Elderly Fall Detection using Lightweight Convolution Deep Learning Model

Neeraj Varshney[1]

[1]GLA University, Mathura.
neeraj.varshney@gla.ac.in[1]

Abstract: Old people, who are living alone at home face serious problem of Falls while moving from one place to another and sometime life threading also. In order to prevent this situation, several fall monitoring systems based on sensor data were proposed. However, there was an issue of misclassification to identify the fall as daily life activities and also routine activity as fall. Towards this end, a deep learning based model is proposed in this paper by using the data of heart rate, BP and sugar level to identify fall along with other daily life activities like walking, running jogging etc. For accurate identification of fall accidents, a publicly accessible data collection and a lightly weighted CNN model are used. The model reports proposed and 98.21 % precision.

Keywords: Deep learning, Fall detection, activity recognition, CNN, data collection

1. Introduction

Behavioral pattern of anybody can be easily identify through the various activities performed. As per the statistics [1], at least 103.5 million people are over age of 60 in India and many of such people are living alone at their home [2].One major concern is to monitor their daily activates, to avoid any kind of casualty with them like fall while moving from here to there or fall while trying to sit or get up or many more such situation [3-5].

Fall is one of the serious issue for the old people, especially those who are crossing the age of 65 or 70. Falls are commonly understand as "unconsciously down to the ground, floor or some other level down [6], with the exclusions being the deliberate change in position with respect to the wall, furniture and other accessories"[7,8].The major threat with the detection of fall is create the difference between fall and other routine activities because in case of sensor, few activities are similar to the fall like bending, sitting, jumping [9-11].

The common characteristic of the generally proposed solutions are placed the sensor on either on waist or/and chest because these locations are suitable for sensory system to detect fall and other ADL [12]. Fall and ADL both are the part of same bouquet because fall occur while performing the daily activities [13].

2. Literature review

During the last few decades, many researchers have tried to carry out the fall detection activity. Various factors are associated with fall and all the factors are known as risk factors. Truly, the event of fall involves interaction of various factors? Kangas et al. [14] proposed fall detection in which they attached Midsection, wrist, and head accelerometer of three various healthy people. They carried out three standardized, downward, backward, and lateral falls to a floor mattress. They research four algorithms with increasing complexity, Kangas1a and Kangas1d. The learning machine has an effective and precise fall-detection algorithm Hussain et al. [15]. They identified basic features like maximum, minimum and mean amplitude, kurtosis, skewness and achieved an 99.98%accuracywhile using the Support Vector Machine (SVM) classifier. Hakim et al. [16] for everyday activity classification, a proposed threshold algorithm focused on fall detection and supervised machine learning. They used four classification algorithms such as SVM, decision-making arborescence, Nearest Neighbor Classifiers, and Discriminant Analysis. The proposed algorithm for fall-detection has also been developed and generated for the wearable fall-detection system. Fall event based on the threshold defined and used in two HMM forms to differentiate between fall and ADL. Rashidpour et al. [17] introduced a method to identification of fall using data collect through the motion sensor deploy in the mobile phone due to the ease of access and application.

