



Multi-model weighted voting method based on convolutional neural network for human activity recognition

Kangyue Ouyang¹ · Zhongliang Pan¹

Received: 22 November 2022 / Revised: 20 September 2023 / Accepted: 10 October 2023 /
Published online: 1 November 2023

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2023

Abstract

In recent years, human activity recognition (HAR) has been widely used in medical rehabilitation, smart home and other fields. Currently, the recognition performance highly depends on feature extraction and effective algorithm. On the one hand, traditional manual feature extraction and classification algorithms hinder the improvement of HAR. On the other hand, the latest deep learning technology can automatically process data and extract features, but it faces the problems of poor feature quality and information loss. In order to solve this problem, this paper proposes a new recognition method using only wearable sensor data. In the feature extraction stage, the axis information of each sensor is extracted separately into one-dimensional data, and information of all axes is integrated into a two-dimensional graph. Then, two deep convolutional neural network models are designed to train the features based on one-dimensional data and two-dimensional graph respectively. Finally, weighted voting method is used to get the classification results. Experiments have shown that the average recognition accuracy of the method in this paper is about 3% higher than that of other HAR deep neural network methods, which shown the advantage of the method in this paper in obtaining better recognition result with limited data.

Keywords Human activity recognition · Convolutional neural networks · Feature extraction · Two-dimensional graphs · Sensor data · Weighted voting

1 Introduction

Human activity recognition (HAR) has become an important part of many fields such as motion detection, medical services and smart home [1, 2]. In professional sport area, HAR can help athletes improve their daily training efficiency [3, 4]. In the field of medical service, HAR technology plays an important role in monitoring various chronic

✉ Zhongliang Pan
panzhongliang@m.scnu.edu.cn

¹ School of Electronics and Information Engineering, South China Normal University, Guangzhou 510006, China

diseases such as Parkinson's disease. For special diseases, early identification of patients with certain behavior patterns can even prevent serious consequences [5].

In general, HAR uses two following recognition modes: Video-based recognition mode and sensor-based recognition mode. Video-based recognition mode usually uses camera to obtain images or dynamic video [6]. Its major defects are personal privacy issue and high requirements in terms of device and calculation.

In contrast, sensor-based recognition is the current mainstream. This method can also be divided into two sub types based on sensor placement: wearable sensors placed on users and ambient sensors placed in surroundings. Both of these two types use sensors to obtain information about movement trajectory, body temperature, heart rate, etc. In addition, with the development of smart devices and mobile internet, a variety of sensors such as three-axis accelerometers and gyroscopes can be embedded in smart phones, smart watches and other devices, making the collection of human activity information faster and more efficient [7].

A standard HAR process based on wearable sensors has four main steps: collection of human activity data and preprocessing; feature extraction; model training; applying the model to HAR classification. For most of the HAR methods with data from small smart devices, the most challenging problems exist in feature extraction and model training.

For feature extraction, the difficulty is that human activities are similar but different, activities may have confusing patterns in some dimensions. For example, for running and walking, the main difference that accelerometers and gyroscopes can detect is forward speed, which makes it difficult to distinguish these two activities effectively. In addition, most feature extraction work requires manual feature design to extract effective information hidden in original signals [8]. Though manual methods work well in standard laboratory settings, lack of expertized experience which those methods highly rely on greatly restricts the quality of feature extraction. On the other hand, most studies directly use traditional machine learning algorithms to study the activity patterns in model training phase. This kind of approach usually requires structured data which is generated from massive manual work. Thus, the performance is limited by the quality of data processing and it's difficult to take the advantage of high-performance computing.

Instead of manual construction of one-dimensional features, a more popular and effective method is to transform the original sensor data into two-dimensional images [9], then use a specific deep convolutional neural network (CNN) or deep recursive neural network (RNN) to learn two-dimensional image features. Compared with manual feature extraction technology, deep learning technology based on deep neural network does not rely on expertized manual experience to design features. Instead, it learns high-dimensional features automatically. However, the original physical activities of human occur in three-dimensional space, this original information is reduced to one dimension after being recorded by wearable sensors, and then further transformed to high-dimensional features in deep learning model, this can cause the problem of information loss during dimensional transformation.

In view of the possible loss of information caused by the deep learning method, this paper focuses on the research to develop a weighted voting method that can significantly reduce information loss when one-dimensional sensor data is mapped to higher dimension, as shown in Fig. 1. The method in this paper extracted information from each individual axis of the wearable sensor as well as two-dimensional image generated from the axis-combined data. DCNN will be used to these two types of information respectively to train after information extraction. After training, the corresponding weight is

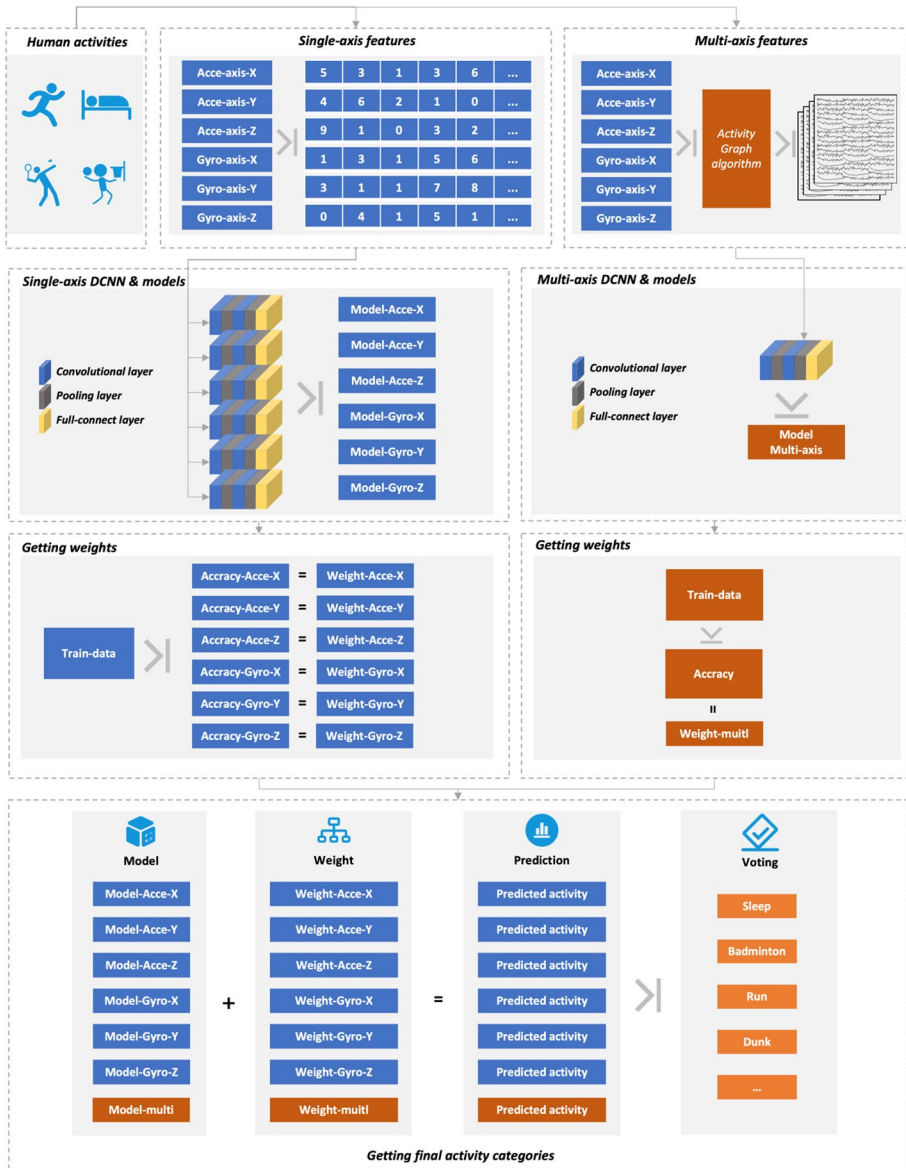


Fig. 1 Multi-model weighted voting framework

allocated according to the contribution of each model to the recognition effect, and the recognition result is finally obtained by weighted voting.

Our main contributions are summarized as follows:

1. A weighted voting method based on axis information from sensor is proposed. The features generated by the method include the information of each axis and the correlated

information between different axes, enriching the information representation of original activities.

