Spatial Temporal Attention Graph Convolutional Networks with Mechanics-Stream for Skeleton-Based Action Recognition

Katsutoshi Shiraki(✉), Tsubasa Hirakawa, Takayoshi Yamashita,
and Hironobu Fujiyoshi

Chubu University, 1200 Matsumotocho, Kasugai-shi, Aichi, Japan
{siraki,hirakawa}@mprg.cs.chubu.ac.jp,
{yamashita,fujiyoshi}@isc.chubu.ac.jp

Abstract. The static relationship between joints and the dynamic importance of joints leads to high accuracy in skeletal action recognition. Nevertheless, existing methods define the graph structure beforehand by skeletal patterns, so they cannot capture features considering the relationship between joints specific to actions. Moreover, the importance of joints is expected to be different for each action. We propose spatial-temporal attention graph convolutional networks (STA-GCN). It acquires an attention edge that represents a static relationship between joints for each action and an attention node that represents the dynamic importance of joints for each time. STA-GCN is the first method to consider joint importance and relationship at the same time. The proposed method consists of multiple networks, that reflect the difference of spatial (coordinates) and temporal (velocity and acceleration) characteristics as mechanics-stream. We aggregate these network predictions as final result. We show the potential that the attention edge and node can be easily applied to existing methods and improve the performance. Experimental results with NTU-RGB+D and NTU-RGB+D120 demonstrate that it is possible to obtain a attention edge and node specific to the action that can explain behavior and achieves state-of-the-art performances.

1 Introduction

Human action recognition has been actively proposed and widely applied to surveillance systems and sports analysis. The skeleton based method is robust to environmental and viewpoint changes. These methods convert the skeletal data into a grid structure or sequence and extract features using a convolutional neural network (CNN) or a recurrent neural network (RNN) [1–13]. Grid structures or sequences cannot represent human skeletons completely. Meanwhile, graph structure can represent the skeleton naturally.

A graph convolutional network (GCN) have been used for skeletal action recognition that inputs skeletons as a graph structure [14–20]. A GCN applies

H. Ishikawa et al. (Eds.): ACCV 2020, LNCS 12626, pp. 341–357, 2021.
https://doi.org/10.1007/978-3-030-69541-5_21

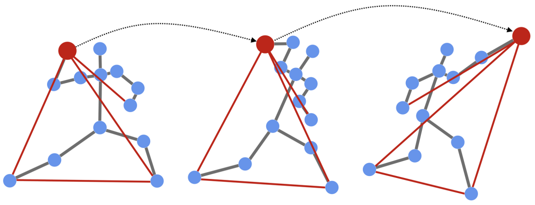


Fig. 1. Important relationships between joints and important joints in throwing.

convolutions to a graph structure and extract features. By using a GCN, features can be acquired that consider relationships between joints, so that complex actions can be recognized. A typical GCN-based method is spatial-temporal GCN (ST-GCN) [14]. ST-GCN achieves high performance by representing skeletons using two graph structures: spatial and temporal graphs. The graph connection in ST-GCN is fixed by human skeleton patterns. However, the important relationships between joints are expected to change depending on each action. For example, when we throw an object with the right arm while using the left leg as an axis foot, the relationship between the left leg and the right arm is important (Fig. 1). Therefore, fixing connection patterns in advance cannot acquire features considering joint relationships specific to action.

To solve this problem, there are methods to automatically obtain graph connection patterns from feature maps [17,18]. These methods acquires the connection pattern from the feature map representing the optimal relationship between joints for each action. However, these methods focus only on the relationships between joints. The important joints differ for each action, and the importance may change over the action frames. Considering the importance of different joints for each action and each frame could improve the recognition performance.

In this paper, we propose spatial-temporal attention graph convolutional networks (STA-GCN) that consider the important joint relationships and the importance of joints that differ for each action. An *attention edge* represents important relationship between joints statically. By convolving the feature map with the attention edge, features considering the important relationships between joints can be acquired. An *attention node* represents the dynamic importance of joints for each frame as a two-dimensional map. The attention edge and node are acquired during forward propagation. The attention node emphasizes important joints in a feature map by attention mechanism. Consequently, our method emphasizes the spatial and temporal features at the same time. We call the combined graph as a *spatial-temporal attention graph* (attention graph). The attention graph enables a network to learn the important joints and relationships between joints. Attention edge and node can be acquired by adding a module called *attetntion branch* to the network. Therefore, it can be applied into existing skeleton-based methods.

In addition, we introduce a multi-modal learning. We use six modals: coordinates, velocity, and acceleration for joint and bone. Existing methods have revealed multi-modal learning is effective [18,20]. However, few methods have examined the effects of modal combinations. Coordinates indicate spatial position, and velocity and acceleration indicate the amount of temporal movement, so the characteristics of each modal are different. Also, since the scales of the coordinates, velocity, and acceleration are different, training on the same network may be difficult. Therefore, we propose a *mechanics-stream structure* based on the spatial-temporal mechanical characteristics and value scale of each modal. Experimental results show the effect of modal combinations on accuracy.

2 Related Work

Skeleton-Based Action Recognition. Skeleton-based action recognition can be divided into two methods: handcrafted feature based and deep learning based. The handcrafted-feature based methods use features designed on the basis of human knowledge [21–23]. The deep-learning based methods can acquire features automatically, many methods using CNNs and RNNs have been proposed due to the recent advances in deep learning [1–13]. The CNN-based methods [1–6] manually convert skeletons into a grid structure and use it as an input. The RNN-based methods [7–13] input skeletons as a sequence and extract features representing temporal relationship between consecutive frames. CNN- and RNN-based methods achieve high recognition accuracy. However, since CNN and RNN represent the input data as a grid structure or sequence, they cannot completely represent human skeletons. To solve this problem, methods representing skeletons as a graph structure have been proposed and have achieved higher accuracy [14–20]. When the skeletons are represented as a graph structure, nodes correspond to joint coordinates, and edges correspond to relationships between joints. The efficient representation between joints can recognize complex actions.

