Multi-convLSTM neural network for sensor-based human activity recognition

Zili Li, Yixin Liu, Xuerong Guo, Ji Zhang
Department of Information Science and Engineering, Chongqing jiaotong University, Chongqing 400007, China
622180070009@mails.cqjtu.edu.cn
allenyplee@qq.com

Abstract. In recent years, human activity recognition (HAR) has attracted a lot of attention due to its wide application, such as indoor positioning and navigation. This paper proposes a MconvLSTM to construct a multi-unit deep network structure, which can effectively improve the accuracy of HAR. Firstly, the input data is dimensionally expanded. Secondly, multiple convLSTM module are used to input data from different sensors to achieve partial weight sharing. Multiple outputs are merged finally. The experimental results show that the partial weight sharing mechanism and dimension expansion effectively improve the extraction of single sensor features, aiming to improve the activity recognition rate. Using public UCI datasets for testing, the accuracy is significantly improved compared to traditional convLSTM network results.

1. Introduction

In recent years, with the development of artificial intelligence and sensor technology, indoor positioning and navigation have begun to receive widespread attention. The low signal availability of the Global Navigation Satellite System (GNSS) in indoor environments, other methods need to be used for mobile real-time positioning and navigation[2]. Consumers use smartphones as indoor positioning and navigation devices. These devices are equipped with various low-cost sensors. In order to reduce the positioning and navigation errors, pedestrian activity recognition[1-3] is an effective method.

Human activity recognition divided into optical device-based activity recognition and sensor-based activity recognition. Optical device-based includes video and image, sensor-based includes wearable devices[5], environmental sensors[6] and hybrid sensors[3]. Sensor-based human activity recognition is not restricted by fixed-point optical collection devices, and user privacy data can be properly saved. We focus on the human activities recognition using smartphones. Smartphones are highly flexible in the using, smartphones-based is more complex than fixed wrist, waist and leg sensors. This article mainly studies human activity recognition on fixed relative position.

The method based on wearable devices uses the Micro-Electro-Mechanical System (MEMS) Obtain acceleration and gyroscope data from multiple parts of the human body, and extract various types of features, including statistical[7], time domain[2], frequency domain[8], etc. Some works also use Principle Component Analysis (PCA)[9], Linear Discriminant Analysis (LDA)[9] and other methods to reduce dimensionality of features[2]. Finally, those features are served as inputs to train a PR model, such as Naive Bayes (NB), support vector machine (SVM)[8], decision tree (DT)[9], to make activity inference in real HAR tasks. However, compared to other wearable devices, activity
recognition based on smartphones is more susceptible to placement and posture, resulting in many hidden features that are difficult to detect and extract manually.

With the rapid development of deep learning in recent years, tremendous achievements have been made in computer vision, natural language processing, and speech recognition. Compared with the traditional manual feature extraction method, deep learning can reduce the workload of feature design, by constructing an end-to-end neural network to automatically learn and capture more high-level and meaningful features. In addition, the deep neural network structure is more suitable for unsupervised learning and incremental learning, and has better scalability. Deep learning is also applied to optical device-based activity recognition[11] and sensor-based activity recognition[10,12], including deep neural network (DNN), Convolutional neural network (CNN)[10], autoencoder, recurrent neural network (RNN)[11]. Ronao[13] proposed a deep convolutional neural network, which uses the one-dimensional time series signals of smartphone sensors to effectively distinguish walking, going upstairs and downstairs and other similar motions. Hannink[14] used a method based on deep convolutional neural networks to convert the abstract information provided by wearable sensors into context-sensitive expert features to distinguish gait. A general depth framework for activity recognition based on the convLSTM model[15], which improves the accuracy of activity recognition by combining two network models. Nils used the fANOVA framework to explore the effect of hyperparameters on convolution and cyclic models[16]. Yu[17] proposed a deep network architecture using residual bidirectional long-short-term memory network (Bi-LSTM), which improves the accuracy of distinguishing activities by connecting positive and negative time nodes. Some works have furthermore focused on lightweight deep learning model for HAR. Agarwal requires less computing power to achieve a good recognition effect[18]. Hassan[20] proposed an effective end-to-end deep neural network (DNN) model to identify human activities from time-sparse data signals.

