International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 2s
DOI: https://doi.org/10.17762/ijritcc.v11i2s.6033
***Article Received**: 18 November 2022 **Revised**: 26 December 2022 **Accepted**: 01 January 2023*

An Imperceptible Method to Monitor Human Activity by Using Sensor Data with CNN and Bi-Directional LSTM

P. Rajesh[1]*, R. Kavitha[2]
[1,2] Department of Computer Science and Engineering
[1,2] Vel Tech Rangarajan Dr.Sagunthala R&D Institute of Science and Technology
Avadi,Chennai,Tamil Nadu India
[1*] prajesh@veltech.edu.in
[2] rkaviha@veltech.edu.in

Abstract— Deep learning (DL) algorithms have substantially increased research in recognizing day-to-day human activities All methods for recognizing human activities that are found through DL methods will only be useful if they work better in real-time applications. Activities of elderly people need to be monitored to detect any abnormalities in their health and to suggest healthy life style based on their day to day activities. Most of the existing approaches used videos, static photographs for recognizing the activities. Those methods make the individual to feel anxious that they are being monitored. To address this limitation we utilized the cognitive outcomes of DL algorithms and used sensor data as input to the proposed model which is collected from smart home dataset for recognizing elderly people activity, without any interference in their privacy. At early stages human activities the input for human activity recognition by DL models are done using single sensor data which are static and lack in recognizing dynamic and multi sensor data. We propose a DL architecture based on the blend of deep Convolutional Neural Network (CNN) and Bi-directional Long Short-Term Memory (Bi-LSTM) in this research which replaces human intervention by automatically extracting features from multifunctional sensing devices to reliably recognize the activities. During the entire investigation process we utilized Tulum, a benchmark dataset that contains the logs of sensor data. We exhibit that our methodology outperforms by marking its accuracy as 98.76% and F1 score as 0.98.

Keywords- Elderly care, CNN, Bi-LSTM, Privacy, Sensor data.

I. Introduction

Activity recognition (AR) has extended its application level in various areas like smart home, health care sectors, surveillances, smart driving system etc. It is particularly useful for designing precise and smart assistive technology or optimizing processes in industries where manual involvement is still prevalent. AR techniques have been created using various methods. Signals from videos, multichannel time-series such as readings from isometric scales, are used as inputs to AR methods. Traditional AR processes were done by recorded videos and by gathering hand crafted images, then by involving those inputs into a technology and recognize the activities which consume more time and may lack in accuracy also. Many methods were difficult to capture, evaluate, and analyze the signal series in order to recognize human behaviors. A typical human activity recognition pipeline uses a sliding-window method to segment data and extracts pertinent assembled features from the segmented sequences. However it is a herculean task to analyze and to recognize the activity due to variations in human behaviors. So far, simple human behaviors have been identified and explored satisfactorily. Recognizing complicated human actions is a difficult task, and significant research is ongoing in this arena. Statistics [1], show that the population of elderly people would be more than 1.4 billion by 2025.Most of the aged people would prefer to lead a healthy and private life. But due to high cost of elderly care centers they would stay into their own home without any human assistance and care [2]. Activities of elderly people need to be monitored to detect early symptoms of any abnormalities in their health. It should also be capable of detecting potential emergencies. Even if some old people live in care centers their activities are being monitored using devices like wearable gadgets and cameras. But those monitoring activities mostly intrude into the privacy of the elderly people [3].The recent pandemic situation caused due to COVID-19 had educated us to invent many techniques to monitor people without directly engaging with any individuals.

The focus of this research is to monitor the human activities from multichannel time-series sensor data and to screen the activities of aged people who live alone without any physical and mental assistance. The proposed approach ensures extreme privacy for the person who is being monitored. We used only the sensor data rather videos and images for AR.

Hence to ensure complete privacy and to monitor the activity an approach based on the combination of CNN and Bi-LSTM is proposed. The investigation is carried out using Tulum [4] data set as a training data set, which was prepared by collecting images from various binary sensors placed in the house and infer a most probable sequence of activities performed by the supervised person for minimum period of time. The statistical information of the dataset is as shown in Table II. Then the collected image patterns are used as input to the proposed model. Initially 10 basic daily activities like dish clean, cook, sweep, sit, sleep, eat, watch television, enter, leave, table work are considered to assess the classifier.

Most of the activity recognition models do not focus on classifying and recognizing the temporal data .Our proposed approach is investigated as follows.

- We labelled the data obtained from the dataset. The data is labelled to get normalized sensor data. The OFF mode sensor data is removed from the dataset during labelling the activity.
- Labelling reduces computation cost of CNN during feature extraction.
- CNN is used to extract the features of temporal sensor data from labelled data.
- A Bi-directional LSTM is coupled with CNN to recognize the activity through its storing capability of present and past data.
- We mainly focused on recognizing the temporal data.
- Finally the proposed model outperforms the existing models with accuracy of 98.76% in recognizing all the ten activities considered.

The structure of this article is as follows: Section 2 of this article contains related work. In Section 3 we converse our proposed architecture .Our methods and materials are explained in Section 4. In Section 5, we have explained the results obtained and compared it with some of the existing models. The article is concluded in Section 6.

II. Related Work

Recently many AR methods are proposed by utilizing the wide spread applications of many ML algorithms. Authors of [5] offered a procedural, unambiguous speculative model and initial approach to process sensor data collected from a smart home domain. To describe domain the knowledge the authors suggested an ordered structure of the human activity analysis model. The dataset was collected from the test bed where the locations of sensors are not dependent. This makes the model to recognize only single sense modality. To distinguish unusual and hazardous patterns, authors [6] established a method for distinguishing human activities from sensor data.