Graph Convolutional Networks. GCN is a CNN that inputs a graph structure [24–26]. Kipf and Welling [26] performed graph convolution in the frequency domain using the graph Fourier transform. GCN are used in many fields because the graph structure can effectively represent various data [27–34]. In [27] and [28], the molecular structure was represented as a graph, and a GCN was applied to molecular classification. In [31] and [32], a GCN was applied to video summarization by connecting objects with high spatial and temporal correlation in the video with edges.

Visual Explanation for Deep Learning. Research into understanding the basis of cognitive judgment in deep learning is being actively pursued and visual explanation using attention maps are widely investigated [35–38]. Visual explanation methods can be divided into two types: bottom-up and top-down. In the bottom-up methods [37,38], an attention map is obtained by using gradient, noise, and class information. Therefore, backpropagation must be performed to obtain the attention map. Since it can be applied to various networks, it is widely

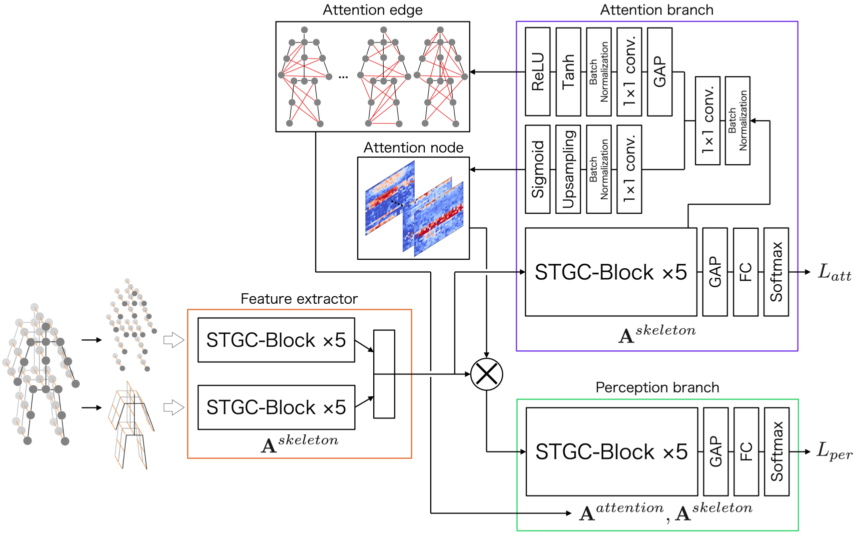


Fig. 2. Spatial-temporal attention graph convolutional networks.

used as a visual explanation method for deep learning. The top-down methods have been studied more actively because they can acquire the attention map during forward propagation [35,36]. ABN [35] is a top-down method that acquires an attention map during forward propagation and applies it to the attention mechanism. ABN has an advantage that the network can be learned while gazing at important areas in image recognition, and realized high recognition performance and analysis of the network decision simultaneously.

3 Proposed Method

We propose a spatial-temporal attention graph convolutional networks (STA-GCN) that considers the static relationships between joints and the dynamic importance of joints. Figure 2 shows the proposed network structure. STA-GCN consists of three modules: feature extractor, attention branch, and perception branch. The feature extractor extracts features of two input modals separately using multiple graph convolutional layers and then connects the feature maps. The attention branch generates an attention edge indicating the static relationship between joints and an attention node indicating a dynamic importance of joints for each frame. The perception branch outputs the final class probabilities by using the feature map obtained from the feature extractor, the attention node, a skeleton pattern, and attention edge.

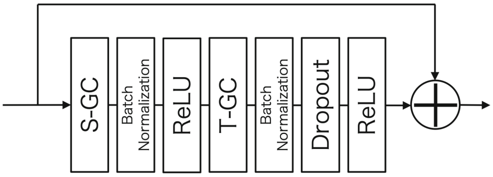


Fig. 3. Spatial-temporal graph convolutional block (STGC-Block).

3.1 Spatial-Temporal Graph Convolutional Block

The Spatial-Temporal Graph Convolutional Block (STGC-Block) shown in Fig. 2. Spatial Graph Convolution (S-GC) and Temporal Graph Convolution (T-GC) are graph convolutions for spatial and temporal graphs, respectively. A spatial graph connects joints in the same frame, and a temporal graph connects the same joints in adjacent frames. After each graph convolution layer, a batch normalization layer and a ReLU are arranged. A dropout is applied after T-GC, and each STGC-Block has a skip connection (Fig. 3).

Here, let $\mathbf{X}_{in} \in \mathbb{R}^{V \times C}$ be a feature input to a graph convolution with node V and dimension C and $\mathbf{A}^{skel} \in \mathbb{R}^{V \times V}$ be a skeleton pattern adjacency matrix, respectively. By using weight matrix $\mathbf{W}^{skel} \in \mathbb{R}^{V \times F}$ with the output dimension F, the graph convolution of $\mathbf{X}_{in}$ is defined by

$$\mathbf{X}_{out}^{skel} = \sum_{q}^{Q} \mathbf{M}_q^{skel} \circ \hat{\mathbf{A}}_q^{skel} \mathbf{X}_{in} \mathbf{W}_q^{skel}, \tag{1}$$

