Triplet attention multiple spacetime-semantic graph convolutional network for skeleton-based action recognition

Yanjing Sun[1,2] · Han Huang[1] · Xiao Yun[1] · Bin Yang[1] · Kaiwen Dong[1]

Accepted: 17 March 2021 / Published online: 24 April 2021

Abstract

Skeleton-based action recognition has recently attracted widespread attention in the field of computer vision. Previous studies on skeleton-based action recognition are susceptible to interferences from redundant video frames in judging complex actions but ignore the fact that the spatial-temporal features of different actions are extremely different. To solve these problems, we propose a triplet attention multiple spacetime-semantic graph convolutional network for skeleton-based action recognition (AM-GCN), which can not only capture the multiple spacetime-semantic feature from the video images to avoid limited information diversity from single-layer feature representation but can also improve the generalization ability of the network. We also present the triplet attention mechanism to apply an attention mechanism to different key points, key channels, and key frames of the actions, improving the accuracy and interpretability of the judgement of complex actions. In addition, different kinds of spacetime-semantic feature information are combined through the proposed fusion decision for comprehensive prediction in order to improve the robustness of the algorithm. We validate AM-GCN with two standard datasets, NTU-RGBD and Kinetics, and compare it with other mainstream models. The results show that the proposed model achieves tremendous improvement.

Keywords Action recognition · Graph convolutional neural network · Spacetime-semantic feature · Triplet attention · Fusion decision

1 Introduction

Human action recognition, playing an important role in video surveillance [12] and human-machine interaction [43], has attracted considerable attention in computer vision in recent years, but lacks full solutions due to complex semantic information [3, 5]. Earlier attempts only encode key points to feature vectors in a single-time step before analysing the time sequence [9]. However, simply encoding the key points as a vector sequence cannot fully express the joint dependence in space. Other attempts design handcrafted features to denote the skeleton [41] and are difficult to generalize to other areas. With the development of deep learning, a large number of action recognition methods are developed based on the convolutional neural network (CNN) [18, 20] and recurrent neural network (RNN) [6, 33]. However, CNN cannot provide satisfactory information representation because the skeleton feature map is not a conventional RGB image and the kernel will convolute substantial empty pixel information in the process of convolution. RNN performs well in processing sequence data but is not suitable for spatial information extraction in each frame [1, 49]. Recently, the graph convolutional network (GCN) has successfully generalized the convolution of RGB images to that of generalized graphs [21, 29, 30] and has been widely used in many fields, which provides a new way to capture the dependencies between joints for the spatial representation of skeleton-based action recognition. Yan et al. [45] established the spatial-temporal graph convolutional network (ST-GCN) that constructed a skeleton map with vertices as joints and skeletons as edges, and used GCN and the temporal convolution network (TCN) to separately extract spatial features and temporal features from skeleton images.

Although ST-GCN and subsequent research approaches exhibit the ability to extract spatial and temporal features,

✉ Xiao Yun
xyun@cumt.edu.cn

[1] China University of Mining and Technology, Xuzhou, China

[2] Xuzhou Engineering Research Center of Intelligent Industry Safety and Emergency Collaboration, Xuzhou, China

there still exist drawbacks in most GCN-based models. For example, (1) existing research extracts features while compressing a large number of video frames, which is not a good way to extract complicated features with long-term time. For instance, the action of sitting down is much shorter and less complex than Tai Chi action with respect to duration and complexity. Existing research handles these two actions in exactly the same manner, which tends to produce false results; (2) As component elements of an action, different frames, joint points, and network features contribute differently for completing the action. Most GCN-based models deal with these features in the same way or only distinguish them with simple attention, which cannot effectively overcome the interference of redundant information. For example, the features of the knee, thigh, shank and foot are very important for the action of walking apart and should be enhanced in the process of identifying the behaviour. Additionally, the ending process of the action of checking time (from a watch) is more important than the beginning in the judgement. The frames and channels containing representative features of the action should be paid more attention.

Recently, the attention mechanism has gradually become widely employed in skeleton-based action recognition. Graph attention [8, 28] specifies different weights for different nodes in a neighbourhood, and convolution attention [14, 44] devotes more attention to those important frames and channels for significant interpretability and effectiveness [14, 40]. However, the past works only focused on applying graph attention or convolution attention alone on skeleton data, and therefore suffer from limitations of feature diversity.

To solve all of the above problems, we propose the triplet attention multiple spacetime-semantic graph convolutional network (AM-GCN) for skeleton-based action recognition. The main contributions in this paper are summarized as follows:

- The proposed multiple spacetime-semantic information extraction model in the AM-GCN network can extract spatial, temporal, as well as semantic information from actions with different completion times and complexity for skeleton-based action recognition, which improves spatial-temporal expression and the ability to distinguish different actions.
- The proposed triplet attention mechanism assigns weights to each frame and channel with convolution attention, and simultaneously to each key point with graph attention.
- We present the proposed fusion decision to comprehensively fuse and predict multiple spacetime-semantic features to improve the robustness of the algorithm. The proposed model achieves outstanding results on two large-scale datasets. We perform extensive experiments to demonstrate the effectiveness of our model.
- Our AM-GCN framework is general and applicable to different graph convolutional networks for skeleton-based action recognition, e.g., AS-GCN [23], 2S-AGCN [35], etc., which only minimally increases computing power and memory consumption. The ablation studies have also been conducted experimentally to validate the effectiveness and general applicability of each proposed component of our framework.

The rest of the paper is organized as follows. Section 2 reviews the related work on skeleton-based action recognition and the graph convolutional neural network. Section 3 presents details of the proposed triplet attention multiple spacetime-semantic graph convolutional network for the skeleton-based action recognition framework. Experimental performance and conclusions are presented in Sections 4 and 5, respectively.

2 Related works

2.1 Skeleton-based action recognition

Traditional skeleton-based action recognition methods usually use hand-crafted features [41] based on visually intuitive characteristics to capture action patterns [15, 42]. For example, Vemulapalli et al. [41] represented 3D skeletons as points in a lie group for human action recognition; Hussein et al. [15] used a temporal hierarchy of covariance descriptors on 3D joint locations. With only hand-crafted features, these methods are not satisfactory in terms of generalization.

With the development of deep learning, human action characteristics learning through neural networks has become mainstream, where the most widely used models are CNN and RNN. CNN-based methods model the skeleton data as pseudoimages [18–20]. For instance, Kim et al. [20] proposed an end-to-end temporal convolution network to achieve action recognition model learning. Ke et al. [18] transformed each skeleton sequence into three clips before using deep neural networks to learn several frames for spatial temporal feature. Shahroudy et al. [34] propose a deep autoencoder-based shared-specific feature factorization network to separate input multimodal signals into a hierarchy of components. RNN-based methods are able to capture the time dependency between consecutive frames, such as bi-RNN [6], Deep LSTM [33], etc. Du et al. [6] proposed a hierarchical bidirectional RNN model to identify the skeleton sequence, which divides the human body into different parts and sends them to different sub-networks (bi-RNN). Different from storing the long-term memory of the entire body in the cell, Shahroudy et al. [33] used a part-based method to store the memory of each part independently and connected them together to form a large cell (Deep LSTM).