Activity acknowledgement, sequence grouping, and illness prediction are the three modules that make up their framework.CNN is used for testing and training the data and produced accuracy of 90.36%. This approach consumes more time in extracting the features since OFF state data is not removed from the input. A multimedia-related acknowledgment system was suggested by [7]. For the state recognition of patients the system is built on features collected from video and audio data acquired which is collected from stored data atmosphere. CNN and SVM are used as a combined approach even though the results reveal that utilizing a mixed modality to accurately categorize and provide enhanced accuracy the privacy is not considered.

Authors of [8] anticipated a technique from the data obtained from a limited count of sensors located in a kitchen setting and analyzed using a CNN and EnsNet ML algorithms. This method had used data from the self collected data set and had failed to annotate the data and the model suffered from memory utilization..To recognize activities, the researchers used hierarchical clustering and the value of accuracy is around 91%. Another study in [9] used Dynamic Bayesian networks (DBNs) to extract correlation between features as they evolve over time. Using hidden semi-Markov models, their method managed to predict the event with the accuracy rate of 94.62%.Bayesian model are capable of recognizing image input, but still it could not classify temporal data.

In the case of activity recognition, certain ensemble approaches have also been used. Authors of [10] unveiled a unique SAT-based splitting approach, authors enhanced random forest method and the accuracy was around 91%.The model's performance is promising but still consumes large memory. In [11] the researchers applied a formerly available Cluster-Based Classifier Ensemble (CBCE) technique, which presents a sustenance procedure for clustering and to address the acknowledgment issues in the formed clusters where the F1 score was around 0.93.The time and storage of this method is acceptable but it could cluster only the stored or static data and denies in recognizing temporal data.

Furthermore, making use of the advancement of deep learning, various models and methods based on neural networks are used to recognize activities. [12] Introduced a technique for labelling a separate, but related collection of activities Then the labelled sensor data is assessed using AlexNet model and produced accuracy as 95.72% .This model could address a limited class or similar classes that occur in a same location. [13] Introduced a resemblance assessment method that labels a big amount of unlabeled data with a little volume of identified data.

The labelled data are then grouped based on the activity and trained using ResNet model were the accuracy is around 93%.This method is similar to previous method and lacks in addressing spatial-temporal data.

Authors of [14] addressed the challenge of data labelling that makes unsupervised ML method difficult. It has been resolved using ensemble cluster method and data augmentation. It also solves the data annotation problem, using frequent sensor excavating techniques. The excavated data is tested with CNN and the accuracy was recorded as 92.63%.CNN layers used in this method extracts features of similar activities first and other activities later which results more time consumption in recognizing unknown or new activity.

Continuous and discreet pattern mining ways, modelling based on reduced feature; probabilistic model and collecting of activity classification through simple mining methods are recently proposed by data scientists [15].These data are collected from smart home with more than one person residing in the test bed mode of activity recognition. For assessing the activity procedures of dissimilar inhabitants, to distinguish a person's actions collected from undisturbed sensor data authors of [16] suggested an approach based on FCA (formal concept analysis) with CNN. Recognizing actions performed by two inhabitants were investigated and the model accuracy was 88.3%.Since two people's activities are used, the model overlaps the activity with the other person. [17] Proposed a factorial HMM model with undeviating Bayesian perseverance with LSTM. The authors didn't make any distinctions amongst the test person and produced accuracy of 92.6%. Researchers' of [18] suggested an upgraded Term Frequency-Inverse Document Frequency technique (TF-IDF) for extracting characteristics and used LSTM alone to recognize the activity and managed to provide accuracy around 91%.

In [19] a method for preparation of real-time activity acknowledgement systems was developed. Sensor based face recognition was used based on LSTM approach, which ascertains the individuals' appearances while maintaining solitude; they do not protect them from privacy attacks based on inference but produced accuracy of 96.35%. Some ways to achieve privacy at the classification level have been proposed, for instance [20] recognized the activity even when the individual tries to cough by referring the stream of voices through mobile devices using deep CNN model and their accuracy rate were 95.7%.

From the literature reviews even though most of the models have produced reasonable accuracy limitations like high time consumption, increased memory usage, lack in privacy, denial in recognizing temporal data etc are found. Most of the existing methods consumed more time in labelling the activity. Few of the works could recognize only similar activity at a faster time and the recognition rate reduces while a newer activity is given as input. We addressed these limitations by generating IPA from the input sensor data which makes feature extraction faster and highest recognition rate in minimum time.

III. PROPOSED METHOD

3.1 CNN

CNN is made up of convolutional layers that extract features from incoming data. Every neural network has some common components like an input layer where the input data is provided to the model, then the hidden layers which has many levels to process the given input, followed by an output layer where the processed input is obtained as output .The structure of neurons that imitate a human brain network is three-dimensional, with inputs and outputs that have width, height, and depth. The CNN layers use filters to fetch all the input features from input data using kernels.

These stratums recite the different metrics from the given input to create feature activation maps (FAM).These maps are the entities which makes the kernel to perform convolution over the data. As indicated in Fig. 1, the input layer with feature signals X (a_1, $a_2 \ldots . a_{n+1)}$ are coupled to the convolution layer with kernel size m.The process of feature extraction is as mentioned from Eqn. (1) to (4).

$$L_1=m_1a_1 + m_2a_2 + m_3a_3 \quad (1)$$

$$L_2=m_1a_2 + m_2a_3 + m_3a_4 \quad (2)$$

$$L_i=m_1n\text{-}1 + m_2a_n + m_3a_{n+1} \quad (3)$$

$$L_i= \alpha \sum (S_n \times H_n, i) \quad (4)$$

Input signal is represented as n, i represents the feature extracted which forms a link between each input S_n,× is the convolution operator. As shown in Fig.1 pooling functions are used to fine tune the obtained feature signals in the pooling layer which is connected with the convolution layer. Multiple convolutional layers aid in the extraction of features from input data at higher levels of abstraction. The convolutional operator is used to determine the dimension of the input data. Hence the 3D data input, two dimensional convolutional layers are naturally recycled for the time-based classification of input vectors. The convolutional process scans the input by passing a filter (kernel) over it and continuously inspecting trivial vectors of the input until it is entirely examined. FAM activations are defined in the filter as Cartesian product (×) of the matrices in existing filter window multiplied by the weights.

