Hindawi
Mathematical Problems in Engineering
Volume 2020, Article ID 5426532, 10 pages
https://doi.org/10.1155/2020/5426532

Research Article

A Novel Multichannel Dilated Convolution Neural Network for Human Activity Recognition

Yingjie Lin and Jianning Wu

College of Mathematics and Informatics, Fujian Normal University, Fuzhou, China

Correspondence should be addressed to Jianning Wu; jianningwu@fjnu.edu.cn

Received 15 January 2020; Accepted 19 June 2020; Published 11 July 2020

Academic Editor: Francesca Vipiana

A novel multichannel dilated convolution neural network for improving the accuracy of human activity recognition is proposed. The proposed model utilizes the multichannel convolution structure with multiple kernels of various sizes to extract multiscale features of high-dimensional data of human activity during convolution operation and not to consider the use of the pooling layers that are used in the traditional convolution with dilated convolution. Its advantage is that the dilated convolution can first capture intrinsical sequence information by expanding the field of convolution kernel without increasing the parameter amount of the model. And then, the multichannel structure can be employed to extract multiscale gait features by forming multiple convolution paths. The open human activity recognition dataset is used to evaluate the effectiveness of our proposed model. The experimental results showed that our model achieves an accuracy of 95.49%, with the time to identify a single sample being approximately 0.34 ms on a low-end machine. These results demonstrate that our model is an efficient real-time HAR model, which can gain the representative features from sensor signals at low computation and is hopeful for the effective tool in practical applications.

1. Introduction

Human activity recognition (HAR) is a typical multiclassification problem, which acquires and analyzes human activity-related data to identify human activity status [1, 2]. It plays an essential role in people's daily life and is widely used in the fields of safety, medical care, smart home, and entertainment. Specific applications include smart home [3, 4], gait analysis [5, 6], security certification [7, 8], health monitoring [9, 10], athlete monitoring [11], and gesture recognition [12, 13]. There are two main methods of human activity recognition: vision-based human activity recognition and sensor-based human activity recognition. Although the vision-based recognition method has been extensively studied and can achieve a high recognition rate, this method is limited by the high acquisition cost of the imaging device, and it is a challenge to collect the image data sometimes, so it is hard to meet the needs of the real-life environment. With the development of smartphones and wearable sensor technologies, smart devices with built-in sensors are characterized by low cost, convenient carrying, and good real-time performance. Therefore, HAR based on sensor signals has become the focus of research in this field.

HAR based on sensor signals includes two methods: the traditional method and the deep learning method. The traditional method based on sensor signal for HAR needs complex preprocessing of the raw data and relies on manual experience to extract the required time-domain features [14–16], frequency-domain features [16–19], and other features [20, 21]. These hand-craft features are shallow features, which would inevitably lose some implicit key features. Deep learning methods can make up for the shortcomings of traditional methods and can dig out automatically the more recognizable inherent features contained in the data by learning the deep nonlinear network structure.

Deep learning is well known as a revolution in machine learning, especially in the field of computer vision [22, 23]

and natural language processing [24, 25]. In recent years, different deep learning methods have been proposed for human activity recognition based on sensor signals, including autoencoders [26], fully connected deep neural network (DNN) [27, 28], recurrent neural network (RNN), convolutional neural networks (CNN), and the hybrid deep learning model. RNN, CNN, and hybrid models are the most widely studied in HAR, and we will introduce them in detail in the second section. RNN, especially Long Short-Term Memory (LSTM), can capture the dynamic time dependence of various motions and helps to explore the pattern features [2]. However, the LSTM takes a longer training time due to numerous parameters that need to be updated during the training process. Compared with RNN, CNN is more able to learn the crucial features contained in recursive patterns [1, 29]. However, most CNNs have a single parameter setting in the convolution process, which dramatically limits the flexibility of the model. Besides, a larger convolution kernel can help to capture more information but increases computation cost for CNN. The application of dilated convolution may be an effective solution, which achieves dilating the receptive field of the convolution kernel without increasing the kernel parameter numbers [23].

Real-time HAR is also a research hotspot in the field. Some methods for this have been proposed to implement this problem [30, 31]. The shortcoming of these works is that it is difficult to maintain the balance between activity recognition accuracy and running time. All of these challenges have led researchers to develop efficient recognition methods with high recognition accuracy and low computational complexity effectively solving these problems.

Based on current research deficiencies, this paper proposes a novel multichannel dilated convolution neural network (MDCNN). The model can get a larger receptive field to extract global features of long-time series from the raw sensor data by using dilated convolution rather than traditional convolution structure. Moreover, the proposed model uses multichannel block convolution operations with different kernel sizes to obtain combined features of multiscale. Through experimental comparison, the proposed model can effectively improve recognition accuracy and achieves real-time HAR effectively.

The rest of this paper is organized as follows: Section 2 provides related work concerning different deep learning methods for HAR. Section 3 describes the fundamentals of CNN, dilated convolution, and multichannel convolution. The framework and training process of the proposed model are introduced in Section 4. Section 5 conducts a series of experiments with the proposed model and discusses the results, while Section 6 gives the conclusion and presents our future work.

2. Related Work

In recent years, various deep learning methods have been proposed for sensor-based HAR. RNN can retain memory and learn sequence data to capture the inherent relationships of time-series data. Chen et al. proposed the LSTM-based method that uses three-axis accelerometer data on the lab public datasets (WISDM) to identify human activities with an accuracy of 92.1% [32]. Guan and Plötz developed ensembles of deep LSTM, which combines sets of diverse LSTM learners into classifier collectives [33]. The experimental result on three standard benchmarks (Opportunity, PAMAP2, and Skoda) demonstrates that Ensembles of deep LSTM outperform individual LSTM networks. However, the deep LSTM takes a longer training time due to numerous parameters that need to be updated during the training process. For enhancing faster learning in early training, Zhao et al. proposed an improved LSTM model: Res-Bidir-LSTM, which also guarantees the validity of information transmission through residual connections and bidirectional cells [34]. The result shows that Res-Bidir-LSTM has increased by around 4% under the public domain UCI dataset and the Opportunity dataset in comparison with previous work. Hammerla et al. explored the three types of deep learning models (deep feed-forward networks (DNN), CNN, and RNN) on the three benchmark datasets in labs [35]. The results found that CNN has better performance than other models on prolonged activities like walking and running.