However, neither CNN nor RNN can fully denote the structure of the skeleton data because skeleton data are embedded in the form of natural graphics rather than vector sequences or 2D grids. Therefore, Yan et al. [45] proposed the spatial-temporal graph convolutional network (ST-GCN), which can adaptively learn the graph structure from skeleton data and outperform previous methods. Based on this groundbreaking work, many researchers have contributed different levels of improvement and innovation [4, 23, 35–37]. For example, Li et al. [23] used an encoder-decoder structure to capture action-specific latent dependencies and high-order polynomials of the adjacency matrix to obtain structural dependencies (AS-GCN). Shi et al. [35] proposed a two-stream adaptive graph convolutional network for robustness enhancement (2S-AGCN). Si et al. [37] explored the method of combining GCN with the long short-term memory (LSTM) to produce a good result (AGC-LSTM). However, only one output is used in these methods to represent all actions, which will cause the model to be unable to obtain good prediction results for actions with different complexity and completion times. To solve this problem, our AM-GCN network sets multi-semantic and multi-temporal features to yield more comprehensive representations of different actions.

2.2 Graph convolution neural network

Traditional neural networks cannot directly process graphs which have more general data structures than image and sequence data. Existing graph convolution models are mainly divided into two architectures: graph neural network (GNN) [16, 32] and graph convolutional network (GCN) [7, 30]. GNN is a combination of graphs and recurrent neural networks in which each node can capture semantic relationships and structural information within its neighbour nodes through iterations and message updates [32]. GCN is based on CNN and can be classified into spectral GCN and spatial GCN. Spectral GCN considers the locality of the graph convolution in the form of spectral analysis and applies a spectral filter in the spectral domain [7]. Spatial GCN uses a graph convolution operation to calculate a new feature vector for each node by aggregating its neighbourhood information [13, 29, 30]. The proposed model follows the spatial GCN framework to learn spatial-temporal features.

Since the edges of the graph are simple and fixed, the weights of neighbour nodes in the GCN process are also simple and fixed. To add interpretable weights to GCN, Lu et al. [28] proposed a graph attention interaction model embedded with the graph attention block to learn the spatial and temporal evolutions of the collective activity and predict the activity labels. Si et al. [38] used a spatial-temporal attention network to selectively focus on discriminative spatial and temporal features. Recently, Yang et al. [47] and Shi et al. [36] also attempted to add the convolution attention mechanism to GCN to achieve better results. However, these works only consider graph attention or convolution attention alone with respect to skeleton data, whereas our algorithm combines them together to enhance information diversity in the attention mechanism.

3 Model architecture

The architecture of the proposed AM-GCN is shown in Fig. 1. First, AM-GCN constructs a graph structure based on the input skeleton information using the GCN-based backbone composed of GCN and TCN to initially extract spatial-temporal features from the graph structure. AM-GCN then constructs multiple spacetime-semantic feature maps through extracted features and judges the importance of different feature maps with the triplet attention mechanism. Finally, the fusion decision is used to classify the input video and obtain the action label.

3.1 Background

Usually, the original skeleton data in one sequence represent a set of joint vectors in one frame [20]. The common structure of the spatial-temporal graph follows the work of ST-GCN [45]. In one frame, the spatial graph is constructed according to the natural skeleton connection relationships of the human body. Between frames, the joints are connected with the same joints of adjacent frames. The structure of the graph is shown in Fig. 2.

Based on the structure of the spatial-temporal graph mentioned above, we perform spatial-temporal graph convolution operations on the graph. In the spatial dimension, the form of the graph convolution operation is defined as:

$$f_{out}\left(v_i\right) = \sum_{v_j \in B_i\left(v_i\right)} \frac{1}{Z_i\left(v_j\right)} f_{in}\left(v_j\right) \cdot w\left(l_i\left(v_{tj}\right)\right), \tag{1}$$

where v denotes the vertex of the graph structure. The neighbour set of the node v_i is defined as $B_i\left(v_i\right) = \left\{v_i \middle| d\left(v_i, v_j\right) \leq D\right\}$, where $d\left(v_i, v_j\right)$ is the minimum path length from v_i to v_j, and D is set to 1 for all cases. Note that a node is a vertex in graph structure and is a keypoint of the human body in the skeleton graph. B_i denotes the convolution sampling area of v_i, and w denotes the weight function. l_i is a mapping function specially designed to map

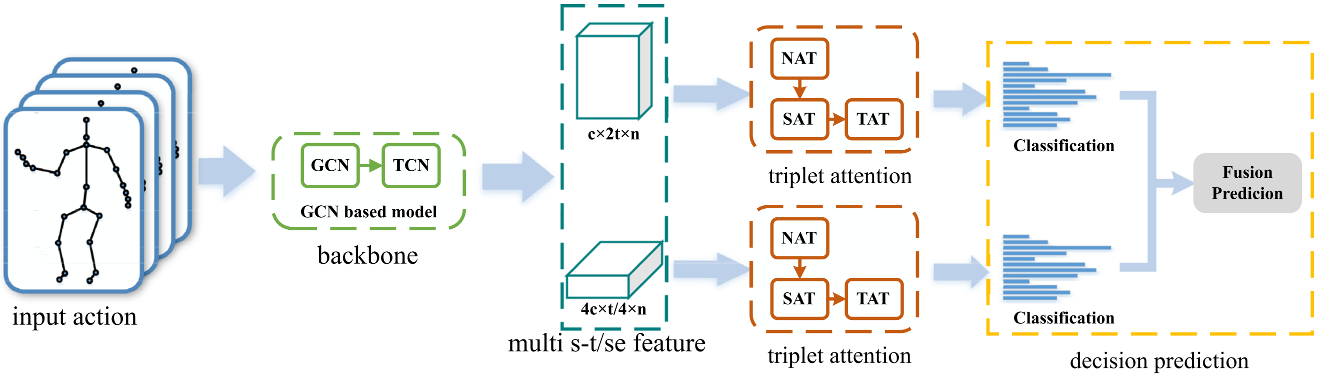


Fig. 1 The architecture of the proposed triplet attention multiple spacetime-semantic graph convolutional network (AM-GCN). NAT, SAT, and TAT respectively represent node attention, channel attention and temporal attention

every vertex with a weight vector. The subset is a neighbour set collection of node v_i, which is divided by the spatial configuration partitioning. As shown in Fig. 2, B_i is divided into three subsets according to the regularity of human action by setting the convolution kernel size to 3, in which V_{i1} is the root node itself (red node in the figure), V_{i2} is a centripetal subset (orange node in the figure), which denotes that the adjacent node is closer to the gravity centre than the root node, and V_{i3} is a centrifugal subset (green node in the figure), which denotes the remaining neighbouring nodes. $Z_i\left(v_j\right)$ denotes the number of V_{ik} contained in the v_j, normalizes the feature representation, and balances the contributions of different subsets to the output.

