Human Motion State Recognition Based on Multi-input ConvLSTM

Zhang yiming 1.a, Wu liuai 1.b
1 School of Electronic and Information Engineering, Lanzhou Jiaotong University, Lanzhou, China
a540364590@qq.com, bkingzyguang@163.com

Abstract. The human body generates acceleration signals during movement. After collecting and processing this signal, the movement state of the human body can be analyzed and the behavior of the human body can be judged. Human motion state recognition has a wide range of applications in fields such as health monitoring, somatosensory games, and user social behavior analysis. In this paper, the UCI data set collected by mobile phone sensors is used to build the model using convolutional long and short-term memory neural network (ConvLSTM) combined with multi-input CNN (Multi-head CNN), and combined with long and short-term memory neural network (LSTM) and convolutional memory Neural network (ConvLSTM) for comparative evaluation. The accuracy of the model in this paper reached 93.75%. Experimental results show that the new algorithm can more accurately classify and recognize the human motion state.

1. Introduction
With the improvement of the accuracy of mobile phone sensors and the development of wearable devices, the data in mobile phone sensors will be a new research field. The use of acceleration sensors and three-axis accelerometers in smart devices to recognize human motion status is also a current research hotspot. Tracking and identifying the motion state of the human body can provide users with health advice and exercise advice. Jennifer R. Kwapisz [1] and others collected accelerometer data of 29 users' daily activities, constructed and processed the data set. Davide Anguita [2] et al. recorded data from the smartphone sensors placed on the waist by 30 subjects engaged in daily activities, and used support vector machines (SVM) to recognize and classify actions. Davide Anguita [3] Use the data in the smartphone to measure the body signals of the human body (exercise state, position, etc.) for auxiliary applications for elderly care. Hache[4] and others established a model for predicting complex activities, which can identify simple activities in complex activities. Jeff Donahue et al. [5] proposed the CNN-LSTM model in 2014, which is an LSTM that uses CNN as the front end and uses this model to generate text descriptions of images. Use CNN to perform and train on image classification, and finally obtain a feature extractor that can be used for subtitle generation. In 2015, Xingjian Shi [8] and others proposed the ConvLSTM model on the basis of FC-LSTM, and applied it to the prediction of regional rainfall, which achieved good results. Since rainfall and human sensor data have space and time dimensions, and the feature attributes of the data are similar, we tried to use the ConvLSTM model to detect human motion state, and achieved good results, and improved on the basis of Multi-ConvLSTM model. The data set uses the UCI-HAR (https://archive.ics.uci.edu/ml/datasets/human+activity+recognition+using+smartphones) data set of the machine learning library of the University of California, Irvine. Using the Multi-head CNN model to build a new model on the basis of ConvLSTM to improve the
accuracy of human motion recognition, and compared with the traditional CNN model, this model has a higher recognition rate and accuracy.

2. Data preparation

2.1. Data analysis

The UCI-HAR data set used the three-axis accelerometer and gyroscope data in the smartphone to collect data from 30 volunteers aged 19-48 at a frequency of 50 Hz. Volunteers performed two activity sequences including walking, going upstairs, going downstairs, sitting, standing, and lying down. The first time the smart phone was placed on the waist, the second time the volunteer placed the smart phone freely. The data is recorded through the developed mobile phone software, and the volunteers’ activities are recorded in video at the same time, and the sports category to which the data belongs is marked later. The data is processed by low-pass filter for noise reduction, and the signal is separated into human body acceleration and gravity signals, containing 10479 samples in total. Table 1 shows the proportions of the six exercise modes.

| ACTION          | THE AMOUNT OF DATA | PERCENTAGE    |
|-----------------|--------------------|---------------|
| WALK            | 1722               | 16.720%       |
| UPSTAIRS        | 1544               | 14.992%       |
| DOWNSTAIRS      | 1406               | 13.652%       |
| SIT             | 1777               | 17.254%       |
| STAND           | 1906               | 18.507%       |
| LIE DOWN        | 1944               | 19.876%       |

2.2. Sliding window

Extracting features from the window is a very effective method to maintain the separability of classes and to represent the features [7]. The sliding window is composed of window size and step size, and the window size is the amount of data processed at one time. The formula for calculating the size of the sliding window is:

\[ \text{window size} = 2^{\text{ceil}(2^{*f})} \]  

Where \( f \) is the sampling frequency of the sensor, the sampling frequency of the sensor in the data set is 50Hz, that is, 128 sampling points are a fixed window, and the time signal in the sliding window is sampled every 2.56 seconds, and there is 50% between two adjacent windows. The overlap rate.

