https://doi.org/10.1007/s11036-019-01445-x

Deep Learning Models for Real-time Human Activity Recognition with Smartphones

Shaohua Wan[1,2] · Lianyong Qi[3] · Xiaolong Xu[4] · Chao Tong[5] · Zonghua Gu[6,7]

Abstract

With the widespread application of mobile edge computing (MEC), MEC is serving as a bridge to narrow the gaps between medical staff and patients. Relatedly, MEC is also moving toward supervising individual health in an automatic and intelligent manner. One of the main MEC technologies in healthcare monitoring systems is human activity recognition (HAR). Built-in multifunctional sensors make smartphones a ubiquitous platform for acquiring and analyzing data, thus making it possible for smartphones to perform HAR. The task of recognizing human activity using a smartphone's built-in accelerometer has been well resolved, but in practice, with the multimodal and high-dimensional sensor data, these traditional methods fail to identify complicated and real-time human activities. This paper designs a smartphone inertial accelerometer-based architecture for HAR. When the participants perform typical daily activities, the smartphone collects the sensory data sequence, extracts the high-efficiency features from the original data, and then obtains the user's physical behavior data through multiple three-axis accelerometers. The data are preprocessed by denoising, normalization and segmentation to extract valuable feature vectors. In addition, a real-time human activity classification method based on a convolutional neural network (CNN) is proposed, which uses a CNN for local feature extraction. Finally, CNN, LSTM, BLSTM, MLP and SVM models are utilized on the UCI and Pamap2 datasets. We explore how to train deep learning methods and demonstrate how the proposed method outperforms the others on two large public datasets: UCI and Pamap2.

Keywords Deep learning · Human activity recognition · Smartphone · Feature extraction

1 Introduction

The purpose of human activity recognition (HAR) is to infer the current behavior and goals of the human body through a series of observations and analyses of human behavior and its environment. HAR research has attracted attention because of its advantages of widespread application in intelligent surveillance systems, healthcare systems, virtual reality interactions, smart homes, abnormal behavior detection and other fields, as well as its ability to provide individualized support and interconnection for different fields. HAR has been an important concern in the computer science community. With the in-depth development of machine learning and deep learning technologies and the release of depth sensors, such as Microsoft Kinect, and NVIDIA's GPUs, it has been possible to quickly identify human behavior using devices such as video and motion sensors.

At present, HAR is mainly performed by external or internal sensing. In the former mode, devices such as cameras are placed at certain fixed predetermined locations, and the inference of activities is entirely dependent on the user's interaction with these devices. In the latter, a specific device, such as an inertial sensor, is placed on a part of the user's body to sense motion. In most cases, a camera is used as an external sensor for HAR. The inference of human activity comes from the camera recording a video sequence. The external sensing method has certain limitations. For example, if the user is not in the sensor range or the subject moves freely in the scene to introduce different degrees

✉ Shaohua Wan
shaohua.wan@ieee.org

Extended author information available on the last page of the article.

Published online: 30 December 2019

of occlusion problems, the activity cannot be recognized. Second, the environment is dynamic and complex, such as weather and daylight in the background, which also increases the difficulty of identification. The installation and maintenance of video sensors bring high costs. In addition, video processing technology is computationally complex and expensive, which makes video-like real-time HAR systems less practical. Wearable sensors for HAR do not have these limitations. The three-axis accelerometer is the most commonly used sensor for recognizing walking, running, jogging and other walking activities.

HAR data mainly come from RGB video sequences, depth images and skeleton nodes. Thus, in recent years, the deployment of video surveillance cameras has dramatically increased. The proposed algorithms were primarily based on RGB video because video at the time was derived from a traditional visible light camera. With the release of depth sensors, depth images are less affected by the external environment than RGB images, which is very helpful for improving the robustness and reliability of HAR. There is another new type of smart-terminal-based wearable product that is also rapidly developing. Currently, smartphones and smart devices, such as smart watches, are more sophisticated, and a large number of high-precision sensors are integrated into these devices. There are a wide variety of devices for measuring different behaviors of the human body, but the principles are the same. Researchers use sensor devices such as accelerometers and gyroscopes integrated internally to record changes in data generated by different behaviors and then send the data to intelligent terminals such as computers via wireless networks or Bluetooth technology, completing a series of complex steps on these terminals. Data processing makes it easy for researchers to analyze and record data on different human behaviors.

The main purpose of this paper is to propose a smartphone-inertial-sensor-based architecture for HAR. The proposed CNN model outperforms the traditional solutions, presenting state-of-the-art results on both the UCI and Pamap2 datasets. At the same time, to reduce the costs of energy consumption and hardware facilities, HAR and deep learning methods are combined to improve the calculation speed and recognition accuracy. The task of behavior recognition is to express human body gestures in RGB-D or wearable sensor data, analyze the interacting objects, and understand the semantics of behavioral actions. Finally, the computer can understand and judge the behavior sequences in video scenes similar to humans. Overcoming the high complexity and variability of human behavior, improving the accuracy of computer recognition and predicting behavioral actions can greatly promote the development of smart products in the fields of intelligent monitoring, smart homes, human-computer interactions, and so forth. Therefore, the modeling of time and space in accordance with the real human behavior structure and accurately recognizing human activity have attracted extensive attention from researchers at home and abroad.

The use of the built-in accelerometer in smartphones for HAR has been well solved in the literature [1–8]. In practical applications, real-time classification and HAR accuracy are challenges. In the smartphone era, smartphones contain a variety of embedded sensors, enabling researchers to collect human physiological signals to monitor daily activities and perform motion analysis, such as with accelerometers, gyroscopes, magnetometers, Bluetooth, Wi-Fi, microphones, light sensors, and cellular radio sensors that can be used to infer activity details. Sensors such as accelerometers, gyroscopes, magnetometers, heart rate, and GPS can be deployed for coarse-grained and context-aware recognition and social interaction between users' locations. Motion sensors (accelerometers, gyroscopes, magnetometers) provide important information to easily identify and monitor the user's activity, such as walking, standing or running. At present, HAR research mainly uses various sensor networks that are composed of sensors to record the user's behavior data and then identifies the different behaviors of the users from the data through the corresponding identification algorithm [9–14]. From the perspective of the type of data used, behavior recognition can be roughly divided into two categories: visual data, such as images and videos collected by cameras, and non-visual data, generally obtained using accelerometers and gyroscopes. Behavioral data measured by force sensors, magnetic sensors, or bioelectric (such as EEG signals, EMG signals, and EEG signals) are generally discrete time series data. Currently, with the widespread use of smartphones, smart watches, smart clothes, and so forth, various embedded sensor devices (such as accelerometers, gyroscopes, and cameras in smartphones) are widespread, providing powerful research tools for HAR studies by [15–22].

