Image and Vision Computing 109 (2021) 104141

Contents lists available at ScienceDirect

Image and Vision Computing

journal homepage: www.elsevier.com/locate/imavis

Multi-stream slowFast graph convolutional networks for skeleton-based action recognition

Ning Sun [a,*], Ling Leng [b], Jixin Liu [a], Guang Han [a]

[a] Engineering Research Center of Wideband Wireless Communication Technology, Ministry of Education,Nanjing University of Posts and Telecommunications, Nanjing 210003, China
[b] College of Communication and Information Engineering,Nanjing University of Posts and Telecommunications, Nanjing 210003,China

ARTICLE INFO

Article history:
Received 20 June 2020
Received in revised form 19 January 2021
Accepted 16 February 2021
Available online 23 February 2021

Keywords:
Action recognition
Graph convolutional network
Human skeleton
SlowFast network
Attention

ABSTRACT

Recently, many efforts have been made to model spatial–temporal features from human skeleton for action recognition by using graph convolutional networks (GCN). Skeleton sequence can precisely represent human pose with a small number of joints while there is still a lot of redundancies across the skeleton sequence in the term of temporal dependency. In order to improve the effectiveness of spatial–temporal feature extraction from skeleton sequence, a SlowFast graph convolution network (SF-GCN) is proposed by implementing the architecture of SlowFast network, which is consisted of the Fast and Slow pathway, in the GCN model. The Fast pathway is a temporal attention embedded lightweight GCN for extracting the feature of fast temporal changes from the skeleton sequence with a high frame rate and fast refreshing speed. The Slow pathway is a spatial attention embedded GCN for extracting the feature of slow temporal changes from the skeleton sequence with a low frame rate and slow refreshing speed. The features of two pathways are fused by using lateral connection and weighted by using channel attention. Based on the aforementioned design, SF-GCN can achieve superior ability of feature extraction while the computational cost significantly drops. In addition to the coordinate information of joints, five high order sequences including edge, the spatial difference and temporal difference of joints and edges are induced to enhance the representation of human action. Six SF-GCNs are implemented for extracting spatial–temporal feature from six kinds of sequences and fused for skeleton-based action recognition, which is called multi-stream SlowFast graph convolutional networks (MSSF-GCN). Extensive experiments are conducted to evaluate the proposed method on three skeleton-based action recognition databases including NTU RGB + D, NTU RGB + D 120, and Skeleton-Kinetics. The results show that the proposed method is effective for skeleton-based action recognition and can achieve the recognition accuracy with an obvious advantage in comparison with the state-of-the-art.

1. Introduction

Action recognition is an attractive and challenging task in computer vision. It plays an important role in many applications such as intelligent video surveillance [1], human-computer interaction [2] and motion analysis [3]. Based on the modality of input data, action recognition can be divided into two categories: image sequence-based methods and skeleton sequence-based methods. In the image sequence-based method, the RGB image and its optical flow [4] are usually used as the spatial and temporal representation of human action, respectively. But, the RGB image are often easily disturbed by factors such as dynamic illumination, changing camera view and occlusion, and the optical flow only represents the pixel-level difference between adjacent frames. Skeleton sequence is an assembly of human joints and bones in the time–space domain. Compared with the RGB image, skeleton has a benefit of the robustness with background changing and computing efficiently dues to its condensed representation. With the development of depth camera like Microsoft Kinect [5] and the human pose estimation algorithm like OpenPose [6], the acquisition of the skeleton sequence becomes more and more convenient. It also greatly promotes the research of skeleton-based action recognition.

Skeleton is a typical non-Euclidean data. There are two paradigms to recognize action from skeleton sequence by using deep neural network (DNN). One is to firstly transform the skeleton to form Euclidean data, and then use those deep learning models including convolutional neural network(CNN) [7] or recurrent neural network(RNN) [8] etc. proved significant efficiency in image analysis tasks to perform feature extraction and classification on them. The advantage of these methods is that it can use the off-the-shelf CNN or RNN model with excellent performance. The disadvantage is the topological relationship of skeleton sequence may be damaged in the transformation from non-Euclidean data to Euclidean data, which will do harm to the subsequent spatial–temporal feature learning.

[*] Corresponding author.
E-mail address: sunning@njupt.edu.cn (N. Sun).

https://doi.org/10.1016/j.imavis.2021.104141
0262-8856/© 2021 Elsevier B.V. All rights reserved.

The other paradigm is to build graph convolutional neural network (GCN), which generalizes traditional CNN to handle graph data, to learn the spatial–temporal feature from non-Euclidean skeleton sequence. There are usually two ways to build GCN. The first one is to implement GCN in the spectral domain [9,10]. Similar to the operation on Euclidean data, skeleton data is firstly transformed from the time–space domain to the frequency domain by using graph Fourier transformation, and multiplied in the frequency domain and then transformed back to the time–space domain, thereby completing the graph convolution operation. The second one is spatial domain GCN [11,12], it aggregates the information of adjacent vertices according to the connection relation of the graph vertex in time–space domain, thus completing the graph convolution operation in the time–space domain.

Compared with image sequence, skeleton sequence has been greatly simplified in the spatial dimension while it still contains a lot of redundancy in the temporal dimension, the data between adjacent frames are highly correlated. However, most of the current GCN-based action recognition methods equally treat each frame of the entire skeleton sequence, such as the ST-GCN [13] and its successor [14–16]. In this way, on the one hand, a large-scale GCN is needed to model the spatial–temporal feature of the entire skeleton sequence, which consumes a large amount of computing and memory resources. If a multi-GCNs architecture is established to learn multiple streams of skeleton sequences as in literature [14], the scale of the network will increase greatly, making the training more difficult. On the other hand, lots of redundant information tends to bias the learning of GCN, which may lead to negative effects to the result of action recognition.

To tackle the aforementioned issue, inspired by the SlowFast network for image sequence in literature [17], a SlowFast graph convolution network (SF-GCN) is proposed for skeleton-based action recognition, which generalizes the idea of SlowFast network to the GCN. The SF-GCN is composed of two pathways, one is the Fast pathway, which is focus on extracting fast changing feature like temporal motion of the skeleton from the high temporal resolution sequence with a high frame rate. The other is the Slow pathway, which is focus on extracting slow changing feature like spatial detail of skeleton from the low temporal resolution sequence with a low frame rate. To achieve such purpose, two different GCN models are used to implement the two pathways instead of using two identical CNN models as in literature [17]. The Fast pathway is designed very lightweight for less information redundancy because spatial feature can be provided via the Slow pathway. The intermediate features of two pathways are fused by using lateral connection. Furthermore, spatial and temporal attentions are embedded in the two pathways for improving the ability of feature extraction, and channel attention is used to weight the importance of the output of two pathways for final fusion. So, the SF-GCN can achieve better spatial–temporal feature extraction performance than the related GCN-based models with much lower computational cost.

In addition to the coordinate information of joints provided by the skeleton sequence, the higher order data derived from joint's position is learned by the proposed SF-GCN for action recognition. They are the edge (the direction and length of the bones), spatial difference and temporal difference of joints and edges. In the proposed method, six SF-GCNs are used to learn and fuse the spatial–temporal feature from the aforementioned six skeleton sequences, which is called multi-stream SlowFast graph convolutional networks (MSSF-GCN), as shown in Fig. 1.