There are two advantages to CNN: local dependence and scale invariance [1, 36]. Local dependence means that the signal at the current time may be related to the signal around this point, and scale invariance refers to the fact that the research object does not change in the amplitude or frequency of the synchronization [37]. Zeng et al. proposed an original CNN model for accelerometer data, in which each axis of acceleration is input to a separate convolution layer and a pooling layer, respectively, to extract features [36]. However, due to the fact that the model only considering the acceleration data and model structure is too simple, it is tough to extract crucial features.

Ronao and Cho constructed three layers of CNN, automatically extract robust features from the raw data to raise the accuracy of HAR, and get the UCI dataset and WISDM dataset [38]. They further improve the performance of their model by using additional information from the fast Fourier transform (FFT) of the raw data. However, both the convolution and the pooling operations of the method are performed in a single channel; this single parameter setting dramatically limits the flexibility of the parameters, so the network is unable to extract efficient global and local features at multiple scales. Mohammad et al. presented the multiple CNN pipelines with the structure of late fusion and bypassing connections [39]. This model can comfortably accommodate multiple sensors and signal representations such as the time-domain data, FFT information, and spectrogram, achieving higher performance for six publicly available datasets. However, it is computationally expensive compared with earlier methods due to the employment of bypassing connections from all layers [39].

Besides single models, the hybrid deep learning model combines CNN and RNN which are also proposed in a few works. Ordóñez and Roggan proposed a generic deep framework for activity recognition based on convolutional

and LSTM recurrent units [40], where CNN acts as a feature extractor and LSTM models the temporal dynamics of the extracted feature maps. However, this complex network framework suffered from low efficiency and can hardly meet real-time requirements in practice applications.

Most of the current research is carried out offline, and some works have realized HAR in real time. Inoue et al. proposed a deep recurrent neural network (DRNN) for HAR with a high recognition rate and a high throughput [30]. However, despite reducing the training time by parallel processing using the GPU, this network was still very large [41]. Cao et al. proposed a Group-based Context-aware human activity recognition (GCHAR) classification method to achieve HAR in real time, which used a hierarchical group-based classification scheme and context awareness to enhance the classification performance [31]. The result shows that training time and testing time are shorter than other comparison algorithms. The classification accuracy is 94.16%, which is slightly lower than the deep learning algorithm. Therefore, the core of the current work is to achieve a model with high recognition accuracy and low computational complexity.

3. Methodology

3.1. Convolutional Neural Network for HAR. CNN is a multilayered deep network structure consisting of the input layer, the convolutional layer, the pooling layer, the fully connected layer, and the output layer. Among these layers, the alternating convolutional and pooling layers constitute the most prominent structure. Various studies in the field of computer vision have shown that a multilayer CNN structure consisting of a convolutional layer and a pooling layer can extract image features with different levels. At the bottom of the CNN, it generally learns basic features such as local textures and lines of the image. As the network layer deepens, the model learns more and more complex features, and its recognition ability is also raised from identifying the contour of the object enough to identify the entire image.

In CNN, the convolution and pooling operation are performed in sequence: the output of the convolution operation is used as the input of the pooling operation, and then the pooled layer result is used as the input of the next convolution layer and so on and finally sent into the Softmax layer.

Considering that the sensor data belongs to a one-dimensional time series, the input of the proposed multichannel dilated convolution network is one-dimensional time-series data, so its convolution kernel adopts a one-dimensional structure. The output of each convolutional layer and pooling layer is also corresponding to a one-dimensional feature vector, where the accelerometer and gyroscope time-series data inputs are expressed as

$$x = [x_1, \ldots, x_N], \tag{1}$$

where N denotes the length of the time window.

In the convolutional layer, CNN uses the convolution kernel to cope with the input data. Each convolutional layer is connected to the data in the local receptive field of the previous layer to extract local features in the local receptive field. Each particular convolution kernel can extract a differential feature. The data obtained after the convolution operation of a convolution kernel is a feature map, so we can obtain multiple feature maps through multiple convolution kernels to extract multiple features. The output of the convolutional layer is

$$c_i = \sigma\left(\sum_{j=1}^{J} \left(w^j\right)^T x^j_{i:\ i+l-1} + b^j\right), \tag{2}$$

where σ is the activation function. The Restricted Linear Unit (ReLU) function [42] is widely used in deep learning to improve the performance of a deep neural network for nonlinear transformation. b^j is the bias term for the jth feature map, l is the kernel size, and w^j is the weight for the feature map j.

In the pooling layer, CNN aggregates the local features of a particular region to obtain the scale-invariant feature transform. The pooling operation reduces the dimension of processing data and the computational cost while extracting useful information. The pooling operation used in this paper, max pooling, is characterized by outputting the maximum value among a set of nearby inputs, given by

$$p^j_i = \max_{r \in R}\left(c^j_{i \times T + r}\right), \tag{3}$$

where R is the pooling size and T is the pooling stride. With the stacking of convolutional layers and pooling layers, this sparse connection method can significantly reduce the number of parameters while extracting the deep features of the input data layer by layer. The obtained multichannel feature map information is first converted into a 1-dimensional vector and then input into the Softmax layer. The converted 1-dimensional vector form is $p = [p_1, \ldots, p_I]$, where I is the number of units in the last pooling layer. The number of the Softmax layer neurons is consistent with the number of activity categories. The Softmax layer gets the probability distribution of each type of activity, and the type of activity identified by the model is the activity type corresponding to the highest probability. The process is expressed as

$$q_k = \frac{\exp(p)}{\sum_{k=1}^{N_c} \exp(p)}, \tag{4}$$

where c is the activity class and N_c is the total number of activity classes. Forward propagation is performed through the above process, which gives the error values of the network.