This paper mainly identifies two different HAR datasets, UCI and Pamap2, and uses different algorithm models, such as CNN, LSTM, BLSTM, MLP and SVM, for each dataset. The content of this paper is arranged as follows: the first section is the introduction, briefly introducing the research background and basic knowledge of HAR. The second section introduces relevant research on HAR at home and abroad. The third section introduces several deep learning classification models. The fourth section is the experimental section. It introduces two different HAR datasets, presents the preprocessing of the UCI HAR dataset and Pamap2 and describes the experimental process and experimental results in detail. The fifth section is a summary of the full text.

The main contributions of this paper are as follows:

- A method based on CNN is proposed to extract human activity features with cross-domain knowledge, which can capture the differences of the same activity.
- We detail the training processes for deep, convolutional, and SVM models.
- In more than 3,000 experiments, we introduce the suitability of each method for different activities in HAR, analyze the impact that each model's hyperparameters have on performance.
- With these experiments, we show that the proposed convolutional networks outperform the other solutions on 2 public datasets.

2 Related work

HAR can benefit various applications, such as smart health services and smart home applications. Many sensors have been utilized for human activity recognition, such as wearable sensors, smartphones, radio frequency (RF) sensors (WiFi, RFID), LED light sensors, cameras, etc. Owing to the rapid development of wireless sensor network, a large amount of data has been collected for the recognition of human activities with different kind of sensors. Conventional shallow learning algorithms, such as support vector machine and random forest, require to manually extract some representative features from large and noisy sensory data. However, manual feature engineering requires export knowledge and will inevitably miss implicit features.

Recently, deep learning has achieved great success in many challenging research areas, such as image recognition and natural language processing. The key merit of deep learning is to automatically learn representative features from massive data. This technology can be a good candidate for HAR. Hence, it is of utmost importance to compile the accomplishments and reflect upon them to reach further. The objectives of HAR are to express the human body posture in multiple videos or sensor data, to analyze interactive objects, to understand the semantics of behavioral actions, and finally to make the computer understand the behavior sequence in the video scene like humans do [23–27]. Currently, HAR can be broadly classified into two categories: vision-based and sensor-based HAR. The core processing phase based on vision mainly includes data preprocessing, object segmentation, feature extraction and classifier implementation. Due to the vast market demand and economic value, over the past few decades, a large number of researchers have proposed many video-based HAR technologies, which can realize the rapid recognition of human behavior by using video and motion sensors. However, in vision-based HAR, due to the observer's position and angle, the shadow of the object, the color of the background, and the intensity of the light all have a negative effect on accuracy, particularly when privacy issues become more important. In contrast, smartphones and wearable sensors can overcome this privacy issue and be widely used for HAR in smart homes.

Many HAR systems use accelerometers to identify a wide variety of daily activities, such as standing, walking, sitting, running and lying [28–33]. The authors explored accelerometer data to find repetitive activities and attempted to detect and prevent falls in older people in a smart environment. Most of the above systems use a number of accelerometers that are fixed at different locations on the human body. However, this method affects people's daily lives due to the attachment of many sensors on the human body and cable connections. Some studies have attempted to explore data from individual accelerometers at the waist. These works report the substantive identification of basic daily activities, such as running, walking, and lying down, but the activities of some complex groups cannot be accurately identified. Therefore, sensors are deployed in two ways in HAR. The first is a multisensor package, such as a three-axis accelerometer or body area network (BSN), and the other is used in conjunction with other sensors, such as gyroscopes, temperature sensors and heart rate monitors. Bao and Intille [34] proposed the earliest HAR system that uses five wearable dual-axis accelerometers and machine learning classifiers to identify 20 activities of daily living, achieving an 84% classification accuracy, which is quite good considering the number of activities involved. Gyros are also used in HAR and have been shown to improve recognition performance when used in conjunction with accelerometers [35–39].

Because cognitive and computing power have become standard on today's smartphones, researchers have begun to use smartphones to replace wearable sensors in HAR. In addition, smartphones support a variety of sensors, such as accelerometers and gyroscopes, and have wireless communication capabilities, thus making smartphones a very useful tool for activity monitoring in smart homes. In addition, smartphones have fast processing capabilities and are easy to deploy, which makes them more practical than other environmental multimode sensors in smart homes. Smartphones contain inertial sensors (e.g., gyroscopes and accelerometers) that use appropriate sensing resources to obtain HAR human motion information. For example, in [30], they collected user data from a chest unit consisting of an accelerometer and a vital sign sensor using a wirelessly connected smartphone and then processed and analyzed the data using different machine learning algorithms. In [40], a HAR system was developed to identify five transport activities, in which data from smartphone inertial sensors were used for classification along with expert hybrid models. In [41], the authors proposed an offline HAR system that uses a smartphone with a built-in three-axis accelerometer sensor. During the experiment, the smartphone was placed in a pocket. In [42, 43], a smartphone mounted at the waist was used to collect data from inertial sensors for activity recognition. In recent years, deep learning methods have been used in HAR to improve recognition performance, such as convolutional neural networks (CNNs), restricted Boltzmann RBM, recurrent neural network (RNN), long short-term memory (LSTM) network, and bidirectional long short-term memory (BLSTM) network. These deep learning algorithms have a significant improvement in accuracy compared to traditional recognition methods.

Table 1 [44] lists some of the common sensors that are available in popular smartphones. In this article, we focus on two types of motion sensors: accelerometers and gyroscopes. Smartphone-based HAR faces more problems and challenges than HAR using wearable sensors. One of the largest problems is the diversity in the location and orientation of smartphones; another problem is that human activity is always a complex process, and the collected data are often mixed with noise. To make their methods more feasible, research has focused on noise removal, feature extraction, and classification methods.

3 Methods

3.1 SVM

Support vector machine (SVM) is a linear classifier based on supervised learning, but it can also be a nonlinear classifier through different kernel functions. The basic idea of SVM learning is to obtain the separation hyperplane that can correctly divide the training dataset and have the largest set interval. For the case of nonlinearity, a linearly indivisible sample of a low-dimensional input space is mapped to a high-dimensional feature space by using a kernel function including a nonlinear mapping algorithm such that it is linearly separable. Because we used a labeled dataset in this experiment, we chose the SVC classifier and linear kernel provided by sklearn and combined them with PCA to reduce the features to simplify the calculation.

