Applied Intelligence (2022) 52:5608–5616
https://doi.org/10.1007/s10489-021-02723-6

Focus on temporal graph convolutional networks with unified attention for skeleton-based action recognition

Bing-Kun Gao[1] · Le Dong[1] · Hong-Bo Bi[1] · Yun-Ze Bi[1]

Accepted: 26 July 2021 / Published online: 17 August 2021

Abstract

Graph convolutional networks (GCN) have received more and more attention in skeleton-based action recognition. Many existing GCN models pay more attention to spatial information and ignore temporal information, but the completion of actions must be accompanied by changes in temporal information. Besides, the channel, spatial, and temporal dimensions often contain redundant information. In this paper, we design a temporal graph convolutional network (FTGCN) module which can concentrate more temporal information and properly balance them for each action. In order to better integrate channel, spatial and temporal information, we propose a unified attention model of the channel, spatial and temporal (CSTA). A basic block containing these two novelties is called FTC-GCN. Extensive experiments on two large-scale datasets, compared with 17 methods on NTU-RGB+D and 8 methods on Kinetics-Skeleton, show that for skeleton-based human action recognition, our method achieves the best performance.

Keywords Graph convolutional networks · Skeleton-based action recognition · Temporal information · Unified attention model

1 Introduction

As a large research hotspot in the field of computer vision, action recognition has important research significance and wide application prospects in many fields, such as intelligent monitoring [1, 2], human–computer interaction [3, 4], virtual reality [5]. The method based on traditional handcraft features [6, 7] is hard to deal with human action recognition in a complex scene. With the great success of deep learning in image classification [8], the application of deep learning in human action recognition has gradually become a development trend, but there are still some difficulties and challenges. There are two kinds of approaches to solving the problem of action recognition based on deep learning: 1) video recognition method for extracting and classifying spatial-temporal features; 2) pose estimation method for extracting skeleton information for retraining. Since neural networks can learn features from data, and this form of learning mode is consistent with the process of human awareness of the world, semantic features learned from neural networks can also be used for action recognition.

Among the various deep learning models, the most common is the convolutional neural networks (CNN) [8] used for image recognition tasks. Recurrent neural networks (RNN) [9] have been widely used in natural language processing (NLP) [10, 11] due to its superiority in time series modeling. Wang et al. proposed a relatively simple method [12], by coding the joint trajectory (distance) and its dynamic information into a texture pattern [13], called joint trajectory maps (JTM). Donahue et al. put forward a network [14] which combines CNN with LSTM, in which the pre-processed depth image data is first sent to the originally designed CNN to get the spatial features, and then the optical flow information in the video data is sent to the LSTM to get the temporal features. Finally, the spatial-order feature and temporal-order feature are fused and the mapping category of $Softmax$ is adopted. Based on the research of human body 3D skeleton motion representation [15], more and more attention has been paid to it. Shao et al. proposed a hierarchical model [16] of body part motion recognition, which decomposes a skeleton into multiple moving rigid bodies according to the motion characteristics of the human body, the rotation speed invariant descriptor

✉ Hong-Bo Bi
bhbdq@126.com

Bing-Kun Gao
gaobk@163.com

[1] NorthEast Petroleum University, Daqing, China

RRV (Rotation and Relative Velocity) is proposed to represent the rotation and velocity invariant of each rigid body in the skeleton, and the motion representation is obtained.

In the case of skeleton-based action recognition, the graph convolutional networks (GCN) [17–20] approach were proposed and gained attention due to its high-performance achievement. GCN is often used for learning tasks, such as graph classification [21–23], graph regression [24], and node classification [25–27]. According to the topological structure of the human body itself, we can construct a graph structure data with a human shape, and treat it as a graph classification task, then we can use graph convolution to process it. Yan et al. proposed the spatial-temporal graph convolutional network (ST-GCN) [28] that is the first time to apply the convolution to 3D human action recognition. The model not only constructs the spatial graph but also constructs the temporal graph in the time dimension according to the human topological structure, then the spatial-temporal features of human body movements are extracted by graph convolution. To reduce the redundant frames in skeleton video frames and extract the most discriminating video frames, the deep progressive reinforcement learning graph convolutional network (DPRL-GCN) model proposed by Tang et al. [29]. Then the keyframe is sent to GCN for recognition and classification. To excavate the node association in the local block of human body structure, Kalpit et al. proposed a part-based graph convolutional network (PB-GCN) [30] to divide the human body skeleton graph into several different subgraphs according to certain principles, and then perform graph convolution operation on each subgraph separately. Finally, the results are fused by a fusion function. To make full use of the potential connection of skeleton node graph topology. Ye et al. designed a joints relation inference network (JRIN) [31] to automatically explore the relationship between skeleton nodes and nodes, the relationship matrix is applied to the adjacency matrix of the original skeleton data to supplement the potential relationship between nodes of the original skeleton topology to better understand the human action. Subsequently, many GCN methods representing a more appropriate spatial graph have been proposed and the performance have been improved dramatically.

The contributions of this paper are summarized as follows:

- We propose a temporal graph convolutional module (FTGCN) which can focus more temporal information and properly balance them for each action.
- To better integrate channel, spatial, and temporal information, we propose a unified attention model of the channel, spatial, and temporal (CSTA).
- Compared with 17 methods on NTU-RGB+D and 8 methods on Kinetics-Skeleton, our method achieves the best performance.