Batch Normalization (BN) is proposed to improve the performance of CNN [43]. The BN layer can improve the data distribution during training and speed up the training of the model. Also, the BN layer has the characteristics of improving network generalization ability, to avoid the problem of overfitting and gradient disappearing during training [44]. Define the input dataset of a hidden layer of the network as $\{\mu_1, \ldots, \mu_m\}$, m is the number of samples in the batch. First, it should compute the mean value $E(\mu)$ and variance $D(\mu)$ by

$$E(\mu) = \frac{1}{m}\sum_{h=1}^{m}\mu_h, \tag{5}$$

$$D(\mu) = \frac{1}{m}\sum_{h=1}^{m}\left[\mu_h - E(\mu)\right]^2. \tag{6}$$

Then, each dimension is normalized to $\widehat{\mu_h}$, whose distribution has the expected value of 0 and the variance of 1:

$$\widehat{\mu_h} = \frac{\mu_h - E(\mu)}{\sqrt{D(\mu) + \varepsilon}}, \tag{7}$$

where ε is a positive number close to zero. Finally, a pair of parameters γ and β are introduced to reconstruct and transform the data; the output data y of the BN layer is as follows:

$$y_h = \alpha\widehat{\mu_h} + \beta. \tag{8}$$

Parameters α and β are learned along with the original model parameters.

In general, CNN finally derives robust features with the invariant character for translation, rotation, and scale from the raw data. It is as a result of the convolution operations of multiple convolution kernel network structure, which extracts the features contained in the data, and the extracted features are abstracted as the number of network layers increases. Also, due to the characteristics of sparse connections and weight sharing, CNN can reduce the number of parameters in model training and avoid overfitting [37].

3.2. Dilated Convolution. In the traditional CNN, the pooling operation can make the convolution kernel get a larger receptive field, but it is not a strict component of CNN actually [40]. Meanwhile, excessive pooling operations tend to result in a large amount of information loss [23]. Dilated convolution can expand the receptive field without pooling, allowing each convolution output to contain a wide range of information, and has been applied to problems that require longer sequence information dependencies such as speech and text. The inertial sensor signal is a typical time series, so we apply the dilated convolution to the human activity recognition model in this paper.

The principle of dilated convolution is to fill a fixed element 0 that will not adjust during the learning process between the original convolution kernels, which achieves the purpose of dilating the receptive field of the convolution kernel without increasing the number of kernel parameters [23]. The dilated convolution operation is a variant of the traditional convolution operation. If we denote r as the dilation factor, the one-dimensional mathematics of the dilated convolution are as follows:

$$z[i] = \sum_{l=1}^{L} x[i + dl]w[l], \tag{9}$$

where $x[i]$ and $z[i]$ denote the input signal and output signal, respectively; l denotes the size of the convolution kernel; d denotes the dilatation rate. One-dimensional dilated convolution is achieved by inserting "0" between the pixels of the convolution kernel. For a $1*k$ convolution kernel, the dilation factor d is k_d, and the size of k_d can be defined as

$$k_d = k + (k-1)(d-1). \tag{10}$$

The convolution kernel transformed by the dilation factor of $d = 3$ can be expressed as shown in Figure 1.

As can be seen from Figure 1, the $1*3$ convolution kernel becomes a $1*7$ dilated convolution kernel after the dilated operation with the dilatation factor $d = 3$.

The function of the convolution kernel is to identify certain features in the time series of the sensor. When a segment of the time series satisfies the identifiable feature of the convolution kernel, according to (9), the calculated results of the segment activate a larger value z in the new feature map and finally achieve the recognition of the features of the time series. Figure 1 reveals the change in the receptive field of the convolution kernel after the addition of the dilated convolution.

Figure 2 shows an example of dilated convolution with a three-layer convolution structure. In the third layer of convolutional layer, the traditional CNN can only capture three inputs before and after the sensor time series. Under the same conditions, dilated CNN can capture seven input data before and after. Also, dilated CNN has no change in the parameter quantity compared with the traditional CNN.

Without reducing the resolution of the feature map through the pooling layer, the dilated convolution can learn more deep essential features, thus effectively avoiding the problem of severe loss of local detail information in the sensor data. Furthermore, the convolution layer uses different dilated factors to get various sizes of convolution kernel receptive fields and then extract activity features of multiscale.

3.3. Multichannel Block Convolution Network Structure. Although traditional CNNs use filters to capture different features of an instance [25], they perform convolution operations in a single channel, which greatly limits the flexibility of parameter settings and cannot extract global and local features on multiple scales effectively. In order to enhance the robustness of the model, CNN can adopt the group convolution, that is, adopt a multichannel structure; each channel uses different convolution kernel sizes, corresponding to extracting features of different scales of the original sensor time series. Therefore, it can be seen as a fusion method of multiscale features. Figure 3 is the diagram of multichannel convolution.

In multichannel CNN, the convolution operations are grouped into multiple branches and carried out separately, and then the fully connected layer concatenates the feature maps of the branches on the channel. By using different kernels, the features of large-scale convolution kernel learning have more global characteristics, while small-scale convolution kernels get features that better reflect local characteristics.

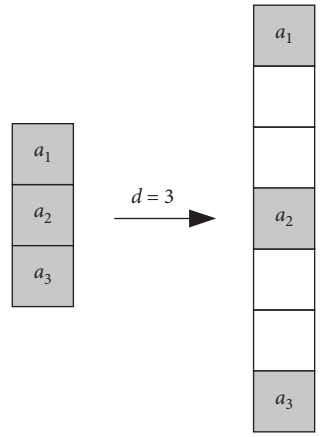

Figure 1: Convolution kernel undergoes dilated operation with dilatation rate $d = 3$.

