NEF-GGCN:Node-Edge Fusion Gate Graph Convolutional Networks For Skeleton-based Medical Action Recognition

Funing Xiang[a], Weibin Liu[b], Zhiyuan Zou[b], XX Xu[c], Weiwei Xing[a,*]

[a] *School of Software Engineering, Beijing Jiaotong University, Beijing, 100044, China*
[b] *Institute of Information Science, Beijing Jiaotong University, Beijing, 100044, China*
[c] *Bestplus Information Technology, Hangzhou, 310052, China*

Abstract

Due to the global outbreak of new coronaviruses, it is imperative that medical personnel follow medical protocols to avoid infection. Although existing graph convolution methods have achieved great success in the field of skeleton-based action recognition, they have limitations in complex actions. There are a large number of hand-to-hand, hand-to-head, or hand-to-foot interactions in medical actions such as hand washing, goggle removal, etc.We define such interactions as near-end actions.To efficiently identify near-end actions using graph convolution methods, we propose a new network model:Node-Edge Fusion Gate Graph Convolutional Networks (NEF-GGCN). In addition, in order to fully utilize the skeletal information, we extract the nodes of the skeleton graph and all the edges connecting the nodes as features inputs to the model. Meanwhile, the weak dependency matrix is designed to enhance the correlation of the end nodes fused with the adjacency matrix through a gating mechanism.Furthermore, we extracted the motion information of joints and bones in the skeleton graph as a temporal attention mechanism to improve the feature representation.Our method is verified on two large datasets and one custom medical datasets, Kinetics, NTU-RGBD and Medical12, it achieves impressive results.

Keywords: `Graph Convolutional Networks(GCN), Action Recognition, Attention, Gate Mechanism`
2010 MSC: 00-01, 99-00

1. Introduction

In recent years, skeleton-based action recognition has attracted widespread attention in video recognition due to its resistance in complex environments, and has become the focus of research. With the increasing maturity of pose estimation algorithms, the prediction and recognition of key points has become quite stable, and a large number of video datasets can be converted into skeleton data based on pose estimation, leading to a rapid increase in the recognition of skeleton-based actions. However, few action recognition methods are available here for the medical field because of the lack of medical behavior-related datasets and the absence of correct standard behavior guidelines.And as the COVID-19 ravage the world, using deep learning methods to monitor the behavior of healthcare workers is significant.

Action recognition can be roughly divided into methods based on artificial design features and methods based on deep learning models [1], the latter can be subdivided into recurrent neural networks based on video sequences and image-based convolutional neural networks. In action video, spatial features can be extracted from the static structure of each frame of image, and spatial features cannot effectively identify distinctive actions. Because a complete action is composed of a set of continuous gesture segments, the temporal and spatial features based on time series have great significance for action recognition.. This fusion of time and space features gave birth to different feature representation structures, such as RGB, optical flow, and skeleton diagrams. Among them, human skeletal structure has become a hot topic of interest, in the physical sense that human action is generated by the sequence of different poses of human skeleton, and therefore naturally as a basic feature of behavior. Compared with the traditional RGB images and optical flow [2], the skeleton not only contains a lot of available features, but also has the advantages of easy calibration and robustness, etc. How to use the depth model to explore the most distinguishing features in the skeleton is a major problem we are solving nowadays. From the time series[3, 4], using RNN-based LSTM to learn spatio-temporal representation, but with non-Euclidean spatial graph structured data as the input of the network, it loses part of the spatial feature information, which makes it skeleton-based action recognition lacks sufficient discrimination in spatial dimension. In the non-Euclidean data modeling method, a new method of spatio-temporal convolution is proposed in [5], which connects the bone and joints in continuous video frames to form a three-dimensional graph structure, and then performs planar spatial convolution and continuous time convolution on the composed set. This modeling method has become a mainstream way of action recognition. In [6, 7], the graph structure is used to model the joint information of the skeleton and used as the input of the neural network, and

[*] Corresponding author
Email address: `wwxing@bjtu.edu.cn` (Weiwei Xing)

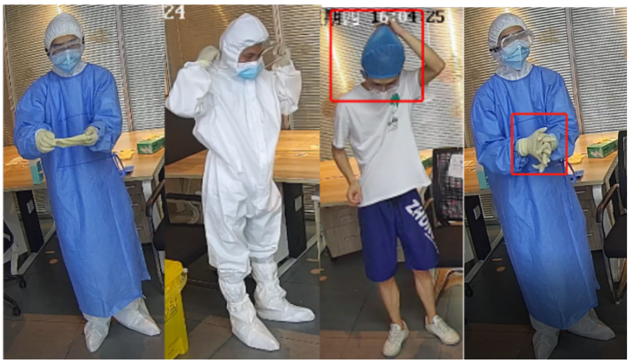

Figure 1: Tip end interactive action of the skeleton in medical field.The actions from left to right are removing gloves, removing mask, removing hat and wash hands.

the action result is obtained through the graph convolution processing. However, [5, 8] found that the bone composition based on the physical fixation of the human body is not the most effective solution. The data-driven composition method combined with the original adjacency matrix in [8] proved that different composition methods affect the accuracy of recognition.