***Article Received**: 18 November 2022 **Revised**: 26 December 2022 **Accepted**: 01 January 2023*

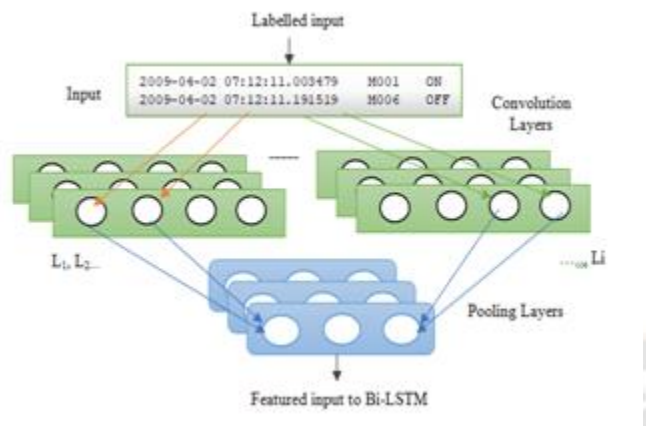

Labelled input
Input
2009-04-02 07:12:11.003479 M001 ON
2009-04-02 07:12:11.191519 M006 OFF
Convolution Layers
L1, L2...
......Li
Pooling Layers
Featured input to Bi-LSTM

Figure 1. Convolutional Neural Network

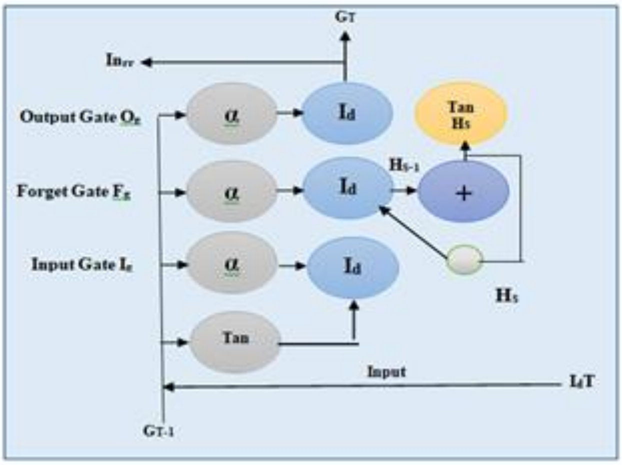


Figure.2 A simple Bi-LSTM Cell

3.2 Bi-LSTM

Every Bi-LSTM layers are prepared with contiguous memory blocks that are tied into a memory cell in a recurrent manner. All those cells are constructed with fewer gates where every gate has responsibilities like when to forget the memory unit prior to the concealed states and update the cells, letting the network to use sequential data of the input.Fig.2 depicts a typical Bi-LSTM cell, Ig represents input_gate, Og as output_gate, Fg is the forget_gate. Their functionalities are input data flow control, to forget the internal state contents when needed, output data flow control respectively. The Eqn. (5) to Eqn. (10) delineates the process. Table 1, explains the parameters and notations used in the Bi-LSTM cell.

TABLE 1. Parameters and notations used in Bi-LSTM

Parameters	Notations
Input feature	If
Input data	Id
Time	T
Input_gate	Ig
Output_gate	Og
Forget_gate	Fg
Bias vector	Bv
Weight metrics	W1, W2
Internal recurrence	Inr
Hidden state	Hs
Current Output	Cto

$$I_g=\alpha\ (W_I I_d T + W_I\ H_{S\text{-}1} + B_I) \tag{5}$$

$$F_g=\alpha\ (W_F\ I_d\ T + W_F\ H_{S\text{-}1} + B_F) \tag{6}$$

$$O_g=\alpha\ (W_O\ I_d T + W_O\ H_{S\text{-}1} + B_O) \tag{7}$$

$$G_T=\alpha\ (W_G\ I_d T + W_G\ H_{S\text{-}1} + B_G) \tag{8}$$

$$Inr= G_T\ I_g\ + F_g \tag{9}$$

$$H_S= O_g\ .tan\ (Inr) \tag{10}$$

We propose a bidirectional LSTM (Bi-LSTM) as an improved method for the customary LSTM model. Fig. 3 depicts the idea of Bi-LSTM [21] the logic behind BLSTM is to make clear predictions by considering the old and newer inputs by to and fro sequences respectively.

$$H_S\ (\text{Forward}) = G\ (W_I \rightarrow I_d T + W_I\ H_{S\text{-}1} + B_I) \tag{11}$$

$$B_I\ (\text{Backward}) = G\ (W_I \leftarrow I_d T + W_I\ H_{S\text{-}1} + B_I) \tag{12}$$

$$Q = G\ (H_S + B_I) \tag{13}$$

Eqn. (11) to Eqn. (13) represents the forward and backward sequences and the same is represented in Fig.3 as forward and backward arrow representation.

