

Motion Complement and Temporal Multifocusing for Skeleton-Based Action Recognition

Cong Wu^{ID}, Xiao-Jun Wu^{ID}, Tianyang Xu^{ID}, Zhongwei Shen^{ID}, and Josef Kittler^{ID}, *Life Member, IEEE*

Abstract—Modeling sequences with spatial-temporal graph convolutional networks has become a mainstream paradigm in skeleton-based action recognition. However, many existing methods adopt redundant or cluttered structures to mine the key action features, thus making it difficult to achieve a balanced or leading performance in accuracy and efficiency. In this paper, we propose a novel framework, referred to as Motion Complement and Temporal Multifocusing Network (MCTM-Net), to capture the relationships within skeleton sequences by means of an efficient decomposition of the spatiotemporal graph model. Specifically, for spatial modeling, we introduce a motion-related relational descriptor that extends the channel dimension so as to enhance the modeling of motion salient regions as a complement to the conventional physical adjacency relationships. An improved parameterized physical relationship model is also proposed to better fit the data characteristics. As for temporal modeling, we propose an efficient multi-focus temporal information acquisition strategy that aggregates the information from multiple temporal spans and adjacent regions. We conduct extensive experiments on multiple representative datasets, including NTU-RGB+D (60&120), Northwestern-UCLA, and UWA3D Multiview Activity II, to validate our innovations. The experimental results show the effectiveness of our method. The code will be available at <https://github.com/cong-wu/MCMT-Net>.

Index Terms—Graph convolutional network, skeleton-based action recognition.

I. INTRODUCTION

WITH the rapid development of computer vision, the problem of devising multimodal representation of visual content has received extensive attention, owing to its

Manuscript received 24 June 2022; revised 14 September 2022 and 12 November 2022; accepted 2 January 2023. Date of publication 12 January 2023; date of current version 8 January 2024. This work was supported in part by the National Natural Science Foundation of China under Grant U1836218, Grant 62020106012, Grant 62106089, and Grant 61672265; in part by the 111 Project of Ministry of Education of China under Grant B12018; in part by the Postgraduate Research and Practice Innovation Program of Jiangsu Province under Grant KYCX22_2299; in part by Engineering and Physical Sciences Research Council (EPSRC) through Multimodal Video Search by Examples (MVSE) under Grant EP/V002740/2; in part by the EPSRC through Joint Academic Data Science Endeavour (JADE2) under Grant EP/T022205/1; and in part by the EPSRC/Defence Science and Technology Laboratory (dstl)/Multidisciplinary University Research Initiatives (MURI) Project under Grant EP/R018456/1. This article was recommended by Associate Editor S. Li. (*Corresponding author: Xiao-Jun Wu.*)

Cong Wu, Xiao-Jun Wu, Tianyang Xu, and Zhongwei Shen are with the School of Artificial Intelligence and Computer Science, Jiangnan University, Wuxi 214122, China (e-mail: congwu@stu.jiangnan.edu.cn; xiaojun_wu_jnu@163.com; tianyang_xu@163.com; shenzw_cv@163.com).

Josef Kittler is with the Centre for Vision, Speech and Signal Processing, University of Surrey, GU2 7XH Guildford, U.K. (e-mail: j.kittler@surrey.ac.uk).

Color versions of one or more figures in this article are available at <https://doi.org/10.1109/TCSVT.2023.3236430>.

Digital Object Identifier 10.1109/TCSVT.2023.3236430

relevance to many different application scenarios [1], [2], [3], [4], [5], [6]. One essential visual information analysis task is human action recognition. For this task, the skeleton sequence is acknowledged to provide a very robust and powerful representation of the human action data.

The merits of skeleton sequences to characterize human action are multifold. First of all, people usually use depth vision sensors to capture skeleton data [7], [8], [9], which overcomes the challenges posed by external environmental conditions such as occlusion and insufficient light. Secondly, compared with the hundreds of thousands of pixels in an RGB image, the data volume of skeleton features is extremely low. Each frame in the skeleton sequence contains only dozens of joint points defined by their physical space coordinates. These characteristics highlight its advantages in practical applications such as autonomous driving, real-time identification, and violence detection [10], [11]. However, these characteristics also give rise to difficulties in skeleton modeling. First, the low amount of data conveyed by its concise description also limits the available information. Therefore, employing more efficient models to exploit the representational capacity of skeleton data is essential. Secondly, the consequence of the non-Euclidean structure of the skeleton implies that conventional deep learning models are handicapped in dealing with skeleton-based data analysis tasks.

Recent studies have indicated that graph convolutional networks (GCN) have inherent advantages in modeling non-Euclidean structural data such as the skeleton [12]. A topological graph structure can be constructed, reflecting the relationship between the adjacent nodes. The interactions between the nodes then can be modeled by continuously aggregating and updating the information conveyed by each feature point according to the established graph structure. Grounded on this premise, Yan et al. [13] proposed a spatiotemporal graph convolutional network based on the natural form of the skeleton. Later, Shi et al. [14] proposed a multi-stream adaptive graph convolutional network, in which a relational schema, based on an embedded similarity representation, was adopted. Also, various feature expression methods (including joints, bones, and corresponding motion information) were devised, and the performance was further improved by fusing the features extracted by the multi-stream network.

Despite the advances made by developing a series of exciting graph convolution networks, the current approaches to modeling skeleton sequences ignore many informative clues, with the consequence that the existing methods fail to model the relationships adequately or introduce too many unnecessary operations when dealing with this task. For example, to exploit the motion information, which is characteristic of the underlying behavioral patterns, existing methods [15], [16]

proposed to construct differential representations on the input side, which are very rough on one hand, and significantly increase the training costs on the other. Although some methods [17], [18], [19] have started to acknowledge the importance of multi-scale modeling in terms of temporal modeling, they either ignore the regional information or the interplay between different time series spans. They also tend to be inefficient or fail to capture the action information comprehensively.