Skeleton-based action recognition is not universal for all actions. In the medical field, hand-to-head or hand-to-hand interaction actions are prevalent, such as the four actions shown in Fig. 1: glove removal, mask removal, hat removal, and hand washing. These actions have the same characteristics as skeletal diagrams in addition to small motion ranges. In this scenario and inspired by [8], we propose a node-edge fusion gate graph convolution neural network, which can adaptively capture the spatial attributes between end joints. In particular, we first proposed a weak dependency matrix, and combined with adjacency matrix though a gating mechanism to effectively collect dependency information between joints of the skeleton. In addition, we extract motion information from the joints and bones in the frame and use it as an attention mechanism to fully characterize the dynamic process. To validate the effectiveness of this method, we experiment on two large datasets and one custom dataset Kinetics, NTU-RGBD and Medical12. Experimental results show that the combination of the motion information of the matrix and the bone adjacency class in the graph convolution leads to a significant performance improvement in skeleton-based action recognition. The contribution in this work are summarized as follows:

1. We propose a novel network model NEF-GGCN, which uses the joints and the degree of interdependence of joints as the input features of the network and effectively identify near-end actions.
2. The gate mechanism is introduced and embedded in the graph convolution layer, which assists the model to learn to selectively focus on discriminative joints and bones.
3. The motion information of the skeleton in NEF-GGCN are devoted to improve the temporal and spatial representation of the features and the robustness of the model.

2. Related Work

2.1. Skeleton-based action recognition

Due to the increasing maturity of human pose estimation techniques, skeletal-based action recognition approaches have been studied extensively in recent years. In the early stages of the deep learning, skeleton-based action recognition modeled the human body by designing handcrafted features [9, 10]. For example, Vemulapalli et al. [9] proposed a new skeletal representations that model the 3D geometric relationships between various body parts using rotations and translations in 3D space, which lies in the Lie group. Fernando et al. [10] use the parameters of learned ranking function as a new video representation. In recent years, deep learning is rapidly emerging and gaining great success in the field of computer vision, which also becomes the mainstream methods for skeleton-based action recognition.According to the type of methods, they can be roughly divided into two categories: the RNN-based methods and the CNN-based methods. RNN-based approaches usually model multiple frames of skeleton data as a sequence of vectors from the temporal dimension. The skeleton inside each frame contains a vector of coordinates of connected joints, which represent the information of spatial dimension. [11, 12, 13, 14, 15]. Du et al. [11] use 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. Song et al. [12] embed a spatiotemporal attention module in LSTM-based model, so that the network can automatically pay attention to the discriminant spatiotemporal region of the skeleton sequence.Zhang et al. [14] introduce the mechanism of view transformation in an LSTM-based model, which automatically translates the skeleton data into a more advantageous angle for action recognition. Si et al. [15] propose a model with spatial reasoning (SRN) and temporal stack learning (TSL), where the SRN can capture the structural information between different body parts and the TSL can model the detailed temporal dynamics.

The CNN-based approach focuses more on the spatial dimension of the skeleton data, which treats the skeleton inside each frame as a unique feature map [16, 17, 18, 19, 20]. Although both models are capable of sequential modeling, while CNN-based method is more about aggregating the overall information from the local information and extracting the hierarchical information from the input. Kim and Reiter [16] re-design the original TCN by factoring out the deeper layers into additive residual terms, which is used to identify actions from a frame-wise skeleton features concatenated temporally across the entire video sequence. Liu et al. [17] develop the sequence-based view invariant transform enhance skeleton, and use CNN to extract robust and discriminative features from color images. Cao et al. [19] build a classification network with stacked residual blocks and design a fully convolutional permutation network to learn an optimized order for the coordinates of body joints in one frame. Thien et al. [20] encode skeleton data to image-based representation for deep convolutional neural networks.

However, It is not optimal to treat the human skeleton data as a two-dimensional grid vector, which is more in line with the characteristics of a graph from the way it is composed by connecting key points and bones. Neither CNN nor RNN can fully represent the graph structure of the skeleton data. To address the drawbacks of the exist works, Yan et al. [5] propose a spatiotemporal graph convolutional network (ST-GCN) which model the skeleton data in several consecutive frames as a spatiotemporal graph. It eliminates the requirement for designing handcrafted transformation rules to transform the skeleton data into vector sequences or pseudo-images, thus achieves better performance. Based on this, Tang et al. [21] further propose a selection strategy of the key frames with the help of reinforcement learning. Li et al. [7] integrate the pose prediction into the action recognition task to help capture more detailed action patterns through self-supervision. Shi et al. [8] proposes a novel composition method which divides the adjacency matrix into three types of graphs to adaptively learn the graph topology for action recognition.To capture the complex spatial-temporal dependencies, Liu et al.[22] present a unified spatial-temporal graph convolutional operator .

2.2. *Graph convolutional neural networks*

CNN is widely used in the field of computer vision because it can effectively extract spatial features. However, the pixels in the image or video data processed by CNN are arranged into a neat matrix. For topological graphs CNN cannot maintain translation invariance on data of Non Euclidean Structure, and it is desired to extract spatial features efficiently for learning on such data structure, so graph convolutional neural networks (GCN) becomes the focus to solve such problems [23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33] .

The graph convolutional neural networks can be divided into two types according to different convolutional approaches: spatial-based convolution and spectral-based convolution. The spatial-based approach refers to the idea of CNN, where the neighbors of each node are weighted and summed. This approach usually requires manual design of the extraction rules for the neighbors. [25, 29, 34]. For example, Niepert et al. [25] propose a normalization algorithm to generate local normalized neighborhood representations for the nodes that covers large parts of the graph according to their distances in the graph. Wang and Gupta [34] represent the input video as a spatio-temporal region graph, and each node in the graph represents a region of interest in the video. In contrast to the spatial-based approach, the spectrum-based approach refers to the traditional way of signal processing by processing the original signal, using the eigenvalues and eigenvectors of the graph Laplacian matrix, and introducing filters to define the graph convolution. [24, 27, 28]. Defferrard et al. cite34-defferrard2016convolutional demonstrated that using recursive Chebyshev polynomials as a filtering scheme is more efficient than previous polynomial filters. kipf and Welling[27] used a first-order approximation of the spectral map convolution that further simplifies this approach.