Hence, the main contributions of this research can be summarized as follows:

(1) As far as our knowledge, the SF-GCN is the first model to expand the idea of SlowFast networks to the GCN, which is composed of two different GCN models to focus on learning slow or fast temporal changes, in respectively. The main advantage of SF-GCN is low computational cost and strong ability of spatial–temporal feature extraction.
(2) Two different GCN models are designed for different purposes in SF-GCN. The Fast pathway is formed by stacking lightweight temporal attention embedded ST-GCN blocks, the Slow pathway is formed by stacking fully-configured spatial attention embedded AGCN blocks. And, the channel attention is used for weighting importance to each channel of the output of two pathways.
(3) Combined by six SF-GCNs, a MSSF-GCN is proposed for skeleton-based action recognition from six streams of skeleton sequences.
(4) Extensive experiments are conducted on three benchmark skeleton-based action recognition databases including NTU RGB + D [18], NTU RGB + D 120 [19] and Skeleton-Kinetics [20], the experimental results show the effectiveness of the proposed method. Compared with state-of-the-art methods, the proposed method can achieve better recognition accuracies, with rates of 89.5%(X-Sub)96.2%(X-View) on NTU RGB + D, 84.4%(X-Sub)/86.1%(X-Set) on NTU RGB + D 120, and 37.0% on Skeleton-Kinetics.

2. Related work

2.1. Graph convolutional network

In the last decade, the DNN achieves a glorious success in the application on the Euclidean data including image and video. Those DNN models are effective for Euclidean data but difficult to apply directly to non-Euclidean data, which is commonly represented as graph. So, there are many recently studies on extending the DNN models for learning graph data.

The general process of graph deep learning can be expressed as the following: Based on the adjacent relationship of the graph data,

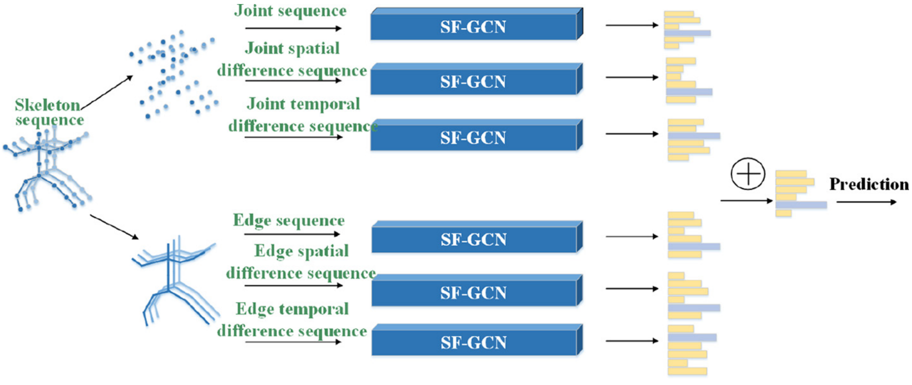

Skeleton sequence
Joint sequence
Joint spatial difference sequence
Joint temporal difference sequence
Edge sequence
Edge spatial difference sequence
Edge temporal difference sequence
SF-GCN
SF-GCN
SF-GCN
SF-GCN
SF-GCN
SF-GCN
⊕
Prediction

Fig. 1. The architecture of the proposed MSSF-GCN for skeleton-based action recognition.

message passing and status updating is performed on each node of the graph. Thus, each node continuously learns the feature of its neighbors. The hierarchical stacking architecture is used to nonlinearize and aggregate the features learned in each layer, and finally learn the complete semantics of the entire graph. For different tasks, there are different forms of graph deep neural network models, of which the most successful model in computer vision tasks is GCN.

The key to build GCN model is how to perform convolution operations on graph data. Its implementation usually falls into two categories: spectral GCN [9,10] and spatial GCN [11,12]. Spectral graph convolution is based on graph Fourier transform, which is an analogy of 1-D signal Fourier transform. The convolution operation of spectral GCN can be computed by taking the inverse Fourier transform of the multiplication between two Fourier transformed graph signals. The convolution operation of spatial GCN works on local neighborhood of nodes and learns the feature of a node from aggregating the information of its neighbors. The main limitation of spectral GCN is that the topological relationship of the nodes in the graph is fixed throughout the training process, because an Eigen decomposition of the Laplacian Matrix of the graph must be firstly performed at the beginning of the training. Spatial GCN does not require the adjacency matrix to be fixed in advance, and the topological relationship of the nodes can even be learned through training based on training samples and tasks. Because of this flexibility, spatial GCN plays a dominant role in skeleton-based action recognition.

2.2. Skeleton based action recognition

Skeleton sequence can effectively represent human action by using a small amount of geometric information such as joint position and adjacent relationship. There are usually two ways to deal with skeleton sequence, one is to firstly transform skeleton sequence into feature vector or matrix in Euclidean domain, and then use the popular deep learning models such as CNN and RNN or long short term memory (LSTM) to feature extraction and classification, which is called Euclidean feature-based method. The other is GCN-based method, which uses GCN to directly learn skeleton sequence in the non-Euclidean domain.

In the Euclidean feature-based methods, one way is assembling the 3D coordinate of the joint into a feature vector in a fixed order, and RNN/LSTM model is used to learn the spatial–temporal feature from the vector sequence. Shahroudy et al. [18] represent the whole body's skeleton with several parts, and each part is represented a concatenated vector of the all joint's coordinates in this part. A P-LSTM is proposed to learn the common temporal patterns of the parts independently, and the features of all parts are combined into a global level representation for action recognition. In the GCA-LSTM [21]proposed by Liu et al., the skeleton sequence is arranged as a matrix of the coordinate of all joints on all frames. And two layers LSTM are used to encode the input matrix and produce an attention for the global action. In the literature [22], the information of edge and surface is derived from joint, and a generic end-to-end RNN based network is designed to accommodate three inputs for action recognition. The other way of Euclidean feature-based method is to perform more complex transformations on skeleton sequence to produce feature image as a representation, and then feeds them into CNN for subsequent processing. In the SkeletonNet [23], skeleton sequence is transformed into two kinds of low-level features with translation, rotation, and scale invariant. And a CNN is used to learn high-level feature from these low-level feature and make the final prediction of action category. Kim et al. [24] propose the Temporal Convolutional Neural Networks (TCN) for action recognition on transformed feature matrix of the skeleton sequence. The feature matrix is composed of the coordinates of all the joints of all the frames in a fixed order. Liu et al. [25] present an enhanced skeleton visualization method for view invariant skeleton-based action recognition. In the literature [7], each channel of the 3D coordinate of a skeleton sequence is transformed into a clip, and a view robust representation is proposed based on the clips. The recent method [26] combines RNN and CNN to build a hybrid network to learn action semantic from the Euclidean feature transformed from the skeleton sequence.