Table 1 Common sensors in popular smartphones

Sensor	Sony Xperia Z5	iPhone 7	Samsung Galaxy S6
Accelerometer	√	√	√
Gyroscope	√	√	√
Light	√	√	√
Proximity	√	√	√
Barometer	√	√	√
GPS	√	√	√

3.2 CNN

A convolutional neural network (CNN) is a deep feedforward artificial neural network that generally consists of multiple different neural network layers with multiple neurons in each layer. In CNNs, there are several important network layers that have different roles, such as the convolutional layer, the pooling layer, and the fully connected layer. The overall structure of the convolutional neural network model proposed in this paper is shown in Fig. 1. This CNN consists of one input layer, one output layer, three convolutional layers (c1, c2, and c3), three pooling layers (p1, p2, and p3) and a fully connected layer (fc1). The input layer is used to receive data and preprocess the data. In the experiments in this paper, we use time series data. The convolutional layer extracts features from the data, and the features extracted from the subsequent convolutional layers are more complex and more efficient. The role of the pooling layer is to reduce the number of features, and maximum pooling (max_pooling) is used in the experiment. Each convolutional layer and one pooling layer cooperate to function as an intermediate layer of the neural network. The final fully connected layer (fc1) is placed before the output layer, which combines the results of the previous layers to calculate the score for each of the last classes. The output layer generates classification results based on the results of the fully connected layers and outputs.

The overall structure of CNNs is described below. Convolution transformations and pooling transformations are included at each level, and each transformed feature vector is expressed as another feature of the input signal characteristics. A pooling layer always follows a convolutional layer, and the most popular is the max-pooling layer. The largest pooling layer takes the maximum of several small features as its own output, thereby shrinking a large amount of data. Two conventional choices for this purpose are to take the average or maximum of small rectangular blocks of the data. After several convolutional and max-pooling layers, the output of these layers is flattened into a one-dimensional vector and used for the classification. At this stage, additional features can

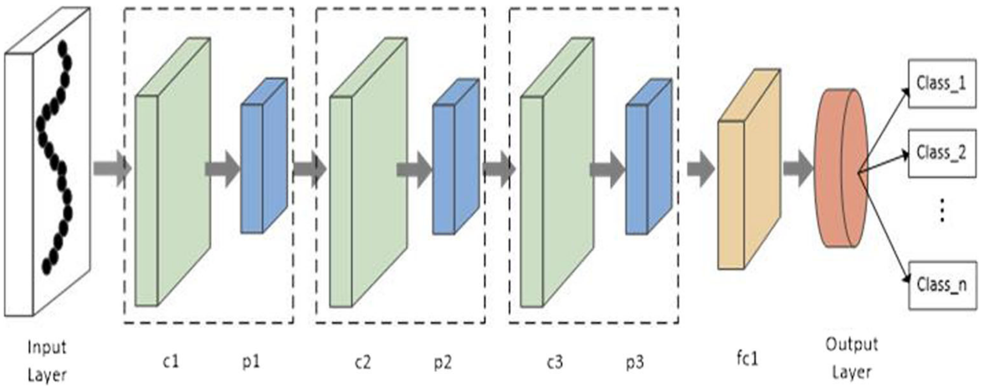


Fig. 1 The architecture of the CNN model

be stacked together with this vector. To learn nonlinear dependencies, CNN has one or more fully connected layers on top of it that perform the classification. Finally, the output of the last layer is passed to a soft-max layer that computes the probability distribution over the predicted classes.

In CNN, activation unit values are computed for each region of the network in order to learn patterns across the input data. The output of convolutional operation is computed as:

$$C_i^{(l,j)} = \sigma(b_j^l + \sum_{m=1}^{M} w_m^{l,j} \chi_{i+m-l}^{l-1,j}) \tag{1}$$

where l is the layer index, σ is the activation function, b is the bias term for the feature map, M is the kernel/filter size, w is the weight of the feature map.

The kernel_sizes of each convolutional layer and pooling layer are 7, 3 and 1, and the convolution depth (depth_size) is 64 steps, stride_size=3, and batch_size=64. The number of hidden units in the fully connected layer is 512. During the training, we used a dropout layer with the following parameters to prevent overfitting: 0.1, 0.25 and 0.5.

Table 2 The activity number and name of Pamap2 dataset

Activity No.	Name	Activity No.	Name
1	Lying	11	Car driving
2	Sitting	12	Ascending stairs
3	Standing	13	Descending stairs
4	Walking	16	Vacuum cleaning
5	Running	17	Ironing
6	Cycling	18	Folding laundry
7	Nordic walking	19	House cleaning
9	Watching TV	20	Playing soccer
10	Computer work	24	Rope jumping

3.3 LSTM

Long short-term memory (LSTM) is a time recurrent neural network that is ideal for modeling time series data due to its design characteristics. The key to LSTM is the state of the cells, with horizontal lines running through the top of the graph. The cell state is similar to a conveyor belt. It is run directly on the entire chain with only a few linear interactions. It is easy to keep the information flowing on it. LSTM has the ability to remove or add information to the state of the cell through a well-designed structure called a "gate." A door is a way to make informed choices pass. They contain a sigmoid neural network layer and a pointwise multiplication operation. The novelty of LSTM is that by increasing the input threshold, forgetting threshold and output threshold, the weight of the self-loop is changed such that the integral scale at different times can be dynamically changed when the model parameters are fixed, which excludes the problem of vanishing gradient or gradient expansion. LSTM has a variety of applications in the technology field. LSTM-based systems can be used in language translation, robotics control, image analysis, document abstraction, speech recognition, image recognition, and handwriting recognition, among others.

3.4 BLSTM

The bidirectional LSTM (BLSTM) model adds a forward-calculation process to the LSTM model. The LSTM model can only speculate the following units through the previous unit, whereas the BLSTM model can be speculated from front to back and can be guessed from the back. BLSTM contains two common RNN structures: a forward RNN that can utilize past information and a reverse RNN that can take advantage of future information. This structure allows BLSTM to use both the previous and next moments of