We argue that it is difficult to thoroughly and efficiently model complex skeleton sequences from a single perspective, so we strengthen and expand the existing methods by building a multi-aspect relational representation model in which the modeling of different perspectives can mutually be reinforced. First, we start by considering the strategies that combine motion patterns with adjacency relations, as motion information is crucial in action sequence modeling, and the channel dimension provides rich feature information. Accordingly, we construct a flexible relational schema capturing motion, together with spatial and channel information. This motion-enhanced relational representation is based on the current features rather than the original skeleton sequence, thus providing more adaptive, rich, and pertinent information. It cannot be overemphasized that the above modeling, as an implicit relationship description, needs to be carried out in conjunction with explicit physical modeling. For the latter, we build a parameterized physical adjacency representation capturing the inherent adjacency relationships of the skeleton structure by means of a trainable sparse matrix that is initialized based on a Gaussian distribution. Simultaneously, we conduct multi-focus modeling in the temporal domain. In particular, by adjusting the temporal receptive field, we extract multiple views for each node based on different temporal distances from the current position. Moreover, we enrich the perceptual granularity of the representation by modeling the relationship between the current node and the corresponding regions from adjacent frames. Finally, we build a spatiotemporal graph convolution module by integrating the above information sources using suitable fusion methods.

To validate our method, we conduct extensive experiments on NTU-RGB+D (60&120) [8], [9], Northwestern-UCLA [7] and UWA3D Multiview Activity II [20] datasets respectively. The experimental results fully support our design, and the final results on multiple benchmarks exceed the mainstream state-of-the-art methods.

The above innovations can be summarized as follows:

- To enhance the quality of spatial modeling, we introduce an extended relational representation of the motion feature, which enables the generation of a data-dependent description; we also propose a parameterized strategy to make the fixed spatial relationship description more robust and adaptive.
- In order to obtain a multi-scale spatiotemporal description, we adopt a temporal modeling method based on multi-scale receptive fields.
- Thanks to the above innovations, we design an effective and efficient spatiotemporal graph convolution module.
- To validate our method, we conduct extensive experiments on NTU-RGB+D (60&120) [8], [9], Northwestern-UCLA [7], and UWA3D Multiview Activity II [20] datasets respectively.

II. RELATED WORK

A. Graph Convolutional Neural Networks

In natural scenes, many commonly used data exist in the form of non-Euclidean structures, such as knowledge graphs, molecular structures, point clouds, voxels, etc. Analogous to the use of Convolutional Neural Networks (CNNs) as the feature extractor for digital images, Graph Neural Networks (GNNs) are a conventional paradigm when dealing with non-Euclidean structured data [21]. Each node in the graph continuously updates its state in the optimization stage according to the relationship with other positions until it reaches equilibrium. Generally, graph convolution includes operations across the spatial domain and spectral domain. The methods focusing on the spatial domain directly employ the convolution operation to model the interaction of the neighboring nodes, which is analogous to the traditional convolutional neural network [13]. Spectral domain graph convolution uses the graph theory to implement convolution operations on topological graphs [12]. Specific forms of graph networks are often required to be constructed in different situations. In our work, we also adopt a spatial graph convolution approach for the skeleton-based action recognition task and extend it to be applicable to spatiotemporal graph structures.

B. Skeleton-Based Action Recognition

1) *Non-GCN Methods*: Before the advent of deep learning, skeleton-based action recognition focused on designing informative hand-crafted features, such as star skeleton [22] or points in a lie group [23]. Later, the community began to develop solutions that transform deep learning frameworks based on pixel-structured data into skeleton-based tasks. Accordingly, [24] proposed to represent a skeleton in an image, which is then fed into a CNN. With the widespread application of Recurrent Neural Networks (RNNs) in video [25], text [26], and other domains, people realized that they are also highly relevant for capturing relational features between frames in an action sequence. An example is the work in [27] which proposed to use RNN to model the long-term contextual information conveyed by a skeleton sequence.

2) *GCN Methods*: As mentioned in Section I, graph convolutional networks offer natural advantages compared with conventional methods when dealing with skeleton modeling. Yan et al. [13] extended the conventional spatial graph convolution and proposed a spatiotemporal graph convolution structure to better reflect the intuitive understanding of spatiotemporal information characterizing actions. Shi et al. [4] proposed a data-driven two-stream structure utilizing multiple deformations of the skeleton, including joint and bone representations of human action. Shi et al. [15] also proposed a four-stream network to combine joint and bone motion features. Most of the following methods are based on this spatiotemporal graph structure. Reference [16] used shift operations to construct a spatiotemporal model with low computational cost. By stacking a complex multi-scale model, [17] achieved a considerable performance gain at the expense of computational overhead. Reference [19] innovatively extended the original relational representation to the channel dimension by proposing a channel-wise topology refinement structure.

Reference [28] proposed a local-and-global attention network for skeleton modeling tasks to enhance spatiotemporal features to capture context, which has been proven effective in action recognition. Given the promise widely attributed to the self-attention mechanism [29], [30], there are grounds to believe that the combination of GCN-based methods and self-attention will prove to be an exciting direction of research. But simply combining the two may not necessarily lead to better results, so more sophisticated designs and experiments are required. And in this article, our focus is on how to fully build an effective spatiotemporal graph convolutional network, rather than expanding the network structure. We believe that the combination of GCN and self-attention is feasible to work, but it is independent of our current innovations.

III. THE PROPOSED APPROACH

In this section, we provide the details of our framework. In Section III-A, we introduce a basic definition of the skeleton graph. Next, in Sections III-B and III-C, we define the proposed graph operations in the spatial and temporal domains, respectively. In Section III-D, we integrate our methodological innovations to obtain a spatiotemporal graph convolutional block and the final network.

A. Skeleton Graph Definition

A skeleton is represented by a graph $G = (V, E)$, where V is a set of skeleton joint points, and E represents a set of corresponding edges. Assuming that G_l is the graph in the l _th layer, then the graph G_{l+1} corresponding to the $(l+1)$ _st layer can be expressed as,

$$G_{l+1} = F(G_l, W_l) = F^U((F^A(G_l, W_l^A)), W_l^U), \quad (1)$$

where $F(\cdot)$ represents a graph operation. This process can be divided into aggregation operation $F^A(\cdot)$ and update operation $F^U(\cdot)$, where W_l^A and W_l^U denote the corresponding parameters.

