# Hybrid Deep Learning-Based Human Activity Recognition (HAR) Using Wearable Sensors: An Edge Computing Approach

**Neha Gaud, Maya Rathore, and Ugrasen Suman**

**Abstract** Due to the growth of Internet of Things (IoT) and advanced sensing based technologies have enabled the development of the miniature-based system. In recent years, the use of wearable and mobile sensors for Human Walking Gesture Recognition has become more popular in various applications, including health care, surveillance, robotics, and industry. The recent growth of edge computing technology for industry 4.0 has provided the opportunity to design the low power and less computationally expensive devices. The edge computing devices cannot support heavy computation and provide great efficiency by reducing the network size and communication latency. Deep learning algorithms have recently demonstrated high performance in HAR. However, the deep learning (DL) models require very high computation systems, which make them ineffective when used on edge devices. In this research, a hybrid deep learning-based model is trained to recognize the various gestures. Three deep learning-based models, namely one-Dimensional Convolutional Neural Network (1D-CNN), Convolutional Neural Network–Long Short-Term Memory (CNN-LSTM), and CNN-Gated Recurrent Unit (CNN-GRU), are designed to test the various human mobility gestures. The WISDM, PAMAP2, and UCI-HAR benchmark datasets were used to assess these models. Among the three datasets, the best accuracies of the models are 99.89%, 97.28%, and 96.78%, respectively, achieved for CNN-LSTM hybrid model. In future, the work can be extended to design an end-to-end edge computing application using Arduino Nano 33 BLE Sensing microcontroller board. The compressed deep learning model will be fused on the Arduino Nano board to recognize various human motion gestures. The research demonstrates the classification of various HAR gestures using hybrid deep learning models.

N. Gaud · U. Suman
School of Computer Science and Information Technology, DAVV, Indore, M.P, India
e-mail: usuman.scs@dauniv.ac.in

M. Rathore (✉)
Christian Eminent College, Indore, M.P, India



A. Swaroop et al. (eds.), *Proceedings of Data Analytics and Management*,
Lecture Notes in Networks and Systems 785,
https://doi.org/10.1007/978-981-99-6544-1_30

**Keywords** Deep learning · Assistive technology · Internet of heath things (IoHT) · Wearable technology · Edge computing

# 1 Introduction

## *1.1 Research Motivation*

HAR includes several techniques, such as wearable, i.e., computer vision and non-wearable, i.e., sensor-based, which can be further segmented into object-tagged, wearable, dense sensing, etc. Before going any further, it should be noted that HAR systems having various inheritance design challenges, namely selection of different types of sensor, a set of criteria for data collecting, the performance of recognition, the amount of energy used, processing power, and adaptability. It is crucial to create a HAR model which is effective and portable while keeping all these factors in mind. A network for mobile-based human activity recognition has been described which uses data from triaxial accelerometers and a long-short memory technique [1]. Current technology relies on Internet connectivity and cloud services to recognize activity patterns using non-parametric ML models in devices like smart watches.

## *1.2 Recent Advancement*

DL-based approaches have recently gained popularity for recognizing human activity because they can employ representation learning techniques which would automatically generate the best features from sensor-generated original data without the need for human involvement and can find hidden patterns in data [2, 8]. The application of edge computing frameworks to perform HAR models at the network's end is still in its initial stages [12]. Recent years have seen the emergence of edge computing as a fresh framework which can shorten Internet connection lag times by relocating processing power from far cloud servers to data sources. It is logical to transfer the design of cloud-based IoT apps to edge-based ones. In this study, we investigate model compression, the building of neural network-based models; however, in future, the implementation will be done on Arduino BLE Nano sensing board.