2. Two kinds of deep convolutional neural networks with different hyperparameters are designed to process single axis features and correlated axes features respectively, which can carry out high-precision human activity recognition based on wearable sensors and improve the efficiency of classification.

A lot of experiments and corresponding analysis were performed on the following public data sets: USCHA [10], UCI [11] and UTD-MHAD [12]. The experiment results show that the method in this paper can improve the recognition accuracy by 3% on average compared with other deep learning methods. At the same time, the recognition accuracy of each sub-category in multi-class recognition scenario is also improved in different degrees.

The rest of this paper is organized as follows: Sec.2 shows the literature review, Sec.3 introduces the method proposed in this paper, Sec.4 introduces the experiment and the corresponding detail discussion, and Sec.5 is the conclusions.

2 Literature review

2.1 Sensing technology

The performance of HAR system depends on the sensor used. Sensors used in HAR mainly include ambient sensors and wearable sensors.

Common ambient sensor includes microphone, Binary sensor and Doppler radar. Ambient sensor shows the connection and interaction between human and the environment, and it can also capture changes of key factors in the environment. On the other hand, due to the disadvantages of high deployment cost, the scalable use of environmental sensors has been greatly affected [13]. Wearable sensor overcomes the disadvantages of expensive and bulky ambient sensor, and it is almost non-invasive to human. These advantages make wearable sensor widely used in HAR field. Among the numerous wearable sensors, three-axis accelerometer and gyroscope are the two that are most widely used and most influential.

Accelerometer The goal of the accelerometer is to measure the acceleration of the target. Such sensors are usually placed on body surface, such as arm [14] and waist [11]. The information collected by the accelerometer is divided into three axes: (X, Y and Z) corresponding to the three standard directions of the three-dimensional world. In an actual three-axis accelerometer data set, gravity components on each axis will be removed in data processing stage sometimes.

Gyroscope The goal of a gyroscope is to measure the angular velocity and direction of the target. Similar to three-axis accelerometer, the gyroscope is also placed on the surface of the subject's body and is usually embedded in a small smart platform such as a smart phone. Similarly, gyroscopes collect information along the X, Y, and Z axis.

2.2 Traditional methods for HAR

The traditional HAR method directly uses or adjusts the traditional machine learning algorithm to learn the extracted features. These methods rely on manual feature extraction to

ensure robustness. The commonly used feature extraction algorithms include Decision tree [15], Multilayer Perceptron [16], Support Vector Machines [11], Logistic regression [17] and Random forest [18].

In terms of specific work, Wang et al. [19] showed a new feature filtering mechanism for human activity recognition based on filtering and wrapper methods. They used this method to select the features with greater gain. Further, they combined this method with Naive Bayes and K-Nearest Neighbor classifiers to recognize human physical activities and achieved good results. In a similar way, Maurer et al. [15] also used the method of feature filtering to recognize human activities. Their practice is to make judgments based on the correlation between features. They assume that features should be independent from each other and highly related to specific activity categories. ProsseBgger and Bouchachia [20] use the improved decision tree to improve the recognition performance of residents' activities in buildings. They expanded the leaf nodes of the decision tree and allow multiple marking of the expanded leaf nodes. Ronao and Cho [21] designed a more complex hybrid framework for recognizing human activities. The advantage of their method is to use random forest algorithm to test the importance of each feature and then select the best feature set according to the importance. After obtaining the features, they use hidden Markov model to classify human activities twice in succession. The first classification separates dynamic activities and static activities, and the second classification identifies the final activity category.

On many public data sets, traditional method has achieved a good baseline effect. However, with more and more types of human activities and the increase of sample size, the traditional manual feature extraction methods have shown many shortcomings. First of all, manual feature extraction requires strong expertized knowledge in the field of signal processing [22]. Obviously, with data expansion, feature extraction that completely relies on expertized knowledge cannot meet the efficiency requirements of HAR. Secondly, features extracted manually cannot avoid the massive loss of key information. Thirdly, features extracted contain a large amount of redundancy information, especially in the recognition of complex activities.

2.3 Deep learning method for HAR

Different from the traditional HAR methods which manually extracting features and applying machine learning methods, deep learning method shows its powerful scalability in the feature engineering stage. Because of the layer-by-layer structure, deep learning can extract both structured and non-structured features. Therefore, the technology can extract new features with higher dimensions and breaks through the limitation of one-dimensional data. There are two commonly used technologies: convolutional neural network which captures the local connection of multimodal activity data [23] and performs well in the recognition of two-dimensional pictures, recurrent neural network which extracts time correlation and retrieve incremental information from time interval, and it is suitable for flow data in human activity recognition.

The use of convolutional neural networks by converting one-dimensional sensor data into two-dimensional images has attracted the attention of many researchers. Jiang and Yin [24] creatively proposed that the original signal can be mapped into pixels and used as part of the image after Fourier transform. They used the corresponding DCNN as the classifier and achieved good results on three data sets. Further, based on the core idea of Jiang and Yin's method, all sensor axis data wave are arranged adjacent to each other at least once

to form a graph, Yang et al. [25]. This research improved data arrangement and generated a new feature graph method, which can extract more correlation information from the original signal. Similarly, Ronao and Cho [21] also proposed a human activity recognition system using a deep convolutional neural network as a classifier. The difference is that they directly extract relevant features from the original data, and use temporary fast Fourier transform technology to process the original data to obtain additional incremental information, which improves the performance of the classifier.

In addition to the convolutional neural network that can process picture format data, LSTM network which is good at processing time series data, also plays an important role in the field of HAR. Tao et al. [26] converted the original data of the sensor into horizontal and vertical components and improved the LSTM neural network, which they called bidirectional long-term and short-term memory (BLSTM). This method is applied to different signal data sequences, and finally significantly improves the performance of the classification algorithm. Moreover, it is also a hot research direction to combine recursive networks with other neural networks to obtain higher performance. For example, in order to recognize human gesture activities, Ordóñez and Roggen [27] used a combination of deep convolution neural network and recursive neural network to jointly extract features which are time dependent. Similarly, Xu et al. [28] also combined recurrent neural network and inception neural network as a hybrid recognition framework. They experimented with different sizes of kernels in the process of feature extraction and achieved remarkable results on public data sets.

In the field of HAR, the method based on deep learning performs well in most laboratory scenarios, but due to the high computing power required by deep learning, it performs poorly in real-time computing scenarios. High latency and high resource consumption are significant disadvantages of this kind of technology [29].

3 Methodology of this paper

3.1 Overview of the method

The core idea of the method in this paper is based on the assumption that each axis of human activity recorded by sensor has unique information, and these axes also have correlated information with each other. We have studied the information on each axis as well as the correlated information between different axes and input these two types of information into deep learning models. Then corresponding weights were assigned to each model, and finally the classification results were given by voting. This method can include more information of original human activities and reduce the loss of information.

As shown in Fig. 1, the steps of the method in this paper are as follows: (1) raw data is collected and preprocessed by sensors (accelerometers and gyroscopes) integrated into smart platform; (2) all axes will be processed into one-dimensional structured data which can be directly used as features of DCNN model. Meanwhile, all axis information will be aggregated to generate a special activity graph which can be used as features of DCNN as well. The algorithm from Yang et al. [25] is used here as the method to generating feature graph; (3) each model after trained will get the prediction accuracy on training data as the weight of the model; (4) the activity classification result on test data is obtained by weighted voting on the prediction of each model.

The rest of this section is organized as follows: Sec.3.2 shows the detail of construction of Single-axis model, Sec.3.3 shows the detail of construction of Multi-axis model, Sec.3.4 introduces the weighted voting model, and Sec3.5 is the evaluation metrics of the method in this paper.

3.2 Construction of Single-axis model

After the preprocessing of the original data obtained by accelerometers and gyroscopes, most of the current deep learning methods only focus on transforming the information from the sensor into various special structured features. For example, Jiang and Yin [24] directly convert the original data into pixel feature maps to express the correlation between adjacent axes. The method from Yang et al. [25] transform the original data into waveform activity graphs by making axes adjacent horizontally and vertically to obtain more correlated information. However, the above methods ignore that while learning more correlations between different axes, the information of each single axis also needs to be learned separately, that is because the correlation learned from axes may causes the loss of information of a single axis.