3. Proposed model

In this Convolution deep learning model, the data received from the sensors are first divided into time-series segments which are the same size as the window frame. The CNN model contains a total of three convolutional layers with dropout layers for regularization and a dense layer. Each convolution layer of model is trained in a similar fashion. Figure 1 shows the Architecture of CNN framework.
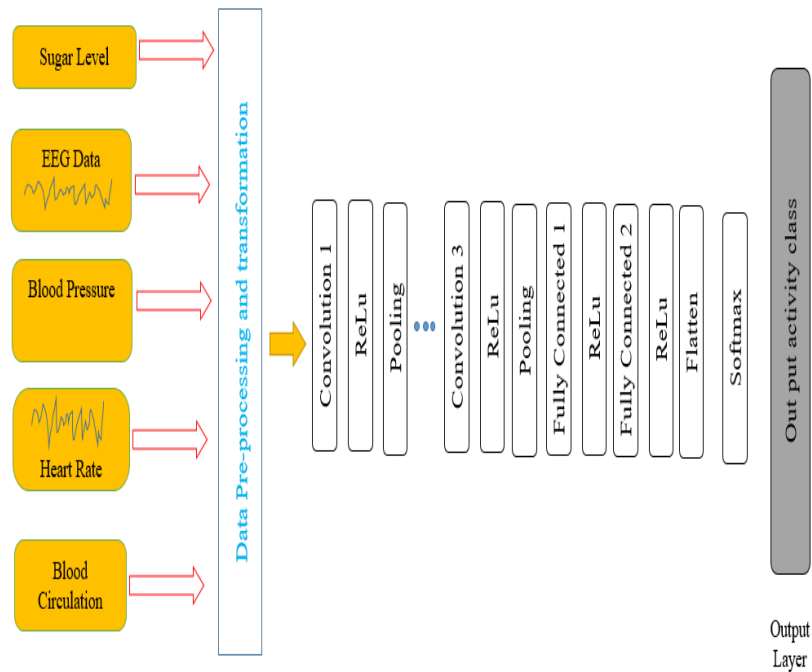
Figure 1. Architecture of CNN framework

Now the whole five input data preprocessed and provide as an input to the CNN model. A batch of segments, each segment sized 1×20, was stored in a 5 dimensions' tensor. Further these segments are converted into 80% training data and rest 20% for testing data. We cheese learning rate of 0.001 and train the CNN model. 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.

Electroencephalogram (EEG) signals has a significant part to play in the diagnosis of the neurological ailments like epileptic seizures besides even neurophysiologists basically rely on this EEG signals. Approximately 1% of population worldwide is affected by epilepsy that has been regarded as a huge neurological ailment. The often happening spikes are helpful in defining the EEG signal that marks the start of epileptic seizures. Epiletic seizures classification is done into two kinds. One is focal epileptic and other is generalized epileptic seizure. The focal epileptic seizures mainly affect the cerebral hemisphere that depicts the symptoms in the respective areas that in turn takes a toll on the mental wellbeing. The latter one is involved with the whole brain that results in unconsciousness showing the symptoms of bilateral motor. These both kinds of epileptic seizures can belie on any human being regardless of age-group.

In this technical work, ICNN is greatly involved in seizure or no seizure detection for specified EEG signal. The network is fed by EEG signal as input for obtaining confidence of every signal as output. In fact, CNN consists of an input and an output layer, and also multiple hidden layers. A convolutional layer, pooling layers and fully connected layers constitutes different hidden layers in CNN. Convolution operation is performed for the input by means of Convolutional layer, whose output is given to following successive layer.

The one single neuron response is utilized for visual stimuli. Convolution networks comprise local or global pooling layers for merging neuron clusters outputs present in one layer into a distinct neuron existing in the following succeeding layer. The average value from each neurons cluster existing in the layer before is engaged by mean pooling. Each neuron in one layer is connected to every other neuron present in another layer using completely connected layers. CNN fundamental concept is identical to that of conventional multi-layer perceptron neural network.

The discussed ICNN includes input layer, convolutional layer, sub-sampling layer and classification layer. This discussed technique achieves obvious advantages for analyzing high-dimensional data. A parameter sharing scheme is greatly utilized through which number of parameters are diminished as well as regulated via convolutional layers. The architectural ICNN block diagram is illustrated in Fig 2
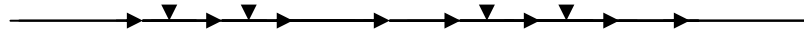
Figure 2. Proposed ICNN diagram

The data transformation into a unified form is performed for sending data to next layer appropriately when the input layer gets EEG signals from training samples. The initial parameters such as local receptive fields scale in addition to different filters are specified in this layer.