## *1.3 Author's Contribution and Novelty*

This research presents hybrid DL models for gesture recognition. The following are the paper's contributions:

- The hybrid deep learning-based models (CNN-1D, CNN-GRU, and CNN-LSTM) are designed for HAR gesture recognition.
- The spatial and temporal features of datasets are used to design the above models.
- Three publicly available benchmark datasets WISDM, PAMAP2, and UCI-HAR are considered to design and test the models.
- Finally, the model which has provided the highest accuracy, i.e., CNN-LSTM is picked.
- The confusion matrix is prepared for all three models over different datasets.

One major objective is to design devices with low power consumption while protecting patient data privacy. This method can also be applied for the evaluation of analysis of various walking techniques, design of prosthesis, and to the rehabilitation of Parkinson's patients and elderly subjects.

### *1.4 Organization of the Paper*

The entire paper is divided into five sections. The first section is introduction, which provides the research motivations, author's contribution, and background information. The second section provides a brief survey of recent state-of-the-art literature. The third section is methodology in which the approach goes over the hardware and software requirements, algorithms, and specific flow tasks for recognizing human gestures. The verification of the model is covered in the fourth section, which includes experiment results and analysis. The last section is conclusion, future work, and limitations.

## 2 Literature Review

Sztyler et al. [3] technique for HAR allows the location of wearable devices on the human body to alter. The technique uses a random forest classifier to combine frequency and gravity-based information in order to identify human activity and calculate the device's orientation. Nevertheless, these methods are not particularly accurate and cannot be used indoors. This study demonstrated an alternate strategy to employ a microcontroller to provide an end-to-end solution to precisely analyze gait speed in a variety of settings, given the constraints of such devices. The gyroscope is used to measure the orientation of moveable body parts, whereas the accelerometer monitors their actual physical acceleration. HAR is a method for separating out very similar human actions using the inputs from both sensors. For the classification of human walking gesture and the study of motion signals, various classifiers have been developed. Sun et al. [4] suggested a CNN-LSTM-ELM network on the opportunity dataset. Activities in the OPPORTUNITY dataset can be separated into gesture and locomotion. They discovered that the ELM classifier generalizes more quickly and effectively than fully linked classifiers [5] has suggested the use of the HAR approach, which is portable, non-interventional, has enhanced accuracy, and is relevant to real-time applications. For the purpose of recognizing complex, contemporaneous, interspersed, and varied human actions, a model based on transfer learning employing (GRUs) has been proposed [6]. Voicu et al. [7] presented a technology for recognizing human physical activity using data from smartphone sensors. Three smartphone-accessible sensors, namely gyroscope, accelerometer, and magnetic sensor, are used in the process to create a classifier. They intend for their proposal to include sitting, jogging, climbing, standing, and descending stairs. The results show that all six activities may be recognized with a high degree of accuracy (86–93%) [9–13].

# 3 Methodology

## *3.1 Technology Used*

To write the code, we have used Google Collab, and it is a collaborative environment, which will be used to produce results in the same file and will provide a CSV file output. Using the TensorFlow library, the deep learning models are designed. To reduce the model size, the TensorFlow-lite library is used and the Google TensorFlow library together with Keras is used for model training and evaluation.

## *3.2 Datasets*

Models are trained on three publicly available datasets: UCI- HAR, PAMAP2, and WISDM.

- UCI-HAR dataset: In the UCI-HAR dataset, six tasks were carried out by the volunteers: walk, downstairs walk, upstairs, standing, sitting, and laying. Using various image processing techniques, a total of 561 characteristics were retrieved from the sensor measurements.
- WISDM dataset: The WISDM dataset consisted of 36 volunteers who did six different activities while wearing smartphones and smart watches. These devices were equipped with accelerometer and gyroscope sensors. These activities were walking, running, ascending stairs, sitting, standing, and lying down.
- PAMAP2: The Physical Activity Monitoring Data Set 2 (PAMAP2) dataset is made up of sensor data obtained from a wearable during a variety of activities and worn on the upper body. It included data from nine sensors, consisting of an accelerometer, gyroscope, and magnetometer. The participants engaged in a variety of activities, such as chores around the house, physical activity, and outdoor pursuits. It produced a total of 52 features, including acceleration, angular speed, and magnetic field readings from the sensors.