Therefore, the method in this paper performs feature processing and model training on each individual axis of the sensor, the goal of which is to obtain unique information of each axis through a separate model. As shown in Fig. 2, take the x-axis on the three-axis accelerometer as an example. First, different kinds of activities recorded by the three-axis accelerometer will be converted into one dimensional data by axis, and all data of x-axis data will be separated to form a structured feature table. Then the feature table will be input into the designed DCNN structure for learning. After learning, the model will be obtained, which is called single-axis model. Since each axis has a model, the names of these models are decided by sensor name and axis name. The example is ‘model-acce-x’, ‘model’, ‘acce’, ‘X’ represents the trained model, sensor name and axis name respectively. By analogy, applying this method to all other axes of all sensors to obtain the corresponding models.

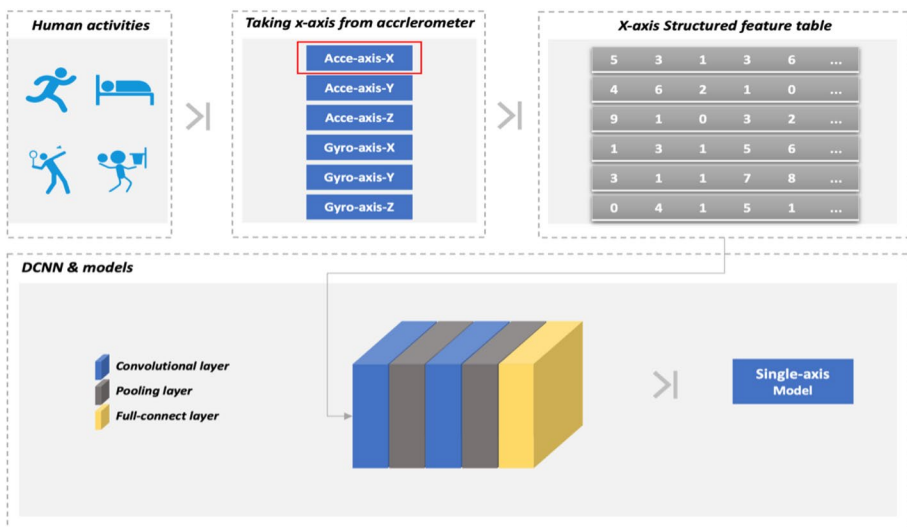


Fig. 2 Generating Single-axis model

3.3 Construction of multi-axis model

In addition to extracting human activity information from individual axis of each sensor, extracting the correlated information of all axes on sensors is also an important research field of HAR deep learning technology. Theoretically, all human physical activities occur in three-dimensional space, and the transformation from the collected information from three-dimensional space into features will inevitably cause the loss of information, so various ways to construct correlation features from all axes of a sensor mainly target to use features with stronger correlation information to reduce information loss.

This paper focuses on two methods to extract the correlation information between different axis. First, Jiang and Yin [24] constructed an activity image method that maps the original sensor data to pixel values, each axis is adjacent to all other axes at least once in the image. This method can greatly enhance the axis-related information hidden in the feature picture. Further, the method from Yang et al. [25] reconstructed and improved the ideas of Jiang and Yin [24]. They converted the original sensor data into waveform graphs instead of pixel values, and expanded the order of axes so that different axes could be adjacent at least once, both horizontally and vertically. Both methods ultimately work well.

Considering the performance on public datasets, the method from Yang et al. [25] are used for extracting information about the correlation between sensor axes, which is called the Multi-axis model. As shown in Fig. 3, first, the data of each axis from the sensor is output as a waveform graph, then all waveform graphs are stacked together according to a specific order. The specific order needs to satisfies that in both horizontal direction and vertical direction, all axes are adjacent at least once, the algorithm for generating specific permutation order and activity graph is shown in Algorithm 1. The final activity graph is generated and is input as features directly into a DCNN of a specific structure for learning. Finally, a model called multi-axis model are gotten, which learns the correlated information between different axes.

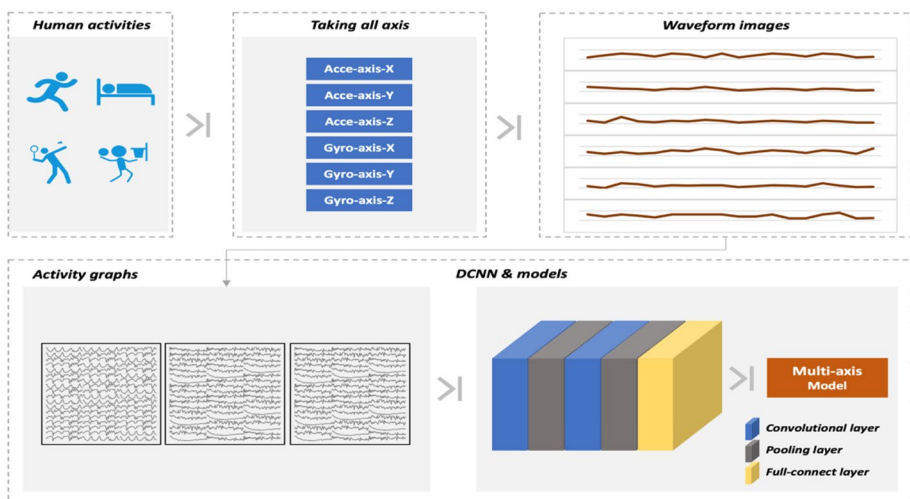


Fig. 3 Generating Multi-axis model

```

Input: Id-list, Output-string;
/* Id-list is a list of the raw axis signal ids. Output-string is a sequence of signal stacks. If there are six raw signals,
Id-list is initialized to [1,2,3,4,5,6], then Output-string is initialized to '1,2,3,4,5,6'. */
Output: Output-string;
Function Complete_flag (Id-list, Current-string):
  // set is a null list;
  for i ∈ Id-list do
    for j ∈ Id-list do
      if i ≠ j then
        Append 'i,j' to set
  return-value = 0;
  for k ∈ set do
    k1= The first element of k; k2= The second element of k;
    if k ∉ Current-string and 'k2,k1' ∉ Current-string then
      return-value = -1
  return return-value

idx = the last item in the Id-list;
add = 1;
while Complete_flag (Id-list, Output-string) = -1 do
  if idx = add then
    add = add + 1;
  if add > length of Id-list then
    add = add - 2;
    Append ',add' to Output-string;
    idx = add, add = 1;
  if 'idx,add' ∈ Output-string or 'add,idx' ∈ Output-string then
    add = add + 1;
  if add > length of Id-list then
    idx = idx + 1, add = 1;
    while 'idx,add' ∈ Output-string or 'add,idx' ∈ Output-string do
      add = add + 1;
      if idx = add then
        add = add + 1
      if add > length of Id-list then
        idx = idx + 1;
        break;
    add = 1;
    Append ',idx' to Output-string;
  else
    Append ',add' to Output-string;
    idx = add, add = 1;
Return Output-string;

```

Algorithm 1(a) Axis sorting

Input: Id-list

/* Id-list is the output of Algorithm 1(a). It indicates the order in which signals are stacked. */

Output: Activity graph

Begin

//Activity graph is a blank graph and be divided into three columns.

// for the first column.

for $i \in \text{Id-list}$ **do**

$j =$ the previous id of i in the Id-list.

// when i is the first id in the Id-list, j is the last id.

the time series waveform of the j_{th} signal sequence is stacked to the bottom of the first column of activity graph.

// for the second column.

for $i \in \text{Id-list}$ **do**

the time series waveform of the i_{th} signal sequence is stacked to the bottom of the second column of activity graph.

// or the third column.

for $i \in \text{Id-list}$ **do**

$j =$ the next id of i in the Id-list.

// when i is the last id in the Id-list, j is the first id.

the time series waveform of the j_{th} signal sequence is stacked to the bottom of the third column of activity graph.