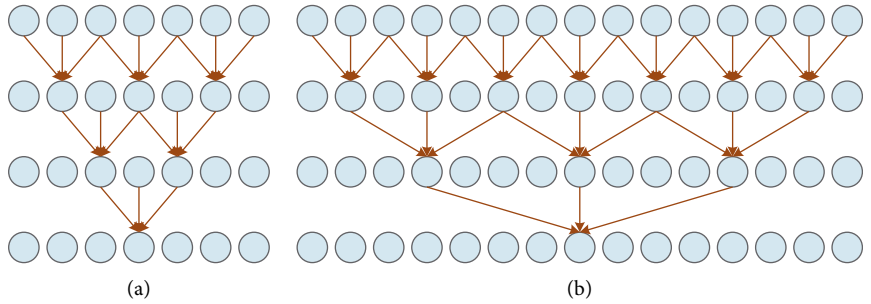

Figure 2: Diagram of dilated convolution and traditional convolution. (a) Traditional convolution. (b) Dilated convolution.

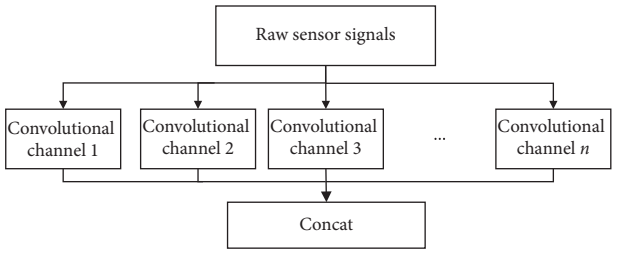

Figure 3: Diagram of multichannel convolution.

4. Principle of Multichannel Dilated Convolution Model

4.1. Model Overview. The MDCNN model proposed in this paper is shown in Figure 4. The whole model composes two parts: feature extraction and classification. The feature extraction part is composed of three dilated convolution channels, Flatten layer, and Concat layer, wherein the dilated convolution channels are the core of MDCNN. Firstly, the sensor data is sent to the dilated convolution channels to extract features of different scales, and the three dilated convolution channels are independent of each other. Then, the Flatten layer "flattened" the other dimensions except for the time dimension into a one-dimensional feature vector and sent it to the Concat layer to concatenate the one-dimensional feature vector of each dilated convolution channel for feature splicing. Finally, in the classification part, the Softmax layer calculates the probability distribution of each type of activity for the feature parameters transmitted from the Concat layer, and the type of activity identified by our model is the activity corresponding to the highest probability in the probability distribution.

The dilated convolution channel 1 is composed of three dilated convolution layers. The model firstly extracts the features with the receptive field increasing sequentially by the dilated convolution layer with dilated factors of 2, 3, and 4, respectively. The BN layer is connected to each of the convolution layers before activation in order to increase the rate of network learning and reduce the risk of overfitting. Finally, the previously obtained feature is flattened into the fully connected layer. The structure of the dilated convolution channel 2 and channel 3 is similar to channel 1, and their convolution kernel sizes are $1*4$ and $1*7$, respectively.

4.2. Model Training of MDCNN. The multichannel dilated convolution model discards the pooling layer on the basis of the traditional CNN, avoiding reducing resolution of the feature map caused by the pooling operation. The proposed model introduces the dilated convolution kernel to increase the receptive field of the convolution kernel and captures the long sequence information on the sensor time series, and the multichannel structure is able to extract features of multiscale.

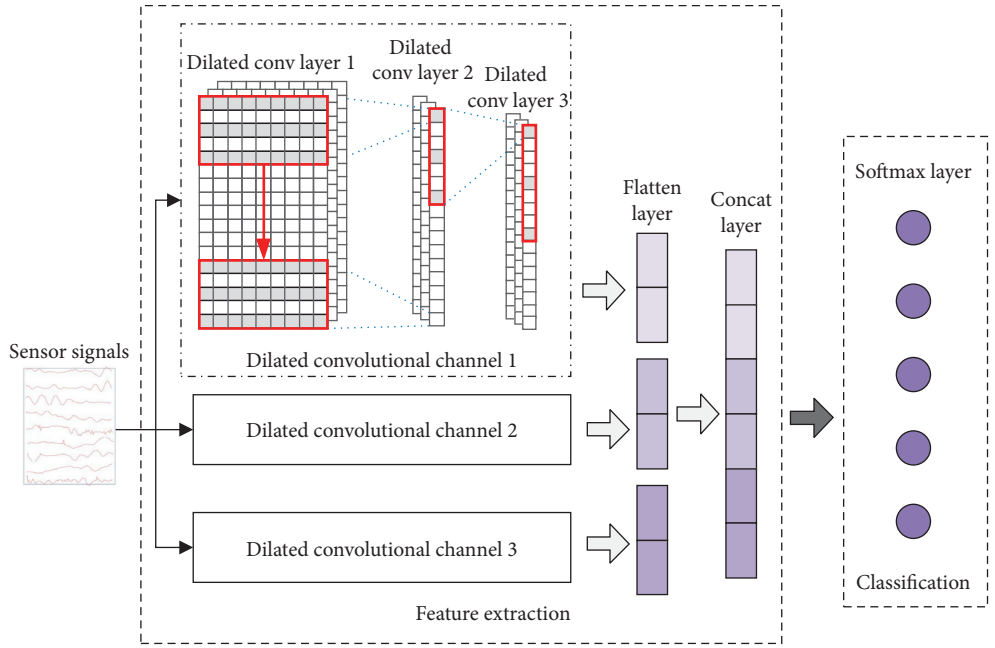

Figure 4: The overall structure of MDCNN.