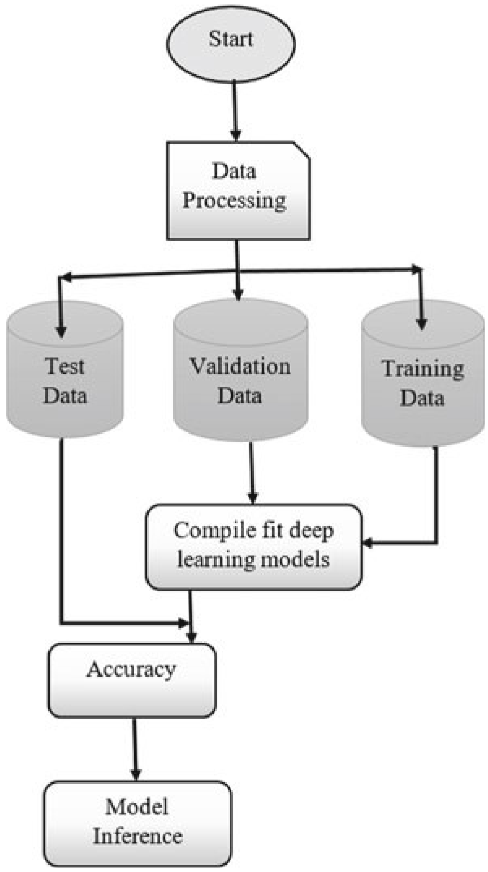


**Fig. 1** Proposed deep learning model flowchart

### *3.3 Development of Hybrid Models*

The stored data are first preprocessed to turn the raw samples into tensors after which the number of observations is divided into training, testing, and validation datasets. Some of the most important functions which have been used in the code are null (). sum to check for missing values and data.dropna() for removing missing or null values. We have created three different models—CNN 1D, CNN-LSTM, and CNN-GRU. Finally, we selected a single model (highest accuracy one) for further checking of performance. Figure 1 shows the proposed methodology flowchart.

### *3.4 Working of DL Models*

**CNN**: The model starts with a 1D convolutional layer of 64 filters with kernel size of 3, activated by ReLU. To extract abstract features from the output of the first layer, the same model is utilized. Following that, a max pooling layer with a pool size of 2 is applied to the output. Number of classes in the output which has been activated by SoftMax is sent to the output of the final max pooling layer which is flattened into a one-dimension vector. Categorical cross-entropy is used as the loss function in the model, ADAM is used as the optimizer, and assessment metrics are used to determine accuracy prior to training. Other two models have used the same hyperparameters. This model is trained for 10 epochs with 32 samples in each batch. While monitoring validation loss, ReduceLR On Plateau was utilized as a call back function with a lower bound of 0.0001 for the learning rate. As a result, each time the learning rate reaches a plateau, the model can decrease it by a factor of 10. This is done to increase the model's overall accuracy.

**CNN-LSTM**: The CNN-LSTM model was created using TensorFlow's Keras API. Its architecture is made up of a Conv1D layer with 64 filters, each with a three-kernel size. Here again we have used the ReLU function which has triggered the activation of neurons in these layers. To identify high-level characteristics from unprocessed data, a sliding window filter is applied to the input in a convolution network. The inputs are processed to the filter several times, creating a map of activation known as a feature map. To represent the temporal sequence of feature maps, we have used hyperbolic tangent (tanh) activation function with a dropout rate of 50%. Again, in this model, we have trained the dataset for 10 epochs with 32 samples in each batch.

**CNN-GRU**: The LSTM-based design described in the aforementioned section is extremely close to the architecture of the model. Feature maps are extracted from the data by the model and are further compressed using a Max-Pool layer. The data is then passed onto a layer called a Gated Recurrent Unit (GRU) after that. In the work, we have utilized 64 GRU units with the hyperbolic tangent (tan) activation function in the sequence layer. The dropout layer is dropped to 50% after the Max-Pool layer which will help in preventing overfitting. Here, we have recorded the model metrics of precision, recall, F1-score, and support values for each gesture category.