In the GCN-based methods, GCN can directly learn the skeleton sequence represented as a spatial–temporal graph. The ST-GCN proposed by Yang et al. [13] is the famous early method that applies graph-based neural networks for skeleton-based action recognition. It proposes several principles in designing convolution kernels to meet the specific demands in skeleton modeling, and is widely used in follow-up work. In the methods of BPLHM [27], Conv-Relation [28] and DGNN [16], the effect of higher order information of joint's coordinate on skeleton-based action recognition is investigated. A two-stream architecture including one joint-based GCN and one edge-based GCN as complementary to each other is proposed in BPLHM. Similarly, DGNN also uses a two-stream model to extract features from the information of joint and edge in skeleton sequence. In Conv-Relation, frame difference of joints and edges are added as higher-order information, and a four-streams model is proposed to learn, predict and fuse from four data. Some efforts are made to learn dynamical graph structure in the training instead of fixing the adjacency matrix of joints before training. In 2s-AGCN [14], the topology of the graph of a skeleton sequence can be either uniformly or individually learned in an end-to-end manner. In GR-GCN [29], a graph regression is firstly used to learn an optimized graph structure over consecutive frames, and the learned spatial–temporal graph is fed to a spectral-based GCN for feature extraction and prediction. Some approaches attempt to embed attention to GCN for better performance. Si et al. [30] propose an attention enhanced graph convolutional LSTM network (AGC-LSTM) for skeleton-based action recognition, the attention module is employed to enhance information of key joints in each AGC-LSTM layer. In Cross-Attention [31], Fan et al. propose a cross-attention module that consists of a self-attention branch and a cross-attention branch to focus on extracting skeleton joints that are not only more informative but also highly related to its context information.

Concurrent with our work, there are two models which aim to improve the recognition accuracy while reducing the computational cost. Zhang et al. [32] propose SGN model by studying the importance of the semantic information and explore the joint-level and frame-level dependencies of the skeleton sequence. In SGN, feature data are performed spatial max pooling along the joints and temporal max pooling along the frames in frame-level module, which greatly reduces the parameters of this model. In the literature [33], inspired by the idea of shift CNN [34], Cheng et al. propose the shift-GCN model composed of novel shift graph operations and lightweight point-wise convolutions for tackling the problem of heavy computational cost and fix receptive field in traditional GCN.

Different from the aforementioned GCN-based method, the proposed SF-GCN applies the idea of SlowFast network in the GCN for skeleton-based action recognition. The Fast pathway adopts lightweight GCN model with significantly reduced feature channels to focus on extracting feature of fast changing from sequence data with a high frame rate. The Slow pathway is made to focus on extracting feature of slow changing from sequence data with a low frame rate. And, three kinds of attentions are embedded to SF-GCN for improving the performance of feature extraction and fusion. This design enables SF-GCN to obtain better recognition results than single GCN model with significantly reduced computation. Based on the coordinate of the joint, five higher order skeleton sequences including edge, the spatial difference and temporal difference of joints and edges are extended. The MSSF-GCN consisting of six SF-GCNs is proposed to perform skeleton-based action recognition on the six streams of input sequences.

3. Multi-stream SlowFast graph convolutional networks for skeleton-based action recognition

The proposed multi-stream SlowFast graph convolutional networks (MSSF-GCN) proposed is composed of six SF-GCNs. Six kinds of skeleton sequences including joint, edge, spatial difference of joints and edges, temporal difference of joints and edges are fed to six SFGCNs for feature extraction, fusion and prediction.

3.1. SlowFast graph convolution network

Referring to the idea of SlowFast network in CNN [17], this paper introduces the architecture of SlowFast network into GCN, which uses two interactive GCNs with different structures to enhance the learning of spatial–temporal feature from skeleton sequence. As shown in Fig. 2, the Fast pathway is a temporal attention embedded lightweight GCN focusing on learning the temporal feature from the skeleton sequence with a high frame rate and fast refreshing speed. The Slow pathway is a spatial attention embedded fully-configured GCN focusing on learning the spatial feature from the skeleton sequence with a low frame rate and slow refreshing speed. Several lateral connection is used to fuse the features from the intermediate layers of two pathways. After weighting importance to all channels of the output feature of Fast and Slow pathway by using the channel attention, the final prediction is made.

3.1.1. Fast and slow pathway

Suppose a skeleton data with N nodes and T frames, its spatial–temporal graph is $\mathbf{G} = (\mathbf{V},\mathbf{F})$, $\mathbf{V} = \{v_{ti} \mid t = 1,\ldots,T, i = 1,\ldots N\}$, $\mathbf{F} \in R^{C\times T\times V}$ is the feature vector with C channels, T frames and V nodes. The Fast pathway is formed by stacking several temporal attention embedded lightweight ST-GCN blocks, as shown in Fig. 3 a). The implementation of each layer is as follows:

$$\mathbf{f}_{out} = \sum_{k}^{K_v} \mathbf{W}_k(\mathbf{f}_{in}\mathbf{A}_k) \otimes \mathbf{S}_T^a \tag{1}$$

where $K_v = 3$ is the number of convolution kernel subsets, the spatial configuration partitioning is used in the Fast pathway; $\mathbf{W}_k$ is the parameter of the k-th subset of the convolution kernel; f_{in} is the input data of the current layer; $\mathbf{A}_k$ is the adjacency matrix corresponding to the k-th subset. ⊗ denotes the matrix multiplication. The implementation of attentions are all base on SEnet [35] in this paper. $\mathbf{S}_T^a \in R^{1\times T\times 1}$ is the importance score of the temporal attention, it can be represented as follows:

$$\mathbf{S}_T^a = \sigma(\mathbf{W}_2(\delta(\mathbf{W}_1(pool(\mathbf{f}))))) \tag{2}$$

where $pool()$ means the average pooling of all channels and joints of $\mathbf{f}$, $\mathbf{W}_1 \in \mathbf{R}^{T\times(T/r)}$ and $\mathbf{W}_2 \in \mathbf{R}^{(T/r)\times T}$ are the weights of two fully connected layers, respectively. r is reduction ratio, σ is the ReLu function, and δ is the Sigmoid function.

The Slow pathway is formed by stacking several spatial attention embedded AGCN blocks with full configuration, as shown in Fig. 3 b). The implementation of each layer is as follows:

$$\mathbf{f}_{out} = \sum_{k}^{K_v} \mathbf{W}_k\mathbf{f}_{in}(\mathbf{A}_k + \mathbf{B}_k + \mathbf{C}_k) \otimes \mathbf{S}_S^a \tag{3}$$

where $\mathbf{A}_k$ is the traditional adjacency matrix; $\mathbf{B}_k$ and $\mathbf{C}_k$ are all the learnable matrix with the same size of $\mathbf{A}_k$, $\mathbf{B}_k$ is the adaptive adjacency matrix, and $\mathbf{C}_k$ is the self-attention matrix. $\mathbf{S}_S^a \in \mathbf{R}^{1\times 1\times V}$ is the importance score of the spatial attention. It also computed according to Eq. (2), where pool()means the average pooling of all channels and frames of f, the weights of two fully connected layers are $\mathbf{W}_1 \in \mathbf{R}^{V\times(V/r)}$ and $\mathbf{W}_2 \in \mathbf{R}^{(V/r)\times V}$, respectively.

As shown in Fig. 3 c), the channel attention is embedded in the top of SF-GCN for weighting importance to all channels of the output feature of two pathways. $\mathbf{S}_c^a \in \mathbf{R}^{C\times 1\times 1}$ is the importance score of the channel attention. It is also computed according to Eq. (2), where $pool()$ means the average pooling of all joints and frames of concatenated feature f of the output of two pathways, the weights of two fully connected layers are $\mathbf{W}_1 \in \mathbf{R}^{C\times(C/r)}$and $\mathbf{W}_2 \in \mathbf{R}^{(C/r)\times C}$, respectively.