The training and optimization of CNN depend on the loss function. The loss function calculates the error between the predicted value and the true value, backpropagates the error from the last layer to each layer of the network through the backpropagation algorithm, and updates the weights. The updated parameters continue to participate in the training, looping back and forth until the loss function value reaches the minimum; that is, the goal of the final training is reached. In this paper, the CNN model training uses a cross-entropy loss function, and it is computed by

$$E_0(x_m, y_m) = -\frac{1}{M}\left[\sum_{i=1}^{M}\sum_{k=1}^{Nc} y_{m,k}\log\left(q_{m,k}\right)\right], \tag{11}$$

where x_m is the training sensor data, $q_{m,k}$ is m-th sample k-th data's predicted label, $y_{m,k}$ is a one-hot vector that represents the label of the k-th data of the m-th sample, M is the total number of samples, and Nc is the total number of label classes.

Large weights can cause the weight vector to get stuck in a local minimum easily since gradient descent only makes small changes to the direction of optimization. This will eventually make it hard to explore the weight space [38]. L_2 regularization is a regularization method that adds an extra term into the cost function that penalizes large weights. For each set of weights, the penalizing term is added to the LOSS function:

$$E = E_0 + \lambda\sum_{\theta}\theta^2, \tag{12}$$

where E_0 is the loss function without L_2 regularization, λ is the regularization coefficient, and θ is the overall weight of the model. In summary, the standardized data training set is input to MDCNN, and the model parameters are trained to obtain the recognition model.

5. Experiment

5.1. Experiment Dataset. We used smartphones dataset (HAR dataset) [45] in the UCI Machine Learning Repository in our experiments. The dataset collected a total of 10,299 sensor data from 30 subjects between the ages of 19 and 48 in lab. The dataset included six modes of action: walking, going upstairs, going downstairs, sitting, standing, and lying down, each subject carrying a smartphone to record sports data. Each subject carries a smartphone to record motion data, and the recorded data is accelerometer data and gyroscope data with a frequency of 50 Hz. The accelerometer data is separated into total acceleration and body acceleration data, and all data are then preprocessed using a noise filter and finally split into 128×9 data windows with 50% overlap between each window. The dataset also offers 561 time and frequency-domain features, but we do not use these features in our experiments. Figure 5 is a schematic diagram showing the structure of a 128×9 sensor data used in the experiment. The dataset is divided into a training set and a test set in a 7 : 3 ratio for the experiment. Table 1 is a description of the composition of the human activity dataset.

5.2. Experiment Result. The experimental environment of this article is a laptop with the CPU of Intel i5-8250U and RAM of 8 GB. The programming language is Python 3.7, and the framework is Keras with Tensorflow backend. In order to make the experimental process more efficient, the sample data was sent to the model experiment in batches with a batch block size of 32. The model used the Adam update rules to optimize training parameters to minimize losses and set the maximum number of training iterations to 150. The learning rate was set to 0.0015. We trained the model and

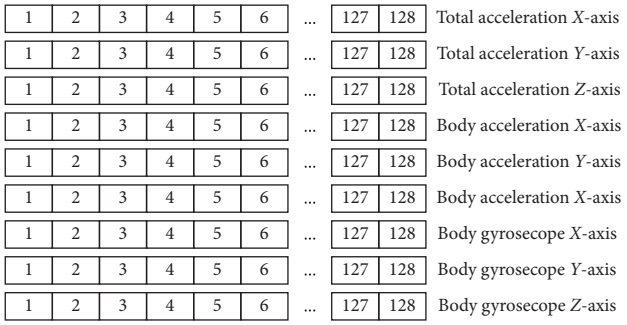

Figure 5: Diagram of the timing structure of the human activity identification dataset.

Table 1: Composition description of human activity dataset for experiment.

Activity types	Sample size	Proportion (%)
Walking	1722	16.72
Walking downstairs	1544	14.99
Walking upstairs	1406	13.66
Sitting	1777	17.24
Standing	1906	18.51
Laying	1944	18.88

tested in the test set and finally got the classification confusion matrix of Table 2.

As can be seen from Table 2, the proposed model achieved excellent recognition results that the accuracy is 95.49%, and the precisions of walking and lying down are over 98.5%. It can be found that the proposed model has a slightly lower F1 score in distinguishing between behavior patterns of sitting and standing, mainly because the two behavior patterns are both static states. The waveform of the signal collected by the sensor at rest is so low that the model cannot extract enough information from the sensor data to distinguish between the two types adequately. At the same time, it may be that CNN has some weaknesses in static activities' identification. The next step is to improve the model further to improve the recognition accuracy for static state activities.

We compare the accuracy of the MDCNN to the other algorithms in literature according to experiment results, which are shown in Table 3. Firstly, compared with traditional methods (SVM; GHAR), our model shows a significant improvement; traditional methods rely heavily on hand-craft features. These hand-craft features from traditional methods are shallow features, which would inevitably lose some implicit key features. Secondly, we conduct experiments to compare neural network models (LSTM, CNN, and DRNN). For the three networks, CNN performs better than RNN or LSTM. CNN has advantages in feature extraction: the convolution kernel extracts abstract high-level gait features through layers, which have a decisive role in the final classification. Compared with RNN, CNN is more able to learn the crucial features contained in recursive patterns in complex cyclic processes such as gait [35].

It can be seen from Table 3 that the proposed model gets the highest recognition accuracy in addition to CNN in [38] and the multiple CNN [39]. The two CNN models incorporate frequency-domain features. The frequency-domain features seem to provide global information that is difficult to obtain in the CNN automatic feature extraction process. CNN is paying more attention to local features rather than global features. It is difficult to extract global information to a limited extent with the traditional CNN convolution kernel length. After adding the dilated convolution structure to the convolution layer, the actual length of the convolution kernel is increased, and the receptive field of the widened convolution kernel can extract longer context information. The experimental results prove that our model has improved over the ordinary CNN model that does not rely on manual features. How to extract more global features from our model would be our future work.

