art methods [4, 9, 12, 13, 28, 29]. The proposed model is trained and tested on these two datasets for activity recognition. Adding features orientation invariant and consecutive point trajectory information as additional input along with the tri-axis data of accelerometer provide strong support to the proposed approach compare to the other methods.

Tables 1 and 2 shown the efficiency of proposed models on activity dataset. Proposed model 5D-CNN achieve 93.04% accuracy on activity dataset, LSTM model achieve 94.63% accuracy, LSTM without information of the location achieve 96.38% and 5D-ConvLSTM model with location achieve.

98.04% validation accuracy. Figures 10, 11, 12 and 13 show the accuracy graph and confusion matrix of the CNN, LSTM, ConvLSTM and ConvLSTM with location model respectively.



	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11
A1	0.93	0.02	0.02	0	0.01	0.02	0	0	0	0	0
A2	0.04	0.93	0	0.01	0	0.01	0	0	0	0	0
A3	0.01	0.01	0.95	0.01	0	0.01	0	0.01	0.01	0	0
A4	0.01	0	0.03	0.95	0	0.01	0	0	0	0	0
A5	0.02	0	0	0	0.97	0	0	0	0	0	0
A6	0.05	0.03	0.04	0.01	0	0.84	0.02	0	0.01	0	0
A7	0.02	0.01	0.01	0	0	0.02	0.95	0	0	0	0
A8	0	0	0	0	0	0	0	0.99	0	0.01	0
A9	0.01	0	0	0	0	0.01	0	0	0.82	0.13	0
A10	0	0	0.06	0	0	0	0	0	0.15	0.78	0
A11	0	0.01	0	0	0	0	0	0.02	0.04	0.01	0.93

Fig. 10 Accuracy chart and Confusion matrix of 5D- CNN Model for Activity Dataset

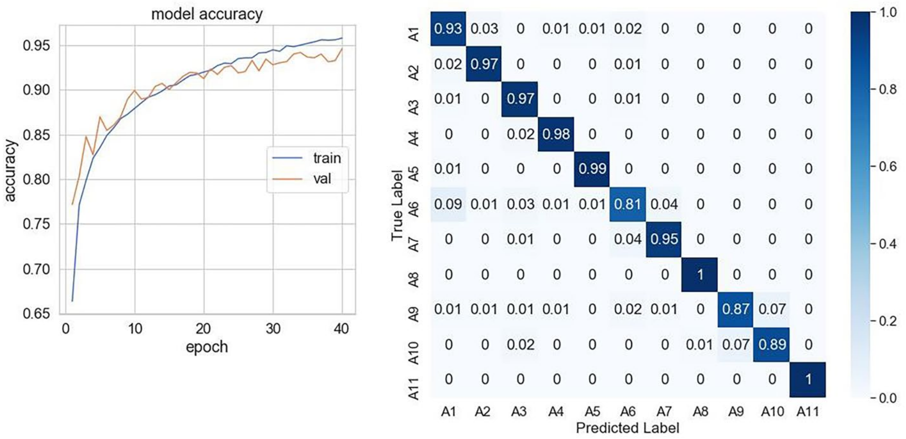


Fig. 11 Accuracy chart and Confusion matrix of LSTM Model for Activity Dataset

All four models are tests with different hyperparameters like number of epochs, number of hidden layers, learning rate, loss function, optimizer, activation function etc. Although we keep learning rate and loss function function fixed for all models. Sparse categorical crossentropy loss function is used for for all models and keep learning rate 0.001.Various possible combinations of epoch, hidden layers and optimizer analyze during the experiment.

Table 3 represents the comparison of proposed models with the approach used in [13]. we find that accuracy is improved using our proposed model and believed that it can be further improved by using more fusion strategy. Table 3 shows that CNN model is ~ 3% better as compared to [13] and LSTM is ~ 4% better whereas ConvLSTM model batter ~ 6% from the state of art method. We also analyze ConvLSTM model with adding additional input location and found that proposed ConvLSTM model is ~ 3% batter with location input.