B. Spatial Graph Convolutional Network

According to [13] and Equation 1, a typical spatial graph convolutional network can be considered as performing a data transformation by means of a standard 2D convolution with a kernel of 1×1 , followed by a multiplication of the resulting feature by a normalized adjacency matrix in the spatial dimension. The formulated spatial graph convolution can be expressed as,

$$f_{out} = \sum_k^K (D^{-1} A_k) \text{Conv}(f_{in}), \quad (2)$$

where f_{in} and f_{out} are the input and output features of the current graph convolution operation, respectively, specified by three parameters: number of channels C , number of frames T , and number of joints in each frame V . $A \in \mathbb{R}^{V \times V}$ represents the adjacency matrix (If the joint points i and j are physically connected, the corresponding $A_{ij} = 1$. Otherwise, $A_{ij} = 0$), and D is the corresponding degree matrix. K denotes the kernel size of the spatial adjacency matrix, with K set to 3.

1) Parameterized Physical Adjacency Representation: A purely physical adjacency relationship only considers the fixed joint positions, connected in pairs, thus ignoring possible relationships with regions. The adjacency representation A can be allowed to be trainable, but this will destroy its initial information content. Some methods [14], [17] have tried to increase the flexibility of the initial representation by introducing a trainable matrix with different initialization strategies. However, as the initialization of the network parameters in deep CNN learning is crucial [31], we need to adopt a careful initialization strategy. To this end, to make full use of the spatial adjacency representation and to enhance the robustness of the network, we introduce a parameterization of the physical relation representation that enhances the inherent expressiveness of the model during training so as to characterize the input features more naturally. Specifically, we introduce a new trainable parameter matrix through a Gaussian distribution-based initialization. Before training begins, each value in the matrix is set to 0 with probability p , and the remaining non-zero positions will take random values from a Gaussian distribution $\mathcal{N}(\mu, \sigma)$. The aim is to endow the graph structure with the ability to extract useful information. Random initialization of the parameters of the different layers of the network is also likely to produce more robust features. At the same time, to facilitate the basic explicit modeling, we add this parameterized matrix to the initial adjacency representation. By rewriting Equation 2, we get

$$f_{out} = \sum_k^K P(D^{-1} A_k) \text{Conv}_k(f_{in}), \quad (3)$$

where $P(\cdot)$ stands for a parameterized operation. The specific structures are shown in the upper part of Figure 1. We will conduct more in-depth verification and analysis of these strategies in the experimental section.

2) Motion Representation: Although the parameterized physical adjacency representation discussed in the previous section provides an improved relationship, its reliance on physical adjacency imposes limitations. Moreover, this representation is input data-independent, i.e. the samples of the entire dataset will share the same set of parameters. The focus during the graph modeling is only on the corresponding information of adjacent regions. The differences between different samples are ignored. Thus, this representation only provides a general basic model, known to be ignoring the modeling of the most critical motion patterns in sequences that convey behavior. As a series of works in [32], [33], and [34] have demonstrated, motion patterns play a crucial role in action recognition tasks. Our proposed motion representation based relational modeling adheres to this idea.

To endow the network with the ability to model implicit dynamic relations, we extract a motion-enriched relationship representation by drawing on the high-dimensional semantic information conveyed by the computed features. In addition, we extend the corresponding relationship matrix to the spatial and channel dimensions, as the channel dimension is often instrumental in enhancing the representational capacity, as shown in [19] and [18]. In theory, we calculate the relationship representation for the current input in terms of a kinematic description of the features in successive frames. The specific