2 Related work

2.1 Skeleton-based action recognition

There are 2 ideas in the field for skeleton-based human action recognition. The early handcraft-based ideas and the current standard deep-learning-based methods. The accuracy of handcraft-based ideas is unacceptable and therefore the deep learning methodology has become the thought methodology during this field for its smart strength and superior performance. There are basically 3 types of network RNNs, CNNs, and GCNs in the recognition of human actions based on deep learning. (1) The RNN-based idea [32] symbolizes the joint coordinates of the skeleton sequence as a vector sequence and feed into the networks. (2) The CNN-based idea [33] converts the skeleton sequence into a corresponding 2D pseudo-image is input into the network, which is similar to the method of image classification. (3) The GCN-based idea takes the joint points of the human body as vertices and the natural connections of the human body as edges to represent the skeleton sequence as a graph. At the same time, it takes into account time information and is widely used due to its superior performance. The first idea to use GCN in this area was ST-GCN [28], which made great progress at the time. Spatial connects the joint points according to the human body structure to form a spatial graph, and connect the same joints in adjacent frames in time, spatially, and temporally the connections form a spatial-temporal graph and send it to the network. However, the spatial graph of ST-GCN [28] is a fixed graph, as it will not demonstrate the relation between the two hands while clapping hands. Therefore, a two-stream adaptive graph convolutional network (2s-AGCN) [34] is proposed.

2.2 Graph convolutional networks

The essential purpose of GCN is to extract the spatial features of the topological graph. There are two main types of graph convolutional neural networks. One type is based on the spatial domain or vertex domain, the other is based on the frequency domain or spectral domain [35]. The method based on spatial convolution directly defines the convolution operation on the connection relationship of each node, which is more similar to the convolution in the traditional convolutional neural network. Different from the spatial perspective method, the spectral perspective method uses the eigenvalues and eigenvectors of the

Laplacian matrix of the graph to study the properties of the graph.

3 Approach

In this section, we will introduce the basic background knowledge of this work. Then describe our proposed focus on the temporal graph convolutional network (FTGCN) module and a unified attention model of the channel, spatial and temporal (CSTA) in detail.

3.1 Graph convolution

We use $G = (V, E)$ to describe the human structure graph in which V is the number of human joints and E is the number of edges connected by these joints, that is, the number of human bones. The adjacent matrix of the human structure graph is defined as $A\{\in\}\{0,1\}^{v\times v}$, where $A_{i,j} = 0$ if the i_{th} and the j_{th} joints are unconnected and 1 otherwise. Let $D \in R^{v\times v}$ be the diagonal degree matrix, where $D_{i,i} = \sum_j A_{i,j}$ is the degree matrix of vertices, and the elements on the diagonal are degrees of each vertex in turn. We use x represent the multidimensional coordinates of joints in human body structure. The adjacency matrix A can be used to aggregate the information of the adjacent nodes. The graph convolution operation of each layer can be expressed as follows:

$$x^{(l+1)} = \sigma\left(\hat{A}x^{l}W^{l}\right) \tag{1}$$

where $\hat{A} = D^{-\frac{1}{2}}(A + I_v)D^{-\frac{1}{2}}$ is the matrix after the adjacency matrix A plus the self-connection matrix I_v and then normalized. x^l is the output tensor of layer l, and $x^0 = x$. W^l is the weight matrix that changes with training. $\sigma(\cdot)$ is used to increase the nonlinearity of the neural network and called the activation function. Following the ST-GCN [28], we will implement a three-partition strategy on the human skeleton graph, that is, the neighbor set is divided into three subsets. The first subset is the root node itself. The second subset is the adjacent node closer to the center of gravity of the human skeleton than the root node. The third subset is adjacent nodes that are farther from the center of gravity of the human skeleton than the root node. In this way, A is accordingly classified to be root node set A_{root}, centripetal group $A_{centripetal}$, and centrifugal group $A_{centrifugal}$, which similar to the movement of body parts. Then there is $\sum_{k=1}^{3} A_k = A$ where $k = \{root, centripetal, centrifugal\}$

ST-GCN [28] is composed of 10 basic blocks. In order to alternately extract spatial and temporal information, each basic block is composed of a spatial GCN and a temporal GCN. In the spatial layer, the convolution operation on the human skeleton structure graph is:

$$x_{out} = \sum_{k=1}^{3} \hat{A}_k x_{in} W_k \tag{2}$$

Where $k = 3$ represents the three partitions mentioned above, each partition performs the same convolution operation, $W_k \in R^{c_{in}\times c_{out}}$ is a weight matrix that can be changed with training, $x_{in} \in R^{v\times c_{in}}$ is the input feature of the spatial layer, and $x_{out} \in R^{v\times c_{out}}$ is the output feature of the spatial layer. c_{in} represents the channel dimension of the input feature, c_{out} represents the channel dimension of the output feature. $\hat{A}_k = {D_k}^{-\frac{1}{2}}(A_k + I_{vk}){D_k}^{-\frac{1}{2}} \in R^{v\times v}$ is the normalized adjacent matrix of each partition. The adjacency matrix A in ST-GCN [28] only considers the natural connection of the human skeleton graph, but the completion of some actions sometimes requires the interaction of non-adjacent joints. In order to solve this problem, the two hand joints that are not adjacent in the clapping action can be connected, AGCN [34] is proposed, constructing the equation as follows:

$$x_{out} = \sum_{k=1}^{3} (\hat{A}_k + B_k + C_k) x_{in} W_k \tag{3}$$

Where $B_k \in R^{v\times v}$ is initially set as a $V \times V$ matrix. The parameters in the matrix are constantly changing with training. It serves as a supplement to the adjacency matrix A. Different joints are connected to different actions during training. $C_k \in R^{v\times v}$ is a data-dependent $V \times V$ matrix, and the parameters in the matrix are normalized to a value of 0-1. If the value is not zero, it means that the two joints are connected to each other, otherwise they are not connected. The larger the value, the stronger the connection strength of the two joints and the higher the correlation with the action.

For the temporal layer, only the same joints of the front and rear frames are connected, so common convolution operations are performed in the time dimension, specifically a $K_t \times 1$ convolution kernel, where K_t is the convolution kernel size in the temporal dimension, set to 9.

3.2 The architecture of the networks

In the early days, the input of this task was only joint information, but the human skeleton graph contains joints and bones at the same time. This makes the two-stream network structure suitable for joints and bones as input. The completion of the action is accompanied by the movement of joints and bones, and the two kinds of information are used as input. Improved recognition performance. In our work, like 2s-AGCN [34]. Joint information is first-order information, bone information is second-order information, and two kinds of information are used as inputs to promote

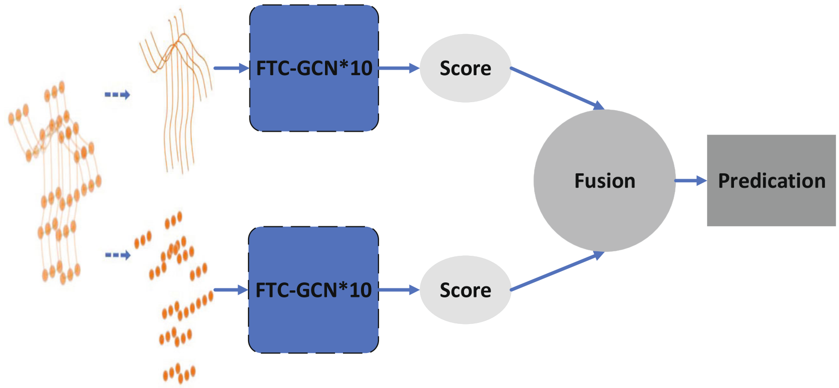

Fig. 1 **The illustration of two-stream fusion architecture**. The top is the bone flow and the bottom is the joint flow. The two-stream scores are fused to get the final prediction result

each other to improve recognition accuracy. The two-stream architecture is shown in Fig. 1, they are trained independently. Given a sample, we first calculate the data of bones based on the data of joints. Then, the joint data and bone data are fed into the joint stream and bone stream, respectively. Finally, the $softmax$ scores of the two streams are added to obtain the fused score and predict the action label. The details of a basic block (FTC-GCN) are listed in Table 1.

A basic block (FTC-GCN) is shown in Fig. 2. In Fig. 1, the joint flow and bone flow both contain 10 basic blocks (FTC-GCN), and the structure is exactly the same. The output channel from the first layer to the fourth layer is 64. The fifth to seventh layers include 128 output channels. The output channels from the eighth layer to the tenth layer are 256. The strides of the fifth and eighth layers is 2 and the other layers are 1. At last, the FC layer is used to generate the final recognition score.

3.3 The FTC graph convolutional networks

As shown in Fig. 2, one basic block is the series of one spatial GCN (FTGCN), one unified attention module (CSTA), and one temporal GCN (TCN). The FTGCN is used to concentrate more temporal information in the spatial layer. BN layer and Relu layer are regular operations. TCN will perform a $K_t \times 1$ convolution operation along the time dimension on the feature map with dimension $C \times T \times V$ obtained in FTGCN. C denote the number of channels, T denote the number of keyframes and V denote the number of joints. Use residual connections to optimize training and gradient propagation.

3.4 Focus on temporal graph convolutional module

Many existing GCN models pay attention to the spatial information and neglect the temporal information. To solve the above problems in this work, we propose a focus on temporal graph convolutional module which can focus more temporal information and properly balance them for each action.