Convolution layer (Cx) performs the convolution of the input data and several layers referred as feature map including the convolution operation outputs from earlier layers generated. The primary objective is the identification of significant features and decrease in the computational complexity of the network. The convolution operation output equation is given below:

$$x_j^l = f(\sum_{i \in mj} x_i^{l-1} * k_{ij}^l + b_j^l) \qquad (1)$$

$$f(x)\frac{1}{1+e^{-x}} \qquad (2)$$

Where x notates convolution layer output value, k indicates kernel (or known as filter), l indicates number of output layer which is obtained by number of kernels, i refers to stride that kernel moves in every computation step, Mj indicates jth feature map produced by different kernels, b indicates bias and f signifies an activation function typically specified in the form of a sigmoid function specified in equation (2). Partaking the identical weights and bias is performed for neuron belonging to similar feature map, even though each output neuron has various receptive fields. By this way, parameter straining are hugely reduced.

An activation function is used after each convolutional layer. Activation function is one meant for mapping an output onto a set of inputs that imparts non-linearity to the network structure. The initial connection weights are assigned to all] feature values provided. Next, a new input pattern is used in addition output is given below

$$y(n) = f(\sum_{i=1}^{i=N} w_i(n)x_i(n) \qquad (3)$$

$$\text{Where } f(x) = \begin{cases} +1 & \text{if } x \geq 0 \\ -1 & \text{if } x < 0 \end{cases} \qquad (4)$$

Where n refers to iteration index

Connection weights are updated based on

$$w_i(n + 1) = w_i(n) + \eta(d(n) - y(n)x_i(n), \quad i = 1,2,..N \qquad (5)$$

Where η refers to gain factor

Then standard deviation is used

$$\sigma = \sqrt{\frac{1}{n}f_i(x_i - \overline{x})^2} \qquad (6)$$

These weighted features are input to the projected ECNN network and classification results with better accuracy are achieved. Each feature map from earlier convolution layer gets sub-sampled in this layer. As seen from Fig 2, Sx + 1sums informative features. The subsample technique is weighted summation computation or

considering maximum value in an n × n area of each feature map. The sub-sample layer output is expressed as below:

$$x_i^l = f(b_j^l \text{downsample}(x_i^{l-1})) + b_j^l \qquad (7)$$

Where xl indicates lth sub-sample layer output value, down sample states subsample function, specifies sub-sample function bias, f and b specifies activation function and bias respectively. The number of training parameters, filtering of noises are balanced via sub-sample layer and over-fitting is prevented in network.

Classification Layer: There is a consistent reduction in output feature maps size when the data is passed through different convolution layers and sub-sample layers. Just one single neuron emerges as a feature vector for each feature map in classification layer and there exist complete connection of vector with a classifier. EEG data segmentation is carried out using certain time window besides acquiring EEG signals plot series.

Range of the time window parameter is between 0.5 s and 10 s (0.5, 1, 2, 5, and 10 s). If time window size is less than or equal to 1 s, then EEG segmentation occurs with no overlap or the data gets segmented during every second with an overlapping happening with previous segments, for ex., a 9-s overlap for a 10-s segment.

CNN framework gets trained in a manner that all the segments lying fully inside the seizure states are used to model the seizure group. Random choice of segments eight times bigger in contradiction to seizure duration is performed from inter-ictal states and is specified on the basis of separation from the seizure state by greater than 1 hour, for modeling the non-seizure class in the course of CNN learning process.

Seizure or no seizure is detected with the help of the trained CNN for every 0.5 s for 0.5-s segment and each second for remaining segments. CNN classification performance is exemplified using seizure/ non-seizure labels with the help of CNN and is assigned through epileptologists. Seizure states are nothing but true positive and true negative rates that introduces segments including a seizure duration. Non-seizure states are nothing but the remaining seizure duration. The classification performance is assessed either by leave-one-out testing or pairwise testing. In the testing stage, the training and testing of the EEG data is done from the final subject that remains. In the case of pairwise testing, the data of an individual subject is taken for the building and testing from every subject separately.

