CNN-LSTM Combined Network for IoT Enabled Fall Detection Applications

Jun Xu\textsuperscript{1,a}, Zunwen He\textsuperscript{1,b} and Yan Zhang\textsuperscript{1,c}

\textsuperscript{1}School of Information and Electronics, Beijing Institute of Technology, Beijing 100081, China

\textsuperscript{a}2220170443@bit.edu.cn; \textsuperscript{b}hezunwen@bit.edu.cn; \textsuperscript{c}zhangy@bit.edu.cn

Abstract. An accidental fall could do a great damage to the health of elderly. Failure to provide timely assistance after a fall may cause injury or even death. In this paper, a fall detection algorithm based on Convolutional Neural Network (CNN)-Long Short Term Memory (LSTM) combined network is proposed, which makes full use of the powerful feature extraction ability of CNN and the excellent time series processing ability of LSTM. Data required by the algorithm is only the resultant acceleration from a low cost three-axis acceleration sensor. The experimental results show that compared with the algorithms based on Support Vector Machine (SVM) and CNN, the proposed algorithm has higher detection accuracy with a small data volume, which is very suitable for Internet of Things (IoT) enabled fall detection applications.

1. Introduction

As the global aging population continues to increase, more and more attention has been paid to the damage of falls on the health of elderly. According to the data from World Health Organization, about 28–35\% of people over the age of 65 fall every year. For people over the age of 70, this number has increased to around 32–42\%. Therefore, an efficient and accurate algorithm of fall detection is of great social significance.

At present, a large number of algorithms have been applied to fall detection, these methods can be divided into two categories according to the deployment methods, i.e., visual-based detection method and wearable device-based detection method. The visual-based method\cite{1}-\cite{2} acquires scene information through the camera, without the need for the elderly to wear additional equipment. Thanks to the rich information acquired by the camera, this kind of approaches can achieve a very high precision. However, video images are greatly affected by environmental factors such as light, and the scope is more limited, so visual-based methods are more suitable for special occasions such as hospital wards. Compared with the visual-based methods, wearable device-based solutions are less affected by the environment and have a wider range of application scenarios. Threshold based detection algorithm\cite{3}-\cite{4} has been widely used in wearable devices, where one or more thresholds are used to determine various motion states of the human body, according to some specific combinations of actions to determine whether a fall event occurred. These methods put high requirements on the installation position and mounting angle of the sensor, thus have a relatively poor robustness. With the development of machine learning technology, SVM\cite{5}-\cite{6}, random forest and other machine learning algorithms are used for fall detection, and get better detection accuracy, but this kind of algorithms
still need to manually extract features for training, which requires a lot of prior knowledge, and the process of feature extraction is cumbersome, meanwhile the computation is large.

In recent years, deep learning has also been increasingly applied to fall detection. Fakhrulddin et al. (2017) [7] proposed an algorithm that directly converts acceleration sensor data into RGB images and classifies them by CNN network, simplifying the process of data preprocessing, with a final detection accuracy of 92.3%. Deep learning based methods need to transmit data to the server through network for processing because it’s difficult to be deployed directly on embedded devices. Although this solution may bring some delays, in such scenarios as nursing home, it facilitates the unified monitoring of the elderly's movement status in the park to ensure timely rescue in case of falls. However, the method in [7] still needs a large amount of data to detect a fall, which will consume a lot of network bandwidth in an IoT system.

In order to solve the problems above, we propose a fall detection algorithm based on CNN-LSTM combined network, which uses neural network to automatically extract features and classify them, avoiding the cumbersome preprocessing process. The raw data from three-axis accelerometer is firstly calculated as the resultant acceleration value to reduce data usage, and then sent to the CNN-LSTM network for classification. It only needs a small amount of data to realize the end-to-end fall detection application, which makes it very suitable for IoT applications. The accuracy evaluated on the MobiAct dataset [8] is 98.98%.

The rest of the paper is organized as follows. Section 2 describes the scheme and network structure used in this study. Section 3 mainly introduces the settings of dataset and experimental parameters. Section 4 shows the results of this study and compares them with other works. The conclusions and future works are drawn in Section 5.

2. CNN-LSTM network architecture

In this study, a CNN-LSTM network is proposed. The network architecture is shown in Figure 1.

The network input layer receives the sequence of the resultant acceleration values from the sensors. Suppose $a_x$, $a_y$, and $a_z$ are the value obtained from 3-axis accelerometer, then the resultant acceleration $a_r$ can be calculated by equation:

$$ a_r = \sqrt{a_x^2 + a_y^2 + a_z^2} \quad (1) $$

In the CNN part, each convolution layer in the network consists of a one-dimensional convolution, a linear activation layer, and a maximum pooling layer. Moreover, batch normalization layer [9] is also applied to the network to optimize the convergence speed and model accuracy. The essence of batch normalization is to use the optimization to change the size of the variance and the mean position, so that the new distribution is more in line with the true distribution of the data, and the nonlinear expression ability of the model is guaranteed.
Two LSTM layers are used to ensure that the model can learn enough features. Highly abstract features are generated by the CNN and LSTM layers, and two fully connected layers with a softmax layer are used to map the learned features to the sample marker space.