The using of ST-GCN and AGCN block to be as the Fast and Slow pathway, respectively, is based on the results of experiments, see Section 4.3 for more details.

3.1.2. Lateral connections

Similar to the model in literature [17], the two pathways in SF-GCN are also fused by using lateral connection. Before introducing the specific configuration of the lateral connection, the related parameters are firstly defined. In the Fast pathway, the temporal stride of sampling is m, which means that it samples one out of every m frames; In the Slow pathway, the temporal stride of sampling is αm, α is the multiple factor. This leads to the fact that the size of the Fast pathway's feature is α time larger in temporal dimension than the size of the Slow pathway's feature in a certain layer. For matching the feature size of two pathways, the number of channels in the Fast pathway is reduced to $1/\alpha$ of the number of channels in Slow pathway in per layer. There are three ways to make a lateral connection:

3.1.2.1. Time-to-channel. Suppose the feature size of a layer in Fast pathway is $\{1/\alpha \times C, \alpha \times T, V\}$, the corresponding feature the size of Slow pathway is $\{C,T,V\}$. In this way, the feature size of Fast pathway is transformed to $\{1/\alpha \times 1/\alpha \times C, T, V\}$=$\{C,T,V\}$. Then, the transformed features are added to the features in Slow pathway.

3.1.2.2. ST-GCN block. In this way, a ST-GCN block is used to downsample features of the Fast pathway to match the feature size of Slow pathway

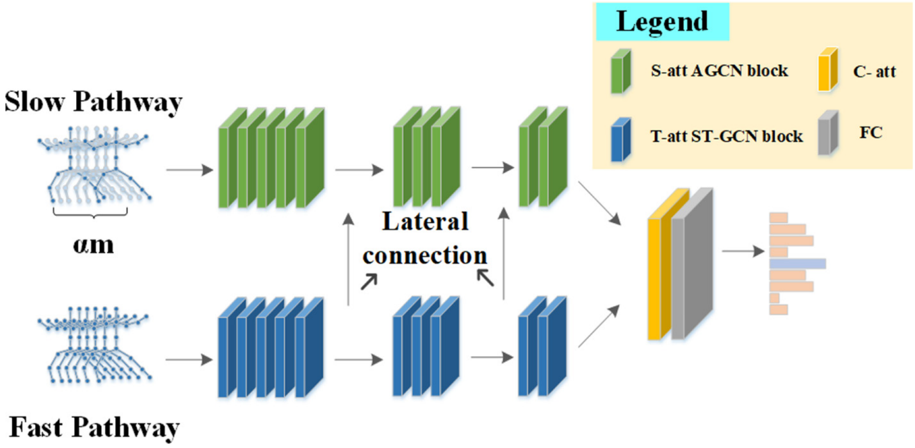
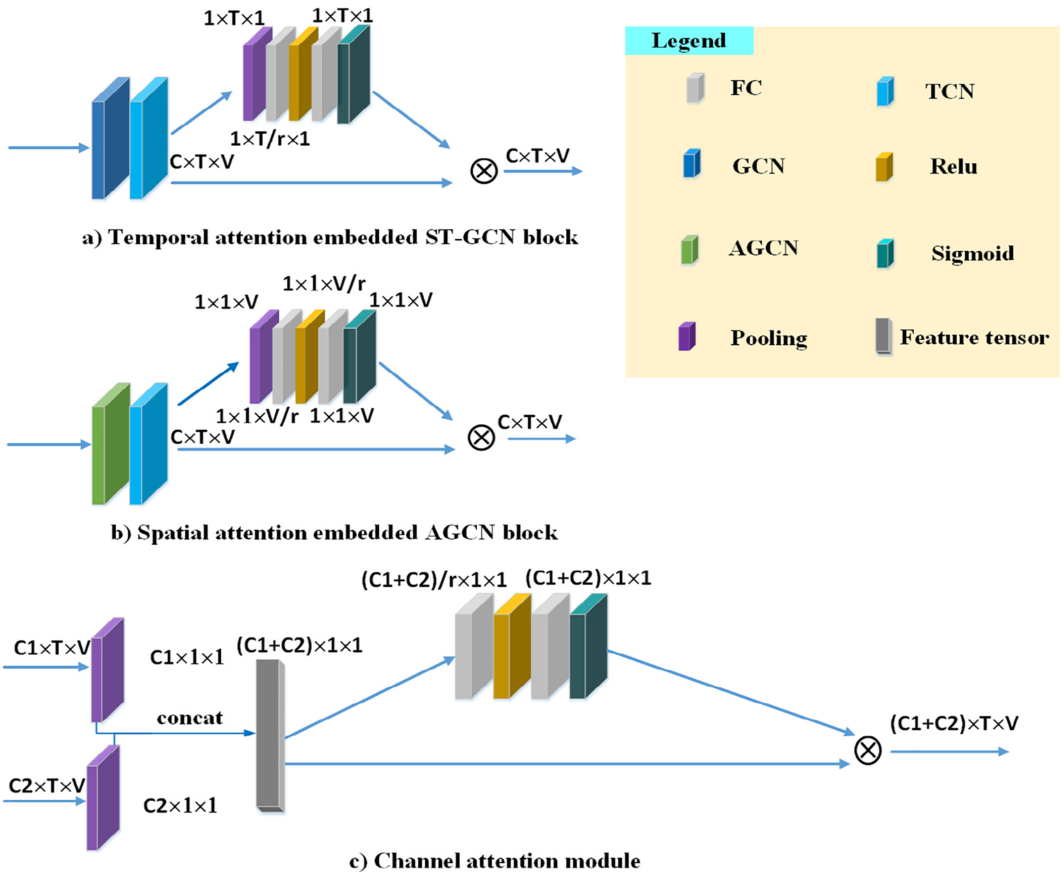


Fig. 2. The architecture of the SF-GCN.



a) Temporal attention embedded ST-GCN block

b) Spatial attention embedded AGCN block

c) Channel attention module

Fig. 3. The architecture of the temporal attention embedded ST-GCN block, spatial attention embedded AGCN block and channel attention module.

in the temporal dimension, that is $\{1/\alpha \times C, T, V\}$. And, it is concatenated with the feature of Slow pathway.

3.1.2.3. AGCN block. The configuration of this way is similar to the way of ST-GCN block, the size of concatenated feature is $\{(1+1/\alpha) \times C, T, V\}$. Based on the experimental results in Section 4.3 b), AGCN block is chosen as the way of lateral connection in SF-GCN.

3.1.3. A instantiation of SF-GCN

A specific implementation of the SF-GCN on the NTU RGB + D database is listed in Table 1. In this example, the size of the input skeleton sequence is$\{N, C, T, V, M\}$, N is batch size, $C = 3$ is size of input feature, $T = 300$ is number of frame, $V = 25$ is the number of joint, $M = 2$ is number of people in one frame, $m = 1$ and $\alpha = 5$.

3.2. Multi-stream SlowFast graph convolution network

In addition to the coordinate of joints, five kinds of the high-order data of joints are computed to feed to six SF-GCNs as the complementary information to each other. Three kinds of attention including spatial, temporal and channel attention are embedded into SF-GCN for better performance. These six SF-GCNs are used to perform feature extraction on the aforementioned six kinds of skeleton sequences. The softmax outputs of the six SF-GCNs are added up as the fused score of final prediction. The entire architecture of six SF-GCNs is called multi-stream SlowFast graph convolution network (MSSF-GCN), as shown in Fig. 1.