Return Activity graph

Algorithm 1(b) Generating activity graph

3.4 Weighted voting model

When all single-axis models and multi-axis model are obtained, it comes to the core steps of the method in this paper, weighted voting. The significance of weighting is that different models essentially represent the summary of some part of the information of the original activity, single-axis models represent the activity information from the corresponding single axis, and multi axis model represents the correlation information of all axes. Each model makes different contributions in recognizing the final correct activity, which means that each model needs to be given different weights to represent the level of contribution. In this method, the classification accuracy of different models on the same training data are made as the weights, which can be obtained completely automatically and avoid the interference of human knowledge. After the weighting stage, the weighted voting of multiple models are carried out to get the classification results. The voting algorithm is shown in Algorithm 2. After the weighted voting, the final classification result of human physical activities can be obtained. The complete workflow of the weighted voting model is shown in Fig. 4 as an example. In the sample workflow, let's assume that we need to recognize four activities: run, badminton, sleep, and dunk. In the stage 1, we use the training data for these four activities to train corresponding models. The single-axis features are one-dimensional arrays that have undergone windowing and segmentation processing. Each axis corresponds to a single-axis model. Since we use both accelerometer and gyroscope sensors, each sensor has three axes (x, y, z). Therefore, we have a total of six single-axis features and their corresponding six single-axis models. The multi-axis feature is a

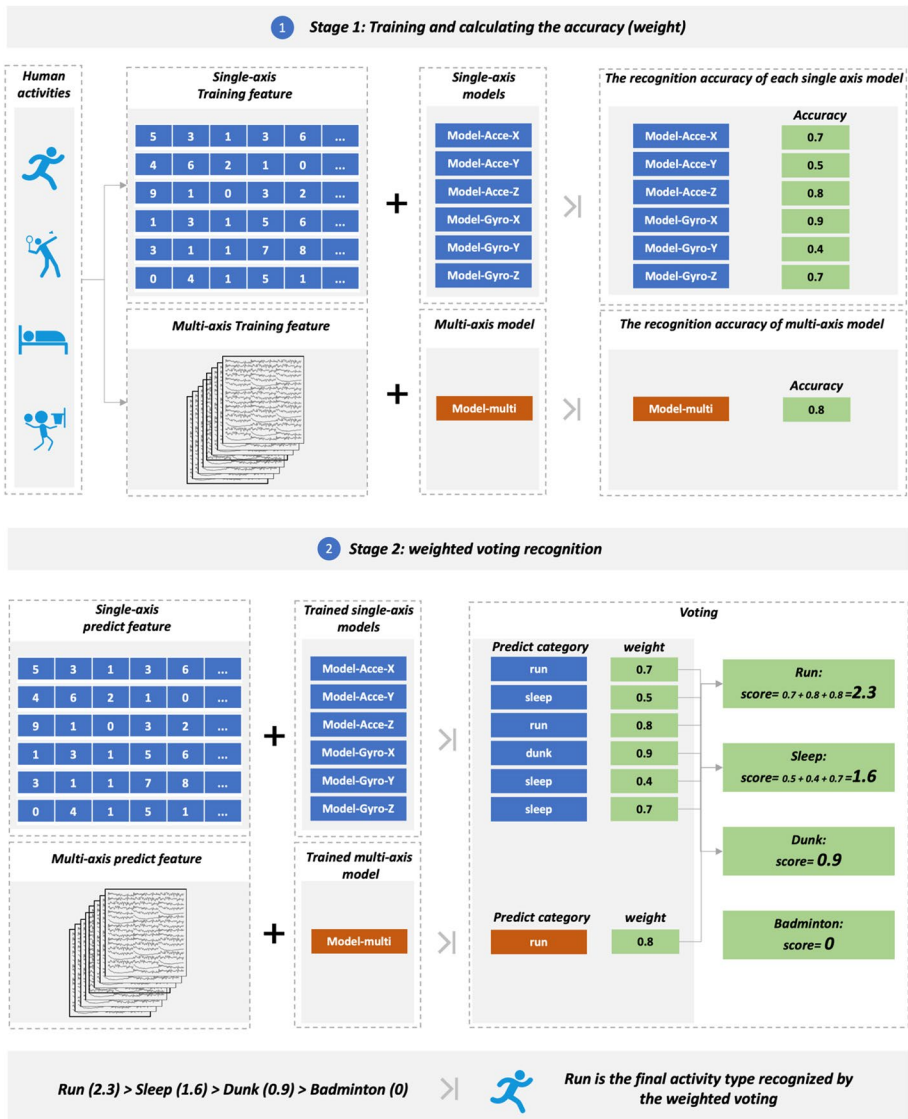


Fig. 4 The workflow of the weighted voting model

two-dimensional image generated by the activity graph algorithm, which aggregates the six single-axis features into one image. The multi-axis feature corresponds to one multi-axis model. So, we have a total of seven models. After training all the models, each model can calculate its accuracy on the training set, which indicates the recognition capability of each model. We use the accuracy of each model as the weight for the subsequent process. The weights for the six single-axis models are 0.7, 0.5, 0.8, 0.9, 0.4, and 0.7, and the weight for the multi-axis model is 0.8. In the stage 2, we need to perform weighted voting on the test dataset (or prediction dataset) to determine the final activity category. Specifically, we use

the models trained in the stage 1 to make predictions on the test features. Each model will output its predicted activity category. The recognition results of the six single-axis models are run, sleep, run, dunk, sleep, sleep, and the recognition result of the multi-axis model is run. We assign the weights obtained in the stage 1 to the recognition results of each model in the stage 2. For each activity, run is recognized by two single-axis models and the multi-axis model. Its weighted score is 2.3 (0.7 + 0.8 + 0.8). Sleep is recognized by three single-axis models. Its weighted score is 1.6 (0.5 + 0.4 + 0.7). Dunk is recognized by one single-axis model. Its weighted score is 0.9. Badminton is not recognized by any model, so its final score is 0. Among the four activities, run has the highest score. Therefore, the final result of the weighted voting recognition is run.

Input: train data feature, train data label, single-axis models, multi-axis model

/* train data feature is the feature used in training the corresponding model.

train data label is the true activity category.

single-axis models are trained single-axis models, multi-axis model is the trained multi-axis model. */

Output: Categories of human activities voted

Weighting step:

- 1) Using single-axis models to predict the results on the training data respectively, and the corresponding accuracy is calculated by using the train data label.
- 2) Accuracy of each single-axis model = weight of each single-axis model.
- 3) Using multi-axis model to predict the results on the training data, and the corresponding accuracy is calculated by using the train data label.
- 4) Accuracy of the multi-axis model = weight of multi-axis model.

Voting step:

Activity_list = [0] * N,

model_list = [single-axis model1, single-axis model2..., multi-axis model]

weight_list = [weight of single-axis model1, weight of single-axis model2..., weight of multi-axis model]

activity_categories = [first activity category, second activity category..., last activity category]

/*Activity_list is a list of numbers. N is the total number of categories of human activities. If there are 3 categories of human activities to be recognized, then Activity_list = [0,0,0] */

for $i \in$ model_list **do**

j = Sequence number of the activity category recognized by the model_list[i]

Activity_list[j] = Activity_list[j] + weight_list[i]

k = Sequence number corresponding to the maximum value of the Activity_list

final_activity_category = activity_categories[k]

Return final_activity_category

Algorithm 2 Weighted voting model

3.5 Evaluation metrics

The goal of HAR field is to recognize the categories of human activity, which is a typical classification problem. As for the evaluation of classification problems, a series of indicators derived from the confusion matrix is the best choice. The confusion matrix

consists of four key elements, TP, FP, FN and TN, which represent the following meanings respectively:

TP (true positive): The number of positive samples is predicted when the real label is positive.

FP (false positive): The number of positive samples is predicted when the real label is negative.

FN (false negative): The number of negative samples is predicted when the real label is positive.

TN (true negative): The number of negative samples is predicted when the real label is negative.

Based on TP, FP, TN, and FN, commonly used classification evaluation metrics such as accuracy, precision, recall, and F1 score can be calculated. In the field of HAR, accuracy is widely used as a general performance evaluation metric, so we also use this metric to compare with other state-of-the-art methods. In addition, in the section where we present the design details of the proposed method, we also use F1 score and AUC as supplementary metrics to better compare the performance with other methods. It is worth noting that the calculation method of accuracy is basically same on binary classification problem and multi classification problem, so there is no need to carry out metric transformation for the HAR multi classification problem. The calculation method of accuracy is shown in Eq. (1). The calculation method of F1 score is shown in Eq. (2).

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \quad (1)$$

$$F1 \text{ score} = \frac{2 \times \frac{TP}{TP+FP} \times \frac{TP}{TP+FN}}{\frac{TP}{TP+FP} + \frac{TP}{TP+FN}} \quad (2)$$

4 Experiments and discussions

4.1 Experimental environment

The experiment was run on a Linux environment with computer hardware using an Intel Core (TM) I7 CPU @ 2.3 GHz and an NVIDIA Tesla P100 GPU with 16GB memory. The experimental software environment was Python 3.7.13, CUDA 11.2, and PyTorch 1.11.0.