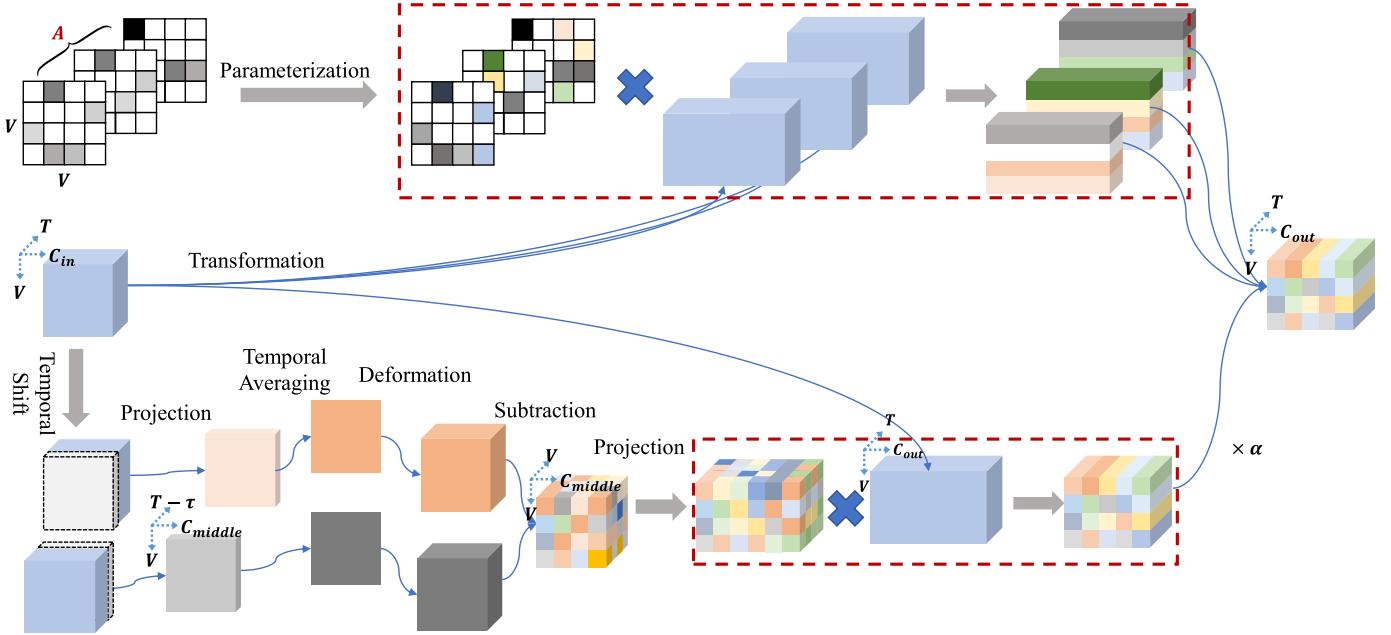


Fig. 1. Spatial graph convolution operation. The red dashed boxes represent the process of graph operations, and the other parts represent the generation of relational descriptions. We construct relational representations with both explicit and implicit approaches. For the explicit part, we use a multi-dimensional parametric physical adjacency representation, as shown in the upper part of the figure. For the implicit part, we compute a motion enhanced representation of the relationships of the elements of the evolving skeleton structure in an action sequence, as shown in the lower part of the picture. In addition, for each convolution operation, we use different transformation features to strengthen the representation capability of the network.

structure is shown in the lower part of Figure 1. We maintain the basic graph convolution operation, i.e., with the help of the relation matrix, to obtain the neighborhood information. The current node features are updated with the aim of establishing the relational representation required for motion perception, as shown by the red dashed box in Figure 1.

Specifically, first, we shift the input features along the time dimension with τ as the stride to obtain $f_{in}^{1:T-\tau}$ and $f_{in}^{\tau:T}$, and project them into a new representation space, respectively. Then we perform the pooling operation on the temporal scale to obtain the corresponding abstract representation of the entire current sequence. Since the dimension of the feature to be processed at this time is $C_m \times V$ ($C_m = \frac{C}{r}$ represents the channel dimension in the current representation space, and r is the corresponding scale of the projection space), in order to obtain the differential representation, we transform its dimension into $C_m \times 1 \times V$ and $C_m \times V \times 1$. Afterward, we replicate these two features in the second and third dimensions to obtain a consistent representation of dimension $C_m \times V \times V$. After this dimensional modification, we subtract these two features to get a difference representation and project it into the output space to define the transformation feature. A concise 2D convolution with a kernel of $(1, 1)$ implements all projections. We denote the motion relation generator as M . Similar to Equation 2, the graph operation here can be expressed as,

$$f_{out} = M(f_{in}) \text{Conv}(f_{in}). \quad (4)$$

This modeling strategy will highlight the positions in the sequence with relatively drastic motion changes. The characterization of these relationships is the essence of graph modeling, aiming at adaptive information enhancement of the salient regions reflecting the motion pattern.

C. Multi-Focus Temporal Convolutional Network

Usually, in the task of action recognition, the spatial feature modeling of each frame of the video is often accompanied by the aggregation operation of the time series information. Different from the spatial graph structure characteristics, from the temporal perspective, the skeleton sequence joint points conform to the Euclidean structure. To perform temporal modeling, [14] chooses a two-dimensional convolution with a convolution kernel of $(9, 1)$. However, this static modeling method is too rigid, reflecting two problems: (1) Actions do not necessarily occur within a particular fixed scale in time. We noticed that, unlike conventional tasks, the input of the skeleton modeling network often exceeds 30 frames, so a temporal modeling strategy combining long and short time is required to obtain multi-scale features. (2) It is easy to ignore the information of adjacent positions simply by considering only the fixed changes of the same position in the sequence. Although our spatial modeling is able to aggregate information from salient regions, it is mainly based on global adjacency representations enhanced by motion patterns, largely ignoring the specific representations of very local information which is crucial in spatiotemporal modeling. Some recent studies [17], [18], [35] have begun to take note of these problems, but they still have failed to propose a comprehensive solution.

Motivated by these shortcomings, we introduce multi-focus temporal modeling to construct a rich information acquisition method by focusing on different scales and different regions, as shown in Figure 2. In particular, we propose several innovations: For any given feature point x_{it} (i and t represent the spatial and temporal index in the skeleton sequence), (1) we aggregate the information from $x_{it'}, t' \in [-k, k]$ by setting different temporal perception scales k . With this design, the temporal modeling operation can take into account

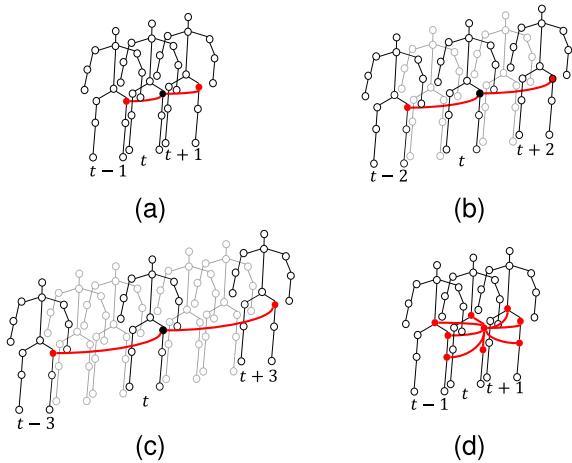


Fig. 2. **Multi-focus temporal graph modeling.** Using the right hip in the skeleton sequence as an example, we illustrate four information acquisition methods. (a) (b) (c) consider the information aggregation of the same position (right hip) between frames with different spans, respectively. (d) extends the modeling of adjacent frames and, in addition, gathers the information conveyed by the adjacent regions.