With a sophisticated graph convolution method, graph construction is also indispensable.There are some works discussing the optimal strategy for graph construction of GCNs. Li et al.[35] assume that the optimal graph topology is a small shifting from the original graph topology and propose to learn a residual graph given the current data.Thakkar et al.[36] divide the skeleton graph into four subgraphs with joints shared across them and learn a recognition model using a part-based graph convolutional network.Guo et al.[37] also propose to learn an attention matrix according to the node features, which is then multiplied to the original adjacency matrices to dynamic adjust the impacting weights between nodes. Wu et al.[38] combine the pre-defined graphs and the learnable graph for graph convolutional layer. The learnable graph is fixed for all of the data samples.Shi et al.[31] propose two types of adaptive graphs, One of them is generated by data-driven generation, where different layers produce different matrices. The other is a data-independent matrix, where a unique matrix is learned separately for different samples.[39] propose a novel Channel-wise Topology Refinement Graph Convolution to dynamically learn different topologies and effectively aggregate joint features in different channels. These works demonstrate that the composition approach is very effective for recognition, either by learning attention weights for graph edges or by using predefined human body-based graphs. Thus, building on these works, we combine the robustness of predefined graphs with the flexibility possessed by data-driven learning of graph structures. We propose to use a gating mechanism to fuse the two kinds of graphs, avoiding the limitations of using different conformations alone.

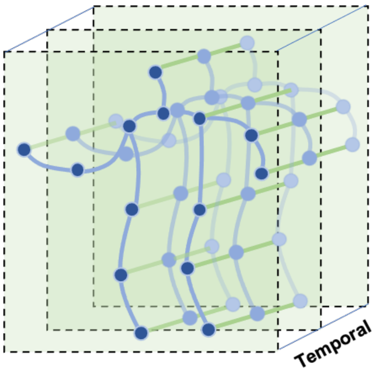

Figure 2: Illustration of the spatiotemporal graph.

3. Method

3.1. *Graph Construction*

The original skeleton in one frame is usually shown with 18 or 24 keypoints and connections between some of the keypoints and represented by vectors. Several research works have shown that using the intrinsic human skeleton as a model input is not the optimal choice[36, 31], which isn't rational for some end joints interaction actions like clap hands, brush teeth as mentioned in Sec 1.Therefor, we proposed a fusion weighted matrix $\boldsymbol{W}$ to model the human skeleton in each frame, as the Equation 1. described, $\boldsymbol{W_{ij}}$ is a $N \times N$ fusion adjacency matrix.We set $\boldsymbol{W_{ii}} = 0$ to eliminate the effects of self connection. Moreover, and give the matrix different parameters α and β according to the distinct connection dependency.For the physical connection of joints, which is describe as α in matrix $\boldsymbol{W}$ and suggested as the blue solid lines in Fig. 3. In addition to intrinsic connections, we assume that there are weakly dependent connections between other unconnected joints., which is described as the dashed orange lines in Fig. 3. Taking tooth brushing as an example, the right hand and the head are not physically connected, but when we use graph convolution for recognition, adding a weakly connected edge between these two points can effectively improve the recognition rate of the model. Therefore, we use the parameter β to represent this connection in the adjacency matrix.

$$\boldsymbol{W_{i,j}} = \begin{cases} 0, \text{ if i=j} \\ \alpha, \text{ if joint } i \text{ and } j \text{ are connected} \\ \beta, \text{ if joint } i \text{ and } j \text{ has weak connected} \end{cases} \quad (1)$$

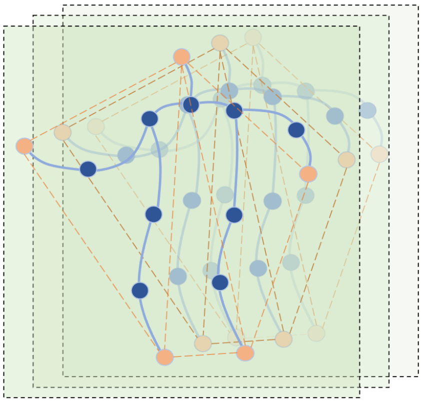

Figure 3: **Illustration of the weak dependency skeleton graph.** The blue line represent the physical connect and the weak dependency is suggested as orange dashed lines.We set different parameters in the fusion adjacency matrix to distinguish these two types of dependencies.