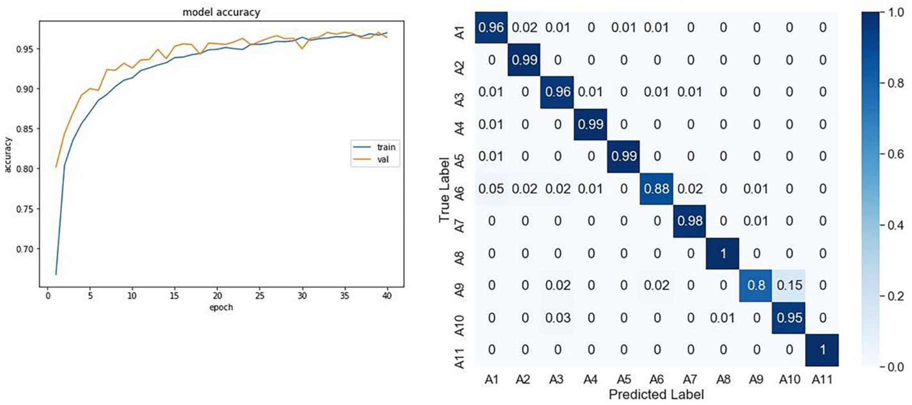
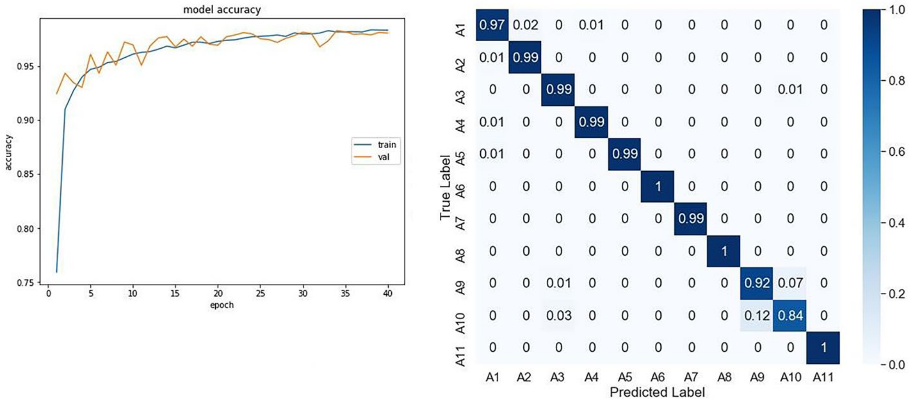


Fig. 12 Accuracy chart and Confusion matrix of ConvLSTM Model for Activity Dataset



	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11
A1	0.97	0.02	0	0.01	0	0	0	0	0	0	0
A2	0.01	0.99	0	0	0	0	0	0	0	0	0
A3	0	0	0.99	0	0	0	0	0	0	0.01	0
A4	0.01	0	0	0.99	0	0	0	0	0	0	0
A5	0.01	0	0	0	0.99	0	0	0	0	0	0
A6	0	0	0	0	0	1	0	0	0	0	0
A7	0	0	0	0	0	0	0.99	0	0	0	0
A8	0	0	0	0	0	0	0	1	0	0	0
A9	0	0	0.01	0	0	0	0	0	0.92	0.07	0
A10	0	0	0.03	0	0	0	0	0	0.12	0.84	0
A11	0	0	0	0	0	0	0	0	0	0	1

Fig. 13 Accuracy chart and Confusion matrix of ConvLSTM Model for Activity Dataset with location input

By analyzing the confusion matrices shown in Figs. 10, 11, 12 and 13, a remarkable performance can be observed in recognition of every activity. Confusion matrices of ConvLSTM model show that walking and writing achieved 100% accuracy whereas while adding the location as input than eating, walking and taking transport achieve 100% accuracy.

Various parameters are tuned for achieving the optimum result. Five convolution layer are used in the CNN model with a learning rate of 0.001 and using Adam optimizer. Whereas the various possible combination of the layers, epoch and optimizer experimented and optimum result received with 40 epochs,128 layers in LSTM and Adam optimizer, whereas 3 convolution layer and 128 LSTM layer are applied for the ConvLSTM model. The Similar combination uses while adding location input.