where $\mathbf{M}^{skel} \in \mathbb{R}^{V \times V}$ is a learning weight matrix for capturing the importance of edges. Q is an optional parameter that indicates the number of hops in the graph structure and can be connected to joints that are hops away. In the case of 1-hop, the graph structure becomes a human skeletal pattern because it connects joints that are one distance away. By increasing the number of hops, we can capture more global features. $\mathbf{M}^{skel}$, $\hat{\mathbf{A}}^{skel}$, and $\mathbf{W}^{skel}$ are defined separately for each Q. $\mathbf{A}_q^{skel} = \mathbf{A}_q + \mathbf{I}$ is calculated by the sum of $\mathbf{A}_q \in \mathbb{R}^{V \times V}$, which indicates the connection relationship with the adjacent node, and the identity matrix $\mathbf{I} \in \mathbb{R}^{V \times V}$, which indicates the loop structure. When graph convolution is performed, the normalized adjacency matrix of $\hat{\mathbf{A}}_q^{skel} = \mathbf{\Lambda}_q^{-\frac{1}{2}} \mathbf{A}_q^{skel} \mathbf{\Lambda}_q^{-\frac{1}{2}}$ is used, where $\mathbf{\Lambda} \in \mathbb{R}^{V \times V}$ is a diagonal matrix with the eigenvalues of graph Laplacian as diagonal components, and the diagonal components are obtained by $\Lambda_{ii} = \sum_j (\mathrm{A}_{ij}^{skel} + \mathrm{I}_{ij})$. The output of the S-GC $\mathbf{X}_{out}^{skel}$ can be represented by a three-dimensional tensor $(F \times T \times V)$, where T indicates a frame. Therefore, T-GC can be implemented by general convolution with arbitrary kernel size $1 \times \varGamma$, as in ST-GCN [14].

3.2 Attention Branch

The attention branch generates the attention edge and node. In the attention branch, the feature map obtained from the feature extractor is input to five STGC-Blocks. The feature map from the five STGC-Blocks passes through global average pooling (GAP), fully connected layers, and the softmax layer to obtain class probabilities. In addition, the attention edge and attention node are generated using the feature map from the last STGC-Block. The output of the last STGC-Block is a three-dimensional tensor $(C \times T \times V)$. To generate attention edges and nodes, the number of dimensions is reduced by using a batch normalization layer and a general convolution layer with a kernel size of 1×1. Then, an attention edge and an attention node are generated by individual processing.

Attention Edge. The attention edge is an adjacency matrix that represents the optimal joint relationship for each action. Attention edges can be expressed as $(K \times V \times V)$ by using the number K of attention edges generated for each action and the node V. K is an optional parameter. The feature map $(C \times T \times V)$ obtained by 1×1 general convolutional layers becomes $(C \times 1 \times V)$ due to the GAP layer. Next, the feature map is extended to $(KV \times 1 \times V)$ by 1×1 convolutional layers. The feature map is converted into $(K \times V \times V)$ and passes through the batch normalization. By passing through the Tanh function, low-value (i.e., insignificant) elements in the feature map are converted to negative values. Finally, by passing through the ReLU, negative elements in the feature map become zero. In short, there is no connection. Therefore, an attention edge with only important connections is generated. The generated attention edge is passed to the perception branch, and the graph is convolved with two graph structures (the skeleton pattern and the attention edge) in the perception branch.

Attention Node. The attention node represents the dynamic importance of joints that differ depending on each action and each frame. The attention node for each action can be expressed as $(T' \times V)$, where T' is frame. The feature map $(C \times T \times V)$ obtained by 1×1 general convolutional layers is reduced to $(1 \times T \times V)$ by the second 1×1 convolutional layer and passes through the batch normalization. At the T-GC layer in attention branch, the number of frames is reduced with one half compared to the feature map obtained from the feature extractor. Therefore, the feature map is extended to $(1 \times T' \times V)$. Then, the upsampling layer interpolates feature map by nearest neighbor. Finally, an attention node is generated by applying the sigmoid function to the feature map $(1 \times T' \times V)$.

The generated attention node is applied to the attention mechanism. The feature map that emphasizes important joints is obtained by reflecting the attention node on the feature map of the feature extractor. Reflecting the attention node is defined by

$$\mathbf{X}'_{out} = M(\mathbf{X}^{FE}_{out}) \cdot \mathbf{X}^{FE}_{out}, \tag{2}$$

where $\mathbf{X}_{out}^{FE}$ is a feature map from the feature extractor, and $M(\mathbf{X}_{out}^{FE})$ indicates an attention node. The feature map $\mathbf{X}'_{out}$ in which important joints are emphasized is used as input for the perception branch.

3.3 Perception Branch

The perception branch inputs a feature map in which important joints are emphasized and performs graph convolution with two graph structures: a human skeleton pattern and an attention edge. Let $\mathbf{A}^{att} \in \mathbb{R}^{K \times V \times V}$ be the attention edge and $\mathbf{X}_{in} \in \mathbb{R}^{V \times C}$ be the input feature map. By using the weight matrix $\mathbf{W}^{att} \in \mathbb{R}^{K \times V \times F}$ with output dimension F, the graph convolution with $\mathbf{A}^{att}$ and $\mathbf{X}_{in}$ is defined as follows:

$$\mathbf{X}_{out}^{att} = \sum_{k=1}^{K} \hat{\mathbf{A}}_k^{att} \mathbf{X}_{in} \mathbf{W}_k^{att}, \tag{3}$$

where K is the number of attention graphs generated for each action. $\hat{\mathbf{A}}^{att}$ is a normalized adjacency matrix, which performs normalization in the same way as Eq. (1). Attention edge emphasizes the relationship of joints for each action, but human actions essentially depend on the skeletal pattern. Therefore in the perception branch, graph convolutions are performed for the both attention edge and the human skeleton pattern. We denote outputs of graph convolution for attention edge and skeleton pattern as $\mathbf{X}_{out}^{att}$ and $\mathbf{X}_{out}^{skel}$, respectively. $\mathbf{X}_{out}^{skel}$ is calculated in the same way as Eq. (1). The output $\mathbf{X}_{out}$ can be obtained by

$$\mathbf{X}_{out} = \mathbf{X}_{out}^{skel} + \mathbf{X}_{out}^{att}. \tag{4}$$

By using multiple STGC-Blocks, a feature map is obtained that considers important relationships between joints and important joints. The acquired feature map passes through the GAP layer and the fully connected layer and is sent to the softmax to obtain the class probability. We use the output from the perception branch as the final class probability of the network.