IV.METHODS AND MATERIALS

The proposed architecture contains CNN which is used for feature extraction and Bi-LSTM for classification in 3 frequencies, Fig.3 provides the details of a single frequency of classification. Every channel is comprised of 3 convolution layers placed on top of each other accompanied through a max-pooling layer. For feature extraction, the convolution layers have around 64 filters to provide uninterrupted mapping and conceptual illustration of sensory input. The results of the Convolution layers and max-pooling layers are further obtained by the ABLSTM layer with configurations with maximum of 128 units and a dropout function of 0.25 percent. As a result, Bi-LSTM layer is highly adapted for adjusting the internal state by using the to and fro sequences. The purpose of the dropout layer is to eliminate over fitting and enhance model accuracy. Within the model, the output from the three channels are normalized and then combined together.

DOI: https://doi.org/10.17762/ijritcc.v11i2s.6033
***Article Received**: 18 November 2022 **Revised**: 26 December 2022 **Accepted**: 01 January 2023*

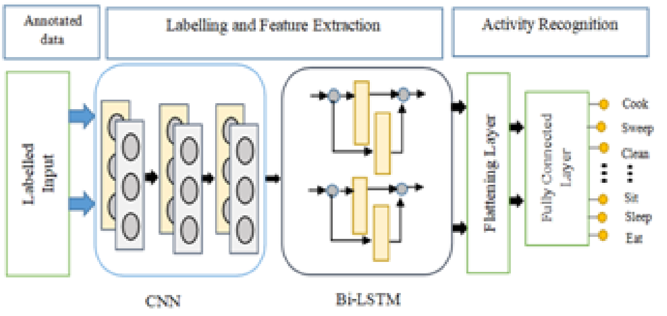


Figure 3. Proposed Architecture

It then proceeds through a FCL and generates the feature by recognizing the activity. The way how the activities are labelled, extracting the features and recognizing the activities are done by using three algorithms. CNN performs the process of extracting the features from the labelled data. All the extracted features are properly examined and fed as input to the next stage of the model. The Bi-LSTM proposed uses the input provided by the CNN that is the feature extraction matrix and performs the process of activity recognition. At the end of the Bi-LSTM layer the flattening layer is deployed for the purpose flattening the output. The fully connected layer classifies the activities.

4.1 Data Set

To conduct the investigation we obtained the features of Tulum data set .This data set contains around 16200 annotations of various activities. The data set was prepared by collecting the data by allowing two residents to stay in the test bed. The data collection is done by incorporating motion sensors (M001-M031) door sensors (D001-D004) and temperature sensors (T001-T005) in and around the test bed where the residents lived around 98 days. The statistical information of the dataset is as shown in Table 2.

TABLE 2. Statistical Information of the Dataset

Descriptions	Metrics
Data set	Tulum
Motion sensors	31 (M001-M031)
Temperature sensors	5 (T001-T005)
Door sensors	4
Activities recorded	14
Samples	17980
Duration of Data collection period	98 Days

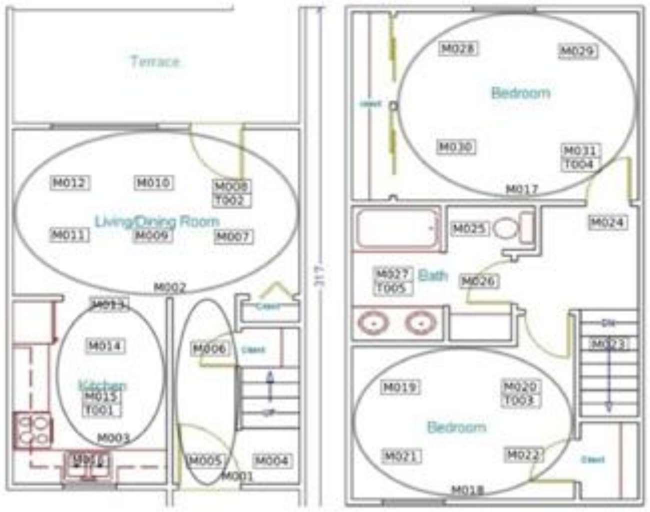
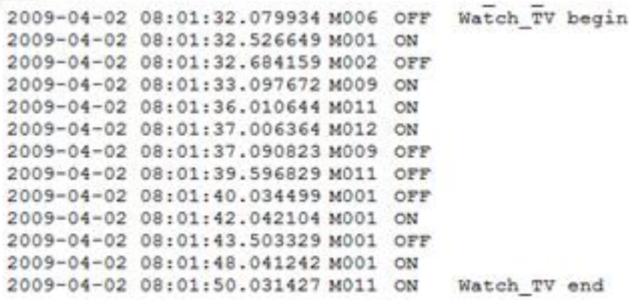


Figure 4. Watch TV sensor data

The layout of the test bed and the locations of the sensors are as shown in Fig.5.The test bed is so constructed with all the basic facilities needed for two persons to live in. The sensors are prompted to gather the information as and when the test person crosses the sensors located. They record their details in the log in four states, like OPEN, CLOSE, ON and OFF. The data set contains the sensor details as date of occurrence, time of the event occurred, sensor ID and sensor status. A sample of sensor data from the data set is shown in Fig.4 for Watch TV activity.