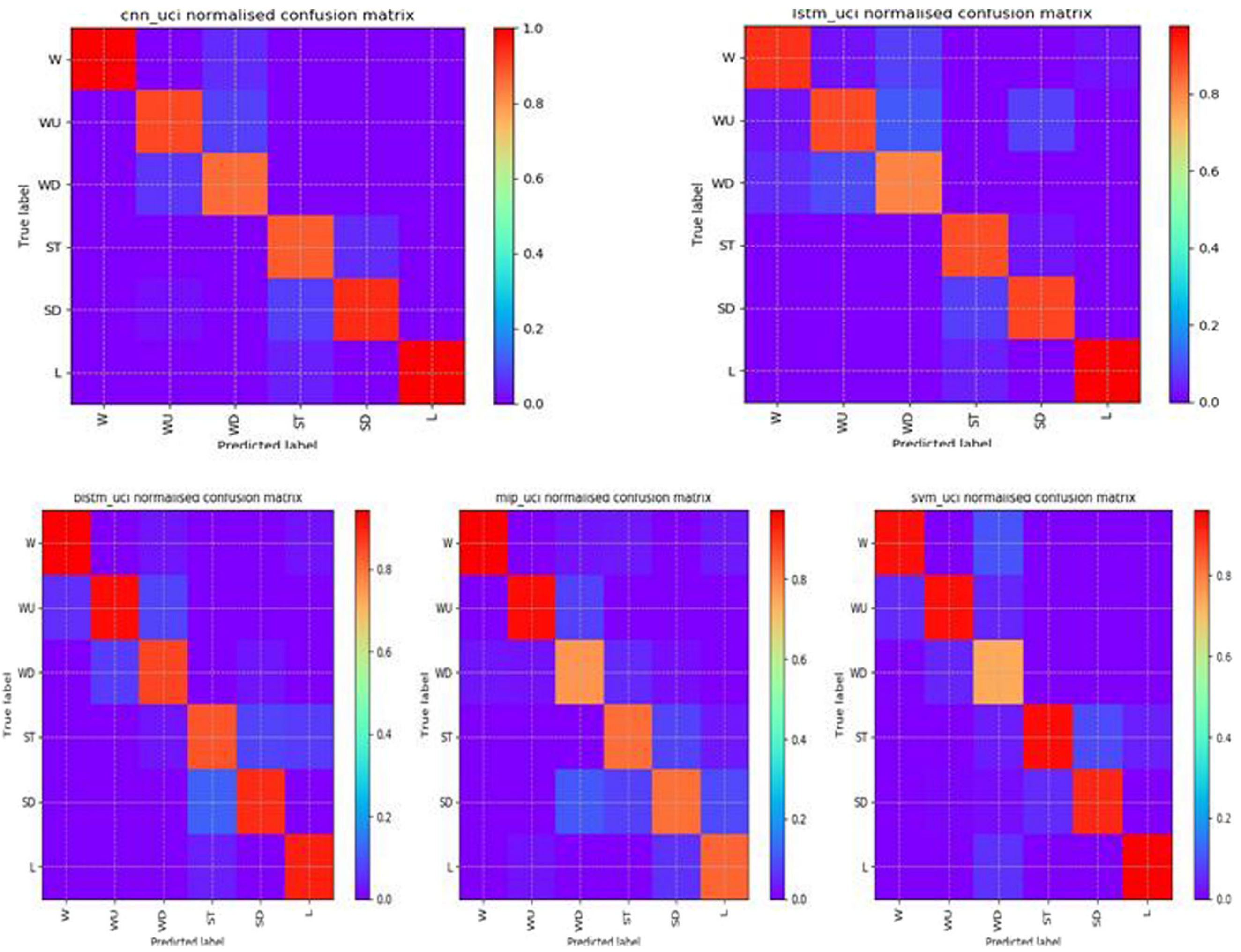


Fig. 2 The confusion matrix of the UCI dataset for five models

information at any time. In some data that have a strong dependence on two-way information, it generally provides better prediction than one-way LSTM.

3.5 MLP

Multilayer perceptron (MLP) is a forward-structured artificial neural network (ANN) used for classification and prediction. The structure of an MLP generally includes three levels of an input layer, a hidden layer and an output layer. In addition to the input nodes, each node is a neuron that uses a nonlinear activation function. MLP uses a supervised learning technique called backpropagation for training. The structure of the MLP in this paper includes two layers: the input layer, the output layer and the middle two hidden layers. The number of hidden cells in the two hidden layers is 256.

4 Experiments and results

4.1 Datasets

4.1.1 UCI HAR dataset

The UCI behavior recognition dataset is a dataset collected by measuring the six daily behaviors of 30 participants. The experiment uses a three-axis embedded accelerometer and a gyroscope operating at 50 Hz. The three component values of the accelerometer and the gyroscope are obtained separately, and the data dimension is 561. The tested behaviors of the participants included WALKING, WALKING_UPSTAIRS, WALKING_DOWNSTAIRS, SITTING, STANDING, and LAYING. The dataset has fewer types of behaviors and data structure specifications and has been widely used in HAR research since its publication.