Tables 4 and 5 shown the evaluation of proposed approach using CNN, LSTM and ConvLSTM model on the WISDM dataset. CNN model achieves 96.52%, LSTM model achieve 97.00% accuracy and.

ConvLSTM model achieved 98.41% accuracy on the WISDM dataset. Figures 14, 15 and 16 show the accuracy graph of all three models for training and testing data under various epochs as well as confusion matrix.

By analyzing the confusion matrices shown in Figs. 14, 15 and 16, a remarkable performance can be observed in the recognition of every activity. Confusion matrices show that standing is classified with 100% accuracy by all three models, all three models discriminate walking and running with grate accuracy. Models are a bit confused with upstairs and

Table 3 Comparison of proposed approach with the state of art method

Method	Activity Dataset (Accuracy %)
Kwon MC et al. [13]	90.00 (without location input)
CNN	93.04 (without location input)
LSTM	94.63 (without location input)
Conv LSTM	96.38 (without location input)
Kwon MC et al. [13]	95.00 (with location input)
Conv LSTM	98.04 (with location input)

Bold entries are proposed models

Table 4 Activity recognition accuracy over the WISDM dataset

Activity	Downstairs	Jogging	Sitting	Standing	Upstairs	Walking
CNN	92%	98%	97%	100%	93%	99%
LSTM	92%	99%	98%	100%	95%	98%
ConvLSTN	96%	99%	99%	100%	95%	100%

Table 5 Result on WISDM Dataset

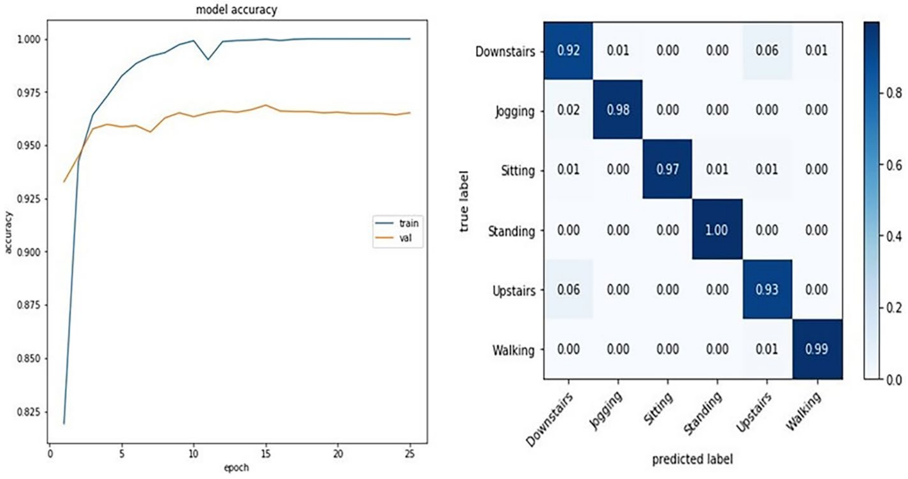
Method	WISDM Dataset [12] (Accuracy %)
CNN	96.52
LSTM	97.00
ConvLSTM	98.41

downstairs. Table 5 shows individual activity recognition accuracy achieved by model-1 and model-2.

Table 6 compare the result of proposed models with the state of art methods. we find that accuracy is improved using our proposed approach. CNN model is ~ 2% better as compared to [30] and ~ 4% from [24] and LSTM model is ~ 3% better then [30] and ~ 4% then [24] whereas ConvLSTM model achieved highest accuracy among the all three proposed models and ~ 4% better as compared to [30].

Table 7 represent two types of errors: a false positive or a false negative. A false positive indicates the false alarm or a type I error and misclassification represent as false negative a type II error. Various performance metrics can be calculated through a confusion matrix.