In (3), the A_k, B_k, and C_k matrices only focus on the possible connections and connection strengths between joints for a certain action without considering time. To pay more attention to the temporal information, we modify (3) into the following form:

$$x_{out} = \sum_{k=1}^{3} (\hat{A}_k + B_k + S_k + \lambda T_k) x_{in} W_k \tag{4}$$

As illustrated in (3), $x_{in} \in R^{C_{in} \times T \times V}$ denote the input feature, $x_{out} \in R^{C_{out} \times T \times V}$ denote the output feature. Here C_{in} and C_{out} denote the number of channels. V denote the number of joints and T denote the length of the skeleton action sequence. As mentioned earlier, in (3), $A_k \in V \times V$ represents the adjacency matrix of the human body structure graph. Before adding to B_k, A_k is normalized to $\hat{A}_k = D_k{}^{-\frac{1}{2}}(A_k + I_{vk})D_k{}^{-\frac{1}{2}} \in R^{v \times v}$, and B_k is initially set to the matrix of $V \times V$ based on A_k. The parameters in the B_k are constantly changing with training. It serves as a supplement to the adjacency matrix A_k. Different joints are connected to different actions during training. $S_k \in R^{v \times v}$ is a data-dependent $V \times V$ matrix, and the parameters in the matrix are normalized to a value of 0-1. If the value is not zero, it means that the two joints are connected or not. The larger the

Table 1 The backbone network of FTC-GCN,which includes ten FTC-GCN blocks

FTC-GCN Block	Block 1	Block 2-4	Block 5	Block 6-7	Block 8	Block 9-10
(C_{in}, C_{out},stride)	(3,64,1)	(64,64,1)	(64,128,2)	(128,128,1)	(128,256,2)	(256,256,1)

The feature dimensions are presented. (C_{in} and C_{out} denote the number of channels)

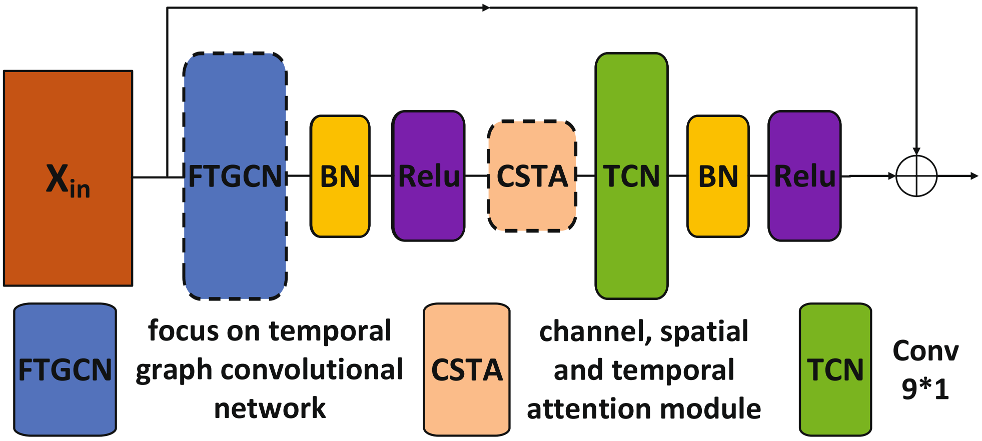
Xin
FTGCN
BN
Relu
CSTA
TCN
BN
Relu
FTGCN
focus on temporal graph convolutional network
CSTA
channel, spatial and temporal attention module
TCN
Conv 9*1

Fig. 2 The illustration of a basic block (FTC-GCN). The FTGCN is used to concentrate more temporal information in the spatial layer, and TCN means 9 × 1 temporal convolution. BN layer and Relu layer are regular operations. CSTA represents the unified attention module. Use residual connections to optimize training and gradient propagation

value, the stronger the connection strength of the two joints and the higher the correlation with the action. $T_k \in R^{v\times v}$ is a dynamic graph, as a supplement to temporal information, we use T_k to concentrate as much temporal information as possible. The spatial information at the beginning of the action changes greatly, and as the action progresses, the time information becomes more and more important. According to this characteristic, we use a λ to adjust the importance of the temporal graph for different layers. Convs denote the 1 × 1 convolution operation. Obtain the S_k matrix through the two Convs in Fig. 3. Convt denote the 9 × 1 convolution operation. Get the $T_k \in R^{v\times v}$ matrix through two Convt

$$S_k = softmax\left(\left(x_{in}^T w_{\theta k}^T\right)\left(w_{\phi k} x_{in}\right)\right) \tag{5}$$

$$T_k = softmax\left(\left(x_{in}^T w_{\eta k}^T\right)\left(w_{\xi k} x_{in}\right)\right) \tag{6}$$

where $w_{\phi k} \in R^{C_{in}\times v}$ and $w_{\theta k} \in R^{C_{in}\times v}$ correspond to the two Convs in Fig. 3. $w_{\theta k}$ and $w_{\phi k}$ denote the 1 × 1 convolution operation with C convolution kernels. $w_{\eta k} \in R^{C_{in}\times v}$ and $w_{\xi k} \in R^{C_{in}\times v}$ correspond to the two Convs in Fig. 3. $w_{\eta k}$ and $w_{\xi k}$ denote the 9 × 1 convolution operation with C convolution kernels.

3.5 Attention module

The channel, spatial, and temporal dimensions often contain redundant information. To better integrate channel, spatial and temporal information, we propose a unified attention model of the channel, spatial and temporal (CSTA).