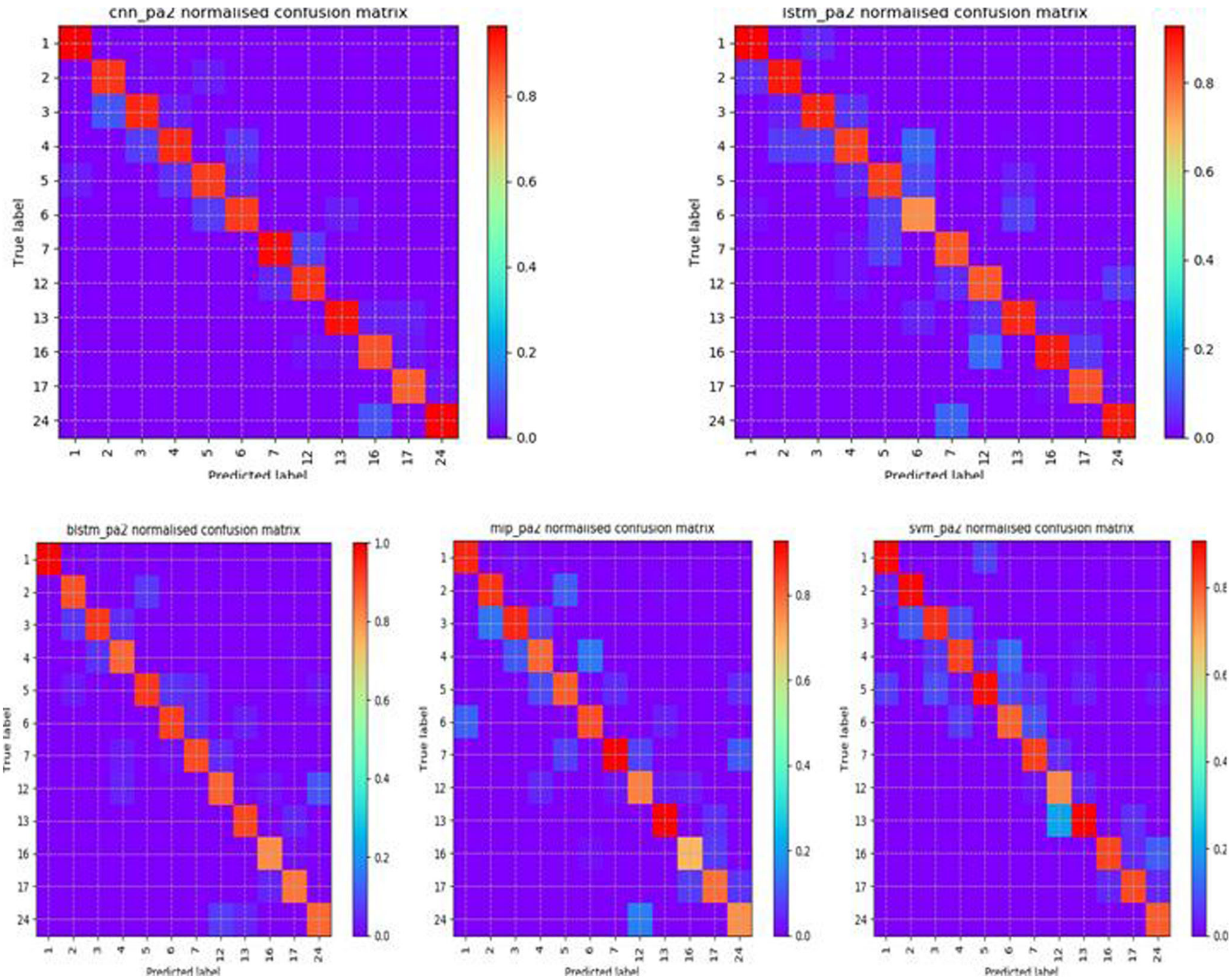


Fig. 3 The confusion matrix of the Pamap2 dataset for five models

4.1.2 Pamap2 dataset

The dataset was measured by Reiss and Strickere et al. and recorded 18 daily physical activities (including 6 optional activities) for 9 subjects (1 female and 8 male); the activity number and names are shown in Table 2. The data include measurements of accelerometers, gyroscopes, magnetometers, temperature, heart rate, and so forth distributed at the hands, chest and ankles of the subject; a total of 54 dimensions; and a sampling frequency of 100 Hz. In the experiment, we eliminated two data dimensions that were not related to the experiment and used the remaining 52-dimensional data as the source of the experimental data. For the final classification, we removed 6 optional activities and only identified and classified 12 daily activities. The activity numbers are sequentially 1, 2, 3, 4, 5, 6, 7, 12, 13, 16, 17, 24.

4.1.3 Dataset preprocessing

The same method was used to preprocess the two datasets used in the experiment. The original different types of data files were merged into one h5df file with a data structure. The training set and test set were formed with sklearn's train_test_split method in a ratio of 7:3. In addition, because the length of the original data is large, it is not conducive to the direct processing of the classification model. Therefore, the data need to be segmented in the preprocessing section, and the segmentation length in the experiment is 25. The data are read using the sliding window mode with coverage

Table 3 The main diagonal values of the confusion matrix for the UCI dataset

Models Activity	CNN	LSTM	BLSTM	MLP	SVM
walking	1.0000	0.9143	0.9429	0.9714	0.9403
walking_upstairs	0.9070	0.8837	0.9268	0.9535	0.9404
walking_downstairs	0.8611	0.8056	0.8611	0.7778	0.7368
sitting	0.8800	0.8800	0.8400	0.8333	0.9439
standing	0.9444	0.8889	0.8889	0.8298	0.9088
laying	1.0000	0.9762	0.9048	0.8438	0.9602
average_acc	0.9321	0.8914	0.8941	0.8683	0.9050

of 50%. For different sampling frequencies of different datasets, we also use the downsampling method to make them consistent and easy to compare.

4.2 Performance metrics

The algorithm evaluation indicators used in this experiment are accuracy, precision, recall, F1-score, confusion matrix, and accuracy/loss. Among the two classification problems, there are the following definitions: Accuracy: For a given test dataset, the ratio of the number of samples correctly classified by the classifier to the total number of samples is the correct rate for the identified samples, and the number of samples identified is accurate.

Precision: The ratio of the number of correctly identified positive samples to the total number of samples identified as positive in the identified sample. The calculation formula is as follows:

Recall: The recall rate is the ratio of the number of positively identified individuals correctly identified to the total number of positive samples in the total sample used. The recall rate is how many positive samples are identified. The calculation formula is

The F1 score, also known as the balanced F score, is the harmonic mean of precision and recall, which combines the results of the precision and recall indicators.

Confusion matrix: In the error matrix, each column of the matrix represents the classifier's category prediction for the sample. Each row of the second matrix expresses the real category to which the version belongs, which can easily indicate whether multiple categories are confusing. In the experiment, because the data size of the two datasets is not the same, we also regularize the confusion matrix and convert the predicted value and the real value in the matrix into corresponding proportions, which is convenient for comparing different datasets.

Accuracy and loss map: Response to the changes in accuracy and loss during the training of the neural network model. Each epoch will generate accuracy and loss value. By plotting the accuracy diagram and the loss diagram, the training of the network model can be visually reflected. The trend can be used to determine whether the model is properly and ideally trained, to detect anomalies in time (such as overfitting or underfitting), and to make adjustments in time.

4.3 Results and discussion

In this section, we validate the two datasets using five models in turn and plot the confusion matrix and the accuracy and loss curves, as well as the final precision, recall, f1_score, and accuracy tables. Through these experimental results, the characteristics and advantages and disadvantages of the model are analyzed. In the experiment, we first trained four neural network models. The CNN, LSTM, BLSTM, and MLP were trained at a learning rate of 0.001 with 200 iterations. The dataset was then classified by SVM.

We have proposed a CNN-based feature extraction approach, which extracts the local dependency and scale invariant characteristics of the acceleration time series. The experimental results have shown that by extracting these characteristics, the CNN-based approach outperforms the state of-the-art approaches. The proposed CNN not only exploits the inherent temporal local dependency of time-series signals, and the translation invariance and hierarchical characteristics of activities, but also provides a way to automatically and data-adaptively extract relevant and robust features without the need for advanced preprocessing or time-consuming feature hand-crafting. Experiments show that more complex features are derived with every

Table 4 The main diagonal values of the confusion matrix for the Pamap2 dataset

Models Activity	CNN	LSTM	BLSTM	MLP	SVM
1	0.9636	0.9273	1.0000	0.8814	0.8983
2	0.8966	0.8966	0.8966	0.8621	0.8966
3	0.9118	0.8824	0.9265	0.8824	0.8529
4	0.9080	0.8488	0.8681	0.8022	0.8298
5	0.8889	0.8519	0.9259	0.8148	0.8889
6	0.8913	0.7500	0.9167	0.8333	0.7917
7	0.9512	0.8293	0.9024	0.9268	0.8333
12	0.8974	0.8205	0.8718	0.7692	0.7436
13	0.9439	0.8785	0.9107	0.9252	0.9065
16	0.8621	0.8966	0.8125	0.6897	0.8276
17	0.8448	0.8276	0.8448	0.7931	0.8190
24	0.9600	0.8933	0.8667	0.7467	0.8000
Average_acc	0.9100	0.8586	0.8952	0.8272	0.8407

Table 5 The performance results for the UCI and Pamap2 datasets

Datasets / Models	UCI Dataset				Pamap2 Dataset			
	precision	recall	f1	acc	precision	recall	f1	acc
CNN	0.9321	0.9282	0.9293	0.9271	0.9166	0.9085	0.9116	0.9100
LSTM	0.8914	0.8899	0.8899	0.8901	0.8651	0.8467	0.8534	0.8586
BLSTM	0.8941	0.8936	0.8935	0.894	0.9019	0.8902	0.894	0.8952
MLP	0.8683	0.8658	0.8661	0.8683	0.8335	0.8217	0.8246	0.8207
SVM	0.9050	0.8986	0.8985	0.905	0.8471	0.8423	0.8376	0.8407

additional layer, but the difference in level of complexity between adjacent layers decreases as the information travels up to the top convolutional layers.