Steps followed in ECNN

1. Procedure epileptic seizure or no seizure
2. For every input signal, showing EEG signal ∈ EEG dataset do
3. Transform input into sub layers
4. Identify EEG seizure features
5. Obtain more meaningful features applying ICNN
6. Carry out training and testing process for dataset provided
7. Replicate predetermined EEG signal label for every signal according to input dataset
8. Identify epileptic seizure or no seizure results with more accuracy

4. Dataset

We test our model on dataset publically available on kaggle. The dataset consists of six activities along with fall. It consists standing, running, falling, walking. Figure 3 and Figure 4 shows the Dataset Activity type and the Signal representation of fall activity.
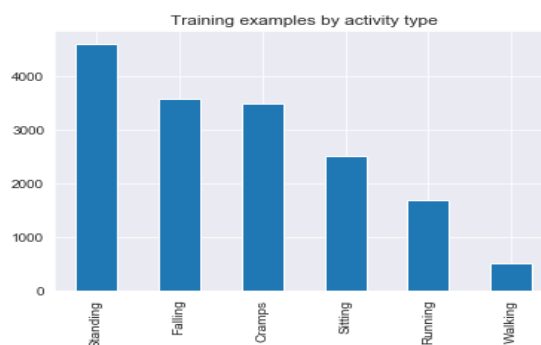
Figure 3. Dataset Activity type
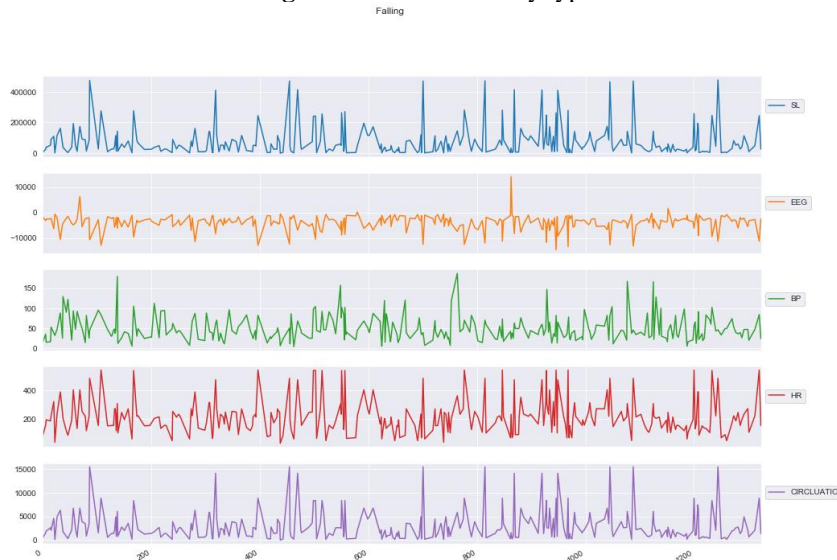
Figure 4. Signal representation of fall activity

Cramps, sitting. The dataset contains five different data values including EEG signal values, heart rate, sugar level, blood pressure and value for blood circulation.