As shown in Fig. 4. Channel attention focuses on important channels, the area of Adaptive Avgpool is 1 × 1, and the dimension of the feature map after passing the channel attention module remains unchanged. Spatial attention focuses on important spatial information. In spatial attention, the spatial information column is averaged and then the convolution operation is performed, the size of the convolution kernel is set according to the number of joint points in different datasets. After spatial attention, the feature map dimension changes from $C \times T \times V$ to $1 \times 1 \times V$. Temporal attention can help the model pay different levels

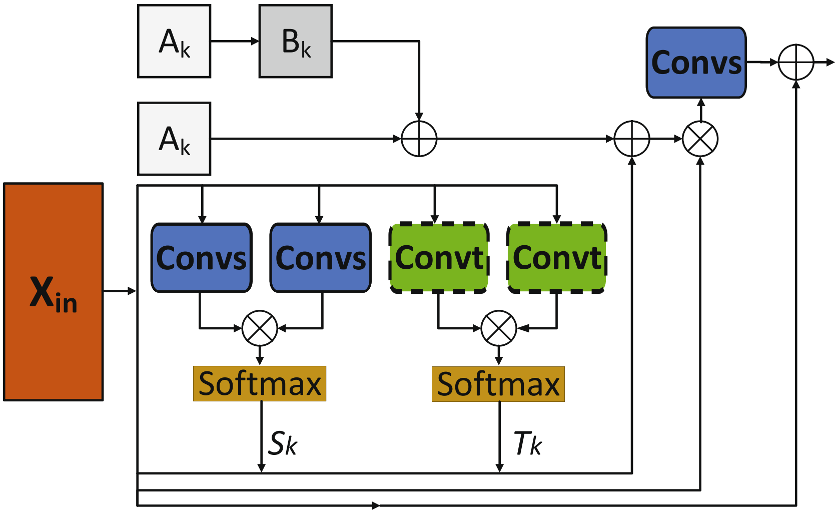

Fig. 3 The illustration of Focus on Temporal Graph Convolutional Networks (FTGCN). A_k is a fixed graph, B_k, S_k and T_k are dynamic graphs. The Convs indicates that the 1 × 1 convolution operation. The Convt denotes the 9 × 1 convolution operation. ⊕ denotes the element-wise addition. ⊗ denotes the matrix multiplication. Use λ to adjust the importance of the temporal graph for different layers. The residual connection is only needed when C_{in} is not the same as C_{out}

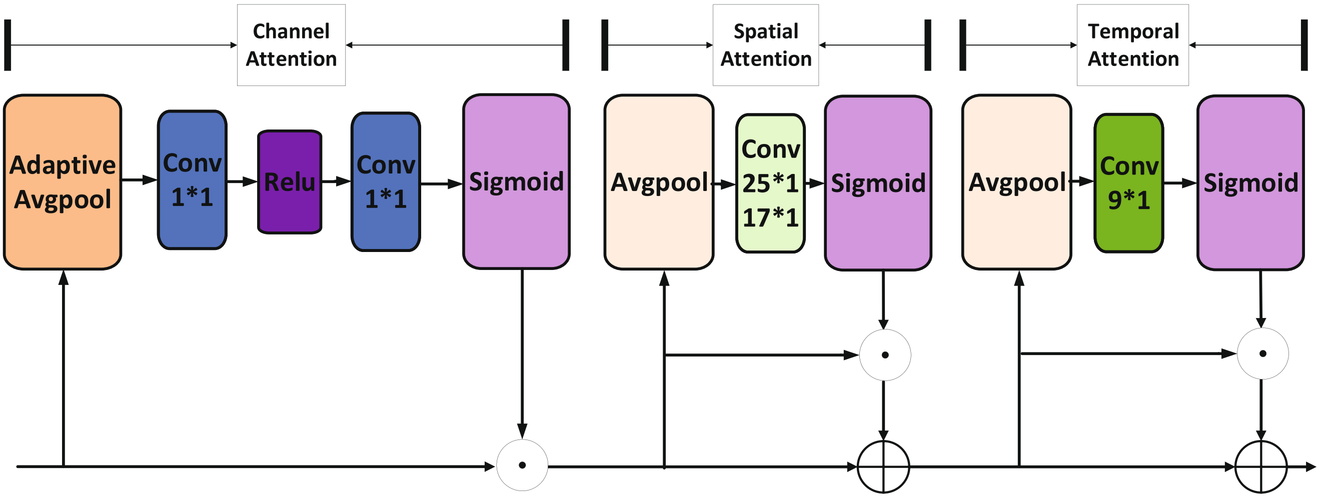

Fig. 4 **Illustration of the unified attention module**. ⊙ denotes the element-wise multiplication. ⊕ denotes the element-wise addition

of attention for each of the frames. After temporal attention, the feature map dimension changes from $1 \times 1 \times V$ to $1 \times T \times 1$. Finally, the dimension becomes $C \times 1 \times 1$ through two fully connected layers.

4 Experiments

We first introduced two large-scale datasets NTU-RGB+D [36] and Kinetics-Skeleton [37] in this field, and then introduced the details of the experiment and some hyperparameters. Then we conducted a large number of experiments on two large-scale datasets and compared 17 methods on NTU-RGB+D [36] and 8 methods on Kinetics-Skeleton [37], and finally we show the results of ablation experiments and tributary results.