## *3.5 Classification*

The accelerometer and gyroscope inputs of various gesture of dataset have fed into the deep learning model, if the input data exceeds a minimum threshold of 1.5, and an inference step will produce the anticipated probability of the data falling into each class of gesture (Squat, Run, Walk, Jump, etc.). The results are represented as confusion matrix and accuracy. Algorithm 1 shows the detailed steps of working.

**Algorithm 1** For HAR's model creation and implementation is shown here

| **Algorithm1**: HAR using wearable sensors. |
| --- |
| **Results**: Classification of accuracy of various human activists. |
| **Initialization**: HAR datasets (WISDM, PAMAP2, and UCI-HAR). |

(continued)

(continued)

| **Algorithm1**: HAR using wearable sensors. |
|---|
| **Step 1**: Pre-Processing of the datasets;<br>**Step 2**: Split the datasets into training, validation set, and testing;<br>**Step 3**: Design of deep learning models CNN-LSTM, 1D-CNN, and CNN-GRU;<br>**Step 4**: Performance analysis of deep learning model based on precision, recall, accuracy, and F1- score;<br>**Step 5**: Select the best model based on accuracy; |

## 4 Experimental Result

This research work has presented the performance of three different deep learning model architectures: CNN-LSTM, CNN 1D, and CNN-GRU, as detailed in the model development section. The data was divided into three groups: training (70%), testing (10%), and validation (20%). The test data is used to examine support, recall, F1-score, precision, and overall accuracy of the models.

### *4.1 Dataset Comparison Result*

Overall, all the models are able to perform with very high accuracy (95%) with the datasets. Among the five gestures, squat and run were easier to identify for the models than the jump and walk gestures.

- Within the six gestures of UCI dataset, namely laying, walk, upstairs walk, and walk downstairs are easier for the model to identify than sitting and standing gestures. CNN has given the best accuracy of 95%. The input data is transformed into tensors with a shape of (5146, 128, 9) and normalized between 0 and 1, while the output data is one-hot encoded with a shape of (7352, 6).
- Within the six features of the WISDM dataset, the standing feature is easier for models to identify as compared to other gestures. In this, CNN-GRU achieved the best accuracy of 97%. The input data was transformed into tensors with a shape of (19,495, 128, 3) and normalized between 0 and 1, while the output data was one-hot encoded with a shape of (19,495, 6).
- In the PAMAP2 dataset, we have a total of 11 features. Rope_Jumping, Cycling, Nordic_Walking, Vaccum_Cleaning, and Ironing are easier for the model to identify as compared to its remaining features. CNN 1D was able to achieve the best accuracy of 99%. The input data is transformed into tensors with a shape of (4370, 128, 39) and normalized **between 0 and 1, while the output data is one-hot encoded with a shape of (4370, 12).**

Figures 2, 3, 4 show the accuracy loss curves for CNN-LSTM model on different datasets UCI-HAR, WISDM, and PAMAP2.

Using USB, the board is loaded with the compressed model header file and the Arduino sketch file (.ino file). A battery is then attached to the board after which the board is reset to activate the new sketch. After that, we will invite the participant to make the various motions, after which the board is free to make inferences in real time. We assessed the board's performance on the participant using 100 gestures (20 for each category). The quantized model's results are fused on the same board