3.4 Learning Method

The network of the proposed method has two branches (attention and perception), and class probabilities are obtained from each branch. The learning error L of the proposed method is calculated by $L = L_{att} + L_{per}$, where L_{att} and L_{per} are the learning errors of the attention and perception branches, respectively. The learning error of each branch is calculated using the cross entropy error.

3.5 Mechanics-Stream

From the joint coordinates of the skeletons, the joint velocity, joint acceleration, bones, bone velocity, and bone acceleration are calculated and input to the network. The joint coordinates and bones contain spatial position information,

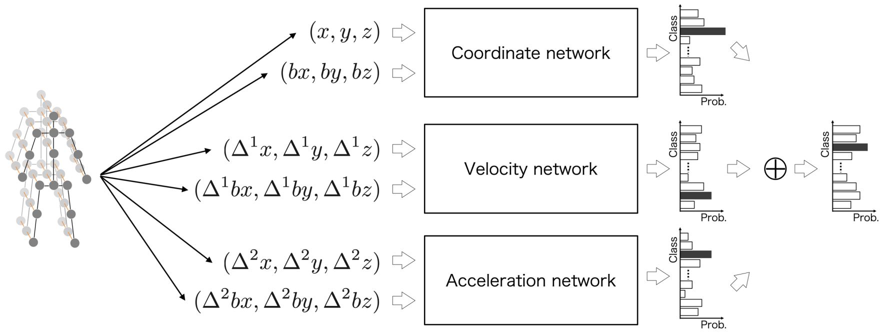

(x, y, z)
(bx, by, bz)
Coordinate network
$(\Delta^1 x, \Delta^1 y, \Delta^1 z)$
$(\Delta^1 bx, \Delta^1 by, \Delta^1 bz)$
Velocity network
$(\Delta^2 x, \Delta^2 y, \Delta^2 z)$
$(\Delta^2 bx, \Delta^2 by, \Delta^2 bz)$
Acceleration network
Class
Prob.
$\oplus$

Fig. 4. Mechanics-stream structure.

while the velocity and acceleration contain information on temporal movement. Therefore, the characteristics of coordinates, velocity, and acceleration are different. In addition, it is considered that learning is difficult even if those are input to the same network because the scales of their values are different. We propose a mechanics-stream structure to solve the problems of different spatial-temporal characteristics and different value scales of each modal.

Figure 4 shows the mechanics-stream structure, which prepares three networks for six modals. Two types of modals are input to one network. The mechanics-stream structure consists of a coordinate network that inputs joint coordinates and bones, a velocity network that inputs joint and bone velocities, and an acceleration network that inputs joint and bone accelerations. The three networks are trained separately. The class probabilities obtained from each network are summed for each class to obtain the final class probability.

Multi-modal. Joint velocity, joint acceleration, bones, bone velocity, and bone acceleration are calculated on the basis of joint coordinates. The joint velocity $\varDelta^1 v_A^t = (\varDelta^1 x, \varDelta^1 y, \varDelta^1 z)$ of a joint $v_A = (x, y, z)$ in the t frame indicates the amount of movement of each joint frame, which is calculated by

$$\varDelta^1 v_A^t = v_A^{t-1} - v_A^t. \tag{5}$$

In addition, the joint acceleration $\varDelta^2 v_A^t = (\varDelta^2 x, \varDelta^2 y, \varDelta^2 z)$ of the joint v_A in the t frame can be obtained by using the formula of linear action with uniform acceleration as follows:

$$\varDelta^2 v_A^t = \frac{(\varDelta^1 v_A^t)^2 - (\varDelta^1 v_A^{t-1})^2}{2}. \tag{6}$$

A bone is the distance in the x, y, and z directions between each joint in the skeletal pattern. Therefore, the direction of the joint can be expressed. The bone $b_{AB}^t = (bx, by, bz)$ at the adjacent joint v_A^t, v_B^t in the skeletal pattern is determined by $b_{AB}^t = v_A^t - v_B^t$. The bone velocity is the amount of movement of the bone in one frame and expresses the angular velocity in the direction of the joint. The bone velocity $\varDelta^1 b^t_{AB} = (\varDelta^1 bx, \varDelta^1 by, \varDelta^1 bz)$ of the bone b^t_{AB} in the t frame can be calculated by the same calculation as Eq. (5) with the bone b^{t-1}_{AB} one frame before. The bone acceleration $\varDelta^2 b^t_{AB} = (\varDelta^2 bx, \varDelta^2 by, \varDelta^2 bz)$ can also be obtained by the same calculation as Eq. (6) with the bone velocity $\varDelta^1 b^t_{AB}$ and $\varDelta^1 b^{t-1}_{AB}$.

4 Experiment

We first evaluate the accuracy when applying the attention graph to a conventional method. In addition, we show the effectiveness of the mechanics-stream structure by changing the combination of modals input to the network. After that, we compare the proposed method with the conventional methods by using two datasets for action recognition.

4.1 Datasets

NTU-RGB+D. NTU-RGB+D [8] is a dataset for action recognition that with 60 different action classes such as daily and exercise action. The skeletons was captured by Microsoft Kinect v2, and the number of joints is 25, the skeleton consists of three-dimensional coordinates (X, Y, Z). This dataset has two evaluation methods: cross-subject and cross-view. Cross-subject divides the data of 40 subjects into training and validation. In cross-view, data taken from the left and right 45 degrees is used for training, and data taken from the front is used for validation. The subjects for training and validation is determined in advance.