4.2 Datasets and pre-processing

There are many kinds of public data sets in the field of HAR. Because the research direction of this paper focuses on using accelerometers and gyroscopes as sensors to collect information, UCI and USCHAD data sets were used to conduct the experiments. These two data sets use accelerometers and gyroscopes to collect data. In addition, the UTD-MHAD data set uses not only accelerometers and gyroscopes but also Kinect camera to collect video data. Since this data set collects more kinds of activities and the

activities themselves are complex, the methods in the HAR field on this data set usually have a significant drop in terms of performance. Therefore, this UTD-MHAD data set can test the performance of the method in this paper in a nearly extreme state, so this data set were also experimented.

In the USCHAD data set, 14 human volunteers aged between 21-49 were arranged to carry out 12 different activities. A MotioNode sensor platform embedded with a three-axis accelerometer and a gyroscope is placed on the buttocks of the subjects to collect activity information. The original sampling frequency of the sensor is 100Hz. In addition, a three-axis magnetometer is also embedded in the platform to collect relevant data. In the experiment, because this type of special data is not within the scope of the study, we eliminate this part of magnetometer data. In the UCI data set, 30 human volunteers aged between 19-48 years old carried out six different activities. A Samsung Galaxy S II smart phone embedded with a three-axis accelerometer and a gyroscope is placed on the waist of the subject to collect activity information. The original sampling frequency of the sensor is 50Hz. In particular, the data recorded from the three-axis accelerometer includes the original version and the version that filters out the gravitational acceleration. For the most complex data set UTD-MHAD, more processing has been done. In the original data set, eight human volunteers completed 27 different types of activities using inertial sensor platforms (integrated with a triaxial accelerometer and a gyroscope) placed on different body parts and a camera placed in the environment. When the inertial sensor platform was placed on the wrist of the experimental object, 21 activities were tested, When the inertial sensor platform was placed on the thigh of the experimental object, 6 kinds of activities were tested. We selected part of the data corresponding to 21 kinds of activities for the experiment and removed the part related to the camera data. This part of the UTD-MHAD data set is called UTD-MHAD 1.

In addition to collecting data and eliminating unnecessary sensor data from the original data set, it is also necessary to perform relevant sliding window segmentation on each data set to generate data samples with the same length of time for subsequent feature construction. And the data selected by the sliding window needs to be divided into two parts: the training data set and the test data set, based on randomly assigned experimental subject IDs. We uniformly set the data ratio of the training set and the test set to the commonly used 7:3. The basic information of datasets and information of data preprocessing is summarized in Table 1, the subjects represents the number of people receiving the experiment, the categories represents the number of activities set by the experiment, the position is the position of the sensor placed in these subjects, the sampling time is the length of the sliding time window at different sensor frequencies, the number of training samples and the number of testing samples shows the number of data samples.

Table 1 Datasets and pre-processing

Dataset	Subjects	Categories	Position	Sampling time	Training samples	Test samples
UACHAD [10]	15	12	Hip	2s	18557	7954
UCI [11]	30	6	Waist	2.5s	7352	2947
UTD-MHAD1 [12]	8	21	Wrist	1s	3014	1293

4.3 Structure of deep neural network

In the method of this paper, the algorithm to learn the hidden information in the feature is DCNN. Typical CNN is composed of convolution layer, pooling layer and full connection layer. The convolution layer performs convolution operation by moving the convolution kernel on the characteristic value. Through convolution operation, it can usually extract the local important features of the data, effectively reduce the dimension of the original data, thus make the neural network learn more important features. The pooling layer has two different strategies. One kernel is used to perform the maximum pooling operation or the average pooling operation to further compress the learned information. The maximum or average usually needs to be selected based on experiments. The method in this paper can get the best effect by using the maximum pooling operation. The full connection layer is usually the last layer in a CNN structure. It receives the neuron output of the previous layer, provides the corresponding number of neurons for matrix operation, and finally applies the activation function to get a result. According to the different characteristics of single axis information and multi axis information, different DCNN structures are used. The DCNN structure corresponding to single-axis model and multi-axis model is shown in Table 2, the DCNN parameters shows the structural parameters of neural networks and the hyperparameters used in training step.

4.4 The performance of weighted voting method

One of the core assumptions of the method in this paper is that the joint weighted voting of multiple models that learn more unique information of different axes can achieve better results than a single model. UCI data set is used to conduct relevant experiments to test the above hypothesis. The information of 9 axes of UCI data set is trained to 9 models, then the UCI test data is predicted to get the classification accuracy of these models. Next, according to the weighted voting method proposed in section 3, the classification accuracy of the weighted voting algorithm is obtained. In the process of the experiment, we found that the models of the three axes that excluded the gravitational acceleration contributed poorly, because most of the information of these three axes are included in the information of those three axes before excluding the gravitational acceleration, thus these three axes after

Table 2 DCNN structure

DCNN Parameters	Single-axis DCNN	Multi-axis DCNN
Convolutional-layer 1 kernel size	1x10	10x10
Convolutional-layer 2 kernel size	1x7	7x7
Convolutional-layer 1 output maps	20	20
Convolutional-layer 2 output maps	30	30
The type of subsampling layer	Max-pooling	Max-pooling
Subsampling-layer 1 kernel size	1x5	5x5
Subsampling-layer 2 kernel size	1x3	3x3
Learning rate	0.0001	0.0001
Batch size	128	256
The number of epochs	1500	1000
The dropout rate	0.1	0.1

excluding the gravitational acceleration would have a negative effect on weighted voting. Therefore, the standard 6 axes information were finally used for weighted voting. The comparison of the results is shown in Fig. 5. It can be seen that the classification accuracy of the single model with the best performance is 83.66%, while after weighted voting, the classification accuracy can reach 88.70%, which proves the correctness of the core view in this paper. The effect of correct weighted voting of multiple models is better than that of single model.

4.5 The ablation experiment based on single-axis models

In Section 4.4, we confirmed that weighted voting of multiple single-axis models can effectively improve recognition performance. Furthermore, we need to explore the individual impact of each single-axis model on the overall performance under the weighted voting scheme. Therefore, we designed ablation experiments by sequentially removing one single-axis model at a time and observing the overall weighted voting results. We selected six single-axis data from two sensors (accelerometer and gyroscope) in the UCI dataset for the experiments. Additionally, we can also incorporate cross-modality in the ablation experiments to explore the role of different single-axis models in recognizing different activities. Specifically, we observe the recognition accuracy of each activity in the ablation experiments.

The results of the ablation experiments for the single-axis models are shown in Fig. 6. We can observe that using all single-axis models for weighted voting results in an overall accuracy of around 88.7%. When we sequentially remove the three single-axis models of the accelerometer, the weighted voting results are 86.8%, 85.5%, and 86.4% respectively. When we sequentially remove the three single-axis models of the gyroscope, the weighted voting results are 85.9%, 87.8%, and 88.2% respectively. After removing the relevant axes