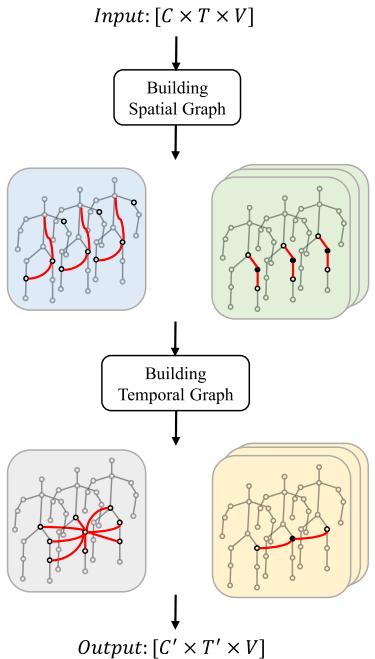


Fig. 3. Operations within a spatiotemporal graph convolutional layer.

information at multiple scales in the same sequence, making it more adaptable to relatively long sequences; (2) for region modeling, we consider the information in a 2D cube adjacent to the current position, that is, to integrate information from $x_{i'i'}, i' \in [-l, l], i' \in [-k, k]$. In the specific implementation, first of all, we add a 2D convolution, with a kernel size of 1, to reduce the number of channels of the input features to a quarter of the original number. Subsequently, we perform a 2D dilated convolution with kernel size 5×1 with varying dilation coefficients $\{1, 2, 3\}$ as the multi-scale modeling operation, and introduce another convolution with a 3×3 kernel to obtain the information of adjacent regions of adjacent frames. We take the current features as the input of these branches and concatenate the obtained features in the channel dimension.

TABLE I
STRUCTURAL DETAILS OF MCTM-NET. THE PARAMETERS OF EACH LAYER INCLUDE THE NUMBER OF OUTPUT CHANNELS AND THE TEMPORAL STRIDE. THE DIMENSIONS OF THE INPUT AND OUTPUT FEATURE CORRESPOND TO THE NUMBER OF CHANNELS C , FRAMES T , AND JOINTS V , RESPECTIVELY

Stage	Layer	Output
Input		$3 \times 64 \times 25$
Feature Extractor	Bn	-
	Block1	$[64, 1]$ $[64, 1]$ $[64, 1]$ $[64, 1]$
	Block2	$[128, 2]$ $[128, 1]$ $[128, 1]$
	Block2	$[256, 2]$ $[256, 1]$ $[256, 1]$
Action Classifier	Global Average Pooling	$256 \times 1 \times 1$
	FC + SoftMax	Categories

In this way, the proposed modeling process is more compact and efficient than similar methods.

D. The Overall Architecture

In order to make a fair comparison, we adopt the structural construction method commonly used by the mainstream methods [13], [14], [16], [19]. Generally speaking, each basic spatiotemporal graph convolution layer is the combination of one spatial graph convolutional operation, one temporal graph convolutional operation, and a residual connection, as shown in Figure 3 (For the sake of simplicity, the residual connection is omitted in this figure). We stack multi-layer spatiotemporal graph convolution modules. The specific operations and the parameters of each layer are shown in Table I (Under different datasets or different frame sampling lengths, the dimension of the feature may change with different frame sampling strategies and data set selection, especially for T and V , but the specific network structure parameters will remain constant.). The final features are obtained by a 10-layer spatiotemporal graph convolution architecture. After uniform pooling, the deep features are sent to the fully connected layer and softmax to obtain the final classification results.

IV. EXPERIMENTS

A. Datasets

1) **NTU-RGB+D:** NTU-RGB+D is currently the most commonly used large-scale dataset for skeleton-based action recognition. NTU RGB+D-60 [8] includes 60 categories, involving a total of 56,880 video samples from 80 views and 40 subjects. NTU RGB+D-120 [9] further expands it to 120 categories involving 114,480 video samples from 155 views of 106 subjects. Actions in these two datasets can be divided into three main categories: daily actions, mutual actions, and medical conditions. Two evaluation methods are used to measure the performance of different methods on these datasets, including cross-subject and cross-view. In the Cross-Subject

evaluation, both datasets select half of the subjects as the training set and the rest as the test set. For Cross-View evaluation, NTU-RGB+D 60 picks all the samples collected by camera 1 for testing and samples collected by cameras 2 and 3 for training, while NTU-RGB+D 120 picks all the samples with even collection setup IDs for training and the others for the test.

2) Northwestern-UCLA: The Northwestern-UCLA Multiview 3D event dataset [7] is a relatively small dataset comprising a total of 1,494 video samples. It contains ten categories: picking up with one hand, picking up with two hands, dropping trash, walking around, sitting down, standing up, donning, doffing, throwing, and carrying. Each category was performed by ten actors. Referring to the mainstream methods [7], [16], [36], we use the cross-view setting for verification. Specifically, the samples collected by the first two cameras are used for training, and the rest are used for testing.

3) UWA3D Multiview Activity II: The UWA3D Multiview Activity II Dataset [20] is another relatively small dataset comprising a total of 30 activities performed by 10 subjects. Each acquisition of each action contains four different views, including the front, left, right, and top. We follow the cross-view evaluation method suggested in [20], that is, training on samples from any two views, then testing on the remaining views. The possible occlusion and similarity exist among some categories, which makes this dataset very challenging.

B. Implementation Details

All experiments are performed on a single GeForce RTX 2080 Ti configured with the Pytorch [37] platform. Moreover, we choose the cross-entropy loss [38] as our classification loss and use Stochastic Gradient Descent with Nesterov Momentum (0.9) [39] as the optimization algorithm. Furthermore, we use the method from [40] to process the NTU RGB+D and NTU RGB+D 120 datasets, with the batch size set to 64 and number of the frame set to 64, and adopt the method from [16] to process the Northwestern-UCLA dataset, with the batch size of 16. In the training and evaluation process, for NTU-RGB+D 60 and NTU-RGB+D 120, we set the initial learning rate to 0.1, the number of training epochs to 65, and the learning rate to 1/10 of the previous epoch at the 35th and 55th epoch. We also used a warm-up strategy for the first five epochs to promote better convergence. For Northwestern-UCLA and UWA3D Multiview Activity II, we adjust the total number of training epochs to 70 and set the learning rate to 1/10 of the previous epoch at the 30th epoch. All experimental results are obtained by training the model from scratch.