The related work illustrates the effectiveness of deep learning in sensor-based human activity recognition. In this paper we propose a deep network structure based on MconvLSTM for real-time human activity recognition to create a reliable indoor positioning and navigation solution. The training process of deep neural network, CNN and LSTM is described in detail, and then we introduce a novel approach of the data expansion and MconvLSTM network construction method. In the course of further experiments, the processing methods of features in different network structures, the influence of data expansion on the recognition results, and the influence of the selection of the number of network layers and the size of the convolution kernel on the model are discussed.

Our contributions as follows: constructing network units according to sensor categories to improve the performance of existing deep learning-based activity recognition technologies. Proposing a method to extend the data dimension to extract more effective time-series features. The research goal is to develop a robust human activity recognition module aimed at enhancing the accuracy of indoor pedestrian positioning and navigation.

2. Background

Multi-layer perceptron (MLP) is an effective method for pattern classification, and is the foundation of deep learning, which processes computing information through a series of interconnected computing nodes. By dividing the computing nodes into different layers and using weights for association, nonlinear input operations are used to transform the input data to achieve a linearly separable goal. The definition of neuron is as follows:

\[ a^{l+1} = \sigma(W^l a^l + b^l) \]  

where \( a^l \) denotes the activation value of layer \( l \), and \( W^l \) denotes the weight matrix. Due to the independence of neurons, time series data (sensor data) cannot be processed well.

Recursive Neural Network (RNN) performs self-feedback by delaying activation of cell values, making it possible to learn the features of time series data. However, the features that are far apart cannot be correlated using RNN in the long-distance sequence. LSTM (Long Short Term Memory networks) effectively solves this problem. LSTM converts neurons into four network units, which are controlled by state gates: write, read, and reset. Convolutional Neural Networks (CNN) are usually used
to extract spatial features from the input sensor data. One-dimensional convolution kernel is used for outlier filtering and feature extraction.

![Diagram of models](image)

Fig.1 Models used in this work. From left to right: (a) Fully connected feed-forward network (b) LSTM network hidden layers containing LSTM cells (c) Convolutional networks that contain layers of convolutions, pooling and flatten.

In indoor activities, the goal of HAR is to determine the status of pedestrians to better provide location services for indoor pedestrians. The human activity recognition problem can be regarded as a pattern recognition problem, and the predefined activity set \( A \) is expressed as:

\[
A = \{a_i\}_{i=1}^{m}
\]  

(2)

where \( m \) is the type of activity set. At the same time, the sensor timing information \( S \) is expressed as

\[
s^k_t = \{d^1_t, d^2_t, d^3_t, \ldots, d^k_t\}, s^k_t \in S
\]  

(3)

The data is collected by a three-axis sensor, where \( k \) is the sensor dimension, and \( d^k_t \) represents the \( k \)-dimensional read data at time \( t \). Build model \( F \) and time sequence information \( S \) to determine the types of human activities, namely:

\[
\hat{A} = \{a_i\}_{i=1}^{m} = F(S), \hat{A} \in A
\]  

(4)

The goal of HAR is to build the model \( F \) by minimizing the discrepancy between predicted activity \( \hat{A} \) and the ground truth activity \( A \). Figure(2) shows a human activity recognition framework based on the MconvLSTM model. The collected sensor data is preprocessed, a dimension expansion algorithm is used to superimpose multiple dimensions of the input data to construct new dimension data. Then the corresponding ConvLSTM modules are constructed according to the number of sensor data, where each module contains three convolutional layers and three LSTM units, put the output of each convLSTM module in parallel into a full connection, and finally output the category through Softmax. In the initialization phase, the weights of each unit are randomly generated, and different convLSTM modules receive their respective sensor data input, and optimize the Loss to improve the classification accuracy of the network model.
3. The proposed method

3.1. Data preprocessing

To solve the problems of data heterogeneity and single data source between the acquisition sensors, the input data needs to be preprocessed. This paper proposes a method of expanding the sensor data dimension to increase the amount of input data and expand the data sample. The Inertial Measurement Unit (IMU) in a smartphone includes an accelerometer and a gyroscope. The combined measurement of multiple sensors can obtain a more accurate classification effect than a single sensor. A sensor is divided into three axes (x, y, z) to collect data. In order to expand the sequence of a single sensor, fully arranging the signals of the sensors in different directions, and numerically standardize the multi-directional sensors.

For input data S, standardize a single signal:

$$S_{\text{ij}}^k = \frac{S_{\text{ij}}}{\sqrt{\sum_{t=i}^{j} S_{t}^k \cdot S_{t}^k}} \quad (5)$$

where k is the type of sensor and S is the input sensor data. Put the standardized data into the newly constructed sequence. As the dimension increases, all extended sequences will be put into the new sequence.