3.2. *Graph Convolution*

According to the key points of the human body calculated by the pose estimation, we can regard the points as a graph node and connect each node in a bones-connected manner. Here, a complete skeleton graph in a frame of image has been constructed, and it can be represented as a vector $\boldsymbol{V}$, $\boldsymbol{V} \in \mathbb{R}^{N*x}$ where N represents the number of key points and x presents the coordinate dimension, $x \in [2, 3]$ and $x \in Z$. A complete action is composed of a serials of consecutive frames.From the perspective of human skeleton, the transformation of different skeletal forms constitutes an action.Although a continuous skeletal image is obtained, how can it be effectively used as the input of image convolution? In ST-GCN, a spatiotemporal skeleton diagram is proposed to model the skeleton diagram from the spatial and temporal dimensions.In spatial dimension, the joints in one frame are connected according to the construct of physical bones.And in temporal dimension, the same key points of the human body are connected between adjacent frames, as indicated by the dark green line in Fig. 2. Each vertex in the graph has a corresponding vector representing the spatial coordinate. Given the definition of the graph above, the operation of the ST-GCN on the point v_i in the spatial graph convolution is formulated as:

$$f_{out}(v_i) = \sum_{v_j \in \mathcal{B}_i} \frac{1}{Z_{ij}} f_{in}\left(v_j\right) \cdot \omega\left(l_i\left(v_j\right)\right) \quad (2)$$

where v denotes the vertex of the graph, which is the elements of the sampling area of the convolution $\mathcal{B}_i$.f represent the feature map.ω is the weighting function that provides a weight vector based on the given input. While the parameters in weight vector is fixed, and the number of vertexes in $\mathcal{B}_i$ is varied.l_i is a mapping function which map all neighbor node according to the special strategy.This strategy separate $\mathcal{B}_i$ into 3 subsets, the first subset is vertex v_i itself.The division of the remaining two subsets is based on the proximity to the gravity centroid and whether it contains neighboring vertexes.Neighboring vertices that are close to the center of gravity will be divided into one subset, and those that are far away will be another subset.Z_{ij} denotes the Cartesian product from v_i to the subset containing v_j, in order to balance the weights of each subset.

To demonstrate the implementation of graph convolution more intuitively, the implement of ST-GCN is formulated as:

$$f_{out} = \sum_{k}^{K_v} W_k\left(f_{in}A_k\right) \odot M_k \quad (3)$$

K_v represents the size of the spatial dimensional convolution kernel, which is set to a fixed value of 3 according to the subset partitioning strategy.$W_k\left(f_{in}A_k\right)$ is the convolution operation, where f_in is a $C \times T \times N$ tensor, representing the input to the network, and the graph convolution is accomplished by performing the Laplace transform on the adjacency matrix. $A_k = \Lambda^{-\frac{1}{2}}\bar{A}_k\Lambda^{-\frac{1}{2}}$, where $A_k = A + I$ and it is similar to the $N \times N$ adjacency matrix.Its elements $\bar{A}_k^{ij}$represent whether or not v_j is connected to the target vertex v_i. $\Lambda_k^{ii} = \sum_j(\bar{A}_k^{ij}) + \alpha$ is the normalized diagonal matrix. α is set to 0.001 to avoid empty rows. $\mathbf{W_k}$ is the weight vector of the 1×1 convolution operation. $\mathbf{M_k}$ is an $N \times N$ attention map.Graph convolution operation dot product with $\mathbf{M_k}$ get the f_{out}.

Based on the ST-GCN, the 2S-AGCN does not use a predefined topological map as the input to the network, and it transforms Eq. 3 into the following form for the purpose of adaptive construct the topology of the graph.

$$f_{out} = \sum_{k}^{K_v} W_k f_{in}\left(A_k + B_k + C_k\right) \quad (4)$$

In this approach, it divide the adjacency matrix into three matrices to satisfy the adaptive requirements. The first matrix A_k, same as the A_k in Eq. 3, represents the adjacency matrix in the physical sense of the skeleton. The second matrix B_k is a data-driven matrix, which is obtained based on the training data learned, and affected by the different network layers. With this matrix is able to break the traditional way of composition and will strengthen the connection between two joints even if these two joints are not connected.The last and most important is the C_k, which is a data-dependent matrix that learns a unique topological map for each type of sample. C_k calculates the similarity of two joints by using two kinds of the normalized embedded Gaussian function θ and ϕ to obtain the $N \times C_eT$ and $C_eT \times N$ matrices, and then multiplies them to obtain an $N \times N$ similarity matrix C_k.The calculation formula is shown as:

$$C_k = softmax\left(\boldsymbol{f_{in}}^{\boldsymbol{T}} \boldsymbol{W_{\theta k}^{T}} \boldsymbol{W_{\phi k}} \boldsymbol{f_{in}}\right) \tag{5}$$

3.3. Gate Mechanism For Matrix Fusion

The spatiotemporal graph convolution for the skeleton data described above is calculated based on a predefined or data-driven graph, and based on these works we try to incorporate weak adjacency matrices to improve the end joints connections.In the medical field, there are many end-joint interactions, so we optimize the predefined skeletal topology and fuse the weak dependency matrix with the adaptive adjacency matrix through a gating mechanism. Our method is formulated as:

$$f_{out} = \sum_{k}^{K_v} W_{k1} f_{in} \left(A_k + \sigma W_{k2} D_k\right) \odot T_k \tag{6}$$

where K_v and A_k are kept the same as mentioned above, K_v is set as a constant 3, A_k is an adaptive matrix, and we fuse the weak adjacency matrix D_k through the σ function as a gating and assign a certain weight through the W_{k2} matrix. At the end, the feature map is obtained by dot-product operation with our motion attention matrix.

As shown in Fig. 4, we take the 3D skeletal diagram as input, and use the gating mechanism implemented by the σ function to obtain the fused adjacency matrix and perform convolution operations, which is computed as Eq. 6.