The identification model also needs to consider the calculation cost. CNN in [38] and the multiple CNN [39] have complex network framework, which incurs expensive computational costs and hardly meets real-time requirements in practice applications. Besides, both of them used the FFT feature, while the multiple CNN additionally used the spectrogram. The additional feature extraction consumes much time, which is also a hassle for real-time calculations. In contrast to them, the proposed model achieves almost similar performance using only raw sensor data without any manual features. MDCNN implements real-time HAR, which is difficult for these two complex models. Also, its training time and testing time are superior to other real-time deep learning models: the training time per epoch is about 6.01 s running on a laptop with the CPU, while DRNN took 116.39 s per epoch in the GPU environment. It takes only 15 minutes to complete the training process in our model, and it is hard to be negligible. However, the training process only needs to be run once in a practical application. The device loaded with a pretrained model can identify measured data in real time. In our experiment, MDCNN completed the identification of all samples within 1 s 323 μs; that is, the time to identify each sample is 0.34 ms. Because the frequency of the sensor's data acquisition is 50 Hz, our model is sufficient to achieve real-time HAR. It is because CNN can perform parallel operations well in the training process. Furthermore, the dilated convolution achieves a more efficient convolution operation under the same computational complexity.

In general, the proposed model achieves real-time HAR with high recognition accuracy and low computational complexity. The model can automatically and efficiently mine the deep and highly recognizable essential features embedded in the data. More importantly, MDCNN expands the receptive field by introducing dilated convolution without increasing parameter, so that the model can mine the timing dependency information in the long sequence to some extent, which makes up for the defects of the traditional in time-series problems.

5.3. Network Structure Analysis. This section analyzes the impact of network structure on accuracy in the proposed

Table 2: Confusion matrix of human activity recognition classification.

Activity type		Prediction						Recall (%)	F1 score (%)
		Walking	Downstairs	Upstairs	Sitting	Standing	Laying		
True	Walking	455	15	26	0	0	0	91.73	95.09
	Downstairs	6	462	1	2	0	0	98.09	97.16
	Upstairs	0	2	418	0	0	0	99.52	96.65
	Sitting	0	1	0	427	55	8	86.97	91.14
	Standing	0	0	0	17	515	0	96.80	93.47
	Laying	0	0	0	0	0	537	100	99.26
	Precision (%)	98.70	96.25	93.93	95.74	90.35	98.53	Accuracy: 95.49%	

Table 3: Comparison of accuracy with other models.

Model	Accuracy (%)	Real time
SVM [16]	89.3	True
LSTM [32]	92.1	No
Res-Bidir-LSTM [34]	93.6	No
CNN [38]	94.79	No
CNN + FFT [38]	95.75	No
Multiple CNN [39]	95.5	No
DRNN [30]	95.42	True
GCHAR [31]	94.16	True
Proposed model	95.49	True

Table 4: Recognition accuracy of different network structures.

Network structure	Accuracy (%)
Pool size of 2	95.30
Pool size of 3	95.18
Without pooling	95.49

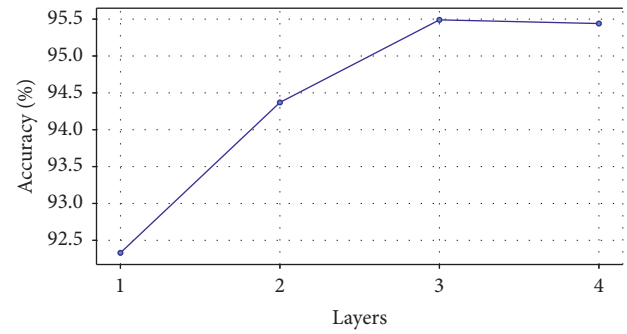

Figure 6: Recognition accuracy of different layers.

model. Firstly, we design an experiment to verify whether the pooling layer is necessary for the proposed model. This experiment was compared by the difference in accuracy between the proposed model and the model with the pooling layer. The pool size of the model with the pooled layer is 2 and 3, respectively. In both sets of experiments, the pooling layer was after the last layer of convolution. The results are shown in Table 4.

It can be seen from Table 4 that the proposed model can achieve higher accuracy than the two models with pooling layers. Also, the accuracy of the model with a large pool size is lower than that of the smaller pool size. The result is because the pooling layer reduces the amount of computation while reducing the resolution, which will lose some of the information useful for classification. As the size of the pool increases, the more information is lost, and the accuracy rate also decreases.

Secondly, we designed a comparison experiment with different layers of MDCNN, which verify the validity of the dilated convolution and analyze the influence of the network depth on the activity recognition accuracy. The experiment results are shown in Figure 6.

As can be seen from Figure 6, the recognition accuracy of MDCNN improves steadily with the increase of the number of layers in 1–3 layers. It is because the advantage of CNN is to mine the nonlinear network structure contained in the raw data. If the network is too shallow, it could not make full use of the powerful fitting model ability of CNN. However, the accuracy of MDCNN recognition of the four-layer network structure is lower than that of the three-layer network. This phenomenon indicates that the deep features extracted by the four-layer MDCNN do not contribute much to the recognition effect and may even extract redundant features, which affects the establishment of the human activity recognition model.

6. Conclusion

This paper proposes an improved multichannel dilated convolution neural network (MDCNN), which not only does not need to extract features manually and reduces the dependence on expert knowledge but also has achieved excellent recognition results in the experiment. At the same time, MDCNN is also a deep learning model that can achieve real-time HAR efficiently. By introducing the structure of dilated convolution and multichannel convolution, MDCNN effectively mines raw sensor data more comprehensively, further extracts more recognizable features, and increases the diversity of feature sets. The experiments also explored the influence of MDCNN structure on recognition accuracy and constructed an ideal human behavior recognition model. It is worth pondering that MDCNN, like other deep learning models, recognizes static activities with lower accuracy than dynamic activities, which requires further improvement. At the same time, the next step will be to apply MDCNN to more complex types of activity recognition.