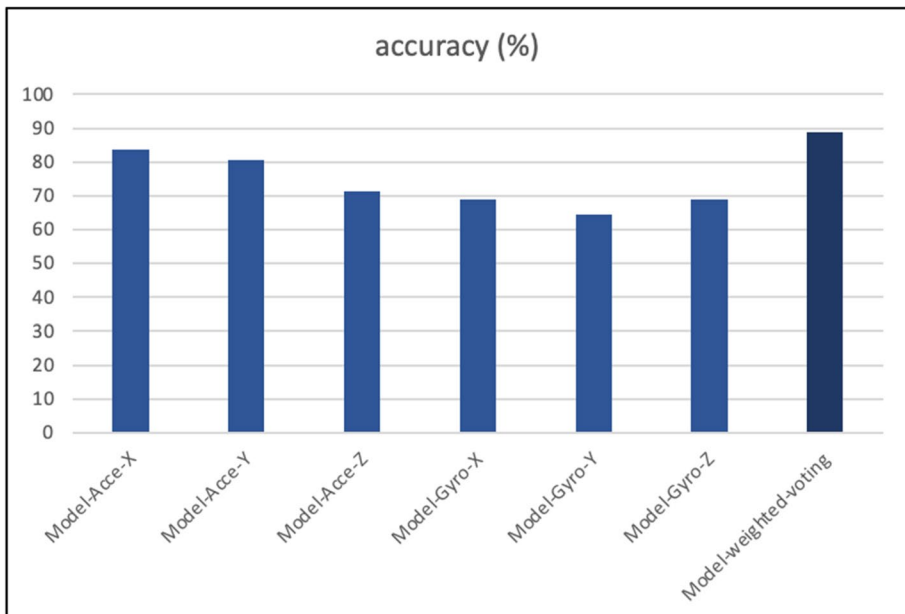


Fig. 5 Performance comparison between different models

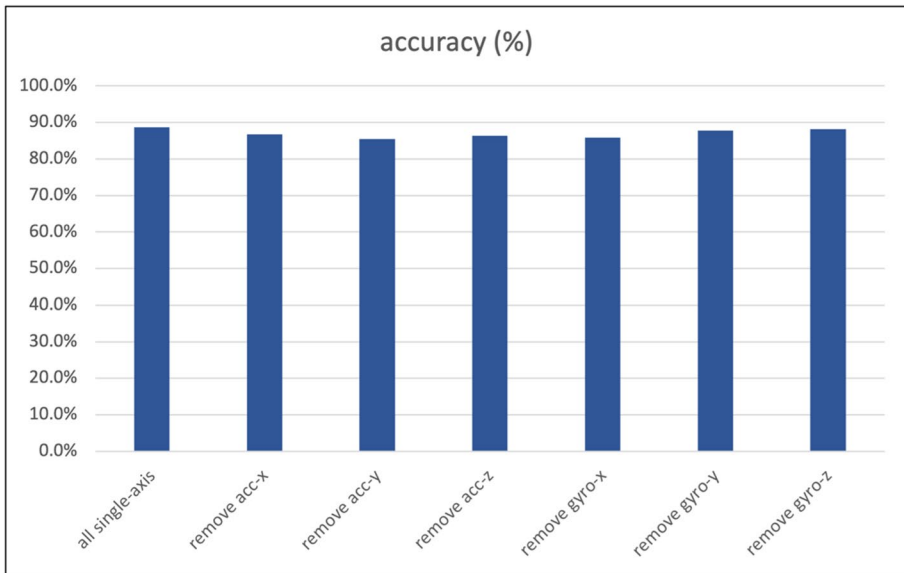


Fig. 6 The results of the ablation experiments for the single-axis models

of the accelerometer, the average recognition accuracy is 86.2%, which is a decrease of 2.5% compared to using all axes. After removing the relevant axes of the gyroscope, the average recognition accuracy is 87.3%, which is a decrease of 1.4% compared to using all axes. This indicates that the accelerometer contains more information for activity recognition compared to the gyroscope. This is because the accelerometer measures the acceleration of an object in multiple directions, while the gyroscope measures the angular velocity generated during turning or rotation. Common human activities such as running and jumping usually involve stretching, moving forward and backward in various directions, while large-scale rotations are relatively less common.

The performance of each activity in the ablation experiments is shown in Fig. 7. It can be observed that in the cross-modality case, the single-axis information has different degrees of impact on different activities. For example, after removing the relevant axes of the accelerometer, the average recognition accuracy for walking, standing, and laying decreases by 3%, 4%, and 4% respectively, while the average recognition accuracy for walking upstairs and sitting does not decrease. This indicates that different activities have different patterns and the distribution of information on different axes is uneven. Therefore, aggregating all single-axis information is valuable and necessary to obtain a high-performance recognition model.

4.6 Comparison between weighted voting method and other methods

After confirming that the weighted voting method can integrate multiple different models and effectively improve classification performance, we also need to test other methods that integrate multiple models. These methods include averaging or maximizing the predictions of each class from multiple models, or performing complex stacking operations on multiple models to obtain the final classification result. Specifically, for multi-classification

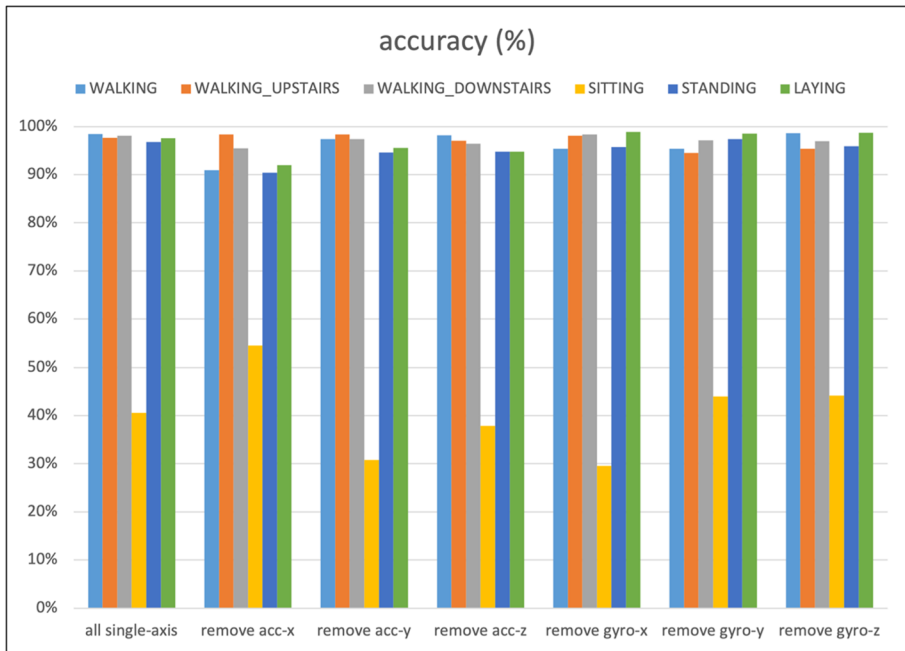


Fig. 7 The recognition accuracy of each activity in the ablation experiments.

Table 3 Comparison between weighted voting method and other methods

	USCHAD			UCI			UTD-MHAD 1		
	Accuracy	F1	AUC	Accuracy	F1	AUC	Accuracy	F1	AUC
Weighted voting method	90.01%	0.88	0.88	92.62%	0.89	0.87	62.57%	0.61	0.83
Averaging method	85.06%	0.82	0.86	89.14%	0.87	0.85	59.52%	0.56	0.79
Maximizing method	86.51%	0.84	0.86	88.30%	0.86	0.84	60.36%	0.57	0.79
Stacking method	87.90%	0.85	0.88	91.07%	0.88	0.85	60.95%	0.59	0.81

problems, the execution process of the averaging method is to calculate the average probability values for each class predicted by the single-axis model and the multi-axis model, finally, the class with the highest average probability is determined as the predicted class. The execution process of the maximizing method is as follows: the probability values predicted by the single-axis model and the multi-axis model for each class are compared, the maximum probability value for each class is selected as the probability value for that class, finally, the class with the highest probability value is determined as the predicted class. The averaging and maximizing methods are simple operations on the predictions of multiple models. We can also use more complex methods at the model level to integrate the prediction results. Stacking is a typical framework for this. Specifically, we use multiple single-axis models and one multi-axis model as weak learners, at the same time, we construct a new neural network that takes the outputs of these weak learners as inputs and returns the final classification result.

Weighted voting, averaging, maximizing, and stacking methods were tested on three public datasets, and the final classification performance is shown in Table 3. It can be observed that among all the methods, the averaging and maximizing methods performed the worst, while the stacking method had slightly lower accuracy than the weighted voting method. This is because the averaging and maximizing methods simply treat all model predictions equally and ignore the differences in classification performance between different models. For example, if two single-axis models have significantly different prediction performances but both predict the probability of the same class as 0.7, theoretically, the output of the better-performing model should be given a higher weight. However, the averaging and maximizing methods cannot achieve this. As for the stacking method, it can be said that weighted voting explicitly utilizes the outputs of multiple models, while stacking implicitly learns this information. However, the overly complex structure and the constraint of input information volume hinder its further improvement in classification ability. The experimental results above demonstrate the effectiveness of the weighted voting method.