C. Ablation Studies

This part will carefully validate the proposed innovations and present the final results achieved with the multi-stream enhancement. It should be noted that, unless otherwise stated, all structures will adopt the optimal parameters by default. We also use the Top-1 accuracy as the primary measure. In addition, we use the Cross-Subject benchmark of the NTU-RGB+D 60&120 datasets to analyze the architectural configuration and perform ablation studies. For the

experiments on the UWA3D Multiview Activity II dataset, we choose $V_1 \& V_2$ as the training views, and V_3 as the test view.

1) Experimental Exploration of the Spatial Graph Convolution Structure: In this part, we separately validate the innovations in spatial modeling. Specifically, we investigate the impact of explicit modeling based on the intrinsic physical structure and implicit modeling based on motion enhancement, with all the results shown in Table II. *i*, *ii*, *iii*, and *iv* correspond to four parameterization strategies. Specifically, (i) directly sets adjacency representation A trainable. (ii) introduces a trainable matrix initialized to 0 and adds it to the initial matrix A . Compared with (ii), (iii) initializes the same mask using a uniform distribution, and (iv) uses a random initialization method based on Gaussian distribution, which is our final strategy. We found that compared with the fixed adjacency representation A , the parameterized representations achieve a significant performance improvement, but there is almost no performance difference between the first three parameterization strategies. The sparse parameterization based on Gaussian distribution achieves the best results, improving the performance to 89.8% on NTU-RGN+D 60, 84.3% on NTU-RGB+D 120, and 70.5% on UWA3D. We believe this sparse representation makes the model more robust by successfully eliminating noise.

Next, we studied implicit modeling. We found that the performance is significantly improved when this perceptual information is added. In particular, when we introduce a control parameter α , which can be adaptively updated during the training process, the model achieves the accuracy of 90.6%, 85.3%, and 70.9% on NTU-RGN+D 60, NTU-RGN+D 120, and UWA3D datasets, respectively. Moreover, although the UWA3D dataset is relatively small and therefore prone to fluctuations, our final scheme still achieves the best performance of 70.9%. Those findings suggest that the ability to capture information in graph structures can be greatly enhanced by adding a data-dependent representation conveying motion clues.

2) The Exploration of Basic Parameters: In this section, we explore all the parameters involved in spatial modeling.

- *Investigation of the Parameterized Representation of Physical Adjacency.* We verify the impact of selecting the three parameters (p , μ , and σ) on the performance, respectively. p represents the degree of sparsity, μ and σ are the mean and variance of the Gaussian distribution, respectively, as described in Section III-B.1. The results are shown in Table III. We first verify the effect of sparsity on accuracy. The results show that appropriate sparsification can significantly improve performance. Secondly, selecting the parameters of the Gaussian distribution is also crucial. If the magnitude is too large, it will cause significant noise interference, and if it is too small, it will easily lead to a suboptimal solution. Interestingly, we investigated the model performance when the mean was 0, and it turned out that this setting resulted in relatively low accuracy because the initialized relational representation favored positive values, not 0. The model achieved optimal performance when these three parameters were set as 0.5, 0.01, and 0.01.

TABLE II
THE IMPACT OF THE SPATIAL MODELING STRATEGIES

Explicit Modeling	Implicit Modeling	NTU-RGB+D 60	NTU-RGB+D 120	UWA3D
$\sum_{i=1}^3 \text{GraphConv}(A_i)$	✗	88.9	83.8	68.1
$\sum_{i=1}^3 \text{GraphConv}(P_i(A_i))$	✗	89.4	84.0	69.5
$\sum_{i=1}^3 \text{GraphConv}(P_{ii}(A_i))$	✗	89.3	84.0	69.3
$\sum_{i=1}^3 \text{GraphConv}(P_{iii}(A_i))$	✗	89.0	83.9	66.9
$\sum_{i=1}^3 \text{GraphConv}(P_{iv}(A_i))$	✗	89.8	84.3	70.5
$\sum_{i=1}^3 \text{GraphConv}(P_{iv}(A_i))$	GraphConv(M)	90.0	84.8	68.1
$\sum_{i=1}^3 \text{GraphConv}(P_{iv}(A_i))$	$\alpha \times \text{GraphConv}(M)$	90.6	85.3	70.9

TABLE III
THE PERFORMANCE OF DIFFERENT PARAMETERIZATIONS OF THE PHYSICAL ADJACENCY REPRESENTATION

p	μ	σ	NTU-RGB+D 60	NTU-RGB+D 120
0	0.01	0.01	90.2	84.9
0.5	0.01	0.01	90.6	85.3
1	0.01	0.01	90.2	84.7
0.5	0.1	0.1	90.3	84.7
0.5	0.001	0.001	90.1	84.6
0.5	0	0.01	90.0	85.0

τ	r	Location	NTU-RGB+D 60	NTU-RGB+D 120
0	2	1 ~ 9	90.3	84.9
1	2	1 ~ 9	90.0	85.0
2	2	1 ~ 9	90.6	85.3
3	2	1 ~ 9	90.2	84.9
2	1	1 ~ 9	90.3	84.6
2	4	1 ~ 9	89.9	84.8
2	8	1 ~ 9	89.4	84.7
2	2	4 ~ 9	89.6	84.3
2	2	8 ~ 9	89.6	84.5

- *Investigation of Different Models of the Motion Relationships.* For the implicit relation expression reflecting action dynamics, as described in Section III-B.2, the configuration options include issues such as the time span τ when the division is made, the channel scale r of the projection space, and the layers where one should add the implicit modeling. We found that the best performance was obtained when choosing $\tau = 2$, $r = 2$, and incorporating this representation at all layers, as Table III shows.