NTU-RGB+D120. NTU-RGB+D120 [39] contains 120 action classes by adding 60 new action classes to NTU-RGB+D. As with NTU-RGB+D, the number of joints is 25, which consists of three-dimensional coordinates (X, Y, Z). While keeping the same shooting direction, the camera height and the distance to a subject are given. This is a large dataset for action recognition with skeletons. There are two evaluation methods: cross-subject and cross-setup. Cross-subject uses the data of the specified subjects for training and the data of the remaining subjects for validation. Cross-setup splits data for training and validation by IDs assigned on the basis of camera height and distance.

4.2 Implementation Details

The output dimensions of the STGC-Block in the feature extractor were 32 dimensions for the first three blocks and 64 dimensions for the remaining two blocks. Feature maps obtained from the different modals are concatenated to form a 128-dimensional feature map. The attention branch passes through two blocks with 128 output dimensions and three blocks with 256 output dimensions. The perception branch sets the same output dimension as the attention branch. In the skeletal pattern, hop Q is set to 3. The number of attention edges generated

Table 1. Accuracy of applying attention edge and attention node to conventional method. Evaluation was performed with cross-subject of NTU-RGB+D.

Attention edge	Attention node	ST-GCN [14]	AS-GCN [17]	2s-AGCN [18]
×	×	81.5	86.8	88.5
✓	×	84.1	86.9	89.2
×	✓	82.8	86.8	89.1
✓	✓	**84.8**	**87.0**	**89.3**

Table 2. Accuracy of an Independent network (Ind.net) and stream structure (X-stream). Evaluation was performed with cross-subject of NTU-RGB+D.

Input data	w/o attention		w/ attention	
	Ind. net	X-stream	Ind. net	X-stream
Joint (coordinate, velocity)	86.0	87.1	86.2	87.2
Bone (coordinate, velocity)	86.1		86.3	
Joint (coordinate, velocity, acceleration)	86.7	87.8	85.7	86.7
Bone (coordinate, velocity, acceleration)	86.7		85.8	
Coordinate (joint, bone)	87.5	89.3	88.6	**90.1**
Velocity (joint, bone)	85.6		87.0	
Coordinate (joint, bone)	87.5	89.1	88.6	89.4
Velocity (joint, bone)	85.6		87.0	
Acceleration (joint, bone)	74.6		77.2	

K for each action is set to 4. The kernel size Γ in temporal graph convolution is 9. The number of input frames is 300. In addition, we used the process implemented by Maosen *et al.* [17] to interpolate the inactive frame in the input frame with the active frame.

4.3 Adaptation of Attention Graph to Conventional Method

We show the effectiveness of the attention graph by applying the attention graph to the conventional methods [14,17,18]. The conventional method used for the experiment is a skeleton-based action recognition method whose code is published by the authors. The adapted method generates an attention graph by adding an attention branch to the hidden layer. The acquired attention graph is adapted to the attention mechanism and graph convolution as in the proposed method.

Table 1 shows the accuracy of applying the attention graph to the conventional methods. The accuracy was improved in most cases where only the attention edge or node was applied to the conventional method. In all conventional methods, the accuracy is highest when both the attention edge and node are applied. These results show that the attention edge and node contribute to improve recognition accuracy, and it is the most effective to apply both simultaneously.

Table 3. Comparison of accuracy on NTU-RGB+D.

Methods	Cross-subject (%)	Cross-view (%)
Lie group [22]	50.1	52.8
Deep LSTM [8]	60.7	67.3
TCN [1]	74.3	83.1
ST-GCN [14]	81.5	88.3
AS-GCN [17]	86.8	94.2
2s-AGCN [18]	88.5	95.1
AGC-LSTM [19]	89.2	95.0
DGNN [20]	89.9	**96.1**
STA-GCN	**90.1**	95.6

Table 4. Comparison of accuracy on NTU-RGB+D120.

Method	Cross-subject (%)	Cross-setup (%)
Soft RNN [40]	36.3	44.9
Dynamic skeleton [41]	50.8	54.7
Spatio-temporal LSTM [9]	55.7	57.9
GCA-LSTM [12]	58.3	59.2
Multi-task learning network [3]	58.4	57.9
FSNet [4]	59.9	62.4
Skeleton visualization [2]	60.3	63.2
Two-stream attention LSTM [13]	61.2	63.3
Multi-task CNN with RotClips [5]	62.2	61.8
Body pose evolution map [42]	64.6	66.9
SkeleMotion [6]	67.7	66.9
TSRJI [43]	67.9	62.8
ST-GCN [14]	72.8	75.4
STA-GCN	**83.9**	**86.5**

4.4 Accuracy with Multi-modal Learning

Table 2 shows the accuracy of an independent network (Ind.net) that inputs multiple modals and the accuracy of stream structure (X-stream). The structure in which the coordinates and velocity are trained in another network has higher accuracy than the structure in which the coordinates and velocity are trained in the same network. These results demonstrate the validity of the mechanics-stream structure. However, when acceleration was added as an input, the accuracy decreased even in the mechanics-stream structure. Acceleration is composed of very small values and does not have enough features for action recognition. Therefore, a model using acceleration achieved poorer result.

4.5 Comparison with State-of-the-Arts

Table 3 shows comparison results in NTU-RGB+D. Compared with the best conventional method (DGNN), STA-GCN improved the cross-subject by 0.2 points, and achieved comparable accuracy in cross-view.

Table 4 shows comparison results in NTU-RGB+D120. STA-GCN significant improbed both cross-subject and cross-setup compared with the best conventional methods. NTU-RGB+D120 contains similar action classes that are difficult to recognize due to the increase the number of classes. The proposed method, which can emphasize specific features by using an attention graph for each action, acquired different features even for similar action classes and contributed to accuracy improvement.