The time graph is constructed by connecting the same joints in continuous frames. Therefore, it is very simple to extend the spatial graph convolutional network into the time domain. Specifically, we perform the convolution on the time dimension of the input feature map: that is, the convolution kernel is $t \times 1$, where t is the size of the convolution kernel. The implementation of the graph

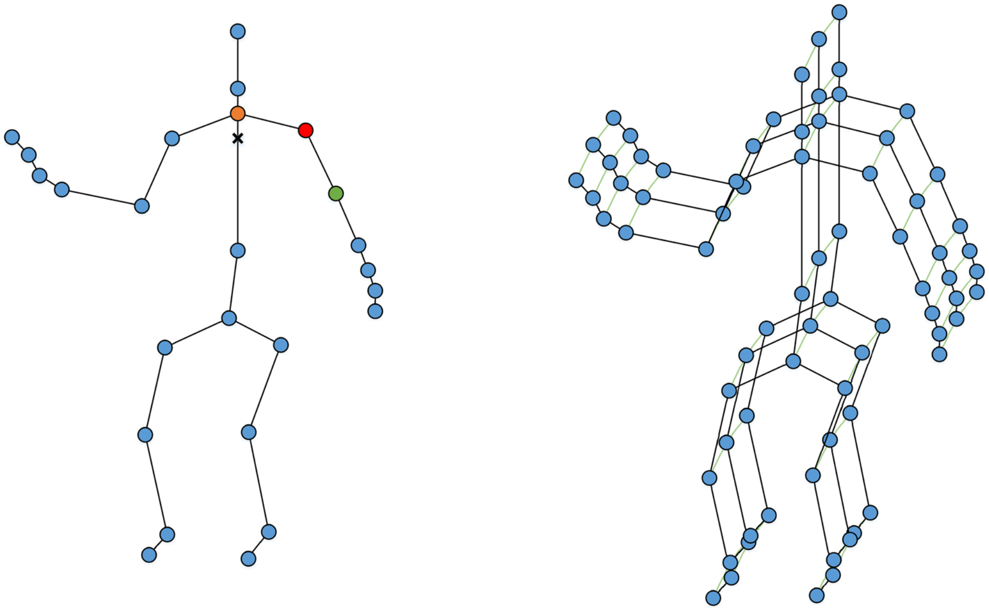


Fig. 2 Skeleton graph (the skeleton graph on the left is the realization form of one frame, and the skeleton graph on the right is the realization form between the adjacent frames. The red, orange and green colours in the left graph denote the root node, the centripetal node, and the centrifugal node)

convolution in the spatial dimension is not straightforward. The output of graph convolution is defined as:

$$f^{out} = \sum_{k=1}^{k} \mu_n \cdot \left(\Lambda_k^{-\frac{1}{2}} A_k \Lambda_k^{-\frac{1}{2}} f_{in} \right) W_k, \tag{2}$$

where f_{in} is the skeleton point input, $A_k = A + I$ is the adjacency matrix with the self-loop added to the graph structure, and $\Lambda_k = \sqrt{D_{ii} D_{jj}}$ is the geometric mean of the degree matrix of the root node and adjacent nodes. In addition, $\Lambda_k^{-\frac{1}{2}} A_k \Lambda_k^{-\frac{1}{2}}$ can be regarded as a convolution kernel, and μ_n denotes the edge weight matrix of the feature map. Thus, the entire convolution process uses edges to aggregate the node information to generate a new node representation.

Li et al. [23] proposed the A-link inference module (AIM) to infer actional links which capture action-specific latent dependencies and the S-link inference module to infer structure links which obtain long-range action links (AS-GCN):

$$\begin{aligned} f^{out} &= \sum_{l}^{L} \sum_{k=1}^{k} \mu_{struc} \cdot \left(\Lambda_k^{-\frac{1}{2}} A_k \Lambda_k^{-\frac{1}{2}} f_{in} \right) W_{struc} \\ &\quad + \lambda \sum_{c}^{C} \sum_{k=1}^{k} \mu_{act} \cdot \left(\Lambda_k^{-\frac{1}{2}} A_k \Lambda_k^{-\frac{1}{2}} f_{in} \right) W_{act}, \end{aligned} \tag{3}$$

where l is the polynomial order, λ is a hyperparameter which establishes a trade-off with respect to the importance between structural features and actional features, and other parameters are the same as in (2).

Furthermore, Shi et al. [35] proposed an adaptive graph convolutional network to adaptively learn the topology of the graph for different GCN layers and combine the second-order information of the skeleton data with the first-order information to improve the recognition performance (2S-AGCN):

$$f^{out} = \sum_{k=1}^{k} f_{in} \left(A_k + B_k + C_k \right), \tag{4}$$

where A_k is the same as the original normalized $N \times N$ adjacency matrix A_k in Eq 2. B_k is also an $N \times N$ adjacency matrix, but the elements of B_k are parameterized and optimized together with the other parameters in the training process. C_k is a data-dependent graph which can learn a unique graph for each sample. 2S-AGCN uses adaptive skeleton points which can obtain a more accurate classification on different actions than ST-GCN and AS-GCN with fixed skeleton points. The experiment results also showed that choosing 2S-AGCN as our backbone achieves the highest accuracy.

3.2 Multiple spacetime-semantic feature

Traditional GCNs exhibit the characteristic of the feature pyramid [24, 27], in which feature maps after different convolution or pooling layers can be regarded as multi-semantic information which is uncovered from different layers to obtain a more global expression. Francisco et al. [31] use a detector to select the candidate regions from each input frame and then identify the objects using a binarization technique to improve the detection of small objects. Neural networks are generally hierarchical, where different layers contain different levels of semantics, and there will be more semantic information in the output feature map than spacetime information. Multi-spacetime information combines spatial information extracted from one single frame and temporal information accumulated frame by frame, without considering which classification may not be correct for long-time actions.

Therefore, the proposed AM-GCN integrates multiple spatial-temporal features with semantic information to obtain correct classification results for both short-time and long-time actions. It naturally uses the pyramid shapes of different layers to construct a feature map with long-time dimension, which retains the original semantic information. The proposed multiple spacetime-semantic feature is not a simple combination of dual outputs, but a feature constructed by different layers of the backbone with different functions. After preliminary feature extraction by the backbone, video input generates semantic and spacetime output via multiple spacetime-semantic feature extraction, respectively. Since we cannot judge whether the video action is multi-semantic or multi-spacetime while the video is being typed, AM-GCN will generate two outputs at the same time and make a comprehensive decision after the subsequent attention mechanism.

GCN-based models are composed of several layers of spatial-temporal graph convolution operators. The time dimensions of the front, middle, and the back layers are t, $t/2$, and $t/4$, where dimension t denotes the specific frames of the video input. Since the convolution process requires compression of the last two dimensions of the feature map to extract semantic information, most models greatly compress the time dimension in the process of obtaining the prediction. In our model, we extract features from the feature maps with three different time dimensions $f^{\lambda} \in \mathbb{R}^{C \times T \times V}$ $(\lambda \in i, j, k)$, representing action information at different resolutions. C, T and V are the channel, the frame, and the number of key points of the feature map. This mode of operation can be implemented in most GCN-based models: we will particularly indicate this part in our experiment.

It is not wise to construct the multiple spacetime feature map by directly up-sampling the time dimension $t/4$ of the original output to the time dimension t of the front layers because the time information after direct up-sampling is randomly interpolated, which does not make much sense for action understanding. In this paper, we adopt the method of gradually concatenating the results of different graph convolutional layers to construct the multiple spatial-temporal feature map. First, we up-sample the time dimension of the feature map f^k to $t/2$ and use 1×1 convolution to keep all the dimensions of f^k consistent with f^j. We also attempt to sample the time dimension of the feature map f^i or f^j. We found that up-sampling f^k first yields a better result than other feature maps. For the specific concatenating process, our experimental result shows that up-sampling the time dimension to $t/2$ is better than t, $3t/2$ and $2t$. We hypothesize that the reason for this is that neither up-sampling nor concatenating will corrupt the information integrity of the video sequence. Thus, how to operate the features while maximizing the retention of the original information becomes a problem worthy of further experimentation. Experiments indicate that the processing method in this model is very effective.

We then concatenate the two time dimensions to obtain a $64 \times t \times n$ intermediate feature map. After that, the intermediate feature map is concatenated with f^i, also based on the time dimension. Finally, we obtain the final multiple spacetime feature map f^{st}. In doing so, we only insert $t/4$ of the time information and maximize the integrity as well as the authenticity of information in comparison with the final total time information $2t$. The specific process is shown in Fig. 3.