![Figure 1 Sliding window example](image-url)
3. Model establishment

3.1. Multi-Head CNN Models

The multi-channel model can have multiple features as input, and each time variable can be input to the model as a separate input. CNN will use the kernel to learn features from each time series. Images generally have two dimensions. The CNN model uses two-dimensional convolution and uses the convolution kernel to move for image processing and analysis. While there is only one dimension in the time series, in theory, a one-dimensional convolution kernel should be used to move on the time series. The Multi-Head CNN (Multi-Head CNN[11]) model can use multiple single-channel CNN models, where each channel is processed with a time series, and the main features of the current time series can be obtained. The feature extraction of each channel is Independent, so as to get independent feature mapping for each time series.

3.2. ConvLSTM Model

Due to the dual attributes of human motion data in time and space, traditional LSTM models cannot predict and analyze spatial data, while CNN has better spatial analysis capabilities. In 2015, Xingjian Shi et al. based on the FC-LSTM model The ConvLSTM model was proposed to make accurate prediction of regional rainfall[9]. Due to the correlation between the spatial dimensions of regions and regions, the traditional LSTM algorithm can only predict the data in the time dimension, so the convolution method is used to change the connection between the input in the traditional LSTM and each gate. It has become a convolution method, and the connection between the states also uses a convolution operation, so that the spatial relationship is processed by convolution, and the data that is affected at the same time and space can be handled well. The calculation process of ConvLSTM is as follows.

\[
\begin{align*}
    i_t &= \sigma(W_i * X_t + W_{hi} * H_{t-1} + W_{ci} * C_{t-1} + b_i) \\
    f_t &= \sigma(W_f * X_t + W_{hf} * H_{t-1} + W_{cf} * C_{t-1} + b_f) \\
    C_t &= f_t * C_{t-1} + i_t * \tanh(H_{t-1} + b_f) \\
    o_t &= \sigma(W_o * X_t + W_{ho} * H_{t-1} + W_{co} * C_t + b_o) \\
    H_t &= o_t * \tanh(C_t)
\end{align*}
\]

Where \(\circ\) is matrix multiplication; \(*\) represents convolution multiplication; \(i\) is the input gate, responsible for the input of information to the memory unit; \(o\) is the output gate, which transmits the processed information to the next neuron; \(f\) is the forgetting gate, responsible for Information filtering takes important information to save, and non-important information is processed and forgotten; \(C\) is a memory unit, used to store information and control output; \(H\) is a hidden layer[11]

3.3. Multi-Head ConvLSTM Model

In the data collected by mobile phones, we divide human motion data into three-axis accelerometer data (Figure 2), gyroscope data (Figure 3), and three-axis acceleration data with gravity (Figure 4). The horizontal axis in the figure all represents time in seconds; the vertical axis in Figure 2 and Figure 4 represents acceleration, in m/s^2; the vertical axis in Figure 3 represents the angular velocity of the gyroscope, in rad/s. Due to the different sensor acquisition methods, the data model of the same action will be different. If the data is simultaneously trained through the neural network, the accuracy of the output result will be greatly biased. Therefore, based on the original ConvLSTM model, we propose a multi-input ConvLSTM model (Multi-Head ConvLSTM) for human body motion state monitoring.
First, through sliding window processing, the data is cut into a sample length of 32, a sample step size of 4, a total of 128 channels, and 3 types of data sets for input. Then initialize the neural network, and use ConvLSTM to build a convolutional neural network. The training algorithm of neural network is divided into two stages, forward propagation algorithm and back propagation algorithm.