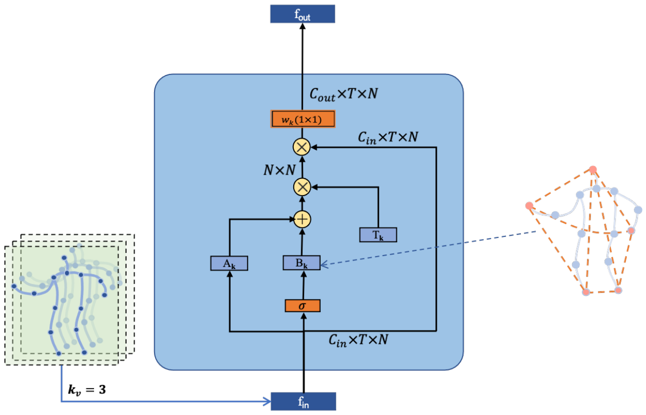
f_{out}
$C_{out} \times T \times N$
$w_k(1\times1)$
$C_{in} \times T \times N$
$N \times N$
T_k
A_k
B_k
σ
$C_{in} \times T \times N$
$k_v = 3$
f_{in}

Figure 4: Illustration of the graph convolutional layer.There are two types graphs in each layer, i.e., A_k and D_k. $w_k(1 \times 1)$ denotes the kernel size of the convolution.σ is the gate function control the fusion result. T_k is the motion attention matrix.

3.4. Temporal Motion Attention

The attention mechanism is most commonly used to enhance the performance of feature in various network structures, and there are formulations for attention modules. In this work , we focus on similar actions such as washing hands and clapping, making phone calls and brushing teeth. From a macro perspective of the skeleton, these actions have a high degree of similarity, with only minor differences in the characteristics of some limbs. Therefore, we extract the motion information of the skeleton between consecutive frames and use it as an attention matrix to improve the robustness of the model.

Temporal motion attention module(TMAM) integrates the fluctuation amplitude and rotation angle characteristics of the same joints between the bones, and strengthens the characterization of limbs movements. It is computed as:

$$T_k = \mu\left(D_i^k\right) + \lambda\left(Max\left(\omega_i^k\right)\right) \tag{7}$$

$$D_i^k = Max\left(\sum_{k}^{K_v} \left|p_i^{k+1} - p_i^k\right|\right) \tag{8}$$

Where $\boldsymbol{D_i^k}$ is the maximum fluctuation range between the two frames of joint i at time t and time t-1 among consecutive K frames. The greater the distance, the greater the weight in the attention matrix.Since the skeleton are connected by multiple joints, the movement of the position in the performance of the action cannot fully describe the limbs language, so we introduce ω_i^k, which represents the angle at which the bone connected to the joint i rotates between the two frames. And adjust the overall weight through the parameters μ and λ.

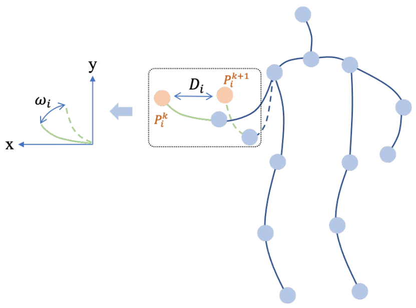

Figure 5: Illustration of the attention module based on motion information.Joints fluctuation distance and bones rotation angle is the basic motion feature for motion attention.

Fig. 6 shows a basic block of NEF-GGCN, which is the series of one spatial GCN , one attention module (TMAM) and one temporal GCN. The convolution along the temporal dimension is the same as the 2S-AGCN, i.e., performing the $K_t \times 1$ convolution on the C × T × N feature maps. Both the spatial GCN and the temporal GCN are followed by a batch normalization layer and a ReLU layer. To stabilize the training and ease the gradient propagation, a residual connection is added for each basic block.

4. Experiments

To demonstrate the generalizability and validity of the method for medical behaviors, we validated our method on three different datasets: NTU-RGBD, Kinetics, and Medical12.

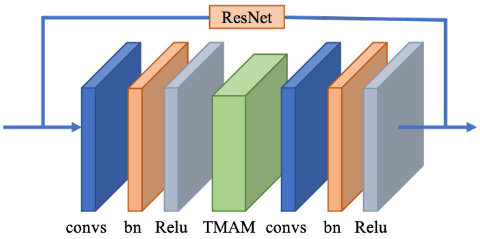

Figure 6: Illustration of network structure.spconv represents the spatial GCN, and temconv represents the temporal GCN, both of which are followed by a BN layer and a ReLU layer. TMAM represents the temporal motion attention module. Moreover, ResNet is added for each block.

4.1. Datasets and Experiment Settings

Kinetics400: Kinetics is a large-scale human action data set containing 300,000 video clips in 400 categories. These video clips come from YouTube videos, and there are many kinds. It only provides raw video clips without frame data. Use the publicly available OpenPose toolbox to estimate the positions of 18 joints on each editing frame. Based on the average joint confidence, two people are selected as a multiplayer clip. We use their published data (Kinetics-Skeleton) to evaluate our model. The data set is divided into a training set (240,000 clips) and a validation set (20,000 clips). According to the evaluation method in, we train the model on the training set and report the top 1 and top 5 accuracy on the validation set.

NTU-RGB+D60: This is the currently largest dataset for action recognition with more than 56 thousand sequences and 4 million frames. The dataset was captured from 40 different human subjects and has 60 classes actions. We use the 3D skeleton data of 25 major body joints. The benchmark evaluations include Cross-Subject (CS) and Cross-View (CV) setting. In the Cross-Subject evaluation, 40320 samples from 20 subjects were used for training and the other 16540 samples were for testing. In Cross-View evaluation, the 37,920 samples captured from camera 2 and 3 were used for training, while the other 18960 samples from camera 1 were for testing.