4.1 Datasets

NTU-RGBD: NTU RGB+D [36] is a large-scale and multi-modality indoor-captured dataset for skeleton-based action recognition, it contains four modalities of data. Here, we

Table 2 Comparison with the state-of-the-art methods on NTU-RGB + D dataset

Methods	Year	Cross-Subject(%)	Cross-View(%)
H-RNN [32]	2015(CVPR)	59.1	64.0
Part-aware LSTM [36]	2016(CVPR)	62.9	70.3
ST-LSTM [39]	2016(ECCV)	69.2	77.7
Two-stream RNN [40]	2017(CVPR)	71.3	79.5
Ensemble TS-LSTM [41]	2017(ICCV)	74.6	81.3
Visualization CNN [42]	2017(CVPR)	76.0	82.6
C-CNN + MTLN [15]	2017(CVPR)	79.6	84.8
Temporal Conv [43]	2017(CVPR)	74.3	84.1
VA-LSTM [44]	2017(ICCV)	79.4	87.6
ST-GCN [28]	2018(CVPR)	81.5	88.3
DPRL [29]	2018(CVPR)	83.5	89.8
PB-GCN [30]	2018(BMVC)	87.5	93.2
RA-GCN [45]	2019(ICIP)	85.9	93.5
AS-GCN [46]	2019(CVPR)	86.8	94.2
2s-AGCN [34]	2019(CVPR)	88.5	95.1
DGNN [47]	2019(CVPR)	89.9	96.1
CGCN [48]	2020(CVPR)	90.3	96.4
FTC-GCN(Ours)	2020	90.4	96.5

Table 3 Comparison with the state-of-the-art methods on Kinetics dataset

Methods	Year	Top-1(%)	Top-5(%)
Feature Enc [7]	2015(CVPR)	14.9	25.8
Deep LSTM [36]	2016(CVPR)	16.4	35.3
Temporal Conv [43]	2017(CVPR)	20.3	40.0
ST-GCN [28]	2018(CVPR)	30.7	52.8
AS-GCN [46]	2019(CVPR)	34.8	56.5
2s-AGCN [34]	2019(CVPR)	36.1	58.7
DGNN [47]	2019(CVPR)	36.9	59.6
CGCN [48]	2020(CVPR)	37.5	60.4
FTC-GCN(Ours)	2020	37.8	60.7

use only the skeleton data, it is formed by three Microsoft Kinect v2 cameras to capture 3D skeleton data at the same time and marked 25 points as joint points. There are 56,880 action clips in 60 classes. The action clips are performed by 40 volunteers whose ages range from 10 to 35. These cameras have the same height but with different horizontal angles. There are two benchmarks for this dataset: (1) Cross Subject (CS), where the subjects are divided into two groups of 20 people each. The training sets included 40,320 samples from 20 subjects and the test sets included 16,560 samples from the remaining 20 subjects. (2) Cross View (CV): Divided by camera angle, the training sets consisted of 37,920 samples captured by camera 2(0°) and 3(45°), while the test sets consisted of 18,960 samples captured by camera 1(-45°).

Kinetics-Skeleton: Kinetics400 [37] is a large-scale dataset that contains about 300,000 video clips in 400 classes from YouTube for human action recognition. The dataset is obtained on Kinetics400 through the OpenPose toolbox, it predicts 18 $2D$ joint nodes and confidence score for each person. The data is divided into training sets and test sets at a ratio of about $12:1$, with each data cut to 300 frames. We report the accuracy of the top 1 and top 5 against the benchmark.

Table 4 The importance of FTGCN and unified Attention (CSTA) were evaluated on NTU-RGB + D dataset

Stream	Model	Accuracy(%)
Joint stream only	AGCN	93.83
	FTGCN wo/λ	94.34
	FTGCN	94.59
	FTGCN-CSTA	95.14
Two stream	AGCN	95.10
Two stream	FTC-GCN	96.50

4.2 Implementation details

Our framework is implemented on PyTorch [38] and the code will be released later (https://github.com/dongle329/FTC-GCN). All experiments use stochastic gradient descent with a Nesterov momentum of 0.9. We use two NVIDIA GeForce 1080Ti GPUs for the model training and the batch size is 16, the weight decay is 0.0001 and the initial learning rate is 0.1.

In our experiment, we trained two flows successively, and each stream occupied two GPUs during training. Finally, we fused the join flow and the bone flow.

For NTU RGB+D [36], the learning rate is divided by 10 at the 30th and 40th epochs. 60 epochs in total. The training time of joint flow on the Cross-Subject benchmark is about 41 hours, the training time of bone flow is about 41 hours. The training time of joint flow on the Cross-View benchmark is about 40 hours, and the training time of bone flow is about 40 hours.

For Kinetics-Skeleton [37], the learning rate is divided by 10 at the 45th and 55th epochs. 65 epochs in total. The training time of joint flow is about 203 hours, and the training time of bone flow is about 202 hours.

4.3 Comparisons to the state-of-the-arts

The results are listed in Tables 2 and 3, respectively. Extensive experiments on two large-scale datasets,

Table 5 Two-stream fusion results on NTU-RGB + D dataset

Methods	Cross-Subject(%)	Cross-View(%)
FTC-GCN (Joint)	87.9	95.1
FTC-GCN (Bone)	88.2	95.0
FTC-GCN (Joint&Bone)	90.4	96.5

compared with 17 methods on NTU-RGB+D [36] and 8 methods on Kinetics-Skeleton [37]. These all show that for skeleton-based human action recognition, our method achieves the best performance, which suggests the superiority of our model.