The metric commonly used are accuracy, precision recall and F-score [21]. Precision indicate the proportion of positive identification was actually correct whereas the recall is



	Downstairs	Jogging	Sitting	Standing	Upstairs	Walking
Downstairs	0.92	0.01	0.00	0.00	0.06	0.01
Jogging	0.02	0.98	0.00	0.00	0.00	0.00
Sitting	0.01	0.00	0.97	0.01	0.01	0.00
Standing	0.00	0.00	0.00	1.00	0.00	0.00
Upstairs	0.06	0.00	0.00	0.00	0.93	0.00
Walking	0.00	0.00	0.00	0.00	0.01	0.99

Fig. 14 Accuracy chart and Confusion matrix of 5D- CNN Model for WISDOM Dataset

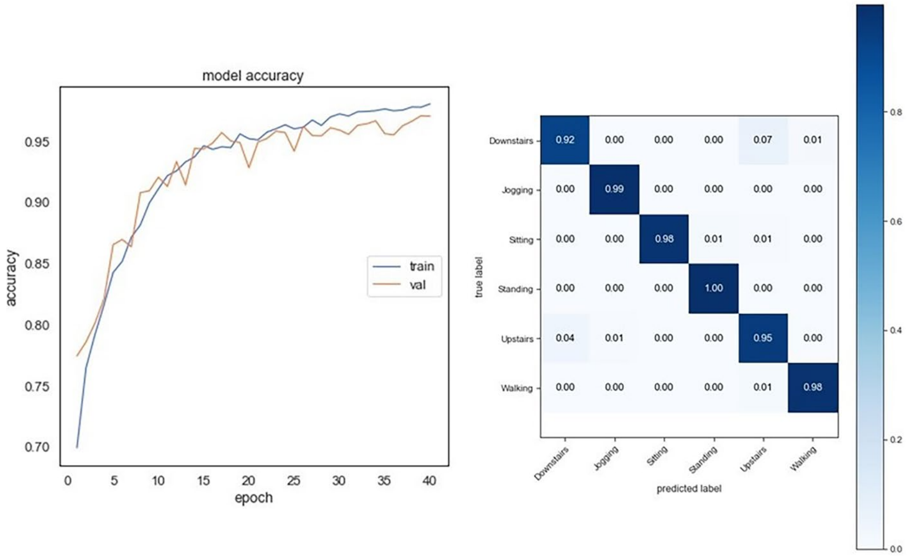


Fig. 15 Accuracy chart and Confusion matrix of LSTM Model for WISDOM dataset

the ratio of correctly detected positive instances to the total number of positive instances. In proposed model we can think of this as representing the correct classification of the activities (Tables 8 and 9).

The F- Score is used to measure the test accuracy and balance the use of precision and recall and can be calculate via harmonic mean of precision and recall. Mathematical

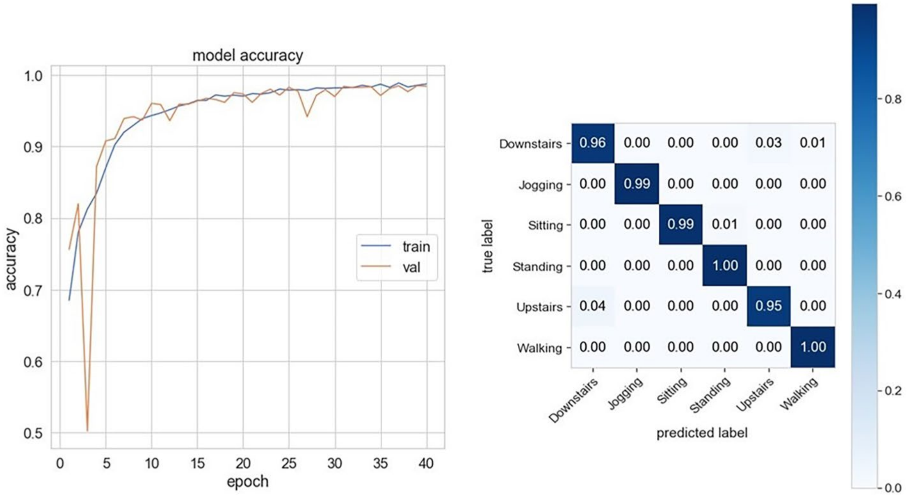


Fig. 16 Accuracy chart and Confusion matrix of ConvLSTM Model

Table 6 Comparison of proposed approach with the state of art method used WISDM dataset