Since the television is located in the living room which contains sensors M002,M007,M008,M009,M010,M011 and M012,the watch television activity is recorded by the above said sensors. The number 2009-04-02 indicates the date of occurrence of activity, 08:01:36 indicates the time, followed by sensor ID and finally the status of the sensor. If the person crosses the sensor it is recorded as ON or OPEN for motion sensor and door sensor respectively. If it is recorded as OFF or CLOSE the person leaves the location. Similarly all the activities are recorded and stored in .CSV file. It contains around 20 classes and we have utilized ten activities for our experiment.



```
2009-04-02 08:01:32.079934 M006 OFF  Watch_TV begin
2009-04-02 08:01:32.526649 M001 ON
2009-04-02 08:01:32.684159 M002 OFF
2009-04-02 08:01:33.097672 M009 ON
2009-04-02 08:01:36.010644 M011 ON
2009-04-02 08:01:37.006364 M012 ON
2009-04-02 08:01:37.090823 M009 OFF
2009-04-02 08:01:39.596829 M011 OFF
2009-04-02 08:01:40.034499 M001 OFF
2009-04-02 08:01:42.042104 M001 ON
2009-04-02 08:01:43.503329 M001 OFF
2009-04-02 08:01:48.041242 M001 ON
2009-04-02 08:01:50.031427 M011 ON   Watch_TV end
```

Figure 5. Sensor layout in test bed during data collection [4]

4.2 Methodology

The experiment was conducted by distributing the process into three stages, in which each stage performs distinct process to recognize the activity.1.Labelling the activity, 2.Feature extraction, and 3.Activity recognition. Labelling the activity is the initial process where all the obtained data are cleaned and labelled, the feature extraction stage analyses the given input labels and extracts the features .The final stage is the activity recognition.

Labelling the activity

We utilized the data log collected from Tulum dataset which consists of sensor data represented as ON, OFF, OPEN, CLOSE whenever the sensor is triggered. Here when the motion sensor or door sensor status if set as ON it indicates that the person is under the surveillance of the respective sensor. To recognize the activity only the ON or OPEN status alone has to be considered. So for the classifier to accept only those states OFF and CLOSE status of the sensors has to be removed from the log. Most of the existing approaches used the OFF state of the sensors which increases the time complexity. We emphasized a simple way to collect only the active state of the sensors and store it in a data log named as Activity Data Cube (ADC)(as act=sst1,sst2,sst3…..sstn) every sst stores parameters as sst= {ast,sno, sstat} where sst represents activity start time,sno is the sensor number, stat is the sensor status. To better understand, the watch TV activity is considered and its sensor log is as shown in Fig.4.The test person when starts to watch TV the sensors M001,M009,M011 and M012 alone is triggered to ON state, But the log contains M001, M002, M006 in the OFF state which should be removed from the sequence sst1,sst2,sst3…..sstn .

To perform the removal of OFF state we assign 1 for ON and 0 for OFF state .When 0 is found in sstn then the 0 value (OFF state) is removed and only the ON status of the sensor is accepted and stored in ADC.For watch TV activity sst1 is generated in the formal sequence as sst1={(08,M006,OFF),(08,M001,ON),(08,M002,OFF),(08,M009,ON), 08,M011,ON),(08,M012,ON),(08,M009,OFF),(08,M011,OFF),08,M001,OFF,(08,M001,ON), (08,M001,OFF), (08,M001,ON),(08,M011, ON)}.Here 08 in all the set indicates time of occurrence of the event followed by the sensor name and status. As specified all the ON and OFF to be numbered as 1 and 0 respectively. Hence sst1 is derived as sst1= {(08,M001,1), (08,M009,1), (08,M011,1), (08,M012,1), (08,M001,1),(08,M001,1), (08,M011,1)} and stored in ADC by labelling watch TV . The same process is done for all the ten activities used in our experiment. Algorithm 1 explains the process of labeling the activity.

Algorithm 1 Activity Labelling

```
Input: sst= {ast, sno, sstat}//sensor activity log details.
Output: act= {sst1, sst2, sst3…..sstn }// labelled activity
Count the activity to the specified sensor
for every act in sst:
    sen_prsn detects the presence of sensor in the log
    initiate to load the activity start
    for every sst in sen_t:
        sen_prsn [{(sst.act, 1, 0)};
        Store only ON (1) sst of every act to ADC
    end for
ADC → (sst, cnt)
```

Algorithm 2: Feature Extraction

```
Input: array_act= {sst1, sst2, sst3…..sstn}
Output: act= {sst1, sst2, sst3…..sstn}
act;
for every n in act:
    new_sen_seq;
    for every sst in act.n:
        tmp LoadTime (sst.ast1, (ADC))
        if tmp > max:
            new_sen_seq [sst]
        end if
    end for
    array_act t {n, new_sen_seq }
end for
return array_act
```

Activity Recognition

After extracting characteristics, we use our proposed algorithm to recognize the activities. The elbow technique was used to choose the clustering results k. We identify the respective cluster that fits to the input test data after clustering, and then determine the relationship among the occurrences in the cluster and the test data. After calculating the ratio (r), choose the n closest occurrences within the cluster and allow them to elect the label of the test data. Let us consider the test sample is represented as tst= {ast, sno, sstat} and the training set sample as trn= {ast, sno, sstat}.We used Levenshtein ratio (Lr) method to identify the similarity ratio between test and training data.Eqn. (14) and (15) represents the method of finding the similarity.

$$L_r = \frac{(S-D)}{S_i} \tag{14}$$

S is the total of length of ast of test and training data. m_n are the weights of time and sequence of the sensor, 24 represents hours in a day.

$$R = (m_n \times 24) - (\mathrm{ts.ast} - \mathrm{tr.ast})/24 \qquad (15)$$

Algorithm 3-Activity recognition

```
Input: act= {sst1, sst2, sst3…..sstn} // labelled activity
Output: Recognized activity
Collect start time of the activity
q=calculate({ast,sstat}) //locate the test sample in cluster
for each e in act:
if cluster(e) == q:
ratio =calcratio(e,ast,m1,m2)     //calculate r
  end if
  Arrange the ratio from highest to lowest values
  for x in range(n):
  max_ratio =ratio (i)        //splay highest ratio
  y= union(max_ratio)       //union of all highest ratio
  Elect the label for the activity
  end for
```

Algorithm 3 delineates the crucial process of the experiment and as an abstract of the analysis, initially all the data set is collected from the resource; all the activities are labelled and need to be grouped for effective classification. Once the data are grouped clusters are formed with respect to the labelled activity. We employed Levenshtein ratio method to find the closest ratio value between test and training data and used those values to label and vote for the activity by incorporating activity recognition model as proposed. Hence the activities are properly classified and the results obtained are discussed as below.