4.3.1 Confusion matrix

The standardized confusion matrices obtained from the UCI and Pamap2 datasets is shown in Figs. 2 and 3. The closer the color of the square on the main diagonal is to red, the higher the prediction accuracy of the category corresponding to the square. There are 6 categories of UCI dataset classification results. The row and column labels of the confusion matrix are in order W(WALKING), WU(WALKING_UPSTAIRS), WD(WALKING_DOWNSTAIRS), ST(SITTING), L(LAYING). There are 12 types of Pamap2 activities, and the activity names in the figure are replaced by the corresponding activity numbers in Table 2. Tables 3 and 4 respectively reflect the probability that each activity class obtained by the UCI and Pamap2 datasets is correctly predicted under five different models (i.e., the main diagonal value of each confusion matrix in Figs. 2 and 3 and count the average correct probability of all classification activities). As shown, CNN achieved the highest average probability value, the performances of the LSTM and BLSTM algorithms were similar, and the effect of MLP was the worst. As a machine learning algorithm, SVM has also achieved good experimental results.

4.3.2 Precision, recall, F1-Score and accuracy

Table 5 lists the values of the four evaluation indicators of precision, recall, f1-score, and accuracy obtained by the five models on the two datasets. Based on these values, it can be found that for the UCI dataset, the performance of the CNN algorithm is optimal, and the values of P, R, F1, and A are higher than those of the other four algorithms. Except for the BLSTM algorithm, the values of the four

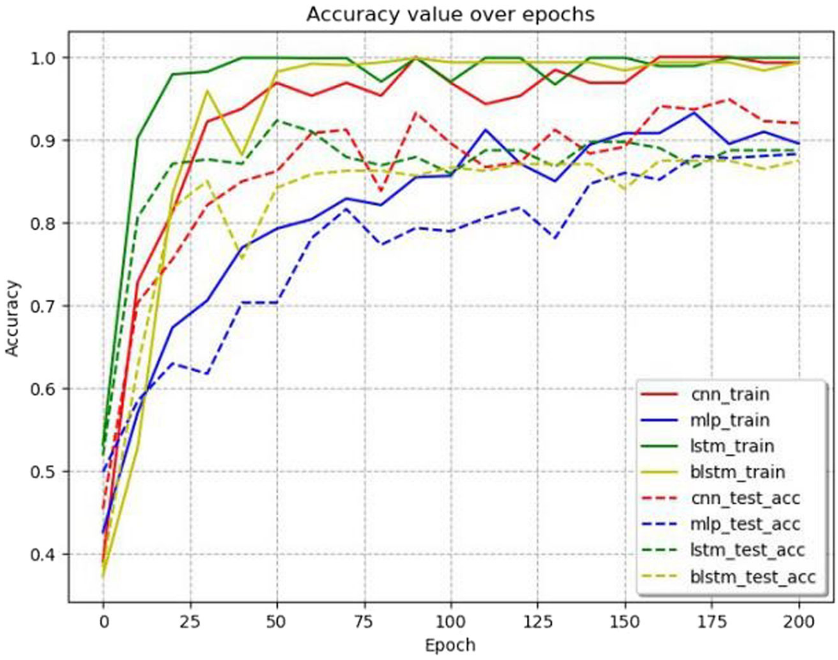
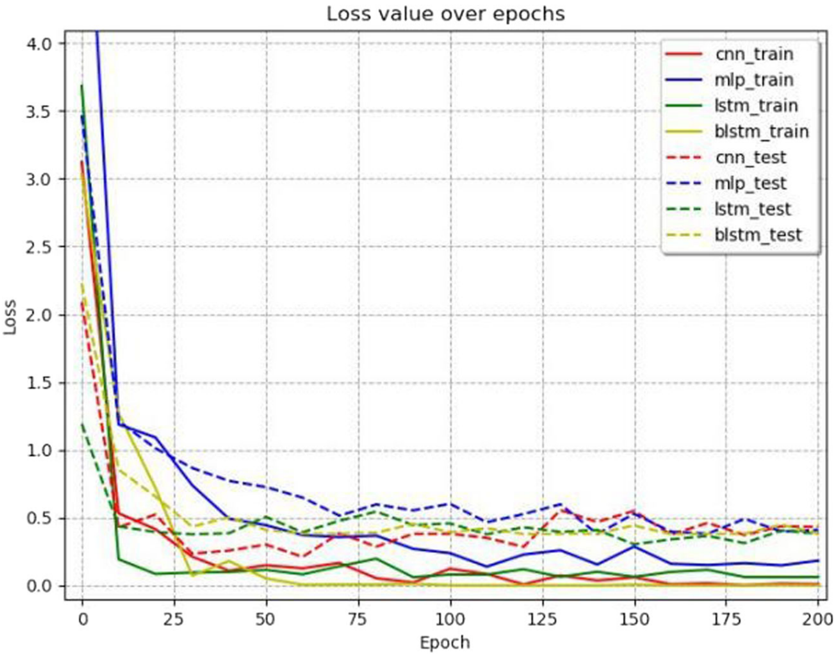


Fig. 4 The accuracy comparison of the four models for the UCI dataset

Fig. 5 The loss comparison of the four models for the UCI dataset

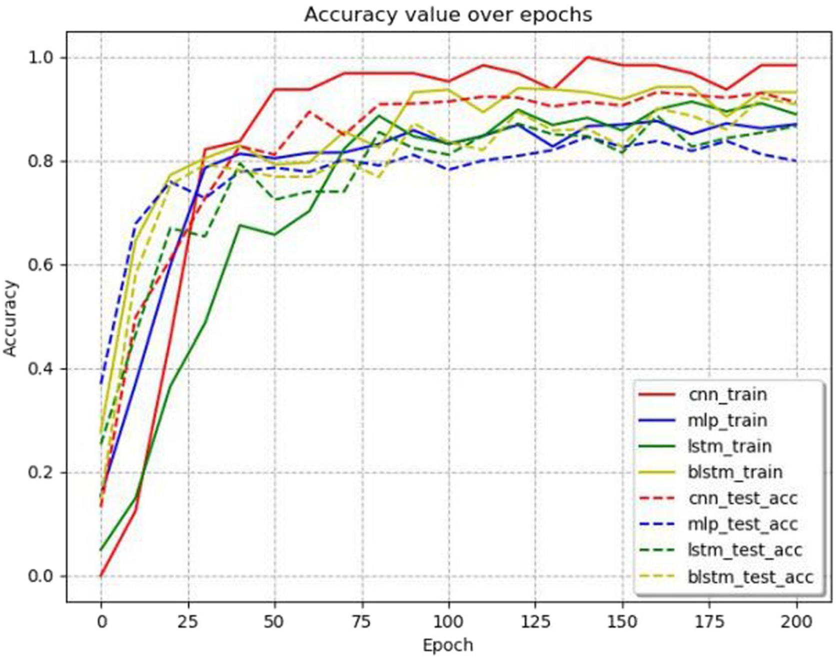


indicators obtained by the other four algorithms on the UCI dataset are lower than the corresponding values obtained on the Pamap2 dataset. This result may be because the Pamap2 dataset has more activity types than the UCI dataset, and the distinction between categories is more complicated. The performance of BLSTM on the Pamap2 dataset shows that the algorithm is more suitable for classifying and identifying long data.