4.4 Ablation study

As shown in Table 4, we conducted ablation experiments about FTGCN, λ and CSTA on NTU-RGB+D [36] .

The accuracy of AGCN is 93.83%. First, we replace the spatial GCN with our proposed FTGCN without the λ parameter, and the obtained accuracy is 94.34%. The increase in accuracy proves the effectiveness of focusing on time information as much as possible. The accuracy of FTGCN of 94.59% proves the effectiveness of the λ parameter, that is the deeper the layer, the more important the temporal information. Finally, the accuracy obtained by adding the CSTA module(FTC-GCN) is 95.14%, which is increased by 0.55%. Combining the scores of joint flow and bone flow to obtain the final accuracy of our network is 96.50%

4.5 The results of two-stream fusion

In this section, we show the results of two streams fused according to different benchmarks on two datasets NTU-RGB+D [36] and Kinetics [37] .

Table 5 shows the two-stream fusion results for different benchmarks on NTU-RGB+D [36] dataset. For cs [36] benchmark, the accuracy of joint flow is 87.9%, the accuracy of bone flow is 88.2%, and the accuracy after fusion is 90.4%. For the cv [36] benchmark, the accuracy of joint flow is 95.1.9%, the accuracy of bone flow is 95.0%, and the accuracy after fusion is 96.5%.

Table 6 shows the two-stream fusion results for different benchmarks on Kinetics [37] dataset. For Top-1 [37] benchmark, the accuracy of joint flow is 36.1%, the accuracy of bone flow is 35.6%, and the accuracy after fusion is 37.8%. Based on this, the two-stream fusion can further boost the performance of the proposed method.

Table 6 Two-stream fusion results of Top-1 accuracy on Kinetics dataset

Methods	Accuracy(%)
FTC-GCN (Joint)	36.1
FTC-GCN (Bone)	35.6
FTC-GCN (Joint&Bone)	37.8

5 Conclusions

We design a novel temporal graph convolutional module(FTGCN) which can focus more temporal information and properly balance them for each action. This approach increases the flexibility and generalization capacity of the model. It is also confirmed that the temporal information of graph is more suitable for the action recognition task than the human-body-based graph. To integrate channel, spatial and temporal information, we propose a unified attention (CSTA) module, which helps the model paying more attention to the important joints, frames and features. In addition, both the FTGCN module and the CSTA module can be easily incorporated into the adaptive graph convolutional networks (AGCN), and significantly improve the performance of AGCN. Due to the contribution of these two modules, our FTC-GCN achieves the best performance compared to the methods listed in the table on the two large-scale action recognition datasets.

References

1. Wang X (2013) surveillance, Intelligent multi-camera video. A Rev Pattern Recognit Lett 34(1):3–19
2. Turaga P, Chellappa R, Subrahmanian VS, Udrea O (2008) Machine recognition of human activities: a survey. IEEE Trans Circ Syst Video Technol 18(11):1473–1488
3. Ellis C, Masood SZ, Tappen MF, LaViola JJ, Sukthankar R (2013) Exploring the trade-off between accuracy and observational latency in action recognition. Int J Comput Vis 101(3):420–436
4. Zhang W, Smith ML, Smith LN, Farooq A (2016) Gender and gaze gesture recognition for human-computer interaction. Comput Vis Image Underst 149:32–50
5. Camporesi C, Kallmann M, Han JJ (2013) Vr solutions for improving physical therapy. In: 2013 IEEE Virtual Reality (VR). IEEE, pp 77–78
6. Vemulapalli R, Arrate F, Chellappa R (2014) Human action recognition by representing 3d skeletons as points in a lie group. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 588–595
7. Fernando B, Gavves E, Oramas JM, Ghodrati A, Tuytelaars T (2015) Modeling video evolution for action recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 5378–5387
8. Krizhevsky A, Ilya S, Geoffrey HE (2017) Imagenet classification with deep convolutional neural networks. Communications of the Acm, USA
9. Greff K, Srivastava RK, Koutnik J, Steunebrink BR, Schmidhuber J (2016) Lstm: A search space odyssey. IEEE Trans Neural Netw Learn Syst 1–11
10. Collobert R, Weston J, Bottou L, Karlen M, Kavukcuoglu K, Kuksa P (2011) Natural language processing (almost) from scratch. J Mach Learn Res 12(1):2493–2537
11. Tarwani KM, Edem S (2017) Survey on recurrent neural network in natural language processing. Int J Eng Trends Technol 48:301–304