We set the original output feature map of the backbone f se as the multiple semantic feature map, which contains significant semantic information and limited spatial-temporal information and achieves a good judgement effect for short-time actions. We thus use the multiple spacetime feature map f^{st} and multiple semantic feature map f^{se} together as our outputs to recognize the input action.

3.3 Triplet attention

For the multiple spacetime-semantic feature map, the time and channel dimensions of the feature map have increased greatly, which inevitably results in redundancy of dimensional information and further affects action analysis. Therefore, it is wise to analyse which channels, frames, and key points are more important during the process of action recognition. In the field of object recognition, Woo et al. [44] effectively paid attention to different channels and spaces; Zhang et al. [48] used self-attention in generative adversarial networks (GAN) and achieved good results. Zhang et al. [50] represent an edge in a graph of a human skeleton by integrating its spatial neighbouring edges and its temporal neighbouring edges, and devise a graph edge convolutional neural network. Gao et al. [11] utilize a hierarchical attentional module with long short-term memory and multi-layer perceptrons to leverage both inter- and intra-frame attention to facilitate visual patterns. The feature maps after the multiple spacetime-semantic feature extraction are $f^{st}, f^{se} \in \mathbb{R}^{C \times T \times V}$. The triplet attention mechanism calculates the key point attention $\mu_n^{1 \times V \times V}$, channel attention $\mu_c^{C \times 1 \times 1}$, and temporal attention $\mu_t^{1 \times T \times 1}$ in turn, adds multiple attentions to f^{st}, f^{se}, and obtains the final output feature map f_{out}^{st}, f_{out}^{se}.

3.3.1 Keypoint attention

In the action modelling process, we add a learnable adjacency matrix n [45] to each spatial-temporal convolution layer to assign different importance values to different joints. By adding this matrix, not only can the weight importance

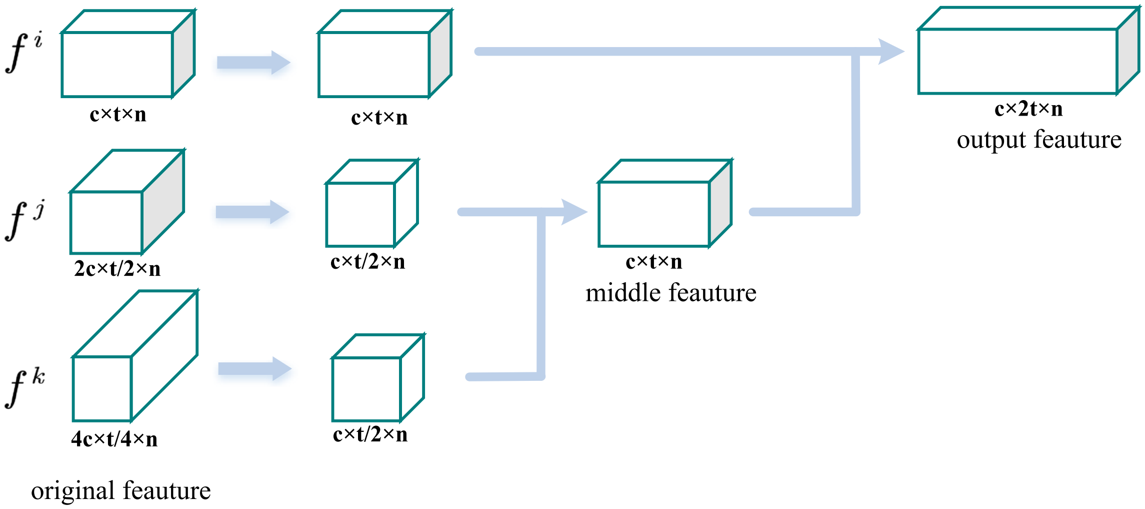


Fig. 3 Multiple spatial-temporal feature map construction process

and the scale node features be learned according to the edges of the space graph, but the edge contribution of the skeleton model can also be scaled for further improving the recognition performance.

3.3.2 Channel attention

In this paper, channel redundancy may occur because we use convolution and pooling layers to extract numerous semantic features for generation of a large number of channels, which will inevitably affect the judgement of networks with respect to action classification. To solve this, we use channel attention to distinguish the importance of different channels.

The feature map of each channel can be regarded as a feature detector, and the purpose of channel attention is to understand which channels truly function adequately [10, 48]. First, we use max-pooling and global average-pooling to aggregate spatial-temporal information and squeeze the time and keypoint dimensions of the input feature map to generate two feature maps $f_{ave}, f_{\max}$ that only retain channel information. Then, we forward these two feature maps to an attention distribution network (consisting of multiple-layer perceptron with one hidden layer) with shared parameters. The output feature vectors can be considered as the completed channel attention μ_c:

$$\begin{aligned}\mu_c &= \sigma\left(Mlp\left(f_{ave}\right) + Mlp\left(f_{\max}\right)\right)\\ &= \sigma\left(W_\alpha\left(W_\beta\left(f_{ave}\right)\right) + W_\alpha\left(W_\beta\left(f_{\max}\right)\right)\right),\end{aligned} \tag{5}$$

For the results of max-pooling and global average-pooling, the weights of the multiple-layer perceptron (MLP) W_α, W_β are shared, and the activation function follows W_β. σ denotes the sigmoid function.

3.3.3 Temporal attention

In this paper, we use a temporal attention mechanism [44] to distinguish the importance of different time series of an action which contribute differently for judging an action.

The time dimension in the graph feature map is the third dimension, and the keypoint dimension is the fourth dimension. These two dimensions correspond to the spatial dimensions of the ordinary convolution feature map. Key point attention has been determined by different weights of the edges in the graph structure, so we use a method to add attention to the time dimension alone. Different from channel attention, temporal attention focuses on choosing the frames that play a greater role in judging action after concatenation. To calculate temporal attention, we still use max-pooling and global average-pooling first for the channel dimension to aggregate channel information and obtain two feature maps that only retain time and key point features: F_{ave},$F_{\max}$. We then concatenate F_{ave},$F_{\max}$ and convolve it through a standard convolution layer. The convolution kernel is set to $t \times 1$ (consistent with TCN) to generate temporal attention μ_t:

$$\mu_t = \sigma\left(Conv^{t\times 1}\left(cat\left[F_{ave}; F_{\max}\right]\right)\right), \tag{6}$$

where σ denotes the sigmoid function and $Conv^{t\times 1}$ denotes a convolution operation with the filter size of $t \times 1$. The convolution kernel is set to $t \times 1$ because what we would like to do is to add attention to the frame, and the convolution kernel can be satisfied with the requirements for extracting attention features. Moreover, we have also attempted to use a larger convolution kernel, but the effect is not satisfactory. We believe that one possible reason for this is that the lack of activation function reduces the attention feature performance so that the abstraction level is also reduced.

3.3.4 Triplet attention

The output after adding the triplet attention mechanism is:

$$\begin{aligned}f_{out}^{st} &= \mu^{att1} f^{st} = \mu_t\left(\mu_c\left(f^{st}\right) \times f^{st}\right) \times \left(\mu_c\left(f^{st}\right) \times f^{st}\right)\\ f_{out}^{se} &= \mu^{att2} f^{se} = \mu_t\left(\mu_c\left(f^{se}\right) \times f^{se}\right) \times \left(\mu_c\left(f^{se}\right) \times f^{se}\right),\end{aligned} \tag{7}$$

where μ_c,μ_t respectively denote the channel attention and the temporal attention, and $f_{out}^{st}/f_{out}^{se}$ denote the feature map with the attention mechanism added. The specific process is shown in Fig. 4.