Data Availability

The data used in this study are from published literature articles and therefore are publicly available.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the Humanities and Social Sciences Fund Project from the Ministry of Education, China (no. 17YJAZH091) and the Excellent Master Degree Thesis Cultivation Project of Fujian Normal University (LWPYS053).

References

[1] J. Wang, Y. Chen, S. Hao, X. Peng, and L. Hu, "Deep learning for sensor-based activity recognition: a survey," *Pattern Recognition Letters*, vol. 119, pp. 3–11, 2019.

[2] H. F. Nweke, Y. W. Teh, M. A. Al-garadi, and U. R. Alo, "Deep learning algorithms for human activity recognition using mobile and wearable sensor networks: state of the art and research challenges," *Expert Systems with Applications*, vol. 105, pp. 233–261, 2018.

[3] A. Jalal, M. A. Quaid, and M. A. Sidduqi, "A triaxial acceleration-based human motion detection for ambient smart home system," in *Proceedings of the 2019 16th International Bhurban Conference on Applied Sciences and Technology (IBCAST)*, pp. 353–358, IEEE, Islamabad, Pakistan, January 2019.

[4] A. Jalal, M. A. K. Quaid, and K. Kim, "A wrist worn acceleration based human motion analysis and classification for ambient smart home system," *Journal of Electrical Engineering & Technology*, vol. 14, no. 4, pp. 1733–1739, 2019.

[5] M. S. Singh, V. Pondenkandath, Z. Bo, P. Lukowicz, and M. Liwicki, "Transforming sensor data to the image domain for deep learning—an application to footstep detection," in *Proceedings of the 2017 International Joint Conference on Neural Networks (IJCNN)*, pp. 2665–2672, IEEE, Anchorage, AK, USA, May 2017.

[6] R. A. Anwary, H. Yu, and M. Vassallo, "An automatic gait feature extraction method for identifying gait asymmetry using wearable sensors," *Sensors*, vol. 18, no. 3, 2018.

[7] M. Muaaz and R. Mayrhofer, "Smartphone-based gait recognition: from authentication to imitation," *IEEE Transactions on Mobile Computing*, vol. 16, no. 11, pp. 3209–3221, 2017.

[8] N. Swati, S. Rajiv, and A. K. Misra, "Towards intelligent human behavior detection for video surveillance," in *Censorship, Surveillance, and Privacy*, pp. 884–917, IGI Global, Hershey, PA, USA, 2019.

[9] M. Alrige and S. Chatterjee, *Toward a Taxonomy of Wearable Technologies in Healthcare*, B. Donnellan, M. Helfert, J. Kenneally, D. VanderMeer, M. Rothenberger, and R. Winter, Eds., pp. 496–504, Springer, Berlin, Germany, 2015.

[10] N. A. Capela, E. D. Lemaire, and N. Baddour, "Feature selection for wearable smartphone-based human activity recognition with able bodied, elderly, and stroke patients," *PLoS One*, vol. 10, 2015.

[11] T. Kautz, B. H. Groh, J. Hannink, U. Jensen, H. Strubberg, and B. M. Eskofier, "Activity recognition in beach volleyball using a deep convolutional neural network," *Data Mining and Knowledge Discovery*, vol. 31, no. 6, pp. 1678–1705, 2017.

[12] Y. Kim and B. Toomajian, "Hand gesture recognition using micro-Doppler signatures with convolutional neural network," *IEEE Access*, vol. 4, pp. 7125–7130, 2016.

[13] J. C. Núñez, R. Cabido, J. J. Pantrigo, A. S. Montemayor, and J. F. Vélez, "Convolutional neural networks and long short-term memory for skeleton-based human activity and hand gesture recognition," *Pattern Recognition*, vol. 76, pp. 80–94, 2018.

[14] M. Shoaib, S. Bosch, O. Incel, H. Scholten, and P. Havinga, "Complex human activity recognition using smartphone and wrist-worn motion sensors," *Sensors*, vol. 16, no. 4, p. 426, 2016.

[15] M. A. Khan, H. M. Siddiqi, and S. Lee, "Exploratory data analysis of acceleration signals to select light-weight and accurate features for real-time activity recognition on smartphones," *Sensors*, vol. 13, no. 10, pp. 13099–13122, 2013.

[16] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. L. Reyes-Ortiz, *Human Activity Recognition on Smartphones Using a Multiclass Hardware-Friendly Support Vector Machine*, J. Bravo, R. Hervás, and M. Rodríguez, Eds., pp. 216–223, Springer, Berlin, Germany, 2012.

[17] Z. He and L. Jin, "Activity recognition from acceleration data based on discrete consine transform and SVM," in *Proceedings of the 2009 IEEE International Conference on Systems, Man and Cybernetics*, pp. 5041–5044, IEEE, San Antonio, TX, USA, October 2009.

[18] D. Figo, P. C. Diniz, D. R. Ferreira, J. M. P. Cardoso, and O. M. Cardoso, "Preprocessing techniques for context recognition from accelerometer data," *Personal and Ubiquitous Computing*, vol. 14, no. 7, pp. 645–662, 2010.

[19] A. Anjum and M. U. Ilyas, "Activity recognition using smartphone sensors," in *Proceedings of the 2013 IEEE 10th Consumer Communications and Networking Conference (CCNC)*, pp. 914–919, IEEE, Las Vegas, NV, USA, January 2013.