Medical12: Medical12 is our custom dataset. We collected the standard medical actions of 20 medical personnel in two different rooms as they entered and left the isolation area. Each person repeated 1-2 sets, and a total of 12 actions such as removing/putting on gloves, removing/putting on goggles and taking off/putting on shoe covers were included in the dataset. During the recording process, the same camera was used to record at all times. We filtered and cut the collected action videos to obtain a total of 654 valid video clips as the dataset. In addition, we used the OpenPose tool to estimate the 18 key point locations of the skeleton in each frame of the video, including only one medical personnel in each action scene.We used the same method as the Kinetics dataset for evaluation.

4.2. Implementation Details

All experiments were extrapolated on the pytorch framework and employed using the stochastic gradient descent algorithm with its initial learning rate set to 0.1, decay weight of 0.0001, batch size of 32, and cross-entropy chosen as the loss function for back-propagating the gradient as in [8].

For the Kinetics dataset, we set the input maximum number of people to 2, and the window size is 150 frames. These 150 frames are randomly selected from the input skeleton sequence, which is consistent with the method of [8], and the rotation and translation are randomly selected. Come to interfere with the joint coordinates a little bit. The learning rate is also set to 0.1 and divided by 10 in the 45_{th} and 55_{th} epochs. A total of 65 epochs of training.

For the NTU-RGBD dataset,we set parameters differently from the Kinetics dataset.For samples with less than 300 frames, we repeat the samples until it reaches 300 frames. The learning rate is set as 0.1 and is divided by 10 at the 30_{th} epoch and 40_{th} epoch. The training process is ended at the 50_{th} epoch.Other parameters remain the same as Kinetics.

For the Medical12 custom dataset, we used the NTU-RGBD-based training results as a pre-trained model with an expanded sample size. For the size of the input tensor, the maximum number of people was set to 1, the window size was set to 300 frames, and random selection was used due to the long duration of some of the actions. The learning rate is initialized to 0.1 and decremented by one decimal place in 30_{th} and 40_{th} epoch, for a total of 45 training epoch.

4.3. Ablation Study

In order to verify the feasibility of NEF-GGCN, we compare the overall performance of the training results in NTU-RGBD and the comparison results are shown in Table 1. We use the 2S-AGCN as the baseline method, we separate the 2S-AGCN as the Js-AGCN and Bs-AGCN according to the [8], and both model fusion with the weak dependency matrix. It shows the subtle lift with the fusion matrix, which reflect the importance of the weak dependency matrix.Based on this model, we proposed NEF-GCN and gradually add gating mechanism and attention mechanism.The results validate our model on the dataset show that our method brings encouraging improvement. Overall, it brings improvements of +1.2% and +1.7% compared with 2s-AGCN with W on CS and CV benchmarks, respectively.

Table 1: Comparisons of the validation accuracy when fuse weak dependency matrix W and with Gate and Temporal Motion Attention Mechanism (TMAM).

Method	X-Sub(%)	X-View(%)
Js-AGCN/With W	87.3 + 0.3	93.7 + 0.2
Bs-AGCN/With W	86.9 + 0.2	93.2 + 0.4
2s-AGCN/With W	87.5	94.2
NEF-GCN/With Gate	87.9	94.5
NEF-GCN/With Gate & TMAM	88.7	95.9
NEF-GGCN	**88.7**	**95.9**

4.4. Results on NTU+RGBD and Kinetics datasets

We compare our model with the other skeleton-based action recognition methods on NTU-RGBD dataset and Kinetics-Skeleton dataset. The results of these two comparisons are shown in Tab 2 and Tab 3. The methods used for comparison include the handcraft-feature-based method [9], RNN-based methods[4, 14], CNN-based methods[16, 18] and GCN-based methods [5, 40, 41, 8].Our model obtain the notable performance with a large margin on both of the datasets, which verifies the effectiveness of our model.

Table 2: Skeleton based action recognition performance on NTU-RGB+D datasets. We report the accuracies on both the cross-subject (X-Sub) and cross-view (X-View) benchmarks.

Method	X-Sub(%)	X-View(%)
Lie Group[9]	50.1	82.8
ST-LSTM[4]	69.2	77.7
VA-LSTM[14]	79.2	87.7
TCN[16]	74.3	83.1
3scale ResNet152[18]	85.0	92.3
ST-GCN[5]	81.5	88.3
Sem-GCN[40]	86.2	94.2
PGCN-TCA[41]	88.0	93.6
2S-AGCN[8]	88.5	95.1
NEF-GGCN	**88.7**	**95.9**

Table 3: Comparisons of the validation accuracy with GCN-based methods on Kinetics Dataset.

Method	Top-1(%)	Top-5(%)
TCN[16]	20.3	40.0
ST-GCN[5]	30.7	52.8
2S-AGCN[8]	36.1	58.7
NEF-GGCN	**36.5**	**59.4**

4.5. Results on Medical12 dataset

The results of the comparison on the public dataset demonstrate the usability of our model. In addition, we also performed comparisons on a custom Medical12 dataset. After data extension and data preprocessing, we compared our model trained on NTU-RGBD as a pre-trained model with other skeleton-based graph convolution methods[5, 8], and the comparison results are shown in Table 4, which proves the better performance of our method on actions with more end-node interactions.