3.4 Fusion prediction

To comprehensively capture the features formed by the multiple spacetime-semantic network and the triplet attention mechanism, we combine the multiple spacetime-semantic network and triplet attention mechanism as the response of AM-GCN. The final output $f_{out} = \left\{f_{out}^{st}, f_{out}^{se}\right\}$ is defined based on (2):

$$\begin{aligned}f_{out}^{st} &= f^{st}\mu^{att1} = \sum_{k=1}^{k}\mu_n \cdot \left(\Lambda^{-\frac{1}{2}} A \Lambda^{-\frac{1}{2}} f_{in}\right) W_1 \mu^{att1}\\ f_{out}^{se} &= f^{se}\mu^{att2} = \sum_{k=1}^{k}\mu_n \cdot \left(\Lambda^{-\frac{1}{2}} A \Lambda^{-\frac{1}{2}} f_{in}\right) W_2 \mu^{att2},\end{aligned} \tag{8}$$

where f_{out}^{st},f_{out}^{se} respectively denote the multiple spacetime and multiple semantic output. f_{out} is the prediction result after the decision.

Because the feature map f_{out}^{st}, f_{out}^{se} will obtain two kinds of prediction results, the final prediction can only provide one action label, so we adopt a fusion prediction method to select the final action label automatically. The prediction method of most GCN-based models is processing the output feature map by a fully connected layer, then inputting it to the softmax function for probability distribution and selecting the class with the highest probability as the final action label. We draw inspiration from the single softmax

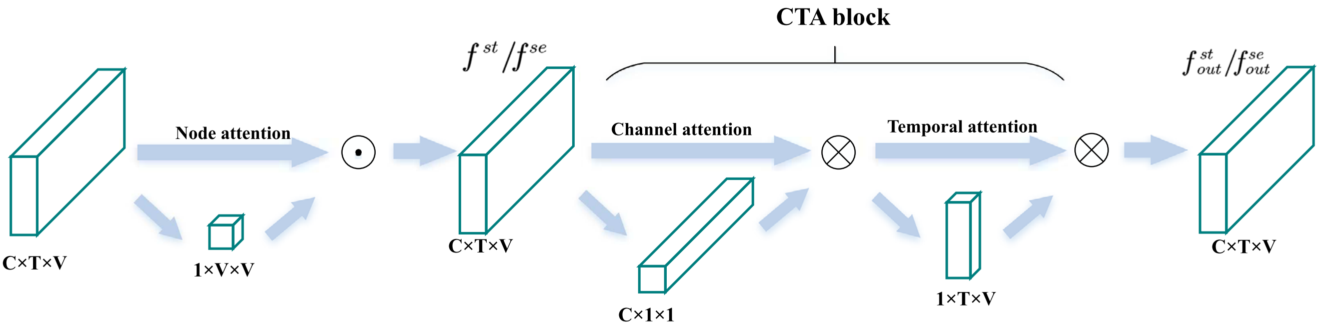


Fig. 4 Process of adding triplet attention mechanism to output feature map

classification and extend it. For multiple feature maps, each feature map is input to softmax for probability distribution, then we compare the two highest probabilities and select the one with the higher probability as the final action label:

$$\begin{aligned} class\ prob &= \max\left(class^{st}, class^{se}\right) \\ class &= \frac{\exp\left(f\right)}{\sum_{i=1}^{N}\exp\left(f\right)}. \end{aligned} \tag{9}$$

Our model calculates the loss function for two original prediction results at the same time. Here, we count and divide the actions with different complexity in the dataset, assigning weights to the multiple-semantic and multi-spacetime losses. After a few experiments on loss weights, we finally set up $loss = 0.65loss^{st} + 0.35loss^{se}$ as the final loss function. The calculated loss function is fed back to the network and the smaller value of the loss function is continuously sought to modify the action of the network recognition results to obtain the optimal model.

Commonly used multi-scale fusion strategies basically rely on the calculated confidence size or non maximum suppression (NMS) to make decisions, which is not applicable for video sequence-based convolution, since there is no confidence parameter in action recognition. To solve this problem, we design this strategy of relying solely on the results of softmax for fusion without introducing an additional confidence parameter. Experiments show that this strategy has achieved good results.

The proposed AM-GCN framework is composed of several layers of spatial-temporal graph convolution operators. The front, middle and back layers respectively include increasing channels with a residual link between each graph convolution layer. The original output of the spatial-temporal GCN is set to semantic output, and the feature map after up-sampling and concatenation is set to spacetime output. The keypoint attention is added after every convolution layer, but the channel and temporal attention are added at the end of the complete structure. We present dropout for regularization, and half of the neurons are closed to avoid overfitting. After that, global pooling was performed on the resulting tensor to obtain the corresponding dimension (semantic or spacetime) feature vector for each sequence.

The specific algorithm flow of the proposed triplet attention multiple spacetime-semantic graph convolutional network for skeleton-based action recognition is shown in Algorithm 1.

Algorithm 1 The algorithm flow of the AM-GCN network.

```
Initialize the network parameters $\theta$ and learning rate $\eta$;
In each training iteration:
Sample m video examples $\left\{x^1, x^2, x^3, ..., x^m\right\}$ from database;
Extract spacetime-semantic features $\left\{f_{st}^1, f_{st}^2, f_{st}^3, ..., f_{st}^m\right\}$ ,$\left\{f_{se}^1, f_{se}^2, f_{se}^3, ..., f_{se}^m\right\}$ of m videos;
Calculate the triplet attention $\mu$, and attach $\mu$ to $f_{st}$ and $f_{se}$;
Obtain the probability of action prediction: $p_i = softmax\left(f^i\right)$;
Use fusion decision to automatically choose the better prediction: $p_i$;
Update AM-GCN network parameters to minimize:
    – $\tilde{V} = \frac{1}{m}\sum_{i=1}^{m} -q_i \log\left(p_i\right)$ ,
    – $\theta^{(k+1)} \leftarrow \theta - \eta\nabla\tilde{V}\left(\theta\right)$ .
Until the AM-GCN network converges, the model is obtained.
```

4 Experiment

4.1 Implementation details

The skeleton information is provided by the NTU-RGBD dataset or extracted by Openpose. We use Pytorch 0.4.1 and train the model on 4 GTX-1080Ti GPUs. We use the stochastic gradient descent (SGD) algorithm to optimize our model, for which the initial learning rate is 0.1, decaying as training deepens. Cross-entropy loss is selected as the loss function to backpropagate gradients, and the weight decay is set to 0.0001. Neither the multiple spacetime-semantic feature nor the triplet attention adds much computing power consumption in comparison with other skeleton-based action recognition models. Training on the NTU-RGBD dataset requires 12 hours (based on ST-GCN), 72 hours (based on AS-GCN) and 20 hours (based on 2S-AGCN) in our synchronized 4-GPU implementation. During the training process, we found that using the triplet attention will greatly accelerate the initial convergence speed of models. We believe that the reason for this is that the triplet attention will help the network to better focus on keypoints, channels and frames which contain more information, making the model converge faster.

4.2 Experimental results and comparison

To verify the performance of the proposed model, we perform a large number of experiments on two skeleton-based action recognition datasets: NTU-RGBD [33] and Kinetics [17]. These two datasets have been widely used in previous skeleton-based action recognition works. At the same time, since the NTU-RGBD dataset is smaller than the Kinetics dataset, we perform exhaustive ablation studies on the NTU-RGBD dataset to verify the effectiveness of the proposed model components.