4.7 Comparison with the baseline method

The method in this paper includes three main parts: single-axis model, multi-axis model and weighted voting, among them, the method of Yang et al. [25] is selected as the multi-axis model. Therefore, the method of Yang et al. [25] can be used as the baseline method of the method in this paper. In order to verify the hypothesis of our method, the weighted voting method combining single-axis model and multi-axis model can obtain more information than the multi-axis model that only focuses on axes correlated information. We need to test the performance comparison between the method in this paper and the baseline method on three public data. The experimental results are shown in the Table 4. Through experiments, it can be seen that the classification performance of the method in this paper on USCHAD and UCI datasets exceeds that of the baseline method (90.01% >87.70%, 92.62% >90.17%). And it is close to the performance of the baseline method on the most complex data set UTD-MHAD 1 (62.57% <64.12%), which is because there are too many types of activities in the UTD-MHAD 1 dataset, and it caused too many categories that need to participate in voting, thus weakens the advantage of voting to a certain extent. Above experiment results prove the effectiveness of the method in this paper, that is, by learning additional single axis information and fusing this part of information with multi axis information, it can achieve better results. In addition to comparing the overall effect on the data set, we also compared the recognition accuracy of the method in this paper with the baseline method on each activity category, as shown in the Fig. 8, it can be seen that the model in this paper can improve the performance of the baseline method in most of the activity categories by weighted voting.

Table 4 Comparison of the baseline method

	USCHAD			UCI			UTD-MHAD 1		
	Accuracy	F1	AUC	Accuracy	F1	AUC	Accuracy	F1	AUC
Proposed method	90.01%	0.88	0.88	92.62%	0.89	0.87	62.57%	0.61	0.83
Baseline method	87.70%	0.85	0.86	90.17%	0.88	0.85	64.12%	0.62	0.85

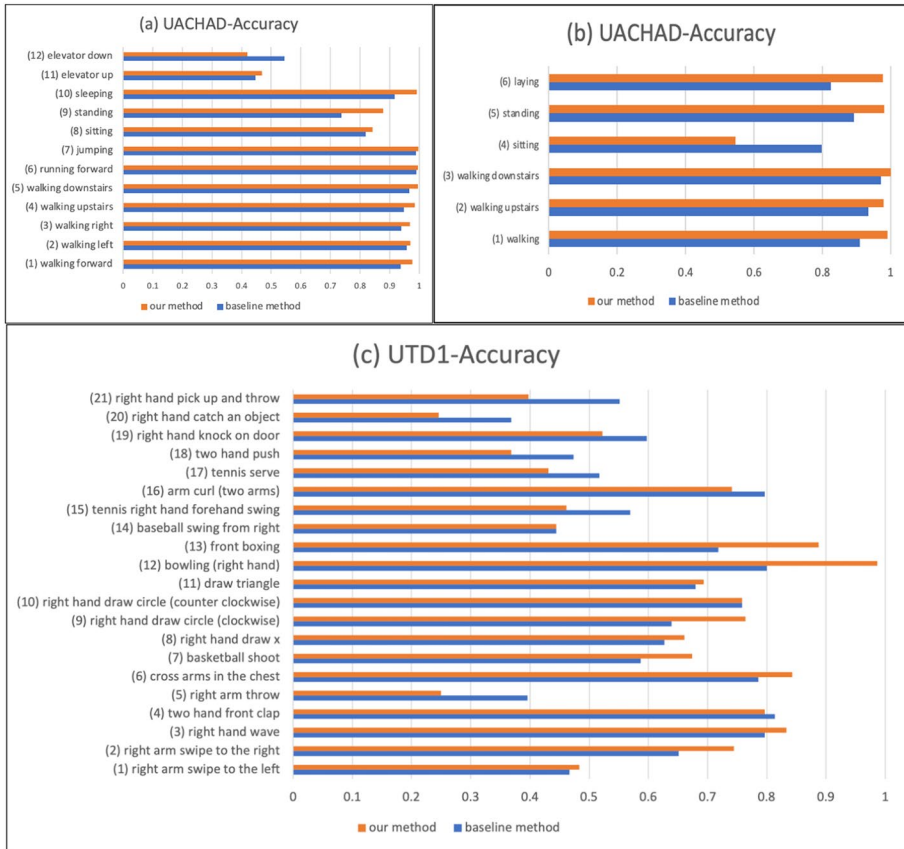


Fig. 8 The performance of sub-categories

4.8 Comparison with other state-of-the-art methods

We also compared the performance of our proposed method with other state-of-the-art methods on three datasets: USCHAD, UCI, and UTD-MHAD 1. Currently, there are three main approaches in the HAR field. The first approach relies on expert knowledge to manually design and extract features, which are then combined with machine learning algorithms for recognition. For example, in the UCI dataset, the original dataset builders manually designed up to 561 features, which include rich time and frequency domain features, providing a good foundation for subsequent researchers to use manual feature construction. The second approach does not involve manual feature design and instead utilizes various deep neural networks to directly extract underlying activity information. The third approach combines the above two methods. Most of these methods construct a hybrid framework to separately handle features and models, and then integrate the advantages of each component to make the final recognition. Our proposed method belongs to this third category.

As shown in Table 5, the method in this paper has also achieved certain advantages over other state-of-art manual methods. On the UACHAD dataset, the special CNN method [30]

with three convolutional layers achieved an accuracy of 86.78%, the DCNN method [31] that processes image information using Fourier Transform achieved an accuracy of 82.55%. The manual method based on J48 Decision Tree [31] obtained the best result (91.37%) and the performance of the method in this paper is close to it (90.01%). On the UCI dataset, the method based on multi class SVM classifier [32] obtained the best result (96.40%) and the performance of the method in this paper ranked second (92.62%), which was much higher than the accuracy of the hybrid approach with SVM and KNN [33] and SVM with PCA method [34]. On the UTD-MHAD 1 dataset, it can be seen that the accuracy of the method in this paper (62.57%) is far better than other algorithms, the methods that came closest to the highest accuracy both used the LSTM-CNN architecture [35, 36], but their accuracies of 56.41% and 57.10% were both below 60%. The above experiments indicate that our proposed method has performance advantages over most state-of-the-art methods based on deep neural networks.

4.9 Discussion on experimental results

Based on all experiments, the HAR method in this paper has achieved better results than the baseline method on USCHAD and UCI datasets (an average increase of about 3%), and it is close to the performance of the baseline method on the most complex data set UTD-MHAD 1. At the same time, the method in this paper has advantages over most state-of-art methods based on deep neural networks. On the basis of this paper, we also need to explore some details of the experiment and ideas for future research.

According to our summary of relevant research, the method in this paper is the first to study single axis information and multi axis correlation information at the same time, and use weighted voting to classify. The advantage of the method in this paper is that it can learn the activity information contained in the limited sensor axis more comprehensively and reduce the loss of information in the process of feature extraction. Also, it realizes a fully automated operation. At the same time, the multi-axis model part of the method in this paper uses the method of Yang et al. [25] as the baseline. The method in this paper integrates single axis information on the basis of the method of Yang et al. [25] and enforces the attention to single axis information. The lack of attention to single axis information may lead to the misrecognition of very similar activities on some specific axes, while the method in this paper theoretically focuses on both local information (single axis) and the whole (multi axis correlation). A potential problem worth further exploration is that we find it unrealistic to only increase the number of corresponding models by increasing the number of single axes of sensors to expect better recognition results. This phenomenon is

Table 5 Comparison with other state-of-the-art methods

	USCHAD	UCI	UTD-MHAD 1
Method in this paper	90.01%	92.62%	62.57%
Other state-of-art methods	76.08% [37] (MLP)	91.31% [38] (RF)	48.57% [17] (ALR)
	91.37% [31] (J48)	96.40% [32] (SVM)	51.42% [39] (Boosting)
	86.78% [30] (CNN)	87% [33] (SVM+KNN)	56.41% [35] (LSTM+CNN)
	82.55% [31] (CNN)	85.4% [34] (SVM+PCA)	57.10% [36] (LSTM+CNN)

MLP Multilayer Perceptron, *J48* J48 Decision Tree, *RF* Random Forest, *ALR* Additive Logistic Regression

confirmed in the experiment of UCI data set. The experiment shows that the final recognition effect of adding 9 single axes information is weaker than that of adding 6 single axes information. In the 9 single axes, 3 single axes with negative effects are highly similar to another 3 axes, which means that simply increasing the sensor axis information cannot guarantee the increase of recognition performance. On the contrary, the addition of redundant information will make redundant models enter the voting system, resulting in 'noise' in the weighted voting process and interfering with the final judgment. This suggests that redundancy must be controlled when choosing to use single axis information of the sensor, rather than simply increase the number.

5 Conclusions

Based on wearable sensor data and deep learning technology, this paper designs a HAR method that extracts single axis information and multi-axes correlation information of sensors at the same time, and recognizes activities by weighted voting. The result shows that the classification performance of the method in this paper improves the recognition accuracy by about 3% on average compared with other deep learning methods that only consider the multi-axes correlation information. Further, it has a significant performance improvement compared with other state-of-art methods based on traditional manual designed features, which shows the advantages of the recognition method in this paper in extracting more sensor information to features and reducing information loss.