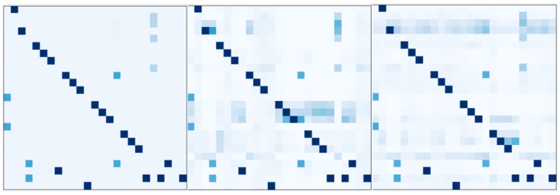

Figure 7: Example of the adaptive adjacency matrices. Different rows shows different subsets. The first matrix is the adjacency matrices of the human-body-based graph in the NTU-RGBD dataset. Others are examples of the learned adaptive adjacency matrices of different layers learned by our model.

Visualization: Fig. 7 shows the adaptive matrices we learned in fusing the weak dependency matrices. The first matrix is a graph of the adjacency matrix based on the human body, and the other two are graphs of the adaptive matrices learned by the different network layers. The attention graph visualization of the hat removal action in Medical12 is shown in Fig. 8. The graph contains a skeletal diagram of the three processes of the doctor in the hat removal, and the larger size of the key point in the skeletal diagram represents the larger value of that point in the weight matrix. Therefore, during the hat removal process, the model will pay more attention to the arm that touches the hat.

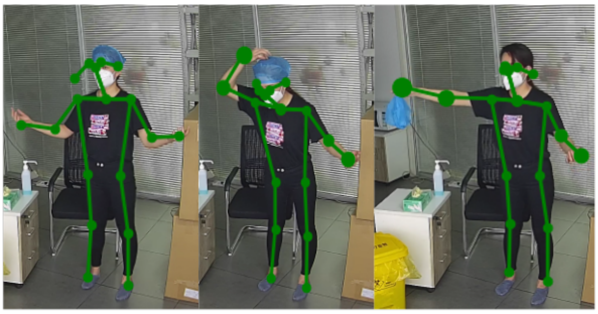

Figure 8: Virtualization of motion attention map on skeleton.Example of the taking off hat, the bigger size of the point the more importer of the joint.

Table 4: Comparisons of the validation accuracy with skeleton-based GCN methods on Medical12 Dataset.

Method	Top1(%)	Top5(%)
ST-GCN[5]	85.2	90.6
2s-AGCN[8]	89.3	95.1
NEF-GGCN	**95.5**	**98.6**

5. Conclusion

In this work, we propose a novel recognition method for action recognition in medical scenarios, which simulates weak connections between joints at the end of the human skeleton through a weighted adjacency matrix and fuses them with a parametric graph structure through a gating mechanism. This method not only improves the dependency between end joints, but also preserves the dependency between the original related nodes. In addition, we introduce a temporal motion attention mechanism based on the range of joint fluctuations. This mechanism allows us to train a more general model and adapt it to a larger number of actions. In addition, our model shows impressive results on two large-scale datasets Kinetics, NTU-RGBD, and a custom Medical12 dataset.

6. Acknowledgements

This work is supported by the Bestplus Information Technology, China. The authors would like to thank the anonymous reviewers for their valuable comments and suggestions, which will help to improve the quality of this manuscript.

References

[1] T. Subetha, S. Chitrakala, A survey on human activity recognition from videos, in: 2016 international conference on information communication and embedded systems (ICICES), IEEE, 2016, pp. 1–7.

[2] K. Simonyan, A. Zisserman, Two-stream convolutional networks for action recognition in videos, arXiv preprint arXiv:1406.2199.

[3] J. Liu, G. Wang, L.-Y. Duan, K. Abdiyeva, A. C. Kot, Skeleton-based human action recognition with global context-aware attention lstm networks, IEEE Transactions on Image Processing 27 (4) (2017) 1586–1599.

[4] J. Liu, A. Shahroudy, D. Xu, G. Wang, Spatio-temporal lstm with trust gates for 3d human action recognition, in: European conference on computer vision, Springer, 2016, pp. 816–833.

[5] S. Yan, Y. Xiong, D. Lin, Spatial temporal graph convolutional networks for skeleton-based action recognition, in: Thirty-second AAAI conference on artificial intelligence, 2018.

[6] C. Si, W. Chen, W. Wang, L. Wang, T. Tan, An attention enhanced graph convolutional lstm network for skeleton-based action recognition, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 1227–1236.

[7] M. Li, S. Chen, X. Chen, Y. Zhang, Y. Wang, Q. Tian, Actional-structural graph convolutional networks for skeleton-based action recognition, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2019, pp. 3595–3603.

[8] L. Shi, Y. Zhang, J. Cheng, H. Lu, Two-stream adaptive graph convolutional networks for skeleton-based action recognition, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2019, pp. 12026–12035.

[9] R. Vemulapalli, F. Arrate, R. Chellappa, 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, 2014, pp. 588–595.

[10] B. Fernando, E. Gavves, J. M. Oramas, A. Ghodrati, T. Tuytelaars, Modeling video evolution for action recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 5378–5387.

[11] Y. Du, W. Wang, L. Wang, Hierarchical recurrent neural network for skeleton based action recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 1110–1118.

[12] S. Song, C. Lan, J. Xing, W. Zeng, J. Liu, An end-to-end spatio-temporal attention model for human action recognition from skeleton data, in: Proceedings of the AAAI conference on artificial intelligence, Vol. 31, 2017.

[13] S. Li, W. Li, C. Cook, C. Zhu, Y. Gao, Independently recurrent neural network (indrnn): Building a longer and deeper rnn, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 5457–5466.