4.3.3 Accuracy and loss map

To better reflect the effect of the neural network model in the training process, we have drawn line graphs in the figures for the CNN, LSTM, BLSTM, and MLP algorithms. The curves represent each model. As the number of training iterations increases, the values of accuracy and loss change. For comparison purposes, we performed 200 iterations for

Fig. 6 The accuracy comparison of the four models for the Pamap2 dataset



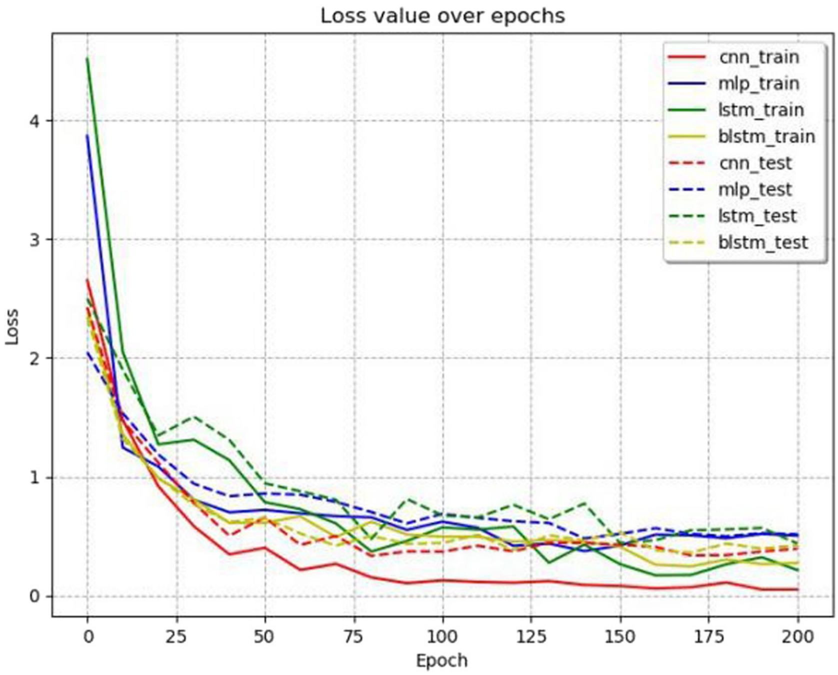


Fig. 7 The loss comparison of the four models for the Pamap2 dataset

each model and draw a point every 10 iterations when drawing, making the line graph clearer but accurately reflecting the trend of the curve. The results are presented in Figs. 4, 5, 6, and 7. Among the four models, MLP performs the worst. CNN is slightly better than LSTM and BLSTM on the Pamap2 dataset, but the performance of the three is similar on the UCI dataset. Overall, the four algorithms performed better on the UCI dataset than on the Pamap2 dataset.

5 Conclusion

In this paper, we compare the advantages and disadvantages of five algorithms, CNN, LSTM, BLSTM, MLP and SVM, in the recognition of human behavior. Through experiments, it can be found that CNN, as a classical neural network algorithm, still has important value in the field of human behavior recognition and is an excellent classification and recognition algorithm. In this paper, five of the algorithms are tested in sequence through two human behavior datasets, and the performance of these models is measured by multiple evaluation indicators. However, there are still some shortcomings in this paper. The structure of the four neural network models used in the experiment can still be further optimized, and more detailed comparison experiments can be conducted.

Acknowledgments This work was supported in part by the National Natural Science Foundation of China under Grant 61672454; by the Fundamental Research Funds for the Central Universities of China under Grant 2722019PY052 and by the open project from the State Key Laboratory for Novel Software Technology, Nanjing University, under Grant No. KFKT2019B17.

Compliance with Ethical Standards

Conflict of interests The authors would like to declare that there are no conflicts of interest with any third party.

References

1. Gao Z, Xuan H, Zhang H, Wan S, Choo KR (2019) Adaptive fusion and category-level dictionary learning model for multi-view human action recognition. IEEE Internet of Things Journal, pp 1–1
2. Ding S, Qu S, Xi Y, Sangaiah AK, Wan S (2019) Image caption generation with high-level image features. Pattern Recognition Letters 123:89–95
3. Gao H, Huang W, Yang X, Duan Y, Yin Y (2018) Toward service selection for workflow reconfiguration: An interface-based computing solution. Futur Gener Comput Syst 87:298–311
4. Xu Y, Yin J, Huang J, Yin Y (2018) Hierarchical topic modeling with automatic knowledge mining. Expert Syst Appl 103:106–117
5. Ding S, Qu S, Xi Y, Wan S (2019) A long video caption generation algorithm for big video data retrieval. Futur Gener Comput Syst 93:583–595
6. Zhang R, Xie P, Wang C, Liu G, Wan S (2019) Classifying transportation mode and speed from trajectory data via deep multi-scale learning. Computer Networks 162:106861
7. Gao H, Duan Y, Miao H, Yin Y (2017) An approach to data consistency checking for the dynamic replacement of service process. IEEE Access 5:11700–11711
8. He L, Chen C, Zhang T, Zhu H, Wan S (2018) Wearable depth camera: monocular depth estimation via sparse optimization under weak supervision. IEEE Access 6:41337–41345