4.6 Visualization of Attention Graph

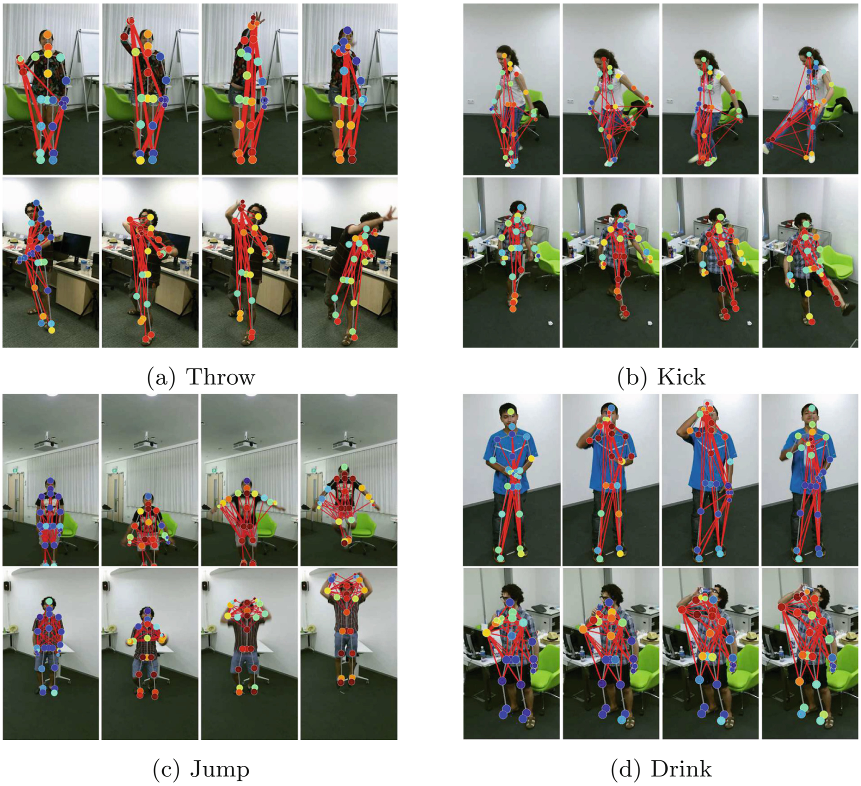
Figure 5 shows the attention graph obtained by the proposed method. We visualized the top 30 attention edges having the highest edge weight as red lines. The color of each joint indicates the value of the attention node: red joints is the most important and blue joints is the least important.

As for the throwing (Fig. 5(a)), the edge is concentrated on the right arm. The attention node shows that the importance of the right hand increases while right hand extends upward and decreases after the throw is completed. In addition, the importance of the left foot gradually increases during the throwing. This is due to the weight shifting to the left foot during the throw.

The attention edge of the kicking (Fig. 5(b)) was concentrated on the le.g. The edges of upper results are concentrated on the right leg and the edges of lower results are concentrated on the left leg because the kick was with right and left legs, respectively. This shows that the proposed method can obtain a connection pattern specific to the action.

The Attention edge of the jumping (Fig. 5(c)) is almost symmetrical. In the attention node, the legs have higher importance just before jumping. The importance of the center of the body increases during the jump. Since there are symmetrica attention edges and body center is high importance, jumping is an action using the whole body conceivably.

The attention edge of the drinking (Fig. 5(d)) is concentrated on the right arm because the right hand holds the drink. Although there was no concentration of edges on the face, the importance of the face as well as the right arm tended to be higher in the attention node. Drinking is similar to throwing, where the right arm is an important joint. Although, when throwing, there are important joints on the left leg, when drinking, few important joints appear on the lower body. This shows that the attention node can also express important joints specific to an action.



(a) Throw

(b) Kick

(c) Jump

(d) Drink

Fig. 5. Visualization of attention graph. (Color figure online)

4.7 Ablation Study

Effect of Each Modal on Attention Graph. To evaluate the effect of each modal on the attention graph, Fig. 6 shows the attention graph of the throwing when one modal is input. Although some attention edges concentrate to joints other than the arms, one attention edge concentrates to the arms for every modal inputs. Therefore, the arm can be regarded as important regardless of the modal input.

The attention node shows the right arm is the most important joint for joint coordinate and bone inputs. However, when the velocity was input, the results showed that the elbow and head had higher importance than the hand. When the joint acceleration is input, only the importance of the arm increases. With bone acceleration, the importance of the whole body increases. From these results, it is considered that the attention node when velocity and acceleration are input tends to strengthen joints with small values. Since the velocity is the amount of movement for one frame, the value of the moved joint (i.e., hands and toes) increases, while the values of body center, elbows, and knees decrease. The attention node played a role in increasing the value of the elbows and knees

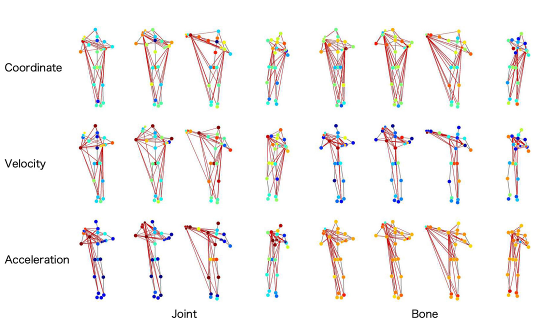


Fig. 6. Attention graph when one modal is input.