3) *The Effect of Different Combinations of Spatiotemporal Modeling:* In order to compare different spatial and temporal modeling methods, as shown in Table IV, we performed a variety of comparisons. All results are given under the same training and validation framework. First, we evaluate several spatial modeling approaches. In particular, compared to the method (a) which simply relies on AGCN, after replacing the spatial modeling method with our proposed method, the performance of (c) reached 89.6% and 84.6%. Similarly, method (d) improved the performance by 0.2% compared to the method (b). We noticed that our proposed method for spatial modeling also impacted the performance in conjunction

with the motion modeling capability. For (c), the temporal modeling only considers the information of the same node in adjacent frames. For (d), the temporal modeling approach uses the region-aware mechanism proposed in [18], and for (e), the temporal modeling uses multi-scale temporal modeling. The results show that our method offers significant advantages over the previous combinations. Compared with our final method (f), the performance of all the other combinations is inferior. As stated before, the UWA3D dataset is relatively small, but as can be seen from the results in the table, our method (f) still achieves the best performance.

In this part, we also compare the accuracy and the overhead of our approach with several mainstream methods. As pointed out in [41] and [42], existing efficiency measures such as FLOPs cannot represent the actual computational cost of the model effectively, due to the possible interference of multiple factors, such as IO overhead, parallelism, etc. Therefore, we choose actual training and test time as the measure of efficiency. The results in Table IV show that, first, compared to method (a), despite the higher computational complexity of training, the difference in the test time of our method (f) is minimal, while the performance improvements, which

TABLE IV

THE ACCURACY OBTAINED BY VARIOUS SPATIOTEMPORAL COMBINATIONS. THE TRAINING AND TEST TIMES SHOW THE AVERAGE TIME COST (MINUTE PER BATCH). ALL THE RESULTS ARE COMPUTED USING UNIFIED EXPERIMENTAL FRAMEWORK

Methods	Spatial	Temporal	NTU-RGB+D 60			NTU-RGB+D 120			UWA3D		
			Acc	Training	Test	Acc	Training	Test	Acc	Training	Test
a	AGCN [14]	AGCN [14]	88.7	2.5	0.4	83.7	4.3	1.2	70.5	0.03	0.01
b	CTR-GCN [19]	CTR-GCN [19]	89.9	12.8	4.6	84.9	17.3	10.6	69.7	0.13	0.07
c	Ours	AGCN [14]	89.6	4.3	0.5	84.6	6.9	0.8	69.3	0.07	0.03
d	Ours	CTR-GCN [19]	90.1	12.7	3.9	85.1	20.3	11.9	70.1	0.12	0.05
e	Ours	Graph2net [18]	89.6	4.0	0.5	84.4	6.5	1.7	68.1	0.07	0.02
f	Ours	Ours	90.6	5.6	0.6	85.3	9.2	2.1	70.9	0.1	0.02
g	CTR-GCN [19]	Ours	89.4	6.5	0.7	84.9	10.0	2.1	70.9	0.1	0.02

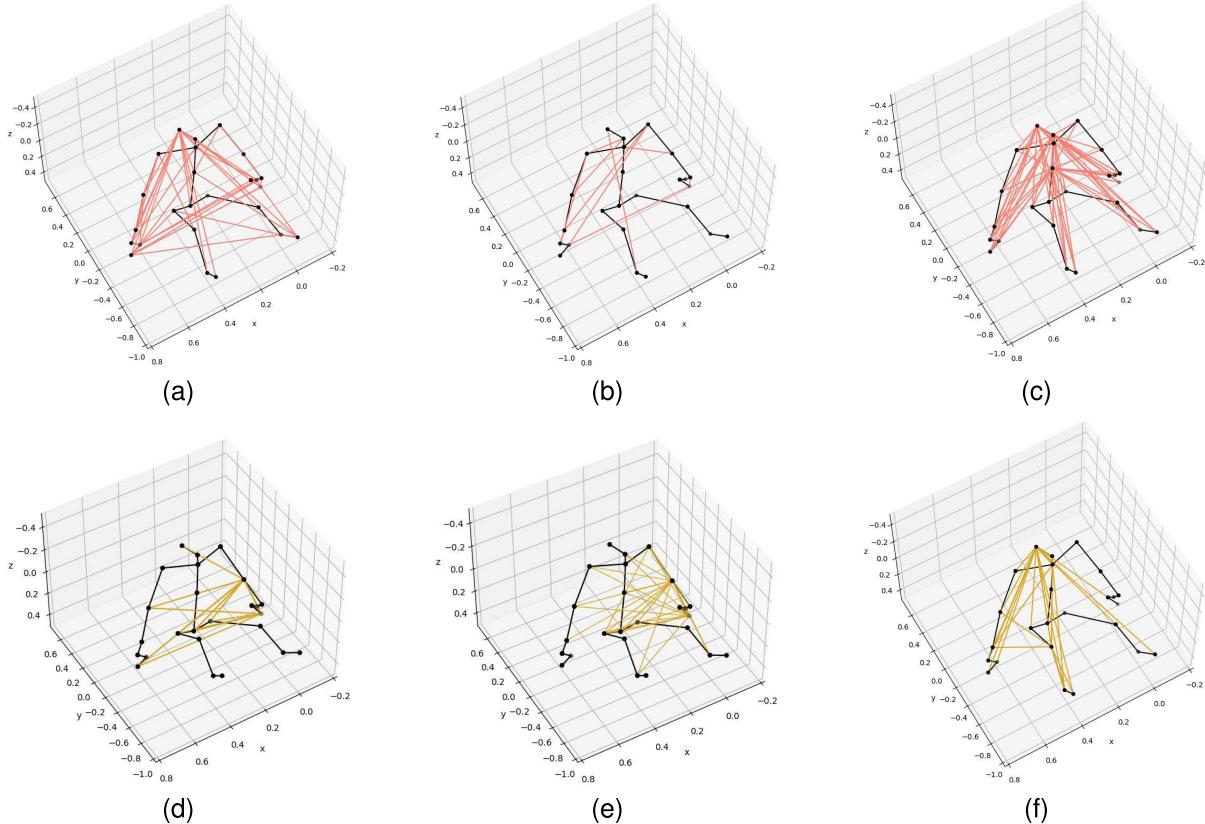


Fig. 4. **Visualization of Spatial Graph Convolutional Modeling.** The black dots and line segments represent a human skeleton structure, and the colored segments represent how the different positions are related. The first row represents the final result of the trainable parameterized matrix, and the second row represents the learned motion representation.

are more critical for practical deployment, are significant. Secondly, compared to the method (b), (d), and (g), our method is more efficient and effective. Compared to methods (c) and (e), our method has an accuracy lead at similar training and test costs. Overall, our method achieves a good balance between performance and efficiency and has obvious advantages over the most advanced methods.