NTU-RGBD [33]: This dataset contains 60 different human action categories, which are divided into three major categories: daily action, mutual action, and health-related action. A total of 56,880 action samples were performed by 40 different subjects. Each action sample contains RGB video, depth map sequence, 3D skeleton data, and infrared video captured by three Microsoft Kinect v2 cameras simultaneously. The 3D skeleton data we focus on consist of 3D positions of 25 body joints per frame. There are two evaluation protocols for this data set: Cross Subject (CS) and Cross View (CV). According to the Cross Subject protocol, the actions performed by 20 subjects constitute the training set, and the remaining actions performed by the remaining 20 subjects are used for testing. For the Cross View protocol, the samples captured by the first two cameras are used for training and the rest are used for testing.

Kinetics [17]: This dataset contains 400 different human action categories, from daily activities to sports scenes and complex interactive actions. There are a total of 240,000 video sequences. Since this dataset only provides RGB video, in order to obtain the keypoint position, we adjust the video to a resolution of 340 × 256 and use the Openpose [2] toolbox to calculate the positions of 18 keypoints per frame and generate 2D pixel coordinates (x, y) and confidence c. Yan et al. [45] used (X, Y, C) vectors to denote each joint, and our model also uses this practice to generate the results. For multiple player situations, we select the two individuals with the highest average key confidence in each sequence. Each video includes 300 frames after clipping. Top-1 and Top-5 are used to identify accuracy.

Because the proposed triplet attention multiple spacetime-semantic graph convolutional network shares weights on different nodes, it is important to keep the proportions of input data on different nodes consistent. In the experiment, the data are first normalized because dimensions of input video are not suitable for the graph convolutional network.

The performance of the proposed AM-GCN is compared in this section with several state-of-the-art methods on the standard datasets NTU-RGBD and Kinetics.

4.2.1 NTU-RGBD dataset

The quantitative results of the comparison test are shown in Table 1. Comparison methods include [6, 25, 33] based on RNN, [18, 20, 26] based on CNN, and [22, 23, 35, 39, 45, 46] based on GCN.

The proposed method is able to improve accuracy for different backbones by only minimally increasing calculation. Our method can outperform ST-GCN by 3.2% and 4.3%, AS-GCN [23] by 1.5% and 1.1%, and 2S-AGCN [35] by 0.9% and 0.6%.

4.2.2 Kinetics dataset

The comparison results of some representative skeleton-based action recognition algorithms [9, 20, 23, 33, 35, 45, 46] on the Kinetics dataset are shown in Table 2. The

Table 1 Comparison results of different algorithms on the NTU-RGBD dataset

Algorithm	Cross subject	Cross view
HB-RNN [6] (2015)	59.1%	64.0%
Deep-LSTM [33] (2016)	60.7%	67.3%
ST-LSTM [25] (2016)	62.9%	70.3%
TCN [20] (2017)	74.3%	83.1%
C-CNN+MTLN [18] (2017)	79.6%	84.3%
Visualization CNN [26] (2017)	76.0%	82.6%
ST-GCN [45] (2018)	81.5%	88.3%
DPRL [39] (2018)	83.5%	89.8%
HCN [22] (2018)	86.5%	91.1%
AS-GCN [23] (2019)	86.8%	94.2%
2S-AGCN [35] (2019)	88.5%	95.1%
C-GCN [46] (2020)	**90.3%**	**96.4%**
Ours	84.7%	92.6%
Ours+	88.3%	95.3%
Ours++	89.4%	95.7%

"+" denotes using AS-GCN as backbone. "++" denotes using 2S-AGCN as backbone. Bold and underlined fonts denote the first and our best performances, respectively

Table 2 Comparison of different algorithms on the Kinetics dataset

Algorithm	Top 1	Top 5
Feature encoding [9] (2015)	14.9%	25.8%
Deep LSTM [33] (2016)	16.4%	35.3%
TCN [20] (2017)	20.3%	40.0%
ST-GCN [45] (2018)	30.7%	52.8%
AS-GCN [23] (2019)	34.8%	56.5%
2S-AGCN [35] (2019)	36.1%	58.7%
C-GCN [46] (2020)	**37.5%**	**60.4%**
Ours	31.4%	54.3%
Ours++	<u>36.3%</u>	<u>59.1%</u>

++ denotes using 2S-AGCN as backbone. Bold and underlined fonts denote the best and second-best performances, respectively

best performance of the proposed model on the Kinetics dataset can reach 36.3%. Since the Kinetics dataset contains many videos which include an occluded skeleton, the improvement of the performance on the Kinetics dataset is not as great as for the NTU-RGBD dataset. Figure 5 demonstrates the visualization of the learned attention maps. It shows that our method can effectively improve the representation of action recognition.

4.3 Ablation study

To test the validity of the multiple spacetime-semantic feature extraction and triplet attention mechanism, an ablation study is performed by removing the proposed component from the entire network on the NTU-RGBD dataset. We use Cross Subject as the experiment evaluation standard.

Table 3 Ablation study with ST-GCN. Bold fonts denote the best performances

Ablation study	Cross subject
ST-GCN	81.5%
ST-GCN+spacetime-semantic scale	84.3%
ST-GCN+attention	83.1%
ST-GCN+spatial-temporal-scale+attention	**84.7%**
Ours	**84.7%**

Multiple spacetime-semantic feature extraction: To verify the performance of multiple spacetime-semantic feature extraction, we compare the multiple spacetime-semantic feature map with the original feature map. As seen from Tables 3 and 4, the accuracy of the model is improved to 84.3% and 88.9% after using the multiple spacetime-semantic feature map, proving the effectiveness of the multiple spacetime-semantic feature. The improvement enabled by the multiple spacetime-semantic feature for 2S-AGCN is less than that for ST-GCN. We hypothesize that this is because the 2S-AGCN is sufficiently powerful and the accuracy is high, and therefore the effect of the multiple spacetime-semantic feature is limited.

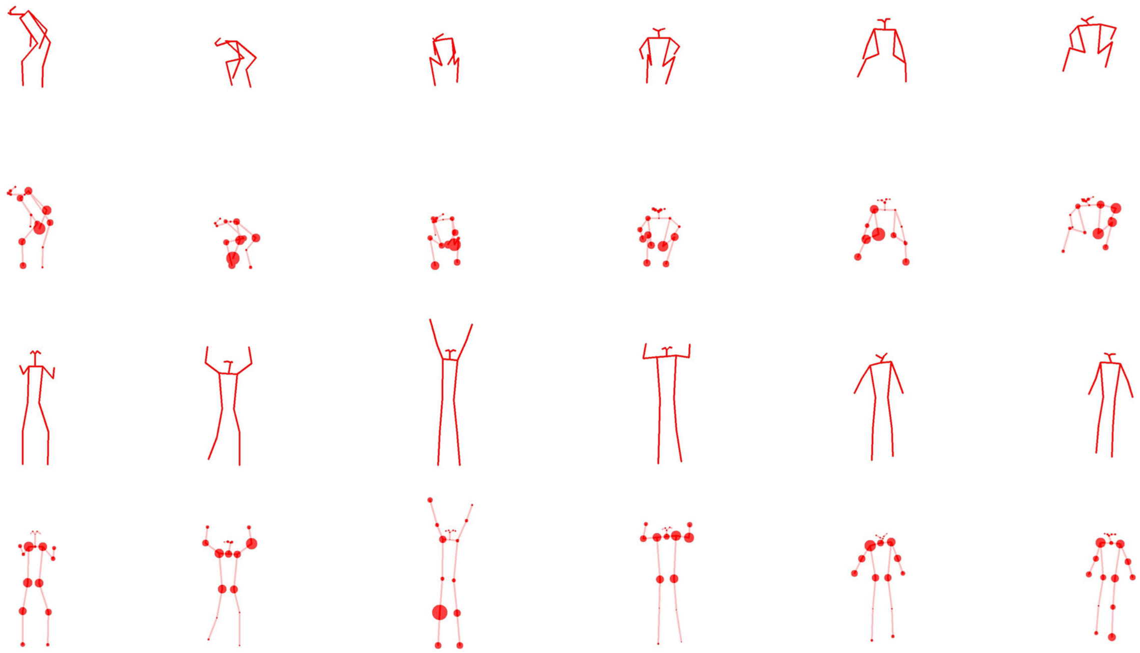


Fig. 5 Visualization of the learned attention maps. The two actions are "Situp" and "Clean and Jerk". The first and third rows show the skeleton map, while the second and fourth rows show the map with attention. The size of the circle represents the importance of the corresponding joint