The first kind of input sequence is joint, which is denoted as $v = (x, y, z)$; The second input sequence is edge, which is a second-order data of joint. Suppose one certain joint is $v_i = (x_i, y_i, z_i)$, and its adjacent joint is $v_{i+1} = (x_{i+1}, y_{i+1}, z_{i+1})$, then the edge can be defined as $e_{vi,\ vi+1} = (x_{i+1} - x_i, y_{i+1} - y_i, z_{i+1} - z_i)$. And, the rest four input sequences are the spatial difference and temporal difference of joints and edges. The temporal difference is the difference between the same joints or edges in the adjacent frames, which can be denoted as $(v^t - v^{t-1})$ or $(e^t - e^{t-1})$. The spatial difference between the joints or the edges is the difference between one certain joint or edge and the center joint v_* or center edge e_* in the same frame, which can be denoted as $(v_i - v_*)$ or $(e_i - e_*)$. In the NTU RGB + D and NTU RGB + D 120 databases, v_* is the joint 2-middle of the spine, and e_* is the edge between joint 1-base of the spine and joint 2-middle of the spine; In the Skeleton-Kinetics database, v_* is the joint 1 located on the neck, and e_* is the edge between joint 1 and joint 0 located on the nose.

4. Experiments and discussion

In this section, we conduct extensive experiments to evaluate performance of the proposed method on three publicly-available skeleton-based action recognition benchmarks; namely, NTU RGB + D database, NTU RGB + D 120 database and Skeleton-Kinetics database.

4.1. Databases

4.1.1. NTU RGB + D database

NTU RGB + D database contains 56,578 skeleton sequences over 60 action classes captured from 40 distinct subjects and 3 different camera view angles. The benchmark evaluations include Cross-Subject (X-Sub) and Cross-View (X-View) setting. In the X-Sub setting, 40,320 samples from 20 subjects are used for training and the other 16,540 samples

Table 1
An instantiation of SF-GCN. The meaning of S:3,12,1 is that the kernel size is 3, the number of channel is 12, and the stride is 1 in spatial dimension; The meaning of T: 9,12,1 is that the kernel size is 9, the number of channel is 12, and the stride is 1 in the temporal dimension. AGCN: F to S means to use AGCN block for lateral connection, and fuse the feature of Fast pathway into Slow pathway. For simplicity, the N and V dimensions of the output features are omitted.

Layer	Fast pathway	Slow pathway	Output size (C × T)
Raw clip	–	–	3 × 300
Sample	Temporal stride:1	Temporal stride:5	Fast: 3 × 300 × 25 Slow: 3 × 60 × 25
0, 1, 2, 3	S: 3, 12, 1 T: 9, 12, 1 T-att	S: 3, 60, 1 T: 9, 60, 1 S-att	Fast: 12 × 300 × 25 Slow: 60 × 60 × 25
4	S: 3, 24, 1 T: 9, 24, 2 T-att	S: 3, 120, 1 T: 9, 120, 2 S-att	Fast: 24 × 150 × 25 Slow: 120 × 30 × 25
Lateral connection (AGCN: F to S)			
5, 6	S: 3, 24, 1 T: 9, 24, 1 T-att	S: 3, 120 + 24 × 2, 1 T: 9, 120 + 24 × 2, 1 S-att	Fast: 24 × 150 × 25 Slow: (120 + 24 × 2)× 30 × 25
7	S: 3, 48, 1 T: 9, 48, 2 T-att	S: 3, 240, 1 T: 9, 240, 2 S-att	Fast: 48 × 75 × 25 Slow: 240 × 15 × 25
Lateral connection (AGCN: F to S)			
8, 9	S: 3, 48, 1 T: 9, 48, 1 T-att	S: 3, 240 + 48 × 2, 1 T: 9, 240 + 48 × 2, 1 S-att	Fast: 48 × 75 × 25 Slow: (240 + 48 × 2)× 15 × 25
T and V average pool, concatenate, C-att, fc			#classes

are for testing. In X-View setting, the 37,920 samples captured from camera 2 and 3 are used for training, while the other 18,960 samples from camera 1 are for testing.

4.1.2. NTU RGB + D 120 database

NTU RGB + D 120 Database extends the NTU RGB + D database by adding 60 additional action classes to the existing ones and 66 more subjects, totaling 113,945 samples over 120 classes captured from 106 distinct subjects and 32 different camera setups. The authors recommend replacing X-View setting with Cross-Setup (X-Set) setting, where 54,468 samples collected from half of the camera setups are used for training and the rest 59,477 samples for testing. In X-Sub setting, 63,026 samples from a selected group of 53 subjects are used for training, and the rest 50,919 samples for testing. We follow this convention and report the top-1 accuracy on all NTU RGB + D databases.

4.1.3. Skeleton-kinetics database database

Skeleton-Kinetics database is adapted from the Kinetics dataset, which contains around 300, 000 video clips with 400 human action classes retrieved from YouTube, using the OpenPose pose estimation toolbox. It contains 240,436 training samples and 19,796 testing samples, where each skeleton graph contains 18 body joints with their 2D spatial coordinates and the prediction confidence score. For the multi-person cases, two people with the highest average joint confidence is selected. Top-1 and Top-5 recognition accuracies are reported as the recommendation.

4.2. Experimental settings

In the all experiments on three databases, the frame number of the input skeleton is $T=300$, temporal stride is $m=1$, multiple factor $\alpha=5$. So, the size of sequence fed to Slow pathway is $S_s=T/m=300$, and the size of sequence fed to Fast pathway is $S_f=T/\alpha m=60$. In the training, the optimizer is the stochastic gradient descent (SGD) [36], the size of mini-batch is 64, the iteration number is 60, and the initial learning rate is 1×10^{-2}. The learning rate is decayed by 10 every 20 epochs. All experiments are developed with PyTorch framework and run on an image processing workstation with an Intel Xeon 2.4 GHz 8 core CPU, 128 GB memory, and two NVIDIA Titan Xp GPUs.

4.3. Evaluation of the SF-GCN

The experiments in this section are all based on the NTU RGB + D database X-View setting, only the joint sequence is used as the input data.

4.3.1. Model configurations of the fast and slow pathway

The impact of four model configurations of Fast and Slow pathway on the performance of action recognition is evaluated in this subsection. They are 1) both pathways are implemented based on ST-GCN block; 2) both pathways are implemented based on AGCN block; 3) Fast and Slow pathway are implemented based on ST-GCN block and AGCN block, respectively; 4) Fast and Slow pathway are implemented based on AGCN block and ST-GCN block, respectively. The accuracies and computational cost (in GFLOPs) of the SF-GCN models with four configurations and two baseline models including ST-GCN and AGCN are listed in Table 2. Due to adopting the lightweight GCN to the Fast pathway and feeding sparse input sequence to the Slow pathway, the computational cost of SF-GCN consisted of two GCN model is much lower than the two baseline GCN model, which is only 16%–37% of that of the two baseline GCN models according to various configurations. The accuracies achieved by SF-GCNs with various configurations are all better than that of the baseline ST-GCN or comparable to that of the baseline AGCN, which show that SF-GCN can obtain excellent recognition accuracy while greatly reducing the computational cost. Furthermore, the accuracy of the third configuration is the best in all configurations, which proves that ST-GCN block is more conducive to extract temporal features from skeleton sequences, and AGCN block is more conducive to extract spatial features from skeleton sequences.