**Algorithm 1 Sensor data expansion**

**Notations**: Sensor signal sequence (SI), Sensor signal sequence dimension (N_s)

**Input**: Sensor signal sequence data (S_i)

$$SI = list, N_s = S_i, length$$

For i = 1 to Ns + 1:

For j = 0 to Ns - i + 1:

If data dimension < 2:

Pass

Else:

Execute eq(5)

SL.append (S_{\text{ij}}^k)

End

End

**Output**: Extended sensor signal sequence (SI)

3.2. Network Architectures
The MconvLSTM sensor module includes three types of structures, namely the input layer, multiple convolution layers and the LSTM layers. The input layer inputs sensor data into the model in a sliding window. The convolutional layer is used to extract space features. Each convolutional layer is convolved by multiple convolution kernels. The activation or output of the convolutional connection is computed as follows:

\[
c_i^l = \sigma (\sum_{m=1}^{M} \psi_m^l \ast x_{i+m-1}^j + b_i^l)
\]

Where \( l \) is the number of layers, \( \sigma \) represents the activation function, in the actual calculation linear unit(ReLU) as the activation function selected, \( \psi_m^l \) is the m-th convolution kernel with feature \( j \), \( x \) is the input unit of length N output from the previous layer, \( b_i^l \) is the offset of the j-th feature.

improving the convergence speed of the network, the input vector \( x_i^j \) is put into standardization layer after convolution:

\[
n_{i}^l = \frac{c_i^l - \mu_c}{\sqrt{\sigma^2 + \epsilon}}
\]

Where \( \mu_c \) and \( \epsilon \) respectively represent the mean and standard deviation of the input vector , and then put into the pooling layer for downsampling after normalization.The pooling layer extracts features from \( n_i^l \), and we use the maxpooling to outputting the maximum value of the input vector as follows:

\[
p_i^l = \max (n_i^l)
\]

where \( R \) is the pooling size and \( T \) is the sliding step size. Multiple convolutional layers, standard layers, and pooling layers are stacked to form a deep neural grid structure. This network structure extracts features through layers, and more complex features can be extracted as the network layer deepens.

After the multi-layer convolutional network is constructed, it will be input into the LSTM unit. The LSTM unit updates the status of the temporal features, the length of multiple units ensures the validity of the temporality. Finally, the outputting of single sensor feature sequence is merged. Unlike the unit structure of convolutional neural networks that use kernels, LSTMs use memory cells to extend RNNs to simplify the temporal relationship in long-term. For a given input time series, LSTMs construct a switch channel mechanism that allows input information to pass through selectively to protect and control the state of the cell including write, read and reset. The cells are running directly on the entire chain which only have a few linear interactions. As input to the LSTM layer:

\[
h_{i}^l = \sigma (\sum_{t=1}^{t+h_{i-1}^l} o_{t+i}^l + b_i^l)
\]

where \( b_i^l \) represents the offset of the unit, \( x_i \) is the input vector to the unit \( t \), and \( o \) is the weight coefficient of each unit. The way of constructing the network with a single sequence is easy to ignore the heterogeneity of different sensor data, so improving the network structure blocks weight sharing between different sensor networks in order to achieve better recognition. The convLSTM network module is established based on the number of sensor types, and more sensors can be introduced to expand the network architecture and enhance the scalability of the network. After the output of the convLSTM module constructed by multiple different sensors, it is constructed in parallel to the fully connected layer:

\[
h_{i}^l = \sigma (h_{i}^l, h_{i}^l, \ldots, h_{i}^l)
\]

Finally, the Softmax classifier combines and identifies the activity categories and outputs the classification results. The classification is as follows:

\[
P(\eta|h) = \arg \max \exp \left( \frac{\eta_h}{L} \right)
\]

where \( \eta \) is the final output category, \( N \) is the total number of activity categories, and \( L \) is the number of layers. When the forward propagation is completed, the loss is calculated, back propagation according to the loss and modify the weights in the network. In order to speed up the network training and reduce the calculation cost, we use the Adam optimizer to update the weights:

\[
m_i = \beta_1 \ast m_{i-1} + (1 - \beta_1) dx
\]
\[ v_i = \beta_2 \cdot v_{i-1} + (1 - \beta_2)dx^2 \quad (13) \]
\[ W_i = W_{i-1} + \frac{-\alpha m}{\sqrt{\nu_i}} \quad (14) \]

where \( \beta \) is the exponential decay rate, \( \alpha \) is the learning rate, and \( W \) is the weight matrix. Repeat the feedforward and feedback until the stop criterion is met (the maximum number of epoch is reached, the loss drops below a certain threshold).