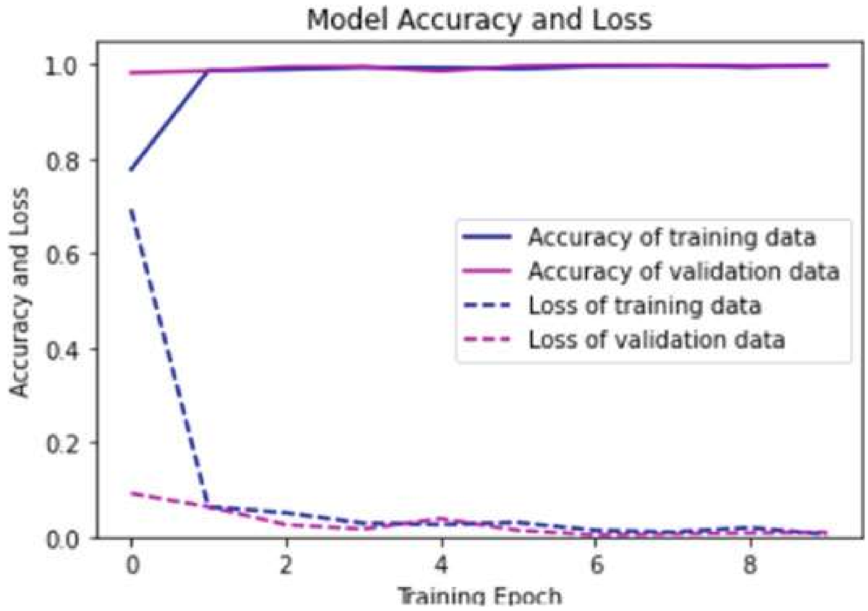


**Fig. 2** CNN-LSTM model (UCI-HAR)

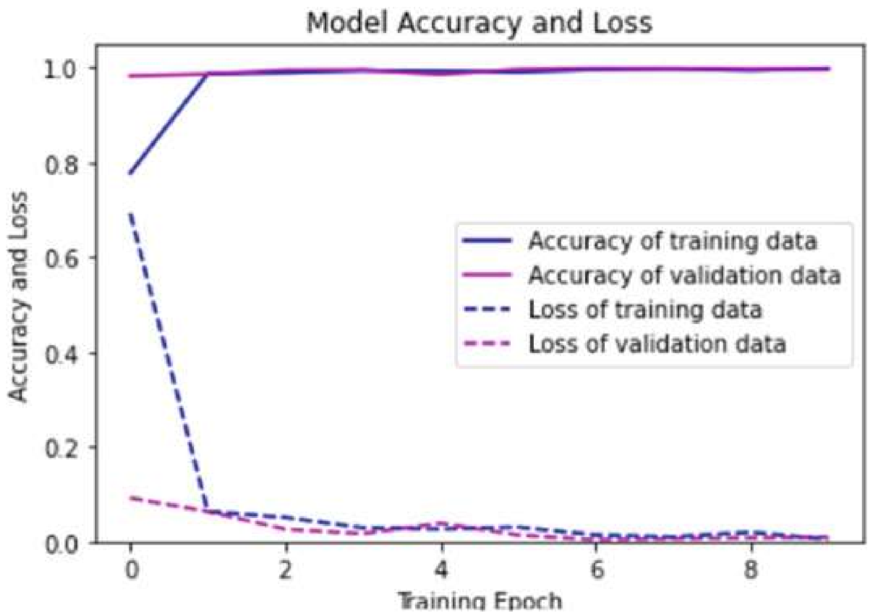


**Fig. 3** CNN-LSTM model (WISDM)

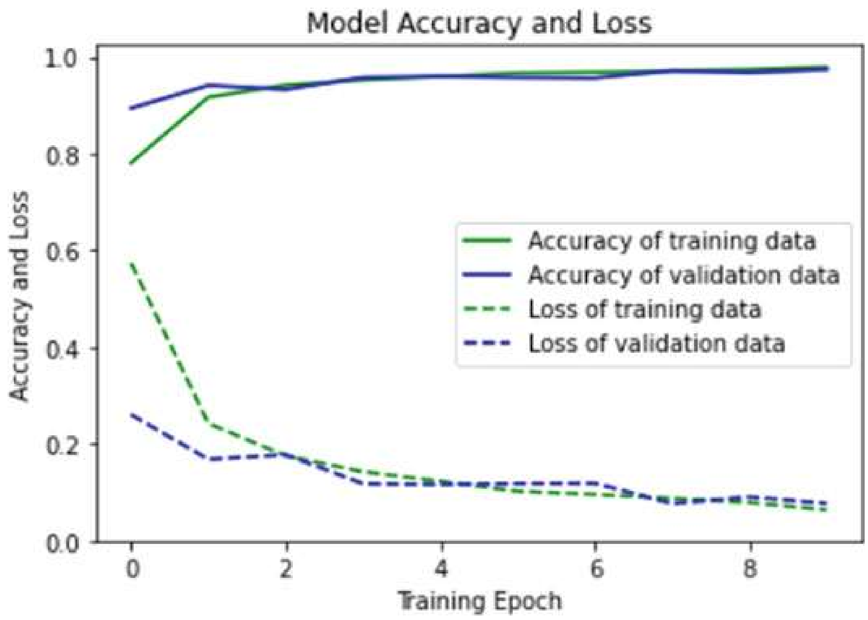


Fig. 4 CNN-LSTM model (PAMAP2)

with similar results to support the edge computing. We found that the majority of the motions had inference times between 100 and 500 ms. By omitting the BLE communication of