Table 4 Ablation study with 2S-AGCN. Bold fonts denote the best performances

Ablation study	Cross subject
2S-AGCN	88.5%
2S-AGCN+spacetime-semantic scale	88.9%
2S-AGCN+attention	89.2%
2S-AGCN+spacetime-semantic scale+attention	**89.4%**
Ours	**89.4%**

Triplet attention mechanism: We also compare the effectiveness of the triplet attention mechanism with the original model in this section. It can be found from Tables 3 and 4 that using the triplet attention mechanism improves the accuracy of the original model to 83.1% and 89.2%, which proves the effectiveness of the triplet attention mechanism. We find that during the experiment, the triplet attention mechanism will greatly accelerate the initial convergence speed and reduce the network fitting time: this proves that the triplet attention will help the model focus on important keypoints, channels and frames, improving the interpretability of the model.

We separately tested the contributions of three sub-attention mechanisms, i.e., keypoint attention, channel attention, and temporal attention, based on the ST-GCN in Table 5. The results show that all of the three sub-attention mechanisms significantly improve the performance. In addition, deleting any one of the sub-attentions will dramatically reduce the performance. After combining all sub-attention mechanisms together, we can utilize all information of the attention mechanism and achieve a superior effect.

4.4 Experiments on long-time action

The proposed model is effective in dealing with long-time action for the reason that the multiple spacetime network is used to extract long-time action features. To prove this, we select ten different long-time actions for an extra experiment, each type of which includes 948 action samples on the NTU-RGBD dataset. The experiment is performed on both the Cross Subject (CS) and Cross View (CV) evaluation criteria. The results of the comparison test are shown in Table 6. It can be determined that the improvement of the proposed method is not obvious for short-time actions (+1%-2%) such as pat on the back, put palms together, etc., but it is much higher for long-time actions (+5%-10%) such as staggering, check time (from watch), touch other persons pocket, and walking apart. It can be verified that ST-GCN exhibits a good effect for most fundamental action types, including most short-time actions, but does not perform well for some long-time actions. The reason for this is that, compared with ST-GCN, the proposed multiple spacetime-semantic feature network is effective in recognizing long-time actions.

Table 5 Ablation study on contributions of sub-attentions

Ablation study	Cross subject
ST-GCN	81.5%
STGCN-KC	82.3%
STGCN-KT	82.6%
STGCN-CT	82.7%
ST-GCN+triplet attention	**83.1%**

STGCN-KC denotes using only the keypoint and channel attentions based on STGCN. STGCN-KT means using only the keypoint and temporal attentions. STGCN-CT denotes using only the channel and temporal attentions. Bold fonts denote the best performances

Table 6 Comparison of specific actions using ST-GCN and the proposed model

Specific action	ST-GCN		Ours	
	xsub	xview	xsub	xview
Tear up paper	39.3%	46.5%	**49.6%**	**77.5%**
Check time (from watch)	72.1%	83.9%	**81.5%**	**94.6%**
Put palms together	85.1%	95.6%	**88.4%**	**97.2%**
Cross hands in front	80.4%	94.0%	**81.5%**	**95.9%**
Staggering	65.2%	87.3%	**70.7%**	**93.4%**
Fan self	82.6%	89.2%	**82.2%**	**94.6%**
Pat on back	91.3%	98.1%	**95.0%**	**99.4%**
Giving object	93.4%	89.6%	**95.2%**	**98.7%**
Touch other persons pocket	83.7%	92.1%	**87.7%**	**97.4%**
Walking apart	99.3%	95.9%	**100%**	**100%**

Top: long-time actions; bottom: short-time actions. Since the dataset includes relatively few categories for long-term actions, the multi-temporal feature improves ST-GCN by approximately 3%-4%. The table shows the improvement for long-term actions. Bold fonts denote the best performances

5 Conclusion

In this paper, we propose a triplet attention multiple spacetime-semantic graph convolutional network (AM-GCN) for skeleton-based action recognition, which can not only capture features of actions with long and short completion times, but also combine the output of the two feature maps and select the best one according to the fusion decision. In addition, we implement three attention mechanisms of channel, temporal, and keypoint in the graph convolution network and combine them as a triplet attention mechanism. The experimentation with the two standard datasets NTU-RGBD and Kinetics shows that the classification accuracy of our model achieves significant improvement as compared with the previous methods.

Acknowledgements This work is supported by the Natural Science Foundation of Jiangsu Province (BK20180640), the National Natural Science Foundation of China (61902404, 512918914, 51734009, 61771417, 61873246), and the State Key Research Development Program (2016YFC0801403).