\[
\begin{align*}
\text{Algorithm 2MconvLSTM construction} \\
\text{Notations:} & \quad \text{weight matrix (W), Epoch(E)} \\
\text{Input:} & \quad \text{Sensor data } D_i = \{ K, I \} \\
\begin{align*}
1 & \quad \text{Gaussian distribution to initialize the } W \\
2 & \quad \text{Use eq(6) to calculate } D_i \\
3 & \quad \text{Standardize the output using eq(7)} \\
4 & \quad \text{Maximum pooling output using eq(8)} \\
5 & \quad \text{if } L \text{ is conv:} \\
& \quad \text{Goto(2)} \\
& \quad \text{Else if } L \text{ is LSTM:} \\
& \quad \text{Goto(6)} \\
6 & \quad \text{Extract temporal features using eq(9)} \\
7 & \quad \text{if } L \text{ is Concat:} \\
& \quad \text{Goto(8)} \\
& \quad \text{Else if } L \text{ is LSTM:} \\
& \quad \text{Goto(6)} \\
8 & \quad \text{Concatenate sensor signal using eq(10)} \\
9 & \quad \text{Calculate the maximum probability and get Loss using eq(11)} \\
10 & \quad \text{if Loss} < \mu \text{ or } E > \text{epoch:} \\
& \quad \text{End} \\
& \quad \text{Else:} \\
& \quad \text{Backpropagation modification weight using eq(12)-(14) and goto(2)} \\
\end{align*}
\end{align*}
\]

\text{Output: MconvLSTM Network structure}

4. Results and Discussions

We conducted several experiments to verify the effectiveness and usability of the MconvLSTM for human activity recognition. Two indicators were used to evaluate the experiment: (1) accuracy, defined as the number of correctly identified activity categories divided by the total number of activity samples. (2) The cost of training and recognition is defined as the time it takes to train the model and the time it takes to test each sample for recognition. All experiments were conducted on Colab, which was an Google's online computing platform equipped with a 2*2.30 GHz CPU and a GPU usable environment.

4.1. Dataset
Public Domain UCI Dataset (HARv2). The dataset comes from the University of California Irvine Data, which uses a smartphone to collect the human activity recognition. Consists of 30 volunteers aged 19-48 years old wearing a smartphone around their waists. Each person has six activities (WALKING, WALKING UPSTAIRS, WALKING DOWNSTAIRS, SITTING, STANDING, and LAYING). Using accelerometer and gyroscope in smartphone, the three-axis linear acceleration and the three-axis angular velocity obtained at a constant rate of 50 Hz, and trim it into a 128 time-step window. The obtained data set was randomly divided into two groups: 70% of the part were generated training data, and the other group was generated test data. The original features were used in our research: three-axis gravity acceleration from the accelerometer and three-axis body acceleration and three-axis angular velocity from the gyroscope.

4.2. Result

In order to verify the applicability and effectiveness of MconvLSTM, this article selected DCNN[13], convLSTM[15], and MconvLSTM for comparison. The convolution kernel size and pool size are set (maxSize, 1) and (2, 1), batch size is 256, dropout rate is 0.2. These parameters can make MconvLSTM achieve a good classification effect, and its classification accuracy rate is 97.11%, the accuracy of DCNN and convLSTM are 94.89% and 96.16%.

| Table 1 Networks Architectures |
|--------------------------------|
|                          | DCNN | ConvLSTM | MconvLSTM |
| input                   | (128, 9) | (128, 9) | 3@(128, 7) |
| CONV1                   | (128, 256) | (128, 256) | 3@(128, 256) |
| pooling                 | (64, 256) | (42, 128) | 3@(42, 128) |
| CONV2                   | (64, 64) | (42, 64) | 3@(42, 64) |
| pooling                 | (32, 64) | (14, 64) | 3@(14, 64) |
| CONV3                   | (32, 32) | (14, 32) | 3@(14, 32) |
| pooling                 | (16, 32) | (4, 32) | 3@(4, 32) |
| LSTM                    | (4, 4, 128) | (4, 128) | 3@(4, 128) |
| LSTM                    | (4, 128) | (4, 128) | 3@(128) |
| OUTPUT                  | (512) | (128) | (384) |
| output                  | (6) | (6) | (6) |