the results, this inference time can be cut even more. Figures 5, 6, and 7 show the performance matrix of compressed CNN-LSTM model over datasets UCI-HAR, WISDM, and PAMAP2, respectively.

Table 1 shows the comparative performance analysis of various deep learning models over different datasets.

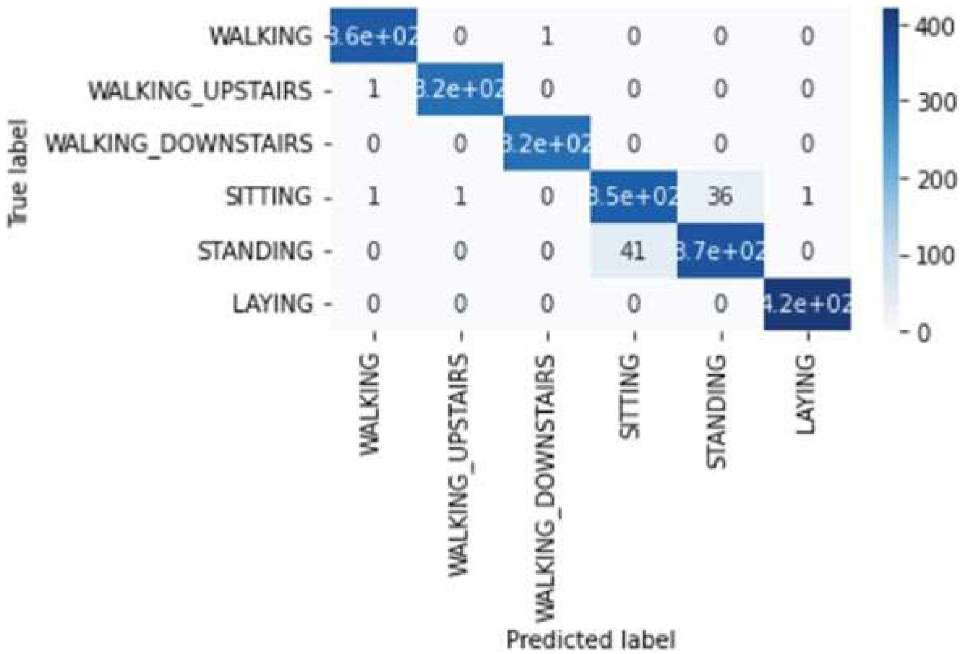


Fig. 5 Confusion matrix for CNN-LSTM compressed model (UCI-HAR)