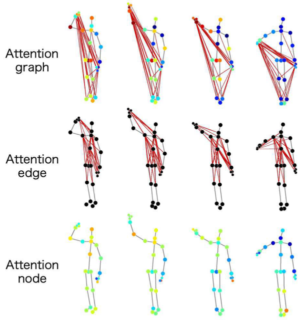


Fig. 7. Interaction of attention edge and node.

around large and important joints. Acceleration has a smaller value scale than coordinates and velocity. For joint acceleration, the value of the toe is very low and is difficult to increase even with the attention node. Therefore, only the joints around the arm that have slight values tended to be emphasized. Bone acceleration has the smallest value scale among all modals. It was difficult to determine which joints were important, resulting in increased whole body values.

Interaction of Attention Edge and Attention Node. To investigate the interaction between the attention edge and the attention node, Fig. 7 shows the attention of the throwing when acquiring only the attention edge, only the attention node, and both. When only the attention edge, the edge concentrate on the right arm, but there is no connection with the foot. The attention edge connects the highly correlated joints, so many edges connect to the relatively moving right and left hands. The relationship between the hand and the foot is also important, but in the upright throwing, the foot does not change relative to the hand movement. In addition, when only the attention node is acquired, a specific joint cannot be emphasized. If the attention node places a strong constraint on a specific joint, the joints are convoluted only by the skeletal pattern, making it difficult to transmit information between joints. In the case of acquiring both an attention edge and node, the importance of the right arm is increasing. Even if a particular joint is regarded as important, there is a connection with various joints by the attention edge and it is easy to transmit information to other joints. Similarly, the hand and foot edges were obtained not only for the highly correlated joints, but also to capture the relationships with important joints obtained from the attention node. Therefore, both important relationships between joints and the importance of joints need to be simultaneously acquired.

5 Conclusions

We proposed a STA-GCN that considers the static relationships between joints for each action and the dynamic importance of joints for each frame. The attention edge expresses the important connection for each action, and the attention node expresses the importance of the joint that differs for each frame. STA-GCN simultaneously generates an attention edge and node, and the attention node is adapted to the feature map by using the attention mechanism. The feature map that emphasizes important joints is graph convolved by using an attention edge. Therefore, our method recognized actions while taking into account important joints and important relationships between joints. In addition, we propose the mechanics-stream structure that considers the mechanical characteristics and scale differences of each modal. The mechanics-stream structure separately inputs coordinates, velocities, and accelerations of joint and bone into the different networks. In the evaluation experiments, we demonstrated our method outperforms conventional methods using two datasets: NTU-RGB+D and NTU-RGB+D120.

Acknowledgement. This paper is based on results obtained from a project, JPNP20006, commissioned by the New Energy and Industrial Technology Development Organization (NEDO).

References

1. Soo Kim, T., Reiter, A.: Interpretable 3D human action analysis with temporal convolutional networks. In: Computer Vision and Pattern Recognition Workshop (CVPRW) (2017)
2. Mengyuan, L., Hong, L., Chen, C.: Enhanced skeleton visualization for view invariant human action recognition. Pattern Recogn. **68**, 346–362 (2017)
3. Ke, Q., Bennamoun, M., An, S., Sohel, F., Boussaid, F.: A new representation of skeleton sequences for 3D action recognition. In: Computer Vision and Pattern Recognition (CVPR) (2017)
4. Jun, L., Amir, S., Gang, W., Ling-Yu, D., Alex, K.: Skeleton-based online action prediction using scale selection network. Trans. Pattern Anal. Mach. Intell. (TPAMI) **42**, 1453–1467 (2019)
5. Qiuhong, K., Mohammed, B., Senjian, A., Ferdous, S., Boussaid, F.: Learning clip representations for skeleton-based 3D action recognition. Trans. Image Process. (TIP) **27**, 2842–2855 (2018)
6. Caetano, C., Sena, J., Brémond, F., dos Santos, J.A., Schwartz, W.R.: SkeleMotion: a new representation of skeleton joint sequences based on motion information for 3D action recognition. In: Advanced Video and Signal-based Surveillance (AVSS) (2019)
7. Yong, D., Wei, W., Liang, W.: Hierarchical recurrent neural network for skeleton based action recognition. In: Computer Vision and Pattern Recognition (CVPR) (2015)
8. Amir, S., Jun, L., Tian-Tsong, N., Gang, W.: NTU RGB+D: a large scale dataset for 3D human activity analysis. In: Computer Vision and Pattern Recognition (CVPR) (2016)
9. Liu, J., Shahroudy, A., Xu, D., Wang, G.: Spatio-temporal LSTM with trust gates for 3D human action recognition. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9907, pp. 816–833. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46487-9_50