Comparing the accuracy of different network structures in the same training data, five cross-validation are used in the experiment to ensure the generalization of the model. Fig(5) show The confusion matrix, where the numbers 0–5 indicate WALKING, UPSTAIRS, DOWNSTAIRS, SITTING, STANDING, and LAYING. Most errors are concentrated in the two states of SITTING and STANDING, the above two activities are highly similar in sensor signals. Compared with DCNN,
convLSTM adds LSTM cells to extract the temporal feature of the sensor. Under the condition that different sensors maintain weight sharing, it is difficult for DCNN and convLSTM to distinguish effective results for similar signals. MconvLSTM constructs multiple convLSTM modules to block weight sharing between different sensors, and more effective feature extraction for a single sensor further reduces classification errors.

### 4.3. Effect of preprocessing

The initial signal preprocessing is to use the data expansion method to expand the single sensor three-axis signal sequence to 4 or 7 dimensions. Table(2) show the accuracy and training cost of the MconvLSTM after using the dimension expansion method. The input data after data expansion effectively represents the features of a single sensor in different dimensions, which improves the classification accuracy. However, high-dimensional signal sequence will increase the training cost of the model. The goal of this paper is to achieve a higher classification effect while saving the model training cost as much as possible.

### 4.4. Kernel size and LSTM

In the convolution layers, the size of the convolution kernel ensures the effectiveness of the network and affects the error of the model. The small-size convolution kernel has the ability to discover refined features, while the large-size convolution kernel has a broader view. The depth of the LSTM unit layer indicates the depth of extraction of the temporal series features of the input data. The deeper layer can capture more features, but too deep layers will increase the training cost of the model and are prone to overfitting. The maximum value and the intermediate value is selected as the size of the convolution kernel, and the number of layers of the LSTM layer is selected from 1 to 3 layers. The results are as follows.

| Table 2 The Accuracy and Training Cost of the MconvLSTM |
|-----------------|-----------------|-----------------|-----------------|
|                 | MconvLSTM (3*3) | MconvLSTM (3*4) | MconvLSTM (3*7) |
| LSTM layers     |                 |                 |                 |
| 3               | 95.14%          | 97.38%          | 97.65%          |
| 2               | 94.12%          | 97.08%          | 97.45%          |
| 1               | 94.09%          | 95.96%          | 96.65%          |
| Kernel size     |                 |                 |                 |
| Max             | 95.14%          | 97.38%          | 97.65%          |
| Max/2            | 94.69%          | 95.52%          | 96.50%          |
| LSTM layers     |                 |                 |                 |
| Max             | 520             | 694             | 784             |
| Max/2            | 303             | 652             | 758             |
| Median-Size     |                 |                 |                 |
| Max             | 520             | 694             | 784             |
| Max/2            | 303             | 652             | 758             |

The increase in the number of LSTM layers on the three models improves the extraction of hidden temporal features and thus improves the classification accuracy. Each additional layer of LSTM layers can effectively improve the accuracy rate by 0.5%~1%, also increased training costs. MconvLSTM performs better in max-size convolution kernels, because the data is split into multiple module inputs during the input stage, and median-size convolution kernels perform better in the unsplit input model (convLSTM), in local areas max-size extraction within can better extract features.

### 5. Conclusion
In this paper, we implement a data expansion method and build an MconvLSTM network based on multiple-sensor inputs, and we use the accelerometer and gyroscope on a smartphone to perform effective, accurate, and data-adaptive human activity recognition (HAR). Convolutional networks present a method to extract relevant and robust features automatically and adaptively without the need for advanced preprocessing, and LSTM extract the inherent temporal features from dependency of time-series signals, more complex features are extracted with every additional layer, but the training costs will increase and the differences in level of complexity between layers will be smaller.

The original signal from multiple sensor is expanded by the pre-processing algorithm to improve the input data dimension to increase the extraction of effective features, sensor data are input into different convLSTM module, which blocks weight sharing among different sensors and strengthens the local characteristics of each sensor. Also wider kernel sizes have also proved beneficial for MconvLSTM which can extract better extract features. Future work introduces a larger dataset to improve the LSTM structure for experiments, and considers building a more lightweight network model for deployment on the mobile.