The loss function in this network is given as follow.

$$\text{Cost} = -\frac{1}{N} \sum_{i=0}^{N-1} [y_i \ln a_i + (1 - y_i) \ln (1 - a_i)] + \lambda \omega_2$$  \hspace{1cm} (2)

where $N$ is the number of categories, $y_i$ is the expected output and $a_i$ is the actual output of neurons. A L2 regularization term is used to improve generalization ability of the network, $\lambda$ is the proportional parameter of the regularization term, and $\omega$ represents the weight of the network.

2.1. 1-D CNN description

CNN has been widely used in many areas for its powerful feature extraction capabilities. Two-dimensional and three-dimensional CNNs play an important role in the fields of image and video processing. Meanwhile, one-dimensional CNNs are best suited for learning the feature in 1-D time series.

In this study, two 1-D convolutional layers are used to extract multiple features in acceleration sequence data. These convolutional layers use the Leaky ReLu activation function to avoid the problem of gradient disappearance. The Leaky ReLu is given as:

$$f(x) = \begin{cases} x, & x > 0 \\ \alpha x, & x \leq 0 \end{cases}$$  \hspace{1cm} (3)

The Leaky ReLu activation function was first proposed in the acoustic model, which has a better performance for 1-D time series. Each layer of convolution is followed by a maximum pooling layer of length 2, which reduces the size of the feature vector, so that the subsequent network obtains higher level abstract feature information.

2.2. LSTM Network description

RNN plays an important role in the processing of speech signals and time series. However, it has a poor ability to deal with the long-term dependency problems, which may lead to problems such as gradient exploding. LSTM solves the long-term dependency problems of the sequence by adding a memory gate to the RNN, resulting in better results. The structure of LSTM is shown in Figure 2.

![Figure 2. Structure of LSTM.](image)

The output response $h_t$ of the nerve cell at time $t$ is determined by the input $x_t$ at that moment and the output response $h_{t-1}$ at the previous moment:

$$h_t = \theta(W_{xi}x_t + W_{hh}h_{t-1} + b_h)$$  \hspace{1cm} (4)

where $\theta(\cdot)$ represents the activation function, $b_h$ represents the offset vector, $W_{xi}$ represents the weight matrix between the input and the neuron, and $W_{hh}$ represents the cyclic weight matrix of the inner self-loop of the neuron. The operations of the various parts of the neuron at time $t$ are as follows.

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i)$$  \hspace{1cm} (5)
where \( \sigma(\cdot) \) denotes the sigmoid function: 
\[
\sigma(x) = \frac{1}{1 + e^{-x}}.
\]
\( W \) denotes a weight matrix between any two units, and \( b \) denotes the offset of a certain unit.

### 3. Experiment setup

#### 3.1. Dataset

In this section, the feasibility and performance of the proposed approach are evaluated on the public MobiAct dataset. The MobiAct dataset uses the built-in LSM330DLC inertial sensor module from Samsung Galaxy S3 smartphone to collect data from three-axis accelerometer, gyroscope and direction meter at a frequency of 200 Hz.

In this experiment, the data set is down sampled to obtain the 50 Hz sampling rate signal, three-axis acceleration data is used to calculate the resultant acceleration. Other data from the gyroscope and direction meter is ignored to make sure that the model uses the least amount of data. Fall behaviour and other six kinds of Activities of Daily Livings (ADL) are selected to train the model. Figure 3 shows the acceleration curve of part of the behaviours.

![Acceleration curve](image)

**Figure 3.** Acceleration curves of different ADLs: (a) the acceleration curve of falling, (b) the acceleration curve of walking, and (c) the acceleration curve of jogging.

Since the fall process lasts about 2 to 4 seconds, we intercept 4 seconds from the moment the fall occurs, i.e., 200 sampling points. The experimental data set is divided into train set (80%) and test set (20%), a total of 5,891 train samples and 1,473 test samples are obtained. During the training process, 15% of the train set is used as validation set to verify the generalization ability of the model.

#### 3.2. Parameter setup

In this experiment, we use the Keras deep learning framework to build the CNN-LSTM network model. The network parameters are shown in Table 1. We also train a pure LSTM and a 1-D CNN with the same parameters to make comparisons.

| Parameter   | Value |
|-------------|-------|
| Epoch       | 300   |
| Optimizer   | Adam  |
| Batch size  | 32    |
| Learning rate | 0.001 |
4. Results and Discussion

We have tested several network structures on MobiAct dataset. The accuracy curve of the training process is shown in Figure 4. The experimental results show that the accuracy of CNN-LSTM network is higher than that of 1D CNN and LSTM. In addition, as shown in the results, the convergence speed of the CNN-based network is much faster than that of the LSTM network.