4.3.2. Configurations of lateral connection

The impact of four configurations of the lateral connections in SF-GCN on action recognition performance is evaluated in this subsection. They are 1) Without using lateral connection; 2) Time-to-channel; 3) ST-GCN block; 4) AGCN block. The results are listed in Table 3. It shows that the way of AGCN block can achieve the best recognition accuracy in comparison with three others ways, which is chosen as the way of lateral connection in the SF-GCN.

4.3.3. Visualization of the learned SF-GCN

The visualization of the skeleton graphs for the top layer of Fast and Slow pathway in SF-GCN is shown in this subsection. Several frames of a certain sequence sample in action class “tear up paper” are shown in Fig. 4. The color of each joint represents the importance of the joint in the learned skeleton graph according to the final action recognition

Table 2
Accuracy and computational cost of SF-GCN with various configurations.

Methods	Accuracy(%)	Computational cost (GFLOPs)
Baseline ST-GCN	88.3	16.34
Baseline AGCN	93.7	18.68
SF-GCN(F:ST-GCN, S:ST-GCN)	90.5	3.02
SF-GCN(F:AGCN, S:AGCN)	93.2	6.16
SF-GCN(F:AGCN, S:ST-GCN)	92.7	5.16
SF-GCN(F:ST-GCN, S:AGCN)	93.6	5.92

Table 3
Recognition accuracy of SF-GCN using various ways of lateral connection.

Configurations of lateral connection	Accuracy(%)
Without using lateral connection	92.9
Time-to-channel	93.2
ST-CGN block	93.4
AGCN block	93.6

result. The redder the color, the more important it is, otherwise the color tends to blue. It can be seen that Fast pathway pays more attention to the joints of hands with fast movement, but less attention to those joints with small movement. However, in the visualization of Slow pathway, there are significantly more red joints compared to Fast pathway, and their colors are constantly changing. This is due to the fact that Slow pathway pays more attention to the extraction of spatial information from different joints. From this, it can be intuitively seen that Fast and Slow pathway respectively focus on the extraction of temporal and spatial information in the skeleton sequence, which also proves the effectiveness of the architecture of SF-GCN.

4.4. Ablation study

The experiments in this section are all based on the NTU RGB + D database X-View setting.

4.4.1. Ablation study of various input sequences

The proposed MSSF-GCN learns spatial–temporal feature from six streams of sequences including joint, edge, the spatial and temporal difference of joints and edges. The results of ablation study on various input sequences are listed in Table 4. The recognition accuracy of SF-GCN based on the input sequence of the joint and edge alone has been 93.6% and 92.5%, respectively, the recognition results on the spatial difference of the joints has reached a higher 94.1%, and other recognition results based on the single input sequence have also exceeded 91.5%. It proves that all six input sequences can provide useful information for skeleton-based action recognition. The fused results of three streams of sequences based on joint and edge can reach 95.6% and 95.2%, respectively, and fusion result of the six streams of sequences is the best accuracy of 96.2%. It proves that the multi-stream sequences are complementary to each other.

Table 4
Recognition accuracy of the proposed method using various input sequences.

Methods	Accuracy(%)
Joint	93.6
Temporal difference of joint (tdj)	91.5
Spatial difference of joint (sdj)	94.1
Edge	92.5
Temporal difference of edge (tde)	91.6
Spatial difference of edge (sde)	91.5
Joint + tdj + sdj	95.6
Edge + tde + sde	95.2
Joint + tdj + sdj + edge + tde + sde	96.2

4.4.2. Ablation study of various attention modules

The recognition accuracies of the proposed method with removing various attention modules are listed in Table 5. It can be seen that the recognition accuracy will decrease after removing any attention module. This verifies that all three kinds of attentions can help improve the recognition performance of the proposed method. Among them, the reduction in recognition accuracy caused by the removal of channel attention is the most obvious, which indicates that channel attention has the greatest impact on the overall performance of the performance.

4.5. Compared with the state-of-the-art

In this section, the proposed method is compared with the state-of-the-art skeleton-based action recognition methods on three benchmarks including NTU RGB + D, NTU RGB + D 120, and Skeleton-Kinetics databases.

4.5.1. Results on NTU RGB + D database

The proposed method is compared with 24 state-of-the-art methods on NTU RGB + D database. These methods are belonging to two categories including Euclidean feature-based method and GCN-based method. The recognition accuracies of all methods are listed in Table 6. Firstly, the proposed method can achieve the recognition accuracy of 96.2% in the experiment based on X-View setting, and the recognition accuracy in the experiment based on X-Sub setting is 89.5%, which is only slightly lower than that of Shift-GCN. When only using three sequences of joints or edges as the input data, the proposed method can also obtain better recognition accuracy than that of almost all the related methods. It shows that extending the architecture of SlowFast network to the GCN model is beneficial to the extraction of spatial–temporal features for

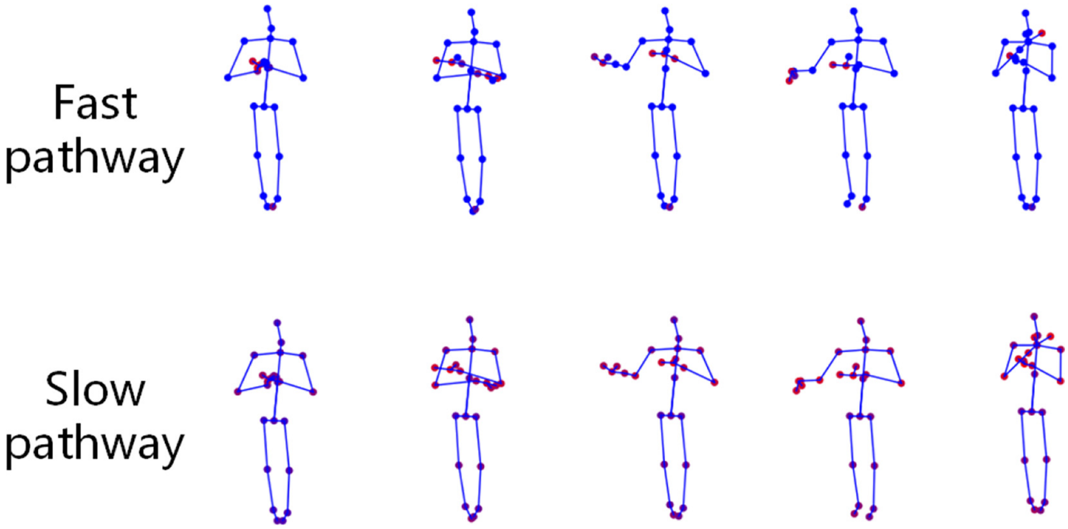


Fig. 4. Visualization of the learned SF-GCN. The top row is the results of Fast pathway, the bottom row is the results of Slow pathway.

Table 5
The recognition accuracies of the proposed method with removing various attention modules. wo/C, wo/T, and wo/S indicates removal of channel, temporal, and spatial attention, respectively.