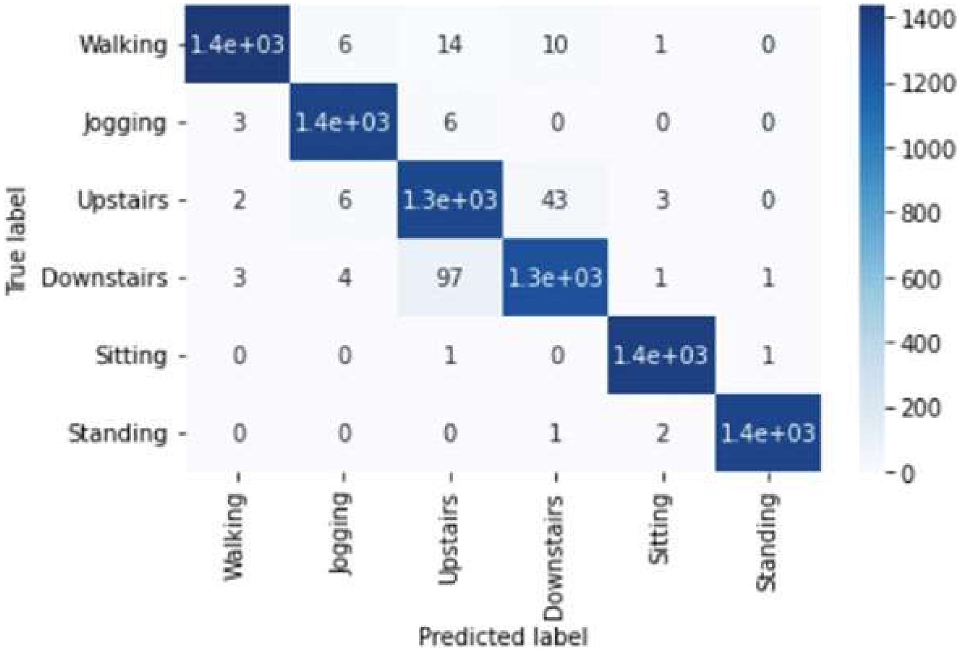


**Fig. 6** Confusion matrix for CNN-LSTM compressed model (WISDM)

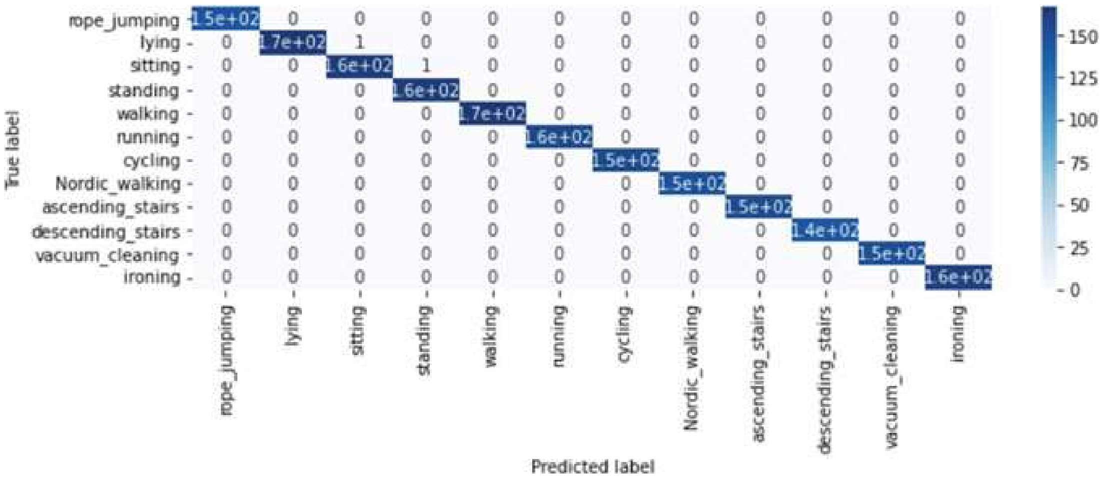


**Fig. 7** Confusion matrix for CNN-LSTM compressed model (PAMAP2)

**Table 1** CNN-LSTM results

| Proposed model | Accuracy on WISDM | Accuracy on PAMAP2 | Accuracy on UCI-HAR |
|---|---|---|---|
| CNN | 97.11 | 96.23 | 95.56 |
| CNN + GRU | 97.32 | 96.59 | 97.79 |
| CNN + LSTM | 99.89 | 97.28 | 96.78 |