![Figure 4. Accuracy curve of the training process.](image)

![Figure 5. Confusion matrix.](image)

The test results on CNN-LSTM network are shown as a confusion matrix in figure 5. It is shown that the CNN-LSTM achieves a classification accuracy of 98.98% in seven types of daily behaviours. As for the fall behaviour, the detection precision and specificity are 98.61% and 99.76%, respectively.

Meanwhile, we compare the results with several other algorithms evaluated on MobiAct dataset. We compare the number of categories, the sampling frequency and the accuracy of ADL classification. It should be noted that some of the algorithms are binary classification method, so the classification accuracy is consistent with the precision of fall detection.

In particular, to evaluate the complexity of the corresponding solutions, the number of sampling points used per seconds and the required sensors are also included in the comparison range to assess whether the model is suitable for IoT applications. The results of each method are shown in table 2.

| Study               | categories | sampling rate | Method   | Classification accuracy | Sensors                      | Points per seconds |
|---------------------|------------|---------------|----------|-------------------------|------------------------------|-------------------|
| Bayat et al. 2014 [5] | 4          | 100 Hz        | SVM      | 91.15%                  | Accelerometer                | -                 |
| Ivan et al. 2018 [10]| 5          | 100 Hz        | DNN      | 89.51%                  | accelerometer magnetometer gyroscope | -                 |
| Xiang et al. 2017 [7]| 2          | 50 Hz         | CNN      | 92.30%                  | Accelerometer                | 150               |
| Ours                | 7          | 50 Hz         | CNN-LSTM | 98.98%                  | Accelerometer                | 50                |

It can be seen from the result that the proposed algorithm achieves a higher precision than most other algorithms. Among the listed deep learning based algorithms, CNN-LSTM requires a least amount of data and sensors, which makes it possible for IoT applications.

5. Conclusion

In this paper, we proposed a fall detection algorithm based on CNN-LSTM network. The average classification accuracy of 7 types of ADLs including fall on MobiAct reaches 98.98%. And the
detection precision and specificity for fall detection are 98.61% and 99.76%. The method does not need to perform artificial feature extraction, the detection results can be directly obtained from the raw data from the low cost three-axis accelerometer, thereby realizing end-to-end application.

It is worth noting that the algorithm requires only the resultant acceleration rather than 3 axis acceleration data, which means a very low data volume. In the following works, the CNN-LSTM network will be deployed on a gateway of an IoT system to achieve an AI-enabled IoT (AIoT) application, which will not only bring more convenience to places such as nursing home but also collect large amounts of measured data for academic research.

Acknowledgement
This work was supported in part by National Natural Science Foundation of China (No. 61871035, 61201192), National High Technology Research and Development Program of China (No. 2015AA01A706).

References
[1] Anishchenko L 2018 Machine learning in video surveillance for fall detection Ural Symposium on Biomedical Engineering, Radioelectronics and Information Technology (USBEREIT) pp 99-102
[2] Ge C, Gu I Y and Yang J 2017 Human fall detection using segment-level cnn features and sparse dictionary learning IEEE 27th Int. Workshop on Machine Learning for Signal Processing (MLSP) pp 1-6
[3] Razum D, Seketa G, Vugrin J and Lackovic I 2018 Optimal threshold selection for threshold-based fall detection algorithms with multiple features 41st Int. Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO) pp 1513-6
[4] Guo H W, Hsieh Y T, Huang Y S, Chien J C, Haraikawa K and Shieh J S 2015 A threshold-based algorithm of fall detection using a wearable device with tri-axial accelerometer and gyroscope Int. Conf. on Intelligent Informatics and Biomedical Sciences (ICIIBMS) pp 54-7
[5] Bayat A, Pomplun M and Tran D A 2014 A Study on Human Activity Recognition Using Accelerometer Data from Smartphones Procedia Computer Science 34 450-7
[6] Wang R, Zhang Y, Dong L, Lu J, Zhang Z and He X 2015 Fall detection algorithm for the elderly based on human characteristic matrix and SVM 15th Int. Conf. on Control, Automation and Systems (ICCAS) pp 1190-5
[7] Fakhruulddin A H, Fei X and Li H 2017 Convolutional neural networks (CNN) based human fall detection on Body Sensor Networks (BSN) sensor data 4th Int. Conf. on Systems and Informatics (ICSIAI) pp 1461-5
[8] Vavoulas G, Chatzaki C, Malliotakis T, Pedidiitis T and Tsiknakis M 2016 The MobiAct Dataset: Recognition of Activities of Daily Living using Smartphones Proc. of the 2nd Int. Conf. on Information and Communication Technologies for Ageing Well and e-Health (ICT4AGEWE) pp 143-51
[9] Ioffe S and Szegedy C 2015 Batch normalization: accelerating deep network training by reducing internal covariate shift Proc. of the 32nd Int. Conf. on Machine Learning (ICML) vol 37 (Lille, France: JMLR.org) pp 448-56
[10] Pires I M, Garcia N M, Pombo N, Flórez-Revuelta F, Spinsante S and Teixeira M C 2018 Identification of activities of daily living through data fusion on motion and magnetic sensors embedded on mobile devices Pervasive and Mobile Computing 47 78-93