Methods	Accuracy(%)
MSSF-GCN wo/C	95.5
MSSF-GCN wo/T	95.7
MSSF-GCN wo/S	95.8
MSSF-GCN	96.2

Table 6
Comparison of the recognition accuracy between the proposed method and the state-of-the-arts on NTU RGB + D database.

Methods	X-View (%)	X-Sub (%)
P- LSTM [18]	70.3	62.9
STA-LSTM [8]	81.2	73.4
Res-TCN [24]	83.1	74.3
SkeletonNet [23]	81.2	75.9
s GCA-LSTM [21]	85.1	77.1
Enhanced skeletion [25]	87.2	80.0
RotClips + MTCNN [7]	87.4	81.1
Beyond Joints [22]	87.6	79.5
SNet + HNet [26]	89.8	84.2
TS + MSSFN [37]	92.3	85.3
ST-GCN [13]	88.3	81.5
STG-IN [38]	88.7	85.8
Si-GCN [39]	89.1	84.2
Cross-Attention [31]	89.3	84.2
BPLHM [27]	91.1	85.4
STGR-GCN [40]	92.3	86.9
GVFE + DH-TCN [41]	92.8	85.3
AS-GCN [15]	94.2	86.8
GR-GCN [29]	94.3	87.5
Conv-Relation [28]	94.5	86.2
SGN [32]	94.5	89.0
AGC-LSTM [30]	95.0	89.2
s-AGCN [14]	95.1	88.5
DGNN [16]	96.1	89.9
Shift-GCN [33]	**96.5**	**90.7**
MSSF-GCN(joint + tdj + sdj)	95.6	87.6
MSSF-GCN(edge + tde + sde)	95.2	88.3
MSSF-GCN(All 6 streams)	96.2	89.5

action recognition from skeleton sequences. Secondly, the GCN-based methods are generally superior to the Euclidean feature-based method methods in terms of recognition accuracy. It indicates that the direct way of learning the topological semantic from skeleton sequence by GCN is more effective than the indirect way of transforming the graph to Euclidean feature and then performing feature extraction. Thirdly, the result of our MSSF-GCN and Shift-GCN show that those skills which have been successfully applied in CNN can be modified and transplanted to GCN, which can also effectively improve the performance of GCN. Fourthly, it can be seen that there are three research hotspots in recent skeleton-based action recognition: 1) Dynamically learn the adjacent relationship of skeleton graph, such as 2S-AGCN [14] and GR-GCN [29], etc.; 2) Model and fuse the higher order information of the skeleton sequences, such as BPLHM [27], Conv-relation [28] and DGNN [16], etc.; 3) Use a variety of attention mechanisms to enhance the spatial–temporal feature extraction of GCN-based model, such as AGC-LSTM [30] and cross-attention [31]. Therefore, the recognition accuracy obtained by these methods is also among the best. Based on these previous work and combined with the architecture of SlowFast network, the proposed method further improves the performance of skeleton-based action recognition.

4.5.2. Results on NTU RGB + D 120 database

Table 7 lists the experimental results on the NTU RGB + D 120 database. The compared related work are the same 2 categories as in the last experiment but only 6 methods since this database is just published in 2019. Similar with the results on NTU RGB + D database, Shift-GCN achieve the best results, the proposed method can both achieve the top level results on X-Sub setting and X-Set setting, no matter whether all six streams of sequences or part of them are input. This proves that the proposed method still has good performance on a larger-scale skeleton action recognition database.In the future work, we intend to study the combination of SlowFast architecture and shift graph operation to further improve the recognition accuracy and reduce the computational cost in the skeleton-based action recognition.

Table 7
Comparison of the recognition accuracy between the proposed method and the state-of-the-arts on NTU RGB + D 120 database.

Methods	X-Set(%)	X-Sub(%)
RotClips + MTCNN [7]*	61.8	62.2
2s GCA-LSTM [21]*	63.3	61.2
GVFE + DH-TCN [41]	79.8	78.3
SGN [32]	81.5	79.2
2s-AGCN [14]	84.9	82.9
Shift-GCN [16]	**87.6**	**85.9**
MSSF-GCN(joint + tdj + sdj)	83.5	83.3
MSSF-GCN(edge + tde + sde)	85.1	82.5
MSSF-GCN(All 6 streams)	86.1	84.4

* Indicates that it is the result re-implemented in literature [19], which is not reported in the original literature.

Table 8
Comparison of the recognition accuracy between the proposed method and the state-of-the-arts on Skeleton-Kinetics database.

Methods	Top-1(%)	Top-5(%)
ST-GCN [13]	30.7	52.8
Conv-Relation [28]	33.1	55.8
BPLHM [27]	33.4	56.2
STG-IN [38]	33.6	59.8
STGR-GCN [40]	33.6	56.1
AS-GCN [15]	34.8	56.5
2s-AGCN [14]	36.1	58.7
DGNN [16]	36.9	59.6
MSSF-GCN(joint + tdj + sdj)	34.7	57.3
MSSF-GCN(edge + tde + sde)	33.4	55.8
MSSF-GCN(All 6 streams)	**37.0**	**59.8**

4.5.3. Results on skeleton-kinetics database

Table 8 lists the experimental results on the Skeleton-Kinetics database. A total of eight GCN-based methods are compared with the proposed methods. Similar to the results on two NTU RGB + D databases, the proposed method also achieved the best recognition accuracy on the Top1 and Top5 results by using the novel GCN-based architecture combined with SlowFast networks, multiple streams of input and multiple attention embedding.

5. Conclusion

In order to extract spatial–temporal feature of human action from a redundant skeleton sequence in a more efficient manner, a SlowFast graph convolution network (SF-GCN) is proposed in this paper. SF-GCN is composed of a Fast pathway and a Slow pathway. Fast pathway pays attention to temporal feature extraction from skeleton sequence with a high frame rate and fast refreshing speed, while Slow pathway pays attention to spatial feature extraction from skeleton sequence with a low frame rate and slow refreshing speed, and the feature extracted from two pathways are fused by several lateral connections. Temporal, spatial, and channel attentions are embedded in SF-GCN for improving the performance. Benefit from this design, SF-GCN can obtain excellent recognition accuracy while greatly reducing the computational cost. A multi-stream SlowFasst graph convolution network (MSSF-GCN) is proposed for feature extraction and fusion from six kinds of input sequences by using six SF-GCNs. Extensive experiments are conducted to evaluate the performance of the proposed method on three benchmarks including NTU RGB + D, NTU RGB + D 120, and Skeleton-Kinetics database. The results show that the proposed method can achieve better results than the related state-of-the-art in skeleton-base action recognition.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

This work was supported by the National Natural Science Foundation of China (Grant Nos. 61471206, 61871445) and the Natural Science Foundation of Jiangsu Province (Grant No. BK20180088), and the Scientific Research Foundation of Nanjing University of Posts and Telecommunications (Grant No. NY218066).