[20] S. Seto, W. Zhang, and Y. Zhou, "Multivariate time series classification using dynamic time warping template selection for human activity recognition," in *Proceedings of the 2015 IEEE Symposium Series on Computational Intelligence*, pp. 1399–1406, IEEE, Cape Town, South Africa, December 2015.

[21] Y.-P. Chen, J.-Y. Yang, S.-N. Liou, G.-Y. Lee, and J.-S. Wang, "Online classifier construction algorithm for human activity detection using a tri-axial accelerometer," *Applied Mathematics and Computation*, vol. 205, no. 2, pp. 849–860, 2008.

[22] R. Meng, S. G. Rice, J. Wang, and X. Sun, "A fusion steganographic algorithm based on faster R-CNN," *Computers, Materials and Continua*, vol. 55, pp. 1–16, 2018.

[23] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," 2016, https://arxiv.org/abs/1511.07122.

[24] D. Zeng, Y. Dai, F. Li, J. Wang, and A. K. Sangaiah, "Aspect based sentiment analysis by a linguistically regularized CNN with gated mechanism," *Journal of Intelligent & Fuzzy Systems*, vol. 36, no. 5, pp. 3971–3980, 2019.

[25] D. Zeng, Y. Dai, F. Li, R. S. Sherratt, and J. Wang, "Adversarial learning for distant supervised relation extraction," *Computers, Materials and Continua*, vol. 55, pp. 121–136, 2018.

[26] A. Wang, G. Chen, C. Shang, M. Zhang, and L. Liu, *Human Activity Recognition in a Smart Home Environment with*

Stacked Denoising Autoencoders, S. Song and Y. Tong, Eds., pp. 29–40, Springer, Berlin, Germany, 2016.

[27] P. Vepakomma, D. De, S. K. Das, and S. Bhansali, "A-wristocracy: Deep learning on wrist-worn sensing for recognition of user complex activities," in *Proceedings of the 2015 IEEE 12th International Conference on Wearable and Implantable Body Sensor Networks (BSN)*, IEEE, Cambridge, MA, USA.

[28] V. Radu, C. Tong, S. Bhattacharya et al., "Multimodal deep learning for activity and context recognition," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 1, no. 4, pp. 1–27, 2018.

[29] N. Y. Hammerla, S. Halloran, and T. Ploetz, "Deep, convolutional, and recurrent models for human activity recognition using wearables," *Journal of Scientific Computing*, vol. 61, pp. 454–476, 2016.

[30] M. Inoue, S. Inoue, and T. Nishida, "Deep recurrent neural network for mobile human activity recognition with high throughput," *Artificial Life and Robotics*, vol. 23, no. 2, pp. 173–185, 2018.

[31] L. Cao, Y. Wang, B. Zhang, Q. Jin, and A. V. Vasilakos, "GCHAR: an efficient Group-based Context-aware human activity recognition on smartphone," *Journal of Parallel and Distributed Computing*, vol. 118, pp. 67–80, 2018.

[32] Y. Chen, K. Zhong, J. Zhang, Q. Sun, and X. Zhao, *LSTM Networks for Mobile Human Activity Recognition*, Atlantis Press, Paris, France, 2016.

[33] Y. Guan and T. Plötz, "Ensembles of deep LSTM learners for activity recognition using wearables," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 1, no. 2, pp. 1–28, 2017.

[34] Y. Zhao, R. Yang, G. Chevalier, X. Xu, and Z. Zhang, "Deep residual Bidir-LSTM for human activity recognition using wearable sensors," *Mathematical Problems in Engineering*, vol. 2018, Article ID 7316954, 13 pages, 2018.

[35] N. Y. Hammerla, S. Halloran, and T. Ploetz, "Deep, convolutional, and recurrent models for human activity recognition using wearables," 2016, https://arxiv.org/abs/1604.08880.

[36] M. Zeng, T. N. Le, B. Yu et al., "Convolutional neural networks for human activity recognition using mobile sensors," in *Proceedings of the 6th International Conference on Mobile Computing, Applications and Services*, pp. 197–205, IEEE, Austin, TX, USA, November 2014.

[37] F. Zhou, L. Jin, and J. Dong, "Review of convolutional neural network," *Chinese Journal of Computers*, vol. 40, pp. 1229–1251, 2017.

[38] C. A. Ronao and S.-B. Cho, "Human activity recognition with smartphone sensors using deep learning neural networks," *Expert Systems with Applications*, vol. 59, pp. 235–244, 2016.

[39] Y. Mohammad, K. Matsumoto, and K. Hoashi, "Primitive activity recognition from short sequences of sensory data," *Applied Intelligence*, vol. 48, no. 10, pp. 3748–3761, 2018.

[40] F. Ordóñez and D. Roggen, "Deep convolutional and LSTM recurrent neural networks for multimodal wearable activity recognition," *Sensors*, vol. 16, no. 1, p. 115, 2016.

[41] I. Amezzane, Y. Fakhri, M. E. Aroussi, and M. Bakhouya, *Comparative Study of Batch and Stream Learning for Online Smartphone-Based Human Activity Recognition*, M. Ben Ahmed, A. A. Boudhir, and A. Younes, Eds., pp. 557–571, Springer, Berlin, Germany, 2019.

[42] G. Hinton and V. Nair, "Rectified linear units improve restricted Boltzmann machines," in *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, Haifa, Israel, June 2010.

[43] S. Ioffe and C. Szegedy, "Batch normalization: accelerating deep network training by reducing internal covariate shift," in *Proceedings of the 32nd International Conference on Machine Learning (ICML 2015*, Lille, France, July 2015.

[44] M. Long and Y. Zeng, "Detecting iris liveness with batch normalized convolutional neural network," *Computers, Materials & Continua*, vol. 58, no. 2, pp. 493–504, 2019.

[45] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. L. Reyes-Ortiz, "A public domain dataset for human activity recognition using smartphones," in *Proceedings of the European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN)*, Bruges, Belgium, 2013.