5.   Result analysis and discussion

Figure 5 illustrates the accuracy and loss of the discussed model. We choose the fall detection dataset [3] which is publically available and achieve accuracy of 98.21%.
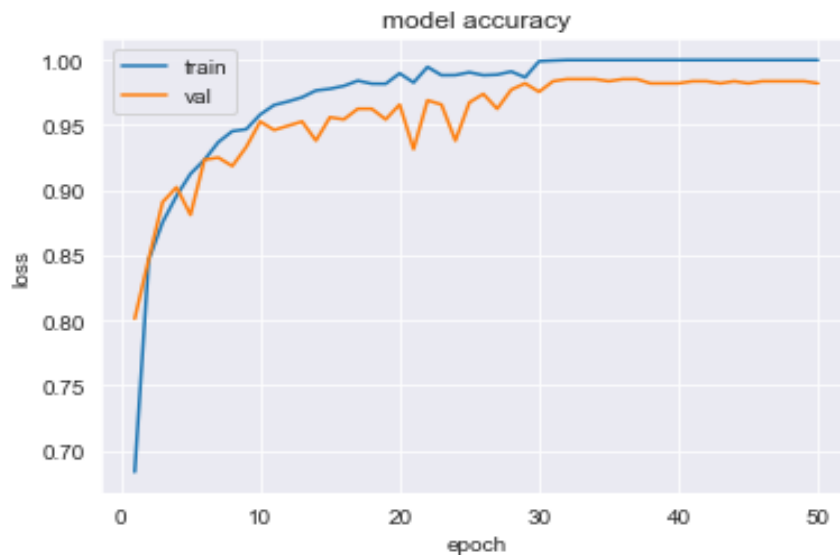
Figure 5. Model accuracy and loss chart

Proposed models is tests with different hyperparameters like number of epochs, number of hidden layers, learning rate, loss function, optimizer, activation function etc. Although learning rate and loss function are fixed throughout experiment. Keep learning rate 0.001 as well as Sparse categorical cross entropy loss function is used. Various possible combinations of epoch, hidden layers and optimizer analyze during the experiment and find that the optimum accuracy achieved with 50 epochs.
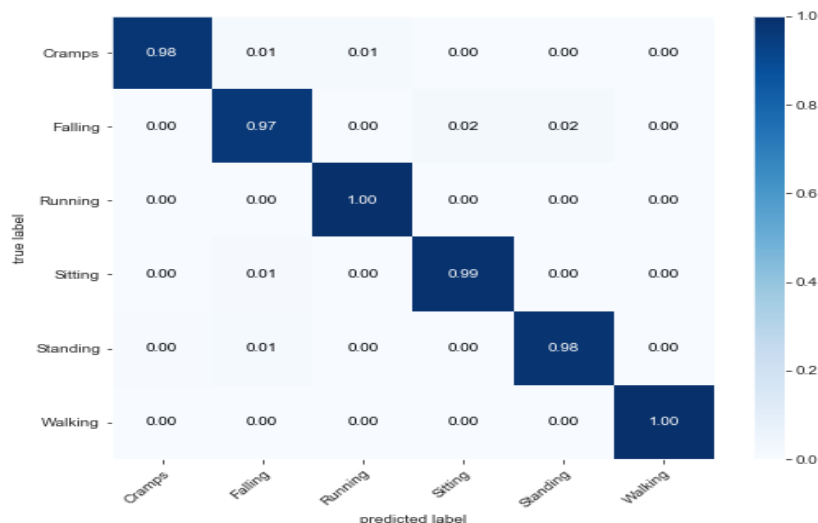
Figure 6. Confusion matrix

By analyzing the confusion matrices shown in Figure 6, a remarkable performance can be observed in the detection of fall. Figure 7 represent the various performance measures like precision, recall, f1-score and support achieved for various activities.

```
Accuracy Score :   0.9820846905537459
Report :
              precision    recall  f1-score   support

           0       0.99      0.98      0.98        85
           1       0.96      0.97      0.96       125
           2       0.96      1.00      0.98        24
           3       0.98      0.99      0.99       125
           4       0.99      0.98      0.99       230
           5       1.00      1.00      1.00        25

    accuracy                           0.98       614
   macro avg       0.98      0.99      0.98       614
weighted avg       0.98      0.98      0.98       614
```

Figure 7. Precision, recall, f-score and support metrics

6. Conclusion

A deep learning based model is proposed to detection of fall and other ADL using the heart rate, blood pressure sugar level and EEG data. Model achieve accuracy of 98.21% while train it upto 50 epochs. proposed model is a lightweight convolution neural network which consist of three convolution layers and dense and finally softmax layer to classify the fall activity. Results specified that the proposed network can distinguish fall from ADL with good accuracy.