The forward propagation algorithm uses the trainable kernel in each convolutional layer to filter the results of the previous layer, and then uses the activation function to output the feature map. The formula of the forward propagation algorithm is as follows:

\[
x'_j = f \left( \sum_{i \in M_j} x'_i \ast k'_i + b'_j \right)
\]  

(3)

Among them, a set of input mappings selected by us, is the offset on the mapping, \(k\) refers to the number of cores used, and "i" and "j" are the two parameters of the kernel. The function of the pooling
layer is to downsample, and collect peripheral neuron output through a kernel [12] The formula is as follows:

\[ x_j^\prime = f(\beta_j^\prime \text{down}(X_j^{(t-1)}) + b_j^\prime) \]  

(4)

In the formula, is the offset of multiplication, is the offset of addition, and "down" is the secondary sampling function using maximum pooling [13]. The fully connected layer is the hidden layer of the multilayer perceptron. The activation function of the output layer uses the multi-class softmax function [14], the formula is:

\[ x_j^\prime = f(\sum_{i\in M_j} x_j^\prime \ast k_j^\prime + b_j^\prime) \]  

(5)

Among them, Z is the metric vector of K, which is the input of the previous layer, and K is the dimension. In this paper, the value of K is 3. The model uses the back propagation function and updates the parameters through the stochastic gradient descent algorithm, making the output result closer to the expected value. And use the classification cross entropy loss function to optimize, the formula is as follows:

\[ \text{loss} = -\sum_{i=1}^{N} \hat{y}_{i1} \log y_{i1} + \hat{y}_{i2} \log y_{i2} + \ldots + \hat{y}_{in} \log y_{in} \]  

(6)

In the formula, N is the number of samples and m is the score. The classification cross-entropy loss function has a good effect in multi-classification problems. In order to prevent over-fitting problems, the Dropout function is also introduced into the model and set to 0.5. In the traditional CNN model, the parameters of each layer are calculated using the following equations:

\[ p = F_N \cdot K_S \cdot P_{PL} + \text{bias} \]  

(7)

Where F_N is the number of convolution kernels, K_S is the size of the convolution kernel, P_PL is the last dimension of the output vector of the previous layer, and bias has the same meaning as F_N. For the Multi-head CNN model, the parameters should be defined according to the input data of each sensor. After experimental testing, the three-axis sensor model in this article uses 32 convolution kernels, the size of the convolution kernel is (1,3); the gyroscope data uses 64 convolution kernels, and the size of the convolution kernel is (1,3); The three-axis accelerometer (including gravity) data model uses 64 convolution kernels, and the size of the convolution kernel is (1,7) to achieve the best results. Each convolution head processes the corresponding time series to obtain the feature mapping sequence of each time series, and finally connect them together. For the Multi-head CNN model, 5 dimensions of input are required. The input equation is as follows:

\[ \text{input}_\text{dim} = (\text{samples}, W_N, W_L, \text{channels}) \]  

(8)

Among them, samples is the sample size, W_N is the length of the sliding window, and W_L is the step length of the sliding window. Channels is the number of channels. Since the time series is univariate, channels =1. The structure of the Multi-head ConvLSTM model thus established is shown in Figure 5.
4. Experimental results and analysis

4.1. Experiment
Use 70% of the data in the dataset for model training, and 30% of the data for testing. The experimental environment uses the TensorFlow tool and the Python language to compile to build the Multi-head CNN model, and add the LSTM and ConvLSTM models for comparison to evaluate the accuracy of the new model. The experiment uses accuracy (Accuracy), recall (Recall) and F1 score (F1-score) as the evaluation indicators of the model. Accuracy is the ratio of the number of samples classified by the model to the total number of samples for a given test set data. The formula is as follows:

\[ \text{Accuracy} = \frac{TP + NP}{ALL} \]  

(9)

Among them, TP represents the positive sample that was retrieved, and TN represents the positive sample that was not retrieved, which is actually a negative sample, and ALL contains all the number of samples.

Recall rate (Recall) represents the correct recognition rate of the model to positive samples, the formula is as follows:

\[ \text{Recall} = \frac{TP}{TP + FN} \]  

(10)

Among them, FN is data that is actually a real sample but has not been monitored. F1 score is a comprehensive evaluation score for the model, using the harmonic average of precision and recall, the formula is as follows:

\[ F1 = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \]  

(11)

Among them, precision represents the accuracy of the model, and the formula is as follows:

\[ \text{Precision} = \frac{TP}{TP + FP} \]  

(12)

FP is the data that is actually a negative sample and is monitored as a positive sample. Table 2 compares the classification results based on the product neural network (CNN) and the convolutional memory neural network (ConvLSTM). It can be seen that the accuracy of Multi-head ConvLSTM is 100% higher than that of ConvLSTM when the same data input is used. 2%, 4% higher than LSTM. The recall rate is 2% higher than ConvLSTM and 1% higher than LSTM. The F1 score is 1% and 3%
higher than the other two models respectively. It can be seen that in the human body data set collected by mobile phones, the effect of Multi-head ConvLSTM on data classification is significantly better than the LSTM and ConvLSTM models.