V. PERFORMANCE EVALUATION

To train our model, we employed the Intel i7, 16GB RAM, and 2.6GHz specifications. TensorFlow, Ubuntu operating system, and Python as programming were used. The obtained results need to be examined for evaluating the proposed model. We primarily used all of the specimens in the deliberated dataset and performed k-fold cross-validations to assess our models performance. The validation is done by selecting different values of k to assess the overall outcome for various forms in training and testing samples. The values are selected in the ratio of 8:2 in which 8 for training and 2 for testing. The cross validation is done as k=3, k=5, k=10.Fig.6 represents the overall accuracy achieved for all the k values.

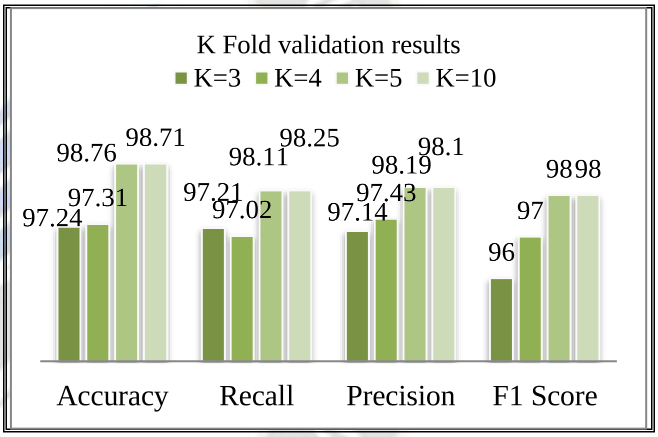


Figure 6.K fold validation results

Activities	Dish clean	Cook	Sweep	Sit	Sleep	Eat	WatchTV	TableWork	Enter	Leave
Dish clean	271	0	0	0	0	0	0	0	0	0
Cook	0	182	0	0	0	0	0	0	0	0
Sweep	0	0	317	0	0	0	0	3	0	0
Sit	0	0	0	212	0	0	0	0	0	0
Sleep	0	0	0	0	173	0	0	0	0	0
Eat	0	0	0	0	0	62	0	0	0	0
WatchTV	0	0	0	9	0	0	278	0	0	0
Tablework	0	0	0	0	0	5	0	36	0	0
Enter	0	0	0	0	0	0	0	0	8	3
Leave	0	0	0	0	0	0	0	0	3	11

Figure 7. Confusion matrix

At k=10 the accuracy is recorded as 98.76% whereas at k=3 the accuracy was 97.24% and 97.01% for k=5. Apart from accuracy, we also used precision, recall, and the F1-score to measure the outcomes. The parameters and the metrics are as shown in Eqn. (16) to Eqn. (19).

Accuracy is obtained from the division of True_Positive (T_P) and True_Negative (T_N) with the summation of False_Positive (F_P) and False_Negative (F_N).Here T_P are the samples which are correctly predicted and True Negative are the samples where the negative values are correctly predicted. For example if the activity Watch TV is correctly recognized by the model as Watch TV, then the sample rate is included as T_P.Similarly if the action predicted is not actually Watch TV then if the prediction is not Watch TV then those samples are considered as T_N.Literally if Watch TV is predicted as cook the it is rated under F_P and vice versa for F_N.So accuracy is the factor of calculating how far the model foretells the positive and false positive classes, Precision fully rates only the positive classes, Recall provides the difference in classifying false and true classes of the recognition. Finally the F1 score is used to measure the accuracy and it is the mean value and combination of precision and recall. These basic machine learning parameters are used to compare our proposed model with other existing models. The results obtained are compared with other models and its pros and cons are discussed in the below section.

Accuracy (A) = (T_P+T_N)/ (T_P+F_P+F_N+T_N)

Precision (P) = (T_P) / (T_P+F_P)

Recall (R) = (T_P) / (T_P+F_N)

F1 Score = (2 * T_P) / (2* T_P+F_P+F_N)

5.1 Results and Discussion

The results obtained by the proposed approach are recorded in the form a confusion matrix as shown in Fig.7. From the data obtained we can clearly say that most of the activities like dish clean, cook, sweep,sit,sleep,eat,watch television, table work are correctly predicted. The values recorded across rows and columns of the matrix clearly express the recognition rate of the above said activities. Few activities are not correctly predicted as per the expectation. For example the Enter and Leave are those categories which are not correctly predicted.The reason may be that those activities occur in the same place where the door sensor is placed. Hence the model misclassifies Enter as Leave and Leave as Enter. Sometimes watch TV and Enter are also misclassified because of the sensor placed in the dining room. But mostly other eight activities are properly classified with the average accuracy of around 98 percent. We have recorded all the considered activities as row versus column, which makes way for easily understanding the confusion matrix. The value in the grey area of the matrix in Fig.8 represents all the correctly classified classes, the True Positive values. To get the overall accuracy for all the selected activities we calculated the average for all the activities .Table 3 provides the overview of all the assessed parameters for every activity.