## 5 Conclusion and Future Extension

This presented work focuses on wearable sensors capable of detecting the wearer's location as they walk have been created and applied in this investigation. The system uses an inertial-based navigation algorithm that was corrected by an EKF. In this study, a compressed deep learning model was used to recognize human movement and gestures. It was decided to use three datasets (WISDM, PAMAP2, and UCI-HAR). For each database, we were able to attain highest accuracy of 99.89%, 97.28%, and 96.78%, respectively, for CNN-LSTM model. We were able to reduce the model size by 10 times while increasing models' average performance to 97% by using the model compression strategies of pruning and quantization. Arduino LED predictor is comfortable to wear, customizable, and most importantly, it will protect data. In the investigation, we placed the compressed model on the board and used it to infer motions in real time. The findings point to a promising new direction in human activity recognition using edge AI devices that are secure, reliable, and low-powered.

**Limitation of work**: HAR systems having various inheritance design challenges, namely selection of different types of sensor, a set of criteria for data collecting, the performance of recognition, the amount of energy used, processing power, and adaptability. The deep learning models having also size is huge which is required to reduce to design less computational machine. It also suffers with intraclass variability, class imbalance problems, etc.

## References


1. Gravina R, Ma C, Pace P, Aloi G, Russo W, Li W, Fortino G (2017) Cloud-based Activity-aaService cyber–physical framework for human activity monitoring in mobility. Futur Gener Comput Syst 75:158–171
2. Greco L, Ritrovato P, Xhafa F (2019) An edge-stream computing infrastructure for real-time analysis of wearable sensors data. Futur Gener Comput Syst 93:515–528
3. Sztyler T, Stuckenschmidt H, Petrich W (2017) Position-aware activity recognition with wearable devices. Pervasive Mob Comput 38:281–295
4. Sun J, Fu Y, Li S, He J, Xu C, Tan L (2018) Sequential human activity recognition based on deep convolutional network and extreme learning machine using wearable sensors. J Sensors
5. Chen J, Sun Y, Sun S (2021) Improving human activity recognition performance by data fusion and feature engineering. Sensors 21(3):692
6. Thapa K, Abdullah Al ZM, Lamichhane B, Yang SH (2020) A deep machine learning method for concurrent and interleaved human activity recognition. Sensors 20(20):5770
7. Voicu RA, Dobre C, Bajenaru L, Ciobanu RI (2019) Human physical activity recognition using smartphone sensors. Sensors 19(3):458
8. Gupta S (2021) Deep learning based human activity recognition (HAR) using wearable sensor data. Int J Inf Manage Data Insights 1(2):100046
9. Dua N et al (2021) Multi-input CNN-GRU based human activity recognition using wearable sensors. Computing 103:1461–1478
10. Dua N et al (2023) A survey on human activity recognition using deep learning techniques and wearable sensor data. In: Machine learning, image processing, network security and data sciences: 4th international conference, MIND 2022, 2023, Proceedings, Springer, pp 52–71

11. Semwal VB et al (2022) Pattern identification of different human joints for different human walking styles using inertial measurement unit (IMU) sensor. Artific Intell Rev 55(2):1149–1169
12. Bijalwan V et al (2022) Wearable sensor-based pattern mining for human activity recognition: deep learning approach. Indus Robot: Int J Robot Res Appl 49(1):21–33
13. Dua N et al (2022) Inception inspired CNN-GRU hybrid network for human activity recognition. Multimed Tools Appl 1–35