| Relational extraction method | Recall | Precision | F_score |
|-----------------------------|--------|-----------|---------|
| Multi-head ConvLSTM         | 0.9273 | 0.9375    | 0.9315  |
| ConvLSTM                    | 0.9023 | 0.9134    | 0.9258  |
| LSTM                        | 0.9136 | 0.8923    | 0.9025  |

4.2. Analysis

Compared with the ConvLSTM model, the traditional LSTM model can only perform data classification judgments in the time dimension. The human body data has both time and space data at the same time, which has obvious disadvantages. It cannot be more effective in the data obtained by the sliding window. Features lead to lower accuracy. The ConvLSTM model is improved on the basis of the LSTM model. The window data of human activities is two-dimensional, and the image is used for convolution comparison and classification. Compared with the traditional LSTM model, there is a significant improvement, and the collection of human body data. It is multi-sensor and multi-angle collection, ConvLSTM can only perform fusion feature classification judgment on multi-sensor data, resulting in a decrease in accuracy. The Multi-head ConvLSTM model proposed in this paper is a good multi-channel training of data, using the most suitable model for different sensor data to input, and then fusing the models to solve the problem of ConvLSTM data fusion confusion and improve classification Accuracy of recognition.

5. conclusion

In this article, a new model Multi-head ConvLSTM is proposed to classify and process human sensor data. This method has a significant improvement compared to the ordinary LSTM method and the ConvLSTM method. This method is mainly to process the multi-sensor signals separately, then perform the feature fusion, and use the combination of convolution and LSTM to process the spatio-temporal data, and obtain good results. Although this model can process sensor signals in multiple channels, each input model still needs to be tested separately to achieve the best results. Therefore, the next step is to find a way to build models and merge features to reduce the workload of multiple experiments.

References

[1] Jennifer R. Kwapisz, Gary M. Weiss, Samuel A. Moore. Activity recognition using cell phone accelerometers. 2011, 12(2):74-82.
[2] Tang T., Zheng L., Weng S., Peng A., Zheng H. (2018) Human Activity Recognition with Smart Watch Based on H-SVM. In: Yen N., Hung J. (eds) Frontier Computing. FC 2016. Lecture Notes in Electrical Engineering, vol 422. Springer, Singapore.
[3] Anguita, D. [et al.]. Energy efficient smartphone-based activity recognition using fixed-point arithmetic. "Journal of universal computer science", 2013, vol. 19, núm. 9, p. 1295-1314.
[4] Mobility Change-of-State Detection Using a Smartphone-Based Approach - Hache, Lemaire, et al. – 2010
[5] Xingjian Shi, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-kin Wong, Wang-chun Woo, “Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting”arXiv:1506.04214
[6] Mikel Canizo, Isaac Triguero, Angel Conde, et al. Multi-head CNN–RNN for multi-time series anomaly detection: An industrial case study. 2019, 363:246-260.
[7] Yang, Jhun-Ying and Wang, Jeen-Shing and Chen, Yen-Ping. Using acceleration measurements for activity recognition: An effective learning algorithm for constructing neural classifiers .
[8] SHI X, CHEN Z, WANG H, et al. Convolutional LSTM network: a machine learning approach for precipitation now casting [C] // International Conference on Neural Information Processing Systems. Cambridge: MIT Press, 2015: 802-810.

[9] N. Srivastava, E. Mansimov, and R. Salakhutdinov. Unsupervised learning of video representations using lstms. In ICML, 2015.

[10] Hochreiter S, Schmidhuber J. Long Short-Term Memory [J]. Neural Computation, 1997, 9(8): 1735-1780.

[11] Mikel Canizo, Isaac Triguero, Angel Conde, Enrique Onieva, Multi-head CNN–RNN for multi-time series anomaly detection: An industrial case study, Neurocomputing, Volume 363, 2019, Pages 246-260, ISSN 0925-2312.

[12] A. Krizhevsky, Convolutional Deep Belief Networks on CIFAR-10.

[13] M. Riesenhuber and T. Poggio, “Hierarchical models of object recognition in cortex,” Nature Neuroscience, vol. 2, no. 11, pp. 1019–1025, 1999.

[14] C. M. Bishop, Pattern Recognition and Machine Learning, Springer, New York, NY, USA, 2006.