[14] P. Zhang, C. Lan, J. Xing, W. Zeng, J. Xue, N. Zheng, View adaptive recurrent neural networks for high performance human action recognition from skeleton data, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 2117–2126.

[15] C. Si, Y. Jing, W. Wang, L. Wang, T. Tan, Skeleton-based action recognition with spatial reasoning and temporal stack learning, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 103–118.

[16] T. S. Kim, A. Reiter, Interpretable 3d human action analysis with temporal convolutional networks, in: 2017 IEEE conference on computer vision and pattern recognition workshops (CVPRW), IEEE, 2017, pp. 1623–1631.

[17] M. Liu, H. Liu, C. Chen, Enhanced skeleton visualization for view invariant human action recognition, Pattern Recognition 68 (2017) 346–362.

[18] B. Li, Y. Dai, X. Cheng, H. Chen, Y. Lin, M. He, Skeleton based action recognition using translation-scale invariant image mapping and multiscale deep cnn, in: 2017 IEEE International Conference on Multimedia & Expo Workshops (ICMEW), IEEE, 2017, pp. 601–604.

[19] C. Cao, C. Lan, Y. Zhang, W. Zeng, H. Lu, Y. Zhang, Skeleton-based action recognition with gated convolutional neural networks, IEEE Transactions on Circuits and Systems for Video Technology 29 (11) (2018) 3247–3257.

[20] T. Huynh-The, C.-H. Hua, T.-T. Ngo, D.-S. Kim, Image representation of pose-transition feature for 3d skeleton-based action recognition, Information Sciences 513 (2020) 112–126. `doi:10.1016/j.ins.2019.10.047`.

[21] Y. Tang, Y. Tian, J. Lu, P. Li, J. Zhou, Deep progressive reinforcement learning for skeleton-based action recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 5323–5332.

[22] Z. Liu, H. Zhang, Z. Chen, Z. Wang, W. Ouyang, Disentangling and unifying graph convolutions for skeleton-based action recognition, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020.

[23] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, P. Vandergheynst, The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains, IEEE signal processing magazine 30 (3) (2013) 83–98.

[24] J. Bruna, W. Zaremba, A. Szlam, Y. LeCun, Spectral networks and locally connected networks on graphs, arXiv preprint arXiv:1312.6203.

[25] M. Niepert, M. Ahmed, K. Kutzkov, Learning convolutional neural networks for graphs, in: International conference on machine learning, PMLR, 2016, pp. 2014–2023.

[26] J. Atwood, D. Towsley, Diffusion-convolutional neural networks, in: Advances in neural information processing systems, 2016, pp. 1993–2001.

[27] T. N. Kipf, M. Welling, Semi-supervised classification with graph convolutional networks, arXiv preprint arXiv:1609.02907.

[28] M. Defferrard, X. Bresson, P. Vandergheynst, Convolutional neural networks on graphs with fast localized spectral filtering, Advances in neural information processing systems 29 (2016) 3844–3852.

[29] T. Kipf, E. Fetaya, K.-C. Wang, M. Welling, R. Zemel, Neural relational inference for interacting systems, in: International Conference on Machine Learning, PMLR, 2018, pp. 2688–2697.

[30] B. Ren, M. Liu, R. Ding, H. Liu, A survey on 3d skeleton-based action recognition using learning method, arXiv preprint arXiv:2002.05907.

[31] L. Shi, Y. Zhang, J. Cheng, H. Lu, Skeleton-based action recognition with multi-stream adaptive graph convolutional networks, IEEE Transactions on Image Processing (2020) 9532–9545.

[32] Y. Yoon, J. Yu, M. Jeon, Predictively encoded graph convolutional network for noise-robust skeleton-based action recognition, Applied Intelligence (2021) 1–15.

[33] W. Xu, M. Wu, J. Zhu, M. Zhao, Multi-scale skeleton adaptive weighted gcn for skeleton-based human action recognition in iot, Applied Soft Computing 104 (2021) 107236.

[34] X. Wang, A. Gupta, Videos as space-time region graphs, in: Proceedings of the European conference on computer vision (ECCV), 2018, pp. 399–417.

[35] R. Li, S. Wang, F. Zhu, J. Huang, Adaptive graph convolutional neural networks, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 32, 2018.

[36] K. Thakkar, P. Narayanan, Part-based graph convolutional network for action recognition, arXiv preprint arXiv:1809.04983.

[37] S. Guo, Y. Lin, N. Feng, C. Song, H. Wan, Attention based spatial-temporal graph convolutional networks for traffic flow forecasting, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 33, 2019, pp. 922–929.

[38] Z. Wu, S. Pan, G. Long, J. Jiang, C. Zhang, Graph wavenet for deep spatial-temporal graph modeling, arXiv preprint arXiv:1906.00121.

[39] Y. Chen, Z. Zhang, C. Yuan, B. Li, Y. Deng, W. Hu, Channel-wise topology refinement graph convolution for skeleton-based action recognition, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 13359–13368.

[40] X. Ding, K. Yang, W. Chen, A semantics-guided graph convolutional network for skeleton-based action recognition, in: Proceedings of the 2020 the 4th International Conference on Innovation in Artificial Intelligence, 2020, pp. 130–136.

[41] H. Yang, Y. Gu, J. Zhu, K. Hu, X. Zhang, Pgcn-tca: Pseudo graph convolutional network with temporal and channel-wise attention for skeleton-based action recognition, IEEE Access 8 (2020) 10040–10047.