TABLE 3.Predicted class wise accuracy of the proposed model

Activities	Accuracy	Precision	Recall	F1 Score
Dish clean	98.97	98.23	98.45	0.97
Cook	98.89	98.41	98.76	0.96
Sweep	98.95	98.62	98.55	0.98
Sit	98.98	98.71	98.69	0.98
Sleep	98.97	97.88	97.36	0.98
Eat	98.92	97.98	98.10	0.98
Watch TV	98.95	98.68	98.21	0.98
Table work	98.99	97.89	98.13	0.98
Enter	97.95	97.31	98.43	0.98
Leave	97.98	97.35	97.58	0.96
Mean Value	**98.76**	**98.11**	**98.19**	**0.98**

We compared our model with few of the existing models that used sensor data as input. The comparison values and its assessed parameters are as shown in Table 4.

TABLE 4. Performance comparison with other Models

S.No	Model	Accuracy [%]	F1-Score
1	Random Forest	83.63	0.83
2	KNN	77.24	0.76
3	SMO	54.70	0.50
4	SVM	72.39	0.71
5	CNN	91.98	0.98
6	RNN	90.34	0.90
7	DeepCNN +LSTM	92.36	0.91
8	**Proposed model**	**98.76**	**0.98**

The results obtained from the proposed approach was compared and analyzed with few of the available ML models. The reason for choosing ML models is that those models have outperformed well in recent times of research in human activity recognition and in line with our proposed method. And also the techniques and parameters used with the proposed approach are similar with the characteristics of ML models. Support Vector Machine, Random Forest and K

Nearest Neighbor ML algorithms are selected for our comparison.

Those ML models performances are measured by conducting grid search and random search with distinct hyperparameters.Table.5 contains the hyperparameter optimization and the optimal hyperparameter part with the accuracy obtained for each of the search for all three ML models.

5.2 Computational Complexities

Mostly inventions or ideas introduced in computing technology will be measured using three important parameters specifically in Artificial Intelligence. We analyzed our proposed model with those three parameters (MET) viz., 1.Memory usage (M), 2.Execution time (E), 3.Time taken to predict (T).

Memory usage (M)

Memory consumption is an important metric in terms of computers. We compared our approach with other DL models, as shown in Table CNN, RNN and DeepCNN with LSTM has consumed more memory even with less trainable parameters.CNN consumes around 92MB of memory while execution whereas 89MB and 76MB are utilized by RNN and DeepCNN respectively. Our proposed model utilized only 36MB of memory which is best in performance of the DL models. Obviously with ML models Random Forest has consumed similar amount of memory as CNN model since all the nodes of the tree has to be loaded into the memory.SVM consumed around 84MB of memory due to its structure and framework.

Execution time (E)

All models including ML and DL took more time to classify the given input image when compared with our proposed approach. The training time of the CNN model is recorded around 34ms for 235387 as trainable parameters.RNN took reduced time than CNN as 22ms for 4381042 trainable parameters whereas DeepCNN with LSTM had achieved better performance than CNN and RNN with 14ms execution time for 510889 trainable parameters. Finally our proposed CNN with ABLSTM had outperformed all other models in terms of execution time with 0.75ms for 861712 trainable parameters.

Time to predict (T)

As the best of the result the prediction time of our proposed model is significantly good, 0.89ms, when compared with other models. Prediction time is in line with the execution time. From the results obtained our approach has better performance in terms of memory, execution time and time taken to predict. The comparison of ML and DL models with proposed approach is depicted in Table.5 and Fig.8

TABLE 5.Computational complexities. Existing Vs proposed models

Models	**Memory**	**Execution time**	**Time to predict**	**Accuracy**
Random Forest	94 MB	79ms	82.54ms	83.63%
SVM	90 MB	84ms	98.33ms	77.24%
KNN	91 MB	82ms	91.48ms	72.39%
CNN	92 MB	34ms	56.31ms	91.98%
RNN	89 MB	22ms	21.43ms	90.34%
DeepCNN	76 MB	14ms	12.58ms	92.36%
Proposed Model	**36 MB**	**0.75ms**	**0.89ms**	**98.76%**

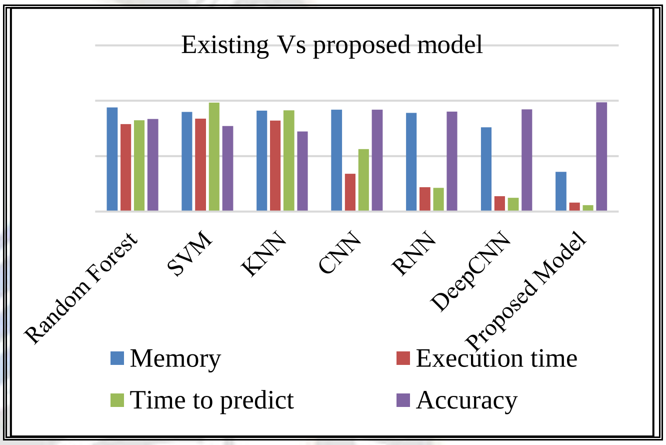


Figure 8. Existing Vs proposed model

VI. Conclusion

Due to the rapid growth of technology especially in the field of artificial intelligence human intervention has vanished. Recently many machine learning models are proposed in making the human interaction with the real world. More specifically in monitoring the human activities has been made more simple and effective. The main reason behind monitoring human activities is to bring higher improvement in the field of defense, smart homes, automated vehicles, elderly people monitoring etc. Even though many methods are proposed for more than a decade only few have proved its excellence in terms of applications, security, reliability and scalability. Many ML algorithms pave way to monitor human activities by utilizing models like CNN, LSTM, and RNN etc. Every model has its own pros and cons. For example maximum of the existing models uses video images, videos, smart phone data, and wearable sensor data for human activity recognition. The common limitations found in those methods are its privacy.