Acknowledgments
This work was supported by the Chongqing Development and Reform Commission(CQDR).

References
[1] Jobanputra C, Bavishi J, Doshi N. Human Activity Recognition: A Survey[J]. Procedia Computer Science, 2019, 155:698-703.
[2] Elhousshi M, Georgy J, Nourredin A, et al. Motion Mode Recognition for Indoor Pedestrian Navigation Using Portable Devices[J]. Instrumentation and Measurement, IEEE Transactions on, 2016, 65(1):208-221.
[3] S. Saeedi, N. El-Sheimy, X. Zhao, and Z. Sayed, “Context aware mobile personal navigation services using multi-level sensor fusion,” in Proc. 24th Int. Tech. Meeting Satellite Division Inst. Navigat. (ION GNSS), Portland, OR, USA, 2011, pp. 1394–1403.
[4] Diane Cook, Kyle D. Feuz, Narayanan C. Krishnan. Transfer learning for activity recognition: a survey[J]. Knowledge & Information Systems, 36(3):537-556.
[5] Chen Y, Xue Y. A Deep Learning Approach to Human Activity Recognition Based on Single Accelerometer[C]// IEEE International Conference on Systems. IEEE, 2015.
[6] Wang A, Chen G, Shang C, et al. Human Activity Recognition in a Smart Home Environment with Stacked Denoising Autoencoders[M]// Web-Age Information Management. Springer International Publishing, 2016.
[7] Figo D, Diniz P C, Ferreira D R, et al. Preprocessing techniques for context recognition from accelerometer data[J]. Personal and Ubiquitous Computing, 2010, 14(7):645-662.
[8] He W, Guo Y, Gao C, et al. Recognition of human activities with wearable sensors[J]. EURASIP Journal on Advances in Signal Processing, 2012, 2012(1):108.
[9] Li Feng, Pan Jingkui. Human Motion Recognition Based on Triaxial Accelerometer[J]. Journal of Computer Research and Development, 2016, 53(3): 621-631.
[10] Wang J, Chen Y, Hao S, et al. Deep Learning for Sensor-based Activity Recognition: A Survey[J]. Pattern Recognition Letters, 2018
[11] Liu J, Shahroudy A, Xu D, et al. Spatio-Temporal LSTM with Trust Gates for 3D Human Action Recognition[J]. 2016.
[12] Chen K, Zhang D, Yao L, et al. Deep Learning for Sensor-based Human Activity Recognition: Overview, Challenges and Opportunities[J]. 2020.
[13] Ronao C A, Cho S B. Human activity recognition with smartphone sensors using deep learning neural networks[J]. Expert Systems with Applications, 2016, 59.
[14] Hanninki J, Kautz T, Pasluosta C F, et al. Sensor-Based Gait Parameter Extraction With Deep Convolutional Neural Networks[J]. 2016.
[15] Ordonez F J, Roggen D. Deep Convolutional and LSTM Recurrent Neural Networks for
Multimodal Wearable Activity Recognition[J]. Sensors, 2016, 16(1).

[16] Hammerla N, Halloran S, Ploetz T, et al. Deep, Convolutional, and Recurrent Models for Human Activity Recognition using Wearables[J]. Learning, 2016.

[17] Zhao Y, Yang R, Chevalier G, et al. Deep Residual Bidir-LSTM for Human Activity Recognition Using Wearable Sensors[J]. Mathematical Problems in Engineering, 2018(PT.17):1-13.

[18] Agarwal P, Alam M. A Lightweight Deep Learning Model for Human Activity Recognition on Edge Devices[J]. 2019.

[19] Tang Y, Teng Q, Zhang L, et al. Efficient convolutional neural networks with smaller filters for human activity recognition using wearable sensors[J]. arXiv, 2020.

[20] Hassan M M, Ullah S, Hossain M S, et al. An end-to-end deep learning model for human activity recognition from highly sparse body sensor data in Internet of Medical Things environment[J]. The Journal of Supercomputing, 2020(4).

[21] Hassan M M, Uddin M Z, Mohamed A, et al. A robust human activity recognition system using smartphone sensors and deep learning[J]. Future Generation Computer Systems, 2018: 307-313.

[22] Uddin M Z, Hassan M M. Activity Recognition for Cognitive Assistance Using Body Sensors Data and Deep Convolutional Neural Network[J]. IEEE Sensors Journal, 2019, 19(19): 8413-8419.