Method	WISDM Data-set (Accuracy %)
Jennifer R et al. [12]	91.7
Schalk et al. [20]	94
Kishor et al.[30]	94.61
Sarbagya et al. [24]	92.22
Yuwen et al. [4]	92.1
Konstantinos et al. [19]	94.18
CNN model	**96.52**
LSTM Model	**97.00**
ConvLSTM Model	**98.41**

Bold entries are proposed models

formulation of precision, recall and F- score and MCC are represented by Eqs. 4,5,6 and 7 respectively [16, 27].

$$\text{Precision} = \frac{\text{TP}}{(\text{TP} + \text{FP})} \tag{6}$$

$$\text{Recall} = \frac{\text{TP}}{(\text{TP} + \text{FN})} \tag{7}$$

$$\text{F} - \text{Score} = \frac{2\text{ X Precision X Recall}}{(\text{Precision} + \text{Recall})} \times 100 \tag{8}$$

$$MCC = \frac{TP\ X\ TN - FP\ X\ FN}{\sqrt{[TP + FP][TP + FN][TN + FP][TN + FN]}} \tag{9}$$

where MCC—> Matthews correlation coefficient
TP—> True positive instances
FP- > False Positive instances
TN- > True Negative Instances

Table 7 Structure of confusion matrix

Confusion Matrix		*Predicted Value*	
		Negative	Positive
Actual Value	Negative	True Negative	False Positive
	Positive	False Negative	True Positive

Table 8 Precision, Recall and F Score and Matthews Correlation Coefficient – Activity Dataset

		A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11	Macro Avg
CNN without location	*Precision*	0.93	0.94	0.95	0.95	0.97	0.84	0.94	0.99	0.84	0.78	0.92	0.91
	Recall	0.85	0.93	0.86	0.96	1.00	0.91	0.96	0.98	0.80	0.83	1.00	0.92
	F-Score	0.89	0.93	0.90	0.96	0.99	0.88	0.95	0.99	0.82	0.81	0.96	0.92
	MCC	0.88	0.93	0.89	0.96	0.99	0.86	0.95	0.98	0.81	0.79	0.96	0.90
LSTM without location	*Precision*	0.93	0.97	0.98	0.98	0.99	0.82	0.95	1.00	0.86	0.89	1.00	0.94
	Recall	0.87	0.95	0.92	0.97	0.98	0.89	0.95	0.99	0.92	0.92	1.00	0.94
	F-Score	0.90	0.96	0.95	0.97	0.99	0.85	0.95	0.99	0.89	0.91	1.00	0.94
	MCC	0.89	0.96	0.94	0.97	0.98	0.84	0.95	0.99	0.89	0.90	1.00	0.94
ConvLSTM without location	*Precision*	0.95	1.00	0.96	0.99	0.99	0.87	0.99	1.00	0.81	0.96	1.00	0.96
	Recall	0.92	0.96	0.92	0.98	0.99	0.95	0.97	0.99	0.97	0.86	1.00	0.96
	F-Score	0.94	0.98	0.94	0.99	0.99	0.91	0.99	1.00	0.88	0.91	1.00	0.96
	MCC	0.93	0.98	0.94	0.98	0.99	0.90	0.98	0.99	0.88	0.90	1.00	0.95
ConvLSTM with location	*Precision*	0.97	0.99	0.99	0.99	0.99	1.00	0.99	1.00	0.80	0.85	1.00	0.96
	Recall	0.97	0.98	0.96	0.99	1.00	1.00	0.97	0.99	0.97	0.91	1.00	0.98
	F-Score	0.97	0.99	0.98	0.99	0.99	1.00	0.98	1.00	0.88	0.88	1.00	0.97
	MCC	0.97	0.98	0.97	0.99	0.99	1.00	0.98	0.99	0.88	0.87	1.00	0.97

Bold entries are proposed models

Table 9 Precision, Recall and F Score and Matthews Correlation Coefficient– WISDOM Dataset