The person being monitored will always suffer from inconvenience that someone is watching. Our proposed method ensures higher accuracy at the same time it ensures the privacy for the person. We used Tulum data set, which is fully based on the sensor data. We implemented a stage by stage process in executing this experiment. Initially we collected the samples, labelled it, extracted the features and finally used CNN coupled with Bi directional LSTM to recognize the activity. At the beginning of this paper we mentioned that the proposal is mostly investigated only for elderly people who live alone. Their privacy is fully restored when our proposed method is utilized. From the results obtained we assure that in the field of human activity recognition our proposed model will definitely prove its outcome and excellence and would support upcoming researches in the area of elderly people tracking.

Conflict of Interest

Authors declare that they have no conflict of interests.

Data Availability

The datasets analyzed during the current study are available in http://casas.wsu.edu/datasets/

REFERENCES

[1]. Ahmad, Taati, and Alex Mihailidis, “Autonomous unobtrusive detection of mild cognitive impairment in older adults,” IEEE Transactions. Biomed. Eng., vol. 62, no. 5, 2015,pp. 1383–1394.

[2]. U. S. C. Euro monitor International, “Living alone statistics,” 2015.[Online]. Available: http://www.statisticbrain.com/living-alone-statistics/

[3]. M. Gochoo, “Device-free non-privacy invasive activity monitoring of elderly people in a smart house,” Ph.D. dissertation, Dept. Electr. Eng., Nat.Taipei Univ. Technol., Taipei, Taiwan, 2017,

[4]. WSU CASAS smart home project. D. Cook. “Learning setting-generalized activity models for smart spaces”. IEEE Intelligent Systems, 2011

[5]. Chen, Nugent, H. Wang, “A knowledge-driven approach for activity recognition in smart homes based on activity profiling”. Future Gener. Comput. Syst. 10, vol.6, 2020,pp 924–941. https://doi.org/10.1016/j10.031

[6]. Chunyu, Chen, Lisha and Xiaohui Peng “A novel random forests based class incremental learning method for activity recognition” Pattern Recog.2018,vol 78,pp2 77–290. DOI.org/10.1016/j.patcog.2018.01.025.

[7]. M. S. Hossain, “Patient state recognition system for healthcare using speech and facial expressions,” J. Med. System, vol. 40, no. 12,2016 p. 272.

[8]. Zhang, and Karunanithi.M," Assessment of activities of daily living Via a smart home environment”, Quantifying Quality of Life. Health Informatics. Springer, Cham. 2022,vol 8,pp342-358,https://doi.org/10.1007/978-3-030-94212-0_20

[9]. Natani, Sharma and Perumal, T. "Sequential neural networks for multi-resident activity recognition in ambient sensing smart homes". Applied Intelligence, vol 51, pp.6014–6028 (2021). https://doi.org/10.1007/s10489-020-02134-z

[10]. C. Hu,Chen,Peng,Yu, C. Gao and Lisha, "A novel feature incremental learning method for sensor-based activity recognition," in IEEE Transactions on Knowledge and Data Engineering, vol. 31, no. 6, pp. 1038-1050, 2019, DOIi: 10.1109/TKDE.2018.2855159.

[11]. Jurek, Nugent, Bi Y, Wu. "Clustering-based ensemble learning for activity recognition in smart homes". Sensors, 2014 Jul 10; vol.4 (7):12, pp285-304. DOI: 10.3390/s140712285. PMID: 25014095.

[12]. Wang, Jindong, Vincent Wenchen Zheng, Yiqiang Chen and Meiyu Huang. “Deep transfer learning for cross-domain activity recognition.” *ICCSE'18* (2018).

[13]. Rashidi, Cook DJ, Holder, Schmitter-Edgecombe,"Discovering activities to recognize and track in a smart environment". IEEE Transaction, Knowledge Data Engineering. 2011; vol.23 (4):pp527-539. DOI: 10.1109/TKDE.2010.148.

[14]. Baghezza, Bouchard, Bouzouane, Vallerand C. "From offline to real-time distributed activity recognition in wireless sensor networks for healthcare: A Review". Sensors. 2021; vol.21 (8):pp2786-2798. https://doi.org/10.3390/s21082786

[15]. Shah, Malik, Khatoon et al. "Human behavior classification using geometrical features of skeleton and Support Vector Machines. Computers, Materials & Continuation, 2018, vol 58. Pp535-553. Doi:10.32604/cmc.2019.07948.

[16]. Hao, Bouzouane, Gaboury, “Recognizing multi-resident activities in non-intrusive sensor-based smart homes by formal concept analysis”. Neuro computing 2018, vol 318, pp75–89, https://doi.org/10.1016/j.neucom.2018.08.033.

[17]. Alemdar, Ersoy “Multi-resident activity tracking and recognition in smart environments”. J. Ambient Intelligence and Humanized Computing. 2017, vol.8, pp513–529.

[18]. Xiong et al “Activity feature solving based on TF-IDF for activity recognition in smart homes”. Complexity 2019, https://doi.org/10.1155/2019/5245373

[19]. Bigham et al “Real-time crowd labelling for deployable activity recognition," in Proceedings of Conf. Computing Supported Cooperation, 2013, pp. 1203_1212

[20]. S. Zhang et al., "Cough trigger: Ear buds IMU based cough detection activator using an energy-efficient sensitivity-prioritized time series classifier," ICASSP 2022 IEEE Intl. Conf. on Acoustics, Speech and Sigl Proc. (ICASSP),pp. 1-5, DOI: 10.1109/ICASSP43922.2022.9746334.

[21]. Grais, Wierstorf, Ward, Plumbley, "Multi-Resolution fully convolutional neural networks for monaural audio source separation". LVA/ICA 2018. Lecture Notes in Comp. Sci, vol 10891. Springer, Cham. https://doi.org/10.1007/978-3-319-93764-9_32