Acknowledgments This work was supported by Guangzhou Science and Technology Project (Grant No. 201904010107), Guangdong Provincial Natural Science Foundation of China (Grant No. 2019A1515010793).

Data availability The datasets analysed during the current study are available in followings: (1) "USC-HAD: A daily activity dataset for ubiquitous activity recognition using wearable sensors," <https://dl.acm.org/doi/pdf/10.1145/2370216.2370438>. (2) "A public domain dataset for human activity recognition using smartphones," https://www.researchgate.net/publication/298412360_A_Public_Domain_Dataset_for_Human_Activity_Recognition_using_Smartphones. (3) "Utd-MHAD: A multimodal dataset for human action recognition utilizing a depth camera and a wearable inertial sensor," <https://ieeexplore.ieee.org/document/7350781>

Declarations

Conflicts of interest If the authors have no relevant financial interests in the manuscript and no other potential conflicts of interest to disclose, a statement to this effect should be included.

References

1. Gupta S (2021) Deep learning based human activity recognition (HAR) using wearable sensor data. *International Journal of Information Management Data Insights* **1**(2):100046
2. Raziani S, Azimbagirad M (2022) Deep CNN hyperparameter optimization algorithms for sensor-based human activity recognition. *Neuroscience Informatics* **2**(3):100078
3. Zhang X (2021) Application of human motion recognition utilizing deep learning and smart wearable device in sports. *Int J Syst Assur Eng Manag.* **12**:835–843
4. Host K, Ivašić-Kos M (2022) An overview of human action recognition in sports based on computer vision. *Heliyon* **8**(6):e09633
5. Raeis H, Kazemi M, Shirmohammadi S (2021) Human activity recognition with device-free sensors for well-being assessment in smart homes. *IEEE Instrumentation & Measurement Magazine.* **24**(6):46–57

6. Azar SM, Atigh MG, Nickabadi A, Alahi A (2019) Convolutional relational machine for group activity recognition, in IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 7884–7893
7. Javed AR, Faheem R, Asim M, Baker T, Beg MO (2021) A smartphone sensors-based personalized human activity recognition system for sustainable smart cities. *Sustainable Cities and Society* **71**:102970
8. Bulling A, Blanke U, Schiele B (2014) A tutorial on human activity recognition using body-worn inertial sensors. *ACM Computing Surveys* **46**(3):1–33
9. Silva DF, Souza VMA, Batista GEAPA (2013) Time series classification using compression distance of recurrence plots, in IEEE 13th International Conference on Data Mining, pp. 687–696
10. Zhang M, Sawchuk AA (2012) USC-HAD: A daily activity dataset for ubiquitous activity recognition using wearable sensors, in Proceedings of ACM International Conference on Ubiquitous Computing Workshop on Situation, Activity and Goal Awareness, 1036–1043
11. D. Anguita, A. Ghio, L. Oneto, X. Parra, J. L. Reyes-Ortiz (2013) A public domain dataset for human activity recognition using smartphones, in 21st European symposium on artificial neural networks, computational intelligence and machine learning, 437–442
12. Chen C, Jafari R, Kehtarnavaz N (2015) Utd-MHAD: A multimodal dataset for human action recognition utilizing a depth camera and a wearable inertial sensor, in IEEE International Conference on Image Processing (ICIP), 168–172
13. Wang J, Chen Y, Hao S et al (2019) Deep learning for sensor-based activity recognition: A survey. *Pattern Recognition Letters* **119**:3–11
14. Zappi P, Lombriser C, Stiefmeier T et al. (2008) Activity recognition from on-body sensors: Accuracy-power trade-off by dynamic sensor selection, in Lecture Notes in Computer Science, 17–33
15. Maurer U, Smailagic A, Siewiorek DP et al. (2006) Activity recognition and monitoring using multiple sensors on different body positions, in International Workshop on Wearable and Implantable Body Sensor Networks, 113–116
16. Catal C, Tufekci S, Pirmir E et al (2015) On the use of ensemble of classifiers for accelerometer-based activity recognition. *Applied Soft Computing* **37**:1018–1022
17. Lara OD, Pérez AJ, Labrador MA et al (2012) Centinela: A human activity recognition system based on acceleration and vital sign data. *Pervasive and Mobile Computing* **8**(5):717–729
18. Feng Z, Mo L, Li M (2015) A random forest-based ensemble method for activity recognition, in 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), 5074–5077
19. Wang A, Chen G, Yang J et al (2016) A comparative study on human activity recognition using inertial sensors in a smartphone. *IEEE Sensors Journal* **16**(11):4566–4578
20. Prossiegger M, Bouchachia A (2014) Multi-resident activity recognition using incremental decision trees, in Adaptive and Intelligent Systems, 182–191
21. Ronao CA, Cho S-B (2014) Human activity recognition using smartphone sensors with two-stage continuous hidden Markov models, in 10th International Conference on Natural Computation (ICNC), 681–686
22. Chen Z, Zhang L, Cao Z, Guo J (2018) Distilling the knowledge from handcrafted features for human activity recognition. *IEEE Transactions on Industrial Informatics* **14**(10):4334–4342
23. Hammerla NY, Halloran S, Plötz T (2016) Deep, convolutional, and recurrent models for human activity recognition using wearables. *Journal of Scientific Computing* **61**(2):454–476
24. Jiang W, Yin Z (2015) Human activity recognition using wearable sensors by deep convolutional neural networks, in Acm International Conference on Multimedia(ACM), 1307–1310
25. Yang P, Yang C, Lanfranchi V, Ciravegna F (2022) Activity graph based convolutional neural network for human activity recognition using acceleration and gyroscope data. *IEEE Transactions on Industrial Informatics* **18**(10):6619–6630
26. Tao D, Wen Y, Hong R (2016) Multicolumn bidirectional long short-term memory for mobile devices-based human activity recognition. *IEEE Internet of Things Journal* **3**(6):1124–1134
27. Ordóñez F, Roggen D (2016) Deep convolutional and LSTM recurrent neural networks for multimodal wearable activity recognition. *Sensors* **16**(1):115
28. Xu C, Chai D, He J et al (2019) Innohar: A deep neural network for complex human activity recognition. *IEEE Access* **7**:9893–9902
29. Chen Z, Jiang C, Xie L (2019) A novel ensemble elm for human activity recognition using smartphone sensors. *IEEE Transactions on Industrial Informatics* **15**(5):2691–2699
30. Chen Y, Xue Y (2015) A deep learning approach to human activity recognition based on single accelerometer, *IEEE Intl Conf Syst Man Cybern*, 1488–1492
31. Jordao A, Nazare AC, Sena J et al. (2018) Human activity recognition based on wearable sensor data: A standardization of the state-of-the-art, CoRR, abs/1806.05226

32. Reiss A, Hendeby G, Stricker D (2013) A competitive approach for human activity recognition on smartphones, in European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, 455-460
33. Arshad M, Sabri MA, Ashraf F et al (2022) Hybrid machine learning techniques to detect real time human activity using UCI Dataset. *EAI Endorsed Transactions on Internet of Things* 7(26):170006
34. Hsu Y-L, Lin S-L, Chou P-H et al. (2017) Application of nonparametric weighted feature extraction for an inertial-signal-based Human Activity Recognition System, *International Conference on Applied System Innovation (ICASI)*, 1718-1720
35. Xia K, Huang J, Wang H (2020) LSTM-CNN Architecture for Human Activity Recognition, *IEEE Access*, pp. 56855-56866
36. Lyu L, He X, Law YW et al. (2017) Privacy-preserving collaborative deep learning with application to human activity recognition, *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, 1219-1228
37. Kwapisz JR, Weiss GM, Moore SA (2011) Activity recognition using cell phone accelerometers. *ACM SIGKDD Explorations Newsletter* 12(2):74–82
38. Casale P, Pujol O, Radeva P (2011) Human activity recognition from accelerometer data using a wearable device, in *Pattern Recognition and Image Analysis*, 289–296
39. Kim H, Kim M, Lee S, Choi YS (2012) An analysis of eating activities for automatic food type recognition, in *Asia-Pacific Signal and Information Processing Association Annual Summit and Conference*, 1-5

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.