		Downstairs	Jogging	Sitting	Standing	Upstairs	Walking	Macro Avg
CNN	*Precision*	0.91	0.99	1	0.99	0.92	0.99	0.97
	Recall	0.92	0.98	0.97	1	0.93	1	0.97
	F-Score	0.92	0.99	0.99	0.99	0.92	0.92	0.96
	MCC	0.95	0.98	0.98	0.99	0.91	0.98	0.97
LSTM	*Precision*	0.94	0.99	1	0.99	0.92	0.99	0.97
	Recall	0.92	0.99	0.98	1	0.95	0.98	0.97
	F-Score	0.93	0.99	0.99	0.99	0.93	0.98	0.97
	MCC	0.92	0.99	0.98	0.99	0.92	0.98	0.96
ConvLSTM	*Precision*	0.98	0.99	1	0.99	0.95	1	0.99
	Recall	0.97	0.99	0.98	1	0.98	0.99	0.99
	F-Score	0.97	0.99	0.99	0.99	0.97	0.99	0.98
	MCC	0.95	1.00	0.99	0.99	0.95	1.00	0.98

7 Conclusion

This article presented the recognition of human activity from accelerometer data. In this work, two additional features, orientation invariant and consecutive point trajectory information are added along with the tri-axis data. Three deep learning architectures CNN, LSTM and ConvLSTM were used for experiment and tested on two different datasets one used in [13] and another one is publically available dataset i.e. WISDM dataset and achieved remarkable results. Also, compared the proposed models with the result of the state of art methods [4, 12, 19, 20, 24, 30] tested on mentioned datasets and found that the proposed models perform well compared to other methods. Results specified that the proposed network can distinguish similar actions with different velocity. In future some more external features can be combined along with tri-axis data and test more complex activity.

References

1. Anjum A, Ilyas MU (2013) Activity recognition using smartphone sensors. In: 2013 ieee 10th consumer communications and networking conference (ccnc). IEEE, pp 914–919
2. 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 1(118):67–80
3. Chen Y, Xue Y (2015) A deep learning approach to human activity recognition based on single accelerometer. In: Systems, man, and cybernetics (smc). IEEE International Conference, pp 1488–1492
4. Chen Y, Zhong K, Zhang J, Sun Q, Zhao X (2016) LSTM networks for mobile human activity recognition. In: 2016 International Conference on Artificial Intelligence: Technologies and Applications Atlantis Press
5. Cowen A (2018) "Exploring Acceleration with a Sensor App: Science Buddies Blog." Science Buddies, Science Buddies, 9 July 2018. https://www.sciencebuddies.org/google-science-journal-app-tutorial-part5-accelerometer#tryaccelerometer
6. Figo D, Diniz PC, Ferreira DR, Cardoso JM (2010) Preprocessing techniques for context recognition from accelerometer data. Pers Ubiquit Comput 14(7):645–662
7. Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Comput 9(8):1735–1780
8. Jiang W, Yin Z (2015) Human activity recognition using wearable sensors by deep convolutional neural networks. In: proceedings of the 23rd ACM international conference on Multimedia, pp 1307–1310