References

[1] A.B. Mabrouk, E. Zagrouba, Abnormal behavior recognition for intelligent video surveillance systems[J], Expert Syst. Appl. (2018) 480–491.
[2] G. Yao, T. Lei, J. Zhong, et al., A review of convolutional-neural-network-based action recognition[J], Pattern Recogn. Lett. (2019) 14–22.
[3] F. Patrona, A. Chatzitofis, D. Zarpalas, et al., Motion analysis: action detection, recognition and evaluation based on motion capture data[J], Pattern Recogn. (2018) 612–622.
[4] K. Simonyan, A. Zisserman, Two-stream convolutional networks for action recognition in videos[C], Neural Information Processing Systems 2014, pp. 568–576.
[5] Z. Zhang, Microsoft kinect sensor and its effect[J], IEEE MultiMedia 19 (2) (2012) 4–10.
[6] Z. Cao, T. Simon, S. Wei, et al., Realtime multi-person 2D pose estimation using part affinity fields[C], Computer Vision and Pattern Recognition 2017, pp. 1302–1310.
[7] Q. Ke, M. Bennamoun, S. An, et al., Learning clip representations for skeleton-based 3D action recognition[J], IEEE Trans. Image Process. 27 (6) (2018) 2842–2855.
[8] S. Song, C. Lan, J. Xing, et al., An end-to-end spatio-temporal attention model for human action recognition from skeleton data[C], National Conference on Artificial Intelligence 2017, pp. 4263–4270.
[9] M. Henaff, J. Bruna, Y. Lecun, et al., Deep convolutional networks on graph-structured data.[J], arXiv preprint arXiv:1506.05163 (2015).
[10] J. Bruna, W. Zaremba, A. Szlam, et al., Spectral networks and locally connected networks on graphs[J], arXiv preprint arXiv:1312.6203 (2013).
[11] T.N. Kipf, M. Welling, Semi-Supervised Classification with Graph Convolutional Networks, 2016.
[12] M. Niepert, M.H. Ahmed, K. Kutzkov, et al., Learning convolutional neural networks for graphs[C], International Conference on Machine Learning 2016, pp. 2014–2023.
[13] S. Yan, Y. Xiong, D. Lin, et al., Spatial temporal graph convolutional networks for skeleton-based action recognition[C], National Conference on Artificial Intelligence 2018, pp. 7444–7452.
[14] L. Shi, Y. Zhang, J. Cheng, et al., Two-stream adaptive graph convolutional networks for skeleton-based action recognition[C], Computer Vision and Pattern Recognition 2019, pp. 12026–12035.
[15] M. Li, S. Chen, X. Chen, et al., Actional-structural graph convolutional networks for skeleton-based action recognition[C], Computer Vision and Pattern Recognition 2019, pp. 3595–3603.
[16] L. Shi, Y. Zhang, J. Cheng, et al., Skeleton-based action recognition with directed graph neural networks[C], Computer Vision and Pattern Recognition 2019, pp. 7912–7921.
[17] C. Feichtenhofer, H. Fan, J. Malik, et al., SlowFast networks for video recognition[J], arXiv: Computer Vision (2019) 6202–6211.
[18] A. Shahroudy, J. Liu, T. Ng, et al., NTU RGB + D: A Large Scale Dataset for 3D Human Activity Analysis[C], Computer Vision and Pattern Recognition 2016, pp. 1010–1019.
[19] J. Liu, A. Shahroudy, M. Perez, et al., NTU RGB + D 120: a large-scale benchmark for 3D human activity understanding.[J], IEEE Trans. Pattern Anal. Mach. Intell. 42 (10) (2019) 2684–2701.
[20] A. Zisserman, J. Carreira, K. Simonyan, et al., The kinetics human action video dataset [J], arXiv preprint arXiv:1705.06950 (2017).
[21] J. Liu, G. Wang, L. Duan, et al., Skeleton-based human action recognition with global context-aware attention LSTM networks[J], IEEE Trans. Image Process. 27 (4) (2018) 1586–1599.
[22] H. Wang, L. Wang, Beyond joints: learning representations from primitive geometries for skeleton-based action recognition and detection[J], IEEE Trans. Image Process. 27 (9) (2018) 4382–4394.
[23] Q. Ke, S. An, M. Bennamoun, et al., SkeletonNet: mining deep part features for 3-D action recognition[J], IEEE Signal Process. Lett. 24 (6) (2017) 731–735.
[24] T.S. Kim, A. Reiter, Interpretable 3D human action analysis with temporal convolutional networks[C], Computer Vision and Pattern Recognition 2017, pp. 1623–1631.
[25] M. Liu, H. Liu, C. Chen, et al., Enhanced skeleton visualization for view invariant human action recognition[J], Pattern Recogn. 68 (68) (2017) 346–362.
[26] M. Naveenkumar, S. Domnic, Deep ensemble network using distance maps and body part features for skeleton based action recognition[J], Pattern Recogn 100 (2020), 107125.
[27] X. Zhang, C. Xu, X. Tian, et al., Graph edge convolutional neural networks for skeleton based action recognition.[J], arXiv: Computer Vision Pattern Recognit. 31 (8) (2019) 3047–3060.
[28] J. Zhu, W. Zou, Z. Zhu, et al., Convolutional relation network for skeleton-based action recognition[J], Neurocomputing (2019) 109–117.
[29] X. Gao, W. Hu, J. Tang, et al., Optimized skeleton-based action recognition via sparsified graph regression[C], ACM Multimedia 2019, pp. 601–610.
[30] C. Si, W. Chen, W. Wang, et al., An attention enhanced graph convolutional lstm network for skeleton-based action recognition[C], Computer Vision and Pattern Recognition 2019, pp. 1227–1236.
[31] Y. Fan, S. Weng, Y. Zhang, et al., Context-aware cross-attention for skeleton-based human action recognition[J], IEEE Access (2020) 15280–15290.
[32] P. Zhang, C. Lan, W. Zeng, et al., Semantics-guided neural networks for efficient skeleton-based human action recognition.[J], arXiv: Computer Vision and Pattern Recognition2019.
[33] K. Cheng, Y. Zhang, X. He, et al., Skeleton-based action recognition with shift graph convolutional network[C], Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition 2020, pp. 183–192.
[34] B. Wu, A. Wan, X. Yue, et al., Shift: a zero flop, zero parameter alternative to spatial convolutions[C], Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2018, pp. 9127–9135.
[35] J. Hu, L. Shen, G. Sun, et al., Squeeze-and-excitation networks[C], Computer Vision and Pattern Recognition 2018, pp. 7132–7141.
[36] L. Bottou, F.E. Curtis, J. Nocedal, et al., Optimization methods for large-scale machine learning[J], SIAM Rev. 60 (2) (2018) 223–311.
[37] F. Meng, H. Liu, Y. Liang, et al., Sample fusion network: an end-to-end data augmentation network for skeleton-based human action recognition[J], IEEE Trans. Image Process. 28 (11) (2019) 5281–5295.
[38] W. Ding, X. Li, G. Li, et al., Global relational reasoning with spatial temporal graph interaction networks for skeleton-based action recognition[J], Signal Process. Image Commun. 83 (2020) 115776.
[39] R. Liu, C. Xu, T. Zhang, et al., Si-GCN: structure-induced graph convolution network for skeleton-based action recognition[C], International Joint Conference on Neural Network 2019, pp. 1–8.
[40] B. Li, X. Li, Z. Zhang, et al., Spatio-temporal graph routing for skeleton-based action recognition[C], National Conference on Artificial Intelligence, 33(01), 2019, pp. 8561–8568.
[41] K. Papadopoulos, E. Ghorbel, D. Aouada, et al., Vertex feature encoding and hierarchical temporal modeling in a spatial-temporal graph convolutional network for action recognition.[J], arXiv preprint arXiv:1912.09745 (2019).