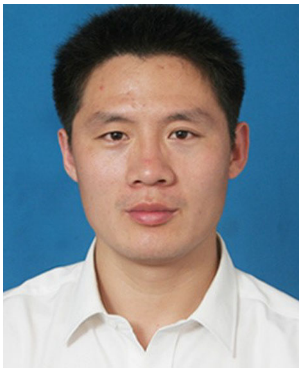
References

1. Cao C, Lan C, Zhang Y, Zeng W, Lu H, Zhang Y (2018) Skeleton-based action recognition with gated convolutional neural networks. IEEE Trans Circuits Syst Video Technol 29(11):3247–3257
2. Cao Z, Simon T, Wei SE, Sheikh Y (2017) Realtime multi-person 2d pose estimation using part affinity fields. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 7291–7299
3. Carreira J, Zisserman A (2017) Quo vadis, action recognition? a new model and the kinetics dataset. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 6299–6308
4. Chen Y, Ma G, Yuan C, Li B, Zhang H, Wang F, Hu W (2020) Graph convolutional network with structure pooling and joint-wise channel attention for action recognition. Pattern Recognit, 103
5. Ding C, Liu K, Cheng F, Belyaev E (2021) Spatio-temporal attention on manifold space for 3d human action recognition. Appl Intell 51(5):560–570
6. 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
7. Duvenaud DK, Maclaurin D, Iparraguirre J, Bombarell R, Hirzel T, Aspuru-Guzik A, Adams RP (2015) Convolutional networks on graphs for learning molecular fingerprints. In: Conference and workshop on neural information processing systems, pp 2224–2232
8. Feng Y, Li K, Gao Y, Qiu J (2020) Hierarchical graph attention networks for semi-supervised node classification. Appl Intell 50(3):1–17
9. 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
10. Fu J, Liu J, Tian H, Li Y, Bao Y, Fang Z, Lu H (2019) Dual attention network for scene segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 3146–3154
11. Gao P, Zhang Q, Wang F, Xiao L, Zhang Y (2020) Learning reinforced attentional representation for end-to-end visual tracking. Inf Sci 517:52–67
12. Gaur U, Zhu Y, Song B, Roy-Chowdhury A (2011) A “string of feature graphs” model for recognition of complex activities in natural videos. In: Proceedings of the IEEE 15th international conference on computer vision, pp 2595–2602
13. Hamilton W, Ying Z, Leskovec J (2017) Inductive representation learning on large graphs. In: Conference and workshop on neural information processing systems, pp 1024–1034
14. Hu J, Shen L, Sun G (2018) Squeeze-and-excitation networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 7132–7141
15. Hussein ME, Torki M, Gowayyed MA, El-Saban M (2013) Human action recognition using a temporal hierarchy of covariance descriptors on 3d locations. In: International joint conference on artificial intelligence
16. i R, Tapaswi M, Liao R, Jia J, Urtasun R, Fidler S (2017) Situation recognition with graph neural networks. In: IEEE International conference on computer vision, pp 4183–4192
17. Kay W, Carreira J, Simonyan K, Zhang B, Hillier C, Vijayanarasimhan S, Viola F, Green T, Back T, Natsev P et al (2017) The kinetics human action video dataset. arXiv:1705.06950
18. Ke Q, Bennamoun M, An S, Sohel F, Boussaid F (2017) A new representation of skeleton sequences for 3d action recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 3288–3297
19. Ke Q, Bennamoun M, An S, Sohel F, Boussaid F (2018) Learning clip representations for skeleton-based 3d action recognition. IEEE Trans Image Process 27(6):2842–2855
20. Kim TS, Reiter A (2017) Interpretable 3d human action analysis with temporal convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition Workshop, pp 1623–1631
21. Kipf TN, Welling M (2017) Semi-supervised classification with graph convolutional networks. In: International conference on learning representations, pp 1–14
22. Li C, Zhong Q, Xie D, Pu S (2018) Co-occurrence feature learning from skeleton data for action recognition and detection with hierarchical aggregation. In: International joint conferences on artificial intelligence, pp 786–792
23. 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
24. Lin TY, Dollár P., Girshick R, He K, Hariharan B, Belongie S (2017) Feature pyramid networks for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2117–2125
25. 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, pp 816–833
26. Liu M, Liu H, Chen C (2017) Enhanced skeleton visualization for view invariant human action recognition. Pattern Recognit 68(8):346–362
27. Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu CY, Berg AC (2016) Ssd: Single shot multibox detector. In: European conference on computer vision, pp 21–37
28. Lu L, Yu R, Di H, Zhang L, Lu Y (2020) Gaim: Graph attention based interaction model for collective activity recognition. IEEE Trans Multimedia 22(2):524–539
29. Monti F, Boscaini D, Masci J, Rodola E, Svoboda J, Bronstein MM (2017) Geometric deep learning on graphs and manifolds using mixture model cnns. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 5115–5124
30. Niepert M, Ahmed M, Kutzkov K (2016) Learning convolutional neural networks for graphs. In: Proceedings of the 33rd international conference on machine learning and data mining, pp 2014–2023
31. Pérez-Hernández F, Tabik S, Lamas A, Olmos R, Herrera F (2020) Object detection binary classifiers methodology based on deep learning to identify small objects handled similarly: Application in video surveillance. Knowl-Based Syst 194:100590
32. Qi S, Wang W, Jia B, Shen J, Zhu SC (2018) Learning human-object interactions by graph parsing neural networks. In: European conference on computer vision, pp 401–417
33. Shahroudy A, Liu J, Ng TT, Wang G (2016) Ntu rgb+ d: a large scale dataset for 3d human activity analysis. In: Proceedings of

the IEEE conference on computer vision and pattern recognition, pp 1010–1019

34. Shahroudy A, Ng TT, Gong Y, Wang G (2018) Deep multimodal feature analysis for action recognition in rgb+d videos. IEEE Trans Pattern Anal Mach Intell 40(5):1045–1058
35. 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 12026–12035
36. Shi L, Zhang Y, Cheng J, Lu H (2020) Skeleton-based action recognition with multi-stream adaptive graph convolutional networks. IEEE Trans Image Process 29:9532–9545
37. Si C, Chen W, Wang W, Wang L, Tan T (2019) An attention enhanced graph convolutional lstm network for skeleton-based action recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1227–1236
38. Song S, Lan C, Xing J, Zeng W, Liu J (2017) An end-to-end spatio-temporal attention model for human action recognition from skeleton data. In: Thirty-first AAAI conference on artificial intelligence, pp 4263–4270
39. 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
40. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN (2017) Kaiser, L.u., Polosukhin, I.: Attention is all you need. In: Conference and workshop on neural information processing systems, pp 5998–6008
41. 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
42. Wang J, Liu Z, Wu Y, Yuan J (2012) Mining actionlet ensemble for action recognition with depth cameras. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1290–1297
43. Wang Y, Zhou L, Qiao Y (2018) Temporal hallucinating for action recognition with few still images. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 5314–5322
44. Woo S, Park J, Lee JY, So Kweon I (2018) Cbam: Convolutional block attention module. In: European conference on computer vision, pp 3–19
45. Yan S, Xiong Y, Lin D (2018) Spatial temporal graph convolutional networks for skeleton-based action recognition. In: Thirty-second AAAI conference on artificial intelligence, pp 7444–7452
46. Yang D, Li MM, Fu H, Fan J, Leung H (2020) Centrality graph convolutional networks for skeleton-based action recognition. arXiv:2003.03007
47. Yang H, Gu Y, Zhu J, Hu K, Zhang X (2020) Pgcn-tca: Pseudo graph convolutional network with temporal and channel-wise attention for skeleton-based action recognition. IEEE Access 8(7):10040–10047
48. Zhang H, Goodfellow I, Metaxas D, Odena A (2018) Self-attention generative adversarial networks. arXiv:1805.08318
49. Zhang S, Yang Y, Xiao J, Liu X, Yang Y, Xie D, Zhuang Y (2018) Fusing geometric features for skeleton-based action recognition using multilayer lstm networks. IEEE Trans Multimed 20(9):2330–2343
50. Zhang X, Xu C, Tian X, Tao D (2020) Graph edge convolutional neural networks for skeleton-based action recognition. IEEE Trans Neural Netw Learn Syst 31(8):3047–3060

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



Yanjing Sun is a professor in School of Information and Control Engineering, China University of Mining and Technology since July 2012. He received the Ph.D. degree in Information and Communication Engineering from China University of Mining and Technology in 2008. He is also a vice director of Coal Mine Electrical Engineering and Automation Laboratory in Jiangsu Province. His current research interests include IBFD communication, embedded real-time system, computer vision, wireless sensor networks, cyber-physical system and so on.

Han Huang received his B.S. degrees in measurement and control technology and instrument from Chongqing University (CQU), China in 2018. He is currently pursuing his master's degree at School of Information and Control Engineering, China University of Mining and Technology. His current research interests are mainly focused on computer vision, action recognition, and human pose estimation.

Xiao Yun received her Ph.D. degrees in Control Science and Engineering from Shanghai Jiao Tong University, China. She is currently an assiociate professor at School of Information and Control Engineering, China University of Mining and Technology. Her current research interests are mainly focused on computer vision, visual tracking, re-identification, and information fusion.

Bin Yang received his B.S. degree in electronic science and technology from Changshu Institute of Technology in 2018. He is a postgraduate in China University of Mining Technology now, and his current research interests include deep-learning-based human action detection and recognition, human pose estimation, and related applications.

Kaiwen Dong received his B.S. degrees in Nanjing Forestry University, China in 2018. He is currently pursuing his Doctor's degree at School of information and Control Engineering, China University of Mining and Technology. His Current research interests are mainly focused on computer vision, object detection and person re-identification.