9. Khan MA, Roy N (2017) TransAct: Transfer learning enabled activity recognition. In: 2017 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops). IEEE, pp 545–550
10. Kim Y (2014) Convolutional neural networks for sentence classification, arXiv preprint
11. Klasnja P, Pratt W (2012) Healthcare in the pocket: Mapping the space of mobile-phone health interventions. J Biomed Inf 45:184–198
12. Kwapisz JR, Weiss GM, Moore SA (2010) Activity recognition using cell phone accelerometers. In: proceedings of the Fourth International Workshop on knowledge discovery from sensor data (at KDD-10). Washington DC
13. Kwon MC, Choi S (2018) Recognition of daily human activity using an artificial neural network and smartwatch. Wirel Commun Mob Comput 2018
14. Lee SM, Yoon SM, Cho H (2017) Human activity recognition from accelerometer data using Convolutional Neural Network. In: 2017 IEEE International conference on big data and smart computing (BigComp). IEEE, pp.131–134
15. Lu J, Zheng X, Sheng M, Jin J, Yu S (2020) Efficient human activity recognition using a single wearable sensor. IEEE Internet Things J 7(11):11137–11146
16. Matthews BW (1975) Comparison of the predicted and observed secondary structure of T4 phage lysozyme. Biochim Biophys Acta Protein Structure 405(2):442–451
17. O'Shea K, Nash R (2015) An introduction to convolutional neural networks. arXiv preprint
18. Pang J (2018) Human Activity Recognition Based on Transfer Learning. Graduate Theses and Dissertations. https://www.digitalcommons.usf.edu/etd/7558
19. Peppas K, Tsolakis AC, Krinidis S, Tzovaras D (2020) Real-Time Physical Activity Recognition on Smart Mobile Devices Using Convolutional Neural Networks. Appl Sci 10(23):8482
20. Pienaar SW, Malekian R (2019) Human activity recognition using LSTM-RNN deep neural network architecture. In: 2019 IEEE 2nd Wireless Africa Conference (WAC). IEEE, pp 1–5
21. Qin Z, Zhang Y, Meng S, Qin Z, Choo KKR (2020) Imaging and fusing time series for wearable sensor-based human activity recognition. Information Fusion 53:80–87
22. Ronao CA, Cho SB (2015) Deep convolutional neural networks for human activity recognition with smartphone sensors. In: International Conference on Neural Information Processing. Springer, Cham, pp 46–53
23. Ronao CA, Cho SB (2016) Human activity recognition with smartphone sensors using deep learning neural networks. Expert Syst Appl 15(59):235–244
24. Shakya SR, Zhang C, Zhou Z (2018) Comparative study of machine learning and deep learning architecture for human activity recognition using accelerometer data. Int J Mach Learn Comput 8:577–582
25. Shoaib M, Bosch S, Incel OD, Scholten H, Havinga PJ (2014) Fusion of smartphone motion sensors for physical activity recognition. Sensors 14(6):10146–10176
26. Singh SP, Lay-Ekuakille A, Gangwar D, Sharma MK, Gupta S (2020) Deep ConvLSTM with self-attention for human activity decoding using wearables. arXiv preprint
27. Sokolova M, Lapalme G (2009) A systematic analysis of performance measures for classification tasks. Inf Process Manage 45(4):427–437
28. Torres-Huitzil C, Alvarez-Landero A (2015) Accelerometer-based human activity recognition in smartphones for healthcare services. In: Mobile Health. Springer, Cham, pp 147–169
29. Voicu RA, Dobre C, Bajenaru L, Ciobanu RI (2019) Human physical activity recognition using smartphone sensors. Sensors 19(3):458
30. Walse KH, Dharaskar RV, Thakare VM (2016) A study of human activity recognition using AdaBoost classifiers on WISDM dataset. The Institute of Integrative Omics and Applied Biotechnology Journal 7(2):68–76
31. Wang A, Chen G, Yang J, Zhao S, Chang CY (2016) A comparative study on human activity recognition using inertial sensors in a smartphone. IEEE Sens J 16(11):4566–4578
32. Wu W, Dasgupta S, Ramirez EE, Peterson C, Norman GJ (2012) Classification Accuracies of Physical Activities Using Smartphone Motion Sensors. J Med Internet 14:e130
33. Xingjian SH, Chen Z, Wang H, Yeung DY, Wong WK, Woo WC (2015) Convolutional LSTM network: A machine learning approach for precipitation nowcasting. In: Advances in neural information processing system, pp 802–810

Authors and Affiliations

Neeraj Varshney[1] · Brijesh Bakariya[2] · Alok Kumar Singh Kushwaha[3] · Manish Khare[4]

Brijesh Bakariya
dr.brijeshbakariya@ptu.ac.in

Alok Kumar Singh Kushwaha
alokkumarsingh.jk@gmail.com

Manish Khare
mkharejk@gmail.com

[1] I K Gujral Punjab Technical University, Kapurthala, Ibban, India

[2] I K Gujral Punjab Technical University, Hoshiarpur Campus, India

[3] Guru Ghasidas, Vishwavidyalaya, Bilaspur, India

[4] Dhirubhai Ambani Institute of Information and Communication Technology, Gandhinagar, India