10. Song, S., Lan, C., Xing, J., Zeng, W., Liu, J.: An end-to-end spatio-temporal attention model for human action recognition from skeleton data. In: Association for the Advancement of Artificial Intelligence (AAAI) (2017)
11. Jun, L., Amir, S., Dong, X., Alex, K., Gang, W.: Skeleton-based action recognition using spatio-temporal LSTM network with trust gates. Trans. Pattern Anal. Mach. Intell. (TPAMI) **40**, 3007–3021 (2018)
12. Liu, J., Wang, G., Hu, P., Duan, L.Y., Kot, A.C.: Global context-aware attention LSTM networks for 3D action recognition. In: Computer Vision and Pattern Recognition (CVPR) (2017)
13. Jun, L., Gang, W., Duan, L.-Y., Abdiyeva, K., Kot, A.C.: Skeleton-based human action recognition with global context-aware attention LSTM networks. Trans. Image Process. (TIP) **27**, 1586–1599 (2018)
14. Yan, S., Xiong, Y., Lin, D.: Spatial temporal graph convolutional networks for skeleton-based action recognition. In: Association for the Advancement of Artificial Intelligence (AAAI) (2018)
15. Si, C., Jing, Y., Wang, W., Wang, L., Tan, T.: Skeleton-based action recognition with spatial reasoning and temporal stack learning. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) ECCV 2018. LNCS, vol. 11205, pp. 106–121. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01246-5_7
16. Thakkar, K., Narayanan, P.J.: Part-based graph convolutional network for action recognition. In: The British Machine Vision Conference (BMVC) (2018)
17. Maosen, L., Siheng, C., Xu, C., Ya, Z., Yanfeng, W., Tian, Q.: Actional-structural graph convolutional networks for skeleton-based action recognition. In: Computer Vision and Pattern Recognition (CVPR) (2019)
18. Shi, L., Zhang, Y., Cheng, J., Lu, H.: Two-stream adaptive graph convolutional networks for skeleton-based action recognition. In: Computer Vision and Pattern Recognition (CVPR) (2019)
19. Si, C., Chen, W., Wang, W., Wang, L., Tan, T.: An attention enhanced graph convolutional LSTM network for skeleton-based action recognition. In: Computer Vision and Pattern Recognition (CVPR) (2019)
20. Lei, S., Yifan, Z., Jian, C., Hanqing, L.: Skeleton-based action recognition with directed graph neural networks. In: Computer Vision and Pattern Recognition (CVPR) (2019)
21. Hussein, M.E., Torki, M., Gowayyed, M.A., El-Saban, M.: Human action recognition using a temporal hierarchy of covariance descriptors on 3D joint locations. In: International Joint Conference on Artificial Intelligence (IJCAI) (2013)
22. Raviteja, V., Felipe, A., Rama, C.: Human action recognition by representing 3D skeletons as points in a lie group. In: Computer Vision and Pattern Recognition (CVPR) (2014)
23. Basura, F., Gavves, E., Oramas, M.J., Amir, G., Tinne, T.: Modeling video evolution for action recognition. In: Computer Vision and Pattern Recognition (CVPR) (2015)
24. Joan, B., Wojciech, Z., Arthur, S., Yann, L.: Spectral networks and locally connected networks on graphs. In: International Conference on Learning Representations (ICLR) (2014)
25. Michaël, D., Xavier, B., Pierre, V.: Convolutional neural networks on graphs with fast localized spectral filtering. In: Advances in Neural Information Processing Systems (NIPS) (2016)
26. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. In: International Conference on Learning Representations (ICLR) (2017)

27. Duvenaud, D., et al.: Convolutional networks on graphs for learning molecular fingerprints. In: Advances in Neural Information Processing Systems (NIPS) (2015)
28. Gilmer, J., Schoenholz, S.S., Riley, P.F., Vinyals, O., Dahl, G.E.: Neural message passing for quantum chemistry. In: International Conference on Machine Learning (ICML) (2017)
29. Yao, L., Mao, C., Luo, Y.: Graph convolutional networks for text classification. In: Association for the Advancement of Artificial Intelligence (AAAI) (2019)
30. Xiang, Z., Junbo, Z., Yann, L.: Character-level convolutional networks for text classification. In: Advances in Neural Information Processing Systems (NIPS) (2015)
31. Wang, X., Gupta, A.: Videos as space-time region graphs. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) ECCV 2018. LNCS, vol. 11209, pp. 413–431. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01228-1_25
32. Qi, S., Wang, W., Jia, B., Shen, J., Zhu, S.-C.: Learning human-object interactions by graph parsing neural networks. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) ECCV 2018. LNCS, vol. 11213, pp. 407–423. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01240-3_25
33. Yao, T., Pan, Y., Li, Y., Mei, T.: Exploring visual relationship for image captioning. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) Computer Vision – ECCV 2018. LNCS, vol. 11218, pp. 711–727. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01264-9_42
34. Yang, X., Tang, K., Zhang, H., Cai, J.: Auto-encoding scene graphs for image captioning. In: Computer Vision and Pattern Recognition (CVPR) (2019)
35. Hiroshi, F., Tsubasa, H., Takayoshi, Y., Hironobu, F.: Attention branch network: learning of attention mechanism for visual explanation. In: Computer Vision and Pattern Recognition (CVPR) (2019)
36. Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., Torralba, A.: Learning deep features for discriminative localization. In: Computer Vision and Pattern Recognition (CVPR) (2016)
37. R., S.R., Michael, C., Abhishek, D., Ramakrishna, V., Devi, P., Batra, D.: Grad-CAM: visual explanations from deep networks via gradient-based localization. In: International Conference on Computer Vision (ICCV) (2017)
38. Chattopadhyay, A., Sarkar, A., Howlader, P., Balasubramanian, V.N.: Grad-CAM++: generalized gradient-based visual explanations for deep convolutional networks. In: Winter Conference on Applications of Computer Vision (WACV) (2017)
39. Jun, L., Amir, S., Mauricio, P., Gang, W., Duan, L.Y., Kot, A.C.: NTU RGB+D 120: a large-scale benchmark for 3D human activity understanding. Trans. Pattern Anal. Mach. Intell. (TPAMI) **42**, 1–18 (2019)
40. Hu, J.F., Zheng, W.S., Ma, L., Wang, G., Lai, J., Zhang, J.: Early action prediction by soft regression. Trans. Pattern Anal. Mach. Intell. (TPAMI) **41**, 2568–2583 (2019)
41. Jianfang, H., Wei-Shi, Z., Jian-Huang, L., Jianguo, Z.: Jointly learning heterogeneous features for RGB-D activity recognition. In: Computer Vision and Pattern Recognition (CVPR) (2015)
42. Liu, M., Yuan, J.: Recognizing human actions as the evolution of pose estimation maps. In: Computer Vision and Pattern Recognition (CVPR) (2018)
43. Caetano, C., Brémond, F., Schwartz, W.R.: Skeleton image representation for 3D action recognition based on tree structure and reference joints. In: Conference on Graphics, Patterns and Images (SIBGRAPI) (2019)