12. Wang P, Li Z, Hou Y, Li W (2016) Action recognition based on joint trajectory maps using convolutional neural networks. Knowl Based Syst 102–106
13. Li C, Hou Y, Wang P, Li W (2017) Joint distance maps based action recognition with convolutional neural networks. IEEE Signal Process Lett 24(5):624–628
14. Donahue J, Hendricks LA, Rohrbach M, Venugopalan S, Guadarrama S, Saenko K, Darrell T (2017) Long-term recurrent convolutional networks for visual recognition and description. In: 2015 IEEE conference on computer vision and pattern recognition (CVPR)
15. Ke Q, Bennamoun M, An S, Sohel F, Boussaid F (2017) A new representation of skeleton sequences for 3d action recognition. In: CVPR, p 2017
16. Shao Z, Li Y, Yao G, Yang J, Wang Z (2018) A hierarchical model for action recognition based on body parts. In: 2018 IEEE international conference on robotics and automation (ICRA)
17. Kipf TN, Welling M (2016) Semi-supervised classification with graph convolutional networks, arXiv:1609.02907
18. Gao H, Wang Z, Ji S (2018) Large-scale learnable graph convolutional networks. In: Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery and data mining, pp 1416–1424
19. Wu F, Zhang T, Souza AHd Jr, Fifty C, Yu T, Weinberger KQ (2019) Simplifying graph convolutional networks, arXiv:1902.07153
20. Chen J, Ma T, Xiao C (2018) Fastgcn: fast learning with graph convolutional networks via importance sampling, arXiv:1801.10247
21. Balcilar M, Renton G, Heroux P, Gauzere B, Adam S, Honeine P (2020) Bridging the gap between spectral and spatial domains in graph neural networks
22. Ma Y, Wang S, Aggarwal CC, Tang J (2019) Graph convolutional networks with eigenpooling. In: Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery and data mining, pp 723–731
23. Zhang M, Cui Z, Neumann M, Chen Y (2018) An end-to-end deep learning architecture for graph classification. In: AAAI, vol 18, pp 4438–4445
24. Bresson X, Laurent T (2017) Residual gated graph convnets, arXiv:1711.07553
25. Wang H, Leskovec J (2020) Unifying graph convolutional neural networks and label propagation, arXiv:2002.06755
26. Huang W, Zhang T, Rong Y, Huang J (2018) Adaptive sampling towards fast graph representation learning. Adv Neural Inform Process Syst 31:4558–4567
27. Sun K, Lin Z, Zhu Z (2019) Adagcn: Adaboosting graph convolutional networks into deep models, arXiv:1908.05081
28. Yan S, Xiong Y, Lin D (2018) Spatial temporal graph convolutional networks for skeleton-based action recognition, arXiv:1801.07455
29. Tang Y, Tian Y, Lu J, Li P, Zhou J (2018) Deep progressive reinforcement learning for skeleton-based action recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 5323–5332
30. Thakkar K, Narayanan P (2018) Part-based graph convolutional network for action recognition, arXiv:1809.04983
31. Ye F, Tang H, Wang X, Liang X (2019) Joints relation inference network for skeleton-based action recognition. In: 2019 IEEE international conference on image processing (ICIP). IEEE, pp 16–20
32. Du Y, Wang W, Wang L (2015) Hierarchical recurrent neural network for skeleton based action recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1110–1118
33. Li B, Dai Y, Cheng X, Chen H, Lin Y, He M (2017) Skeleton based action recognition using translation-scale invariant image mapping and multi-scale deep cnn. In: 2017 IEEE international conference on multimedia and expo workshops (ICMEW). IEEE, pp 601–604
34. Shi L, Zhang Y, Cheng J, Lu H (2019) Two-stream adaptive graph convolutional networks for skeleton-based action recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 12,026–12,035
35. Shuman DI, Narang SK, Frossard P, Ortega A, Vandergheynst P (2013) The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. IEEE Signal Process Magaz 30(3):83–98
36. Shahroudy A, Liu J, Ng TT, Wang G (2016) Ntu rgb+d: A large scale dataset for 3d human activity analysis. 1010–1019
37. Kay W, Carreira J, Simonyan K, Zhang B, Zisserman A (2017) The kinetics human action video dataset
38. Paszke A, Gross S, Chintala S, Chanan G, Yang E, DeVito Z, Lin Z, Desmaison A, Antiga L, Lerer A (2017) Automatic differentiation in pytorch
39. Liu J, Shahroudy A, Xu D, Wang G (2016) Spatio-temporal lstm with trust gates for 3d human action recognition. In: European conference on computer vision. Springer, pp 816–833
40. Wang H, Wang L (2017) Modeling temporal dynamics and spatial configurations of actions using two-stream recurrent neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 499–508
41. Lee I, Kim D, Kang S, Lee S (2017) Ensemble deep learning for skeleton-based action recognition using temporal sliding lstm networks. In: 2017 IEEE international conference on computer vision (ICCV)
42. Liu M, Hong L, Chen C (2017) Enhanced skeleton visualization for view invariant human action recognition. Pattern Recognition
43. Kim TS, Reiter A (2017) Interpretable 3d human action analysis with temporal convolutional networks
44. Jianru X, Wenjun Z, Junliang X, Cuiling L, Nanning Z, Pengfei Z (2017) View adaptive recurrent neural networks for high performance human action recognition from skeleton data. In: Proceedings of the IEEE international conference on computer vision
45. Song YF, Zhang Z, Wang L (2019) Richly activated graph convolutional network for action recognition with incomplete skeletons 2019
46. Li M, Chen S, Chen X, Zhang Y, Wang Y, Tian Q (2019) Actional structural graph convolutional networks for skeleton-based action recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 3595–3603
47. Shi L, Zhang Y, Cheng J, Lu H (2019) Skeleton-based action recognition with directed graph neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 7912–7921
48. Yang D, Li MM, Fu H, Fan J, Leung H (2020) Centrality graph convolutional networks for skeleton-based action recognition, arXiv:2003.03007