References

1. https://www.dailymail.co.uk/indiahome/indianews/article-2785632/VISUAL-EDIT-India-s-senior-citizens-home-alone.html.
2. W. H. O. Ageing and L. C. Unit, "WHO global report on falls prevention in older age," World Heal. Organ., 2008.
3. Özdemir, Ahmet Turan, and Billur Barshan. "Detecting Falls with Wearable Sensors Using Machine Learning Techniques." Sensors (Basel, Switzerland) 14.6 (2014): 10691–10708. PMC. Web. 23 Apr. 2017.
4. Kangas M, Konttila A, Lindgren P, Winblad I, Jamsa T (2008) Comparison of low-complexity fall detection algorithms for body attached accelerometers. Gait & Posture 28: 285–291

5. Hussain F, Umair MB, Ehatisham-ul-Haq M, Pires IM, Valente T, Garcia NM, Pombo N. An Efficient Machine Learning-based Elderly Fall Detection Algorithm. arXiv preprint arXiv:1911.11976. Nov 272019.

6. Hakim A, Huq MS, Shanta S, Ibrahim BS. Smartphone based data mining for fall detection: analysis and design. Procedia Comput. Sci. 105:46-51, 2017 Jan 1.

7. Lim D, Park C, Kim NH, Kim SH, Yu YS. Fall-detection algorithm using 3-axis acceleration: combination with simple threshold and hidden Markov model. Journal of Applied Mathematics. 2014 Sep;2014.

8. Rashidpour M, Abdali-Mohammadi F, Fathi A. Fall detection using adaptive neuro-fuzzy inference system. Int. J.Multimed. Ubiquitous Eng.11:91-106 2016.

9. Noury N, Fleury A, Rumeau P, Bourke AK, Laighin GO, Rialle V, Lundy JE. Fall detection-principles and methods. In2007 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (pp. 1663-1666). IEEE 2007 Aug 22.

10. Zakaria, N.A.; Kuwae, Y.; Tamura, T.; Minato, K.; Kanaya, S. Quantitative analysis of fall riskusing TUG test. Comput. Meth. Biomech. Biomed. Eng. 2013, doi:10.1080/10255842.2013.805211.

11. O'Shea K, Nash R. An introduction to convolutional neural networks. arXiv preprint arXiv:1511.08458. Nov 262015.

12. Singh SP, Lay-Ekuakille A, Gangwar D, Sharma MK, Gupta S. Deep ConvLSTM with self-attention for human activity decoding using wearables. arXiv preprint arXiv:2005.00698. 2020 May 2.

13. Chaudhuri S, Thompson H, Demiris G. Fall detection devices and their use with older adults: a systematic review. Journal of geriatric physical therapy (2001). 37(4):178.Oct 2014;

14. Kangas, Maarit, et al. "Sensitivity and false alarm rate of a fall sensor in long-term fall detection in the elderly." Gerontology 61.1 (2015): 61-68.

15. Hussain, Faisal, et al. "Activity-aware fall detection and recognition based on wearable sensors." IEEE Sensors Journal 19.12 (2019): 4528-4536.

16. Hakim, Abdul, et al. "Smartphone based data mining for fall detection: analysis and design." Procedia Comput. Sci 105 (2017): 46-51.

17. Rashidpour, Roxana, et al. "A Comparative Analysis of Citizenship Education from Derrida and Foucault's Viewpoint: Suggestions for Iranian Educational Philosophers." Iranian Journal of Comparative Education 2.4 (2019): 407-424.

Biography

Neeraj Varshney is presently working as an Assistant Professor in CEA Department of GLA University Mathura. He is pursuing her Ph.D. in Activity Recognition and has done M. Tech, MCA degree. He has vast experience in the field of Teaching and Research. He has published many research papers in reputed International and National Journals. His areas of research include, Digital Image Processing and machine learning, sensor data processing.