9. Ronao CA, Cho S-B (2016) Human activity recognition with smartphone sensors using deep learning neural networks. Expert Syst Appl 59:235–244
10. Hassan MM, Uddin MZ, Mohamed A, Almogren A (2018) A robust human activity recognition system using smartphone sensors and deep learning. Futur Gener Comput Syst 81:307–313
11. Ignatov A (2018) Real-time human activity recognition from accelerometer data using convolutional neural networks. Appl Soft Comput 62:915–922
12. Wang L, Zhen H, Fang X, Wan S, Ding W, Guo Y (2019) A unified two-parallel-branch deep neural network for joint gland contour and segmentation learning. Futur Gener Comput Syst 100:316–324
13. Nweke HF, Teh YW, Al-Garadi MA, Alo UR (2018) Deep learning algorithms for human activity recognition using mobile and wearable sensor networks: State of the art and research challenges. Expert Systems with Applications
14. Lee D-G, Lee S-W (2019) Prediction of partially observed human activity based on pre-trained deep representation. Pattern Recogn 85:198–206
15. Yang Y, Hou C, Lang Y, Guan D, Huang D, Xu J (2019) Open-set human activity recognition based on micro-doppler signatures. Pattern Recogn 85:60–69
16. Saini R, Kumar P, Roy PP, Dogra DP (2018) A novel framework of continuous human-activity recognition using kinect. Neurocomputing
17. Khan MUS, Abbas A, Ali M, Jawad M, Khan SU, Li K, Zomaya AY (2018) On the correlation of sensor location and human activity recognition in body area networks (bans). IEEE Syst J 12(1):82–91
18. Chen Z, Zhang L, Cao Z, Guo J (2018) Distilling the knowledge from handcrafted features for human activity recognition. IEEE Transactions on Industrial Informatics
19. Khalifa S, Lan G, Hassan M, Seneviratne A, Das SK (2018) Harke: Human activity recognition from kinetic energy harvesting data in wearable devices. IEEE Trans Mob Comput 17(6):1353–1368
20. Lv M, Chen L, Chen T, Chen G (2018) Bi-view semi-supervised learning based semantic human activity recognition using accelerometers. IEEE Transactions on Mobile Computing
21. Cheng W, Erfani SM, Zhang R, Kotagiri R (2018) Learning datum-wise sampling frequency for energy-efficient human activity recognition. In: AAAI
22. Rokni SA, Nourollahi M, Ghasemzadeh H (2018) Personalized human activity recognition using convolutional neural networks. arXiv:1801.08252
23. Yin Y, Chen L, Xu Y, Wan J, Zhang H, Mai Z (2019) Qos prediction for service recommendation with deep feature learning in edge computing environment. Mobile Networks and Applications, pp 1–11
24. Gao Z, Wang D, Wan S, Zhang H, Wang Y (2019) Cognitive-inspired class-statistic matching with triple-constrain for camera free 3d object retrieval. Futur Gener Comput Syst 94:641–653
25. Wan S, Zhao Y, Wang T, Gu Z, Abbasi QH, Choo K-KR (2019) Multi-dimensional data indexing and range query processing via voronoi diagram for internet of things. Futur Gener Comput Syst 91:382–391
26. Gao H, Mao S, Huang W, Yang X (2018) Applying probabilistic model checking to financial production risk evaluation and control: A case study of alibaba's yu'e bao. IEEE Transactions on Computational Social Systems 99:1–11
27. Chen Y, Deng S, Ma H, Yin J (2019) Deploying data-intensive applications with multiple services components on edge. Mobile Networks and Applications, pp 1–16
28. Anguita D, Ghio A, Oneto L, Parra X, Reyes-Ortiz JL (2013) A Public Domain Dataset for Human Activity Recognition Using Smartphones. In: Esann
29. Ghio A, Oneto L (2014) Byte the bullet: Learning on real-world computing architectures. In: ESANN
30. Noor MHM, Salcic Z, Kevin I, Wang K (2017) Adaptive sliding window segmentation for physical activity recognition using a single tri-axial accelerometer. Pervasive and Mobile Computing 38:41–59
31. Lee Y-S, Cho S-B (2014) Activity recognition with android phone using mixture-of-experts co-trained with labeled and unlabeled data. Neurocomputing 126:106–115
32. Chen Z, Zhu Q, Soh YC, Zhang L (2017) Robust human activity recognition using smartphone sensors via ct-pca and online svm. IEEE Transactions on Industrial Informatics 13(6):3070–3080
33. Cao L, Wang Y, Zhang B, Jin Q, Vasilakos AV (2018) Gchar: An efficient group-based context—aware human activity recognition on smartphone. Journal of Parallel and Distributed Computing 118:67–80
34. Bao L, Intille SS (2004) Activity recognition from user-annotated acceleration data. In: International Conference on Pervasive Computing. Springer, pp 1–17
35. Wu W, Dasgupta S, Ramirez EE, Peterson C, Norman GJ (2012) Classification accuracies of physical activities using smartphone motion sensors. Journal of Medical Internet Research 14(5):e130
36. Zhao Y, Li H, Wan S, Sekuboyina A, Hu X, Tetteh G, Piraud M, Menze B (2019) Knowledge-aided convolutional neural network for small organ segmentation. IEEE Journal of Biomedical and Health Informatics
37. Ding S, Qu S, Xi Y, Wan S (2019) Stimulus-driven and concept-driven analysis for image caption generation, Neurocomputing. https://doi.org/10.1016/j.neucom.2019.04.095
38. Xu X, Xue Y, Qi L, Yuan Y, Zhang X, Umer T, Wan S (2019) An edge computing-enabled computation offloading method with privacy preservation for internet of connected vehicles. Future Generation Computer Systems 96:89–100
39. Li W, Liu X, Liu J, Chen P, Wan S, Cui X (2019) On improving the accuracy with auto-encoder on conjunctivitis. Applied Soft Computing, p 105489
40. Reyes-Ortiz J-L, Oneto L, Samà A, Parra X, Anguita D (2016) Transition-aware human activity recognition using smartphones. Neurocomputing 171:754–767
41. Kwapisz JR, Weiss GM, Moore SA (2011) Activity recognition using cell phone accelerometers. ACM SigKDD Explorations Newsletter 12(2):74–82
42. Anguita D, Ghio A, Oneto L, Parra X, Reyes-Ortiz JL (2012) Human activity recognition on smartphones using a multi-class hardware-friendly support vector machine. In: International workshop on ambient assisted living, Springer, pp 216–223
43. Wan S, Gu Z, Ni Q (2019) Cognitive computing and wireless communications on the edge for healthcare service robots. Computer Communications. https://doi.org/10.1016/j.comcom.2019.10.012
44. Chen Y, Shen C (2017) Performance analysis of smartphone-sensor behavior for human activity recognition. IEEE Access 5:3095–3110

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

Affiliations

Shaohua Wan[1,2] · **Lianyong Qi**[3] · **Xiaolong Xu**[4] · **Chao Tong**[5] · **Zonghua Gu**[6,7]

[1] School of Information and Safety Engineering, Zhongnan University of Economics and Law,
Wuhan, 430073, China

[2] State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210023, China

[3] School of Information Science and Engineering, Qufu Normal University, Rizhao, 276826, China

[4] School of Computer and Software, Nanjing University of Information Science and Technology,
Nanjing, 210044, China

[5] School of Computer Science and Engineering, Beihang University, Beijing, 100191, China

[6] Department of Applied Physics and Electronics, Umeå universitet, 90187, Umeå, Sweden

[7] College of Computer Science, Zhejiang University, Hangzhou 310027, China