4) Visualization of the Relational Representations: To demonstrate the ability of our approach to constructing a spatial model for relational patterns, we visualize the two spatial modeling strategies respectively. We set a threshold to filter out small correlation patterns. The results are shown in Figure 4. First, we select the parameter matrices from the three random spatiotemporal graph convolution blocks. The results are shown in the first row of Figure 4, i.e. (a)(b)(c). Adding the Gaussian distribution-based sparse representation can significantly strengthen the key relationships. The final training results also confirm that this parameterization enables

TABLE V
THE RESULT OBTAINED BY THE MULTI-STREAM FUSION

Methods	Modalities	Acc (%)
MCTM-Net	J	90.6
MCTM-Net _{En}	J	91.0
MCTM-Net _{En}	J & B	92.4
MCTM-Net _{En}	J & B & JM & BM	92.8

learning a more flexible relational representation. This is reflected in different layers.

Second, since the relational representation of the motion-based part depends on channel dimension, we select three representations of different channel dimensions of the fixed layer for visualization. The result is shown in the second row of Figure 4. The results show that the areas exhibiting movement have been strengthened. For example, (d) and (e) correspond to the motion enhancement reflecting the interaction of the hand and the leg, while (f) also shows the interaction of additional movement areas. At the same

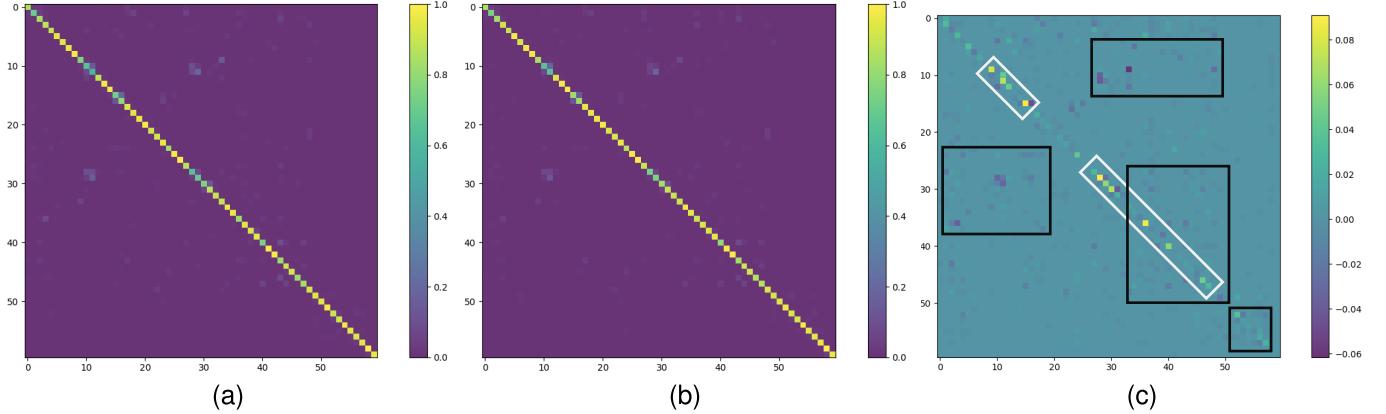


Fig. 5. **Visualization of the Confusion Matrix.** (a): confusion matrix of AGCN. (b): confusion matrix of MCTM. (c): the difference of subtracting (a) from (b).

TABLE VI

A COMPARISON OF OUR APPROACH WITH THE STATE OF THE ART METHODS ON NTU-RGB+D 60&120 AND NORTHWESTERN-UCLA

Methods	NTU-RGB+D 60		NTU-RGB+D 120		Northwestern-UCLA
	X-Sub	X-View	X-Sub	X-Set	
Actionlet ensemble [43]	-	-	-	-	76.0
Lie Group [23]	50.1	52.8	-	-	74.2
H-RNN [27]	59.1	64.0	-	-	78.5
PA-LSTM [8]	62.9	70.3	25.5	26.3	-
SkeMotion [44]	-	-	67.7	66.9	-
STA-LSTM [45]	73.4	81.2	-	-	-
Visualize CNN [46]	76.0	82.6	-	-	-
VA-LSTM [47]	79.2	87.7	-	-	-
Ensemble TS-LSTM [48]	-	-	-	-	89.2
Ind-RNN [49]	81.8	88.0	-	-	-
DPRL [50]	83.5	89.8	-	-	-
HCN [51]	86.5	91.1	-	-	-
Multi CNN+RotClips [52]	-	-	62.2	61.8	-
TSRJI [53]	-	-	67.9	62.8	-
VA-fusion [54]	89.4	95.0	-	-	-
Hybrid [55]	79.4	84.1	-	-	93.1
GeomNet [56]	93.6	96.3	86.5	87.6	-
ST-GCN [13]	81.5	88.3	-	-	-
AGCN [14]	88.5	95.1	82.9	84.9	-
AS-GCN [57]	86.8	94.2	-	-	-
SGN [40]	89.0	94.5	79.2	81.5	93.3
AGC-LSTM [36]	89.2	95.0	-	-	-
DGNN [15]	89.9	96.1	-	-	-
SGCN [58]	89.4	95.7	-	-	-
MV-HPGNet [59]	-	-	83.5	85.4	93.3
MV-IGNet [59]	-	-	83.9	85.6	93.1
Shift-GCN [16]	90.7	96.5	85.9	87.6	94.6
DC-GCN+ADG [60]	90.8	96.6	86.5	88.1	95.3
MS-G3D [17]	91.5	96.2	86.9	88.4	-
Dynamic GCN [61]	91.5	96.0	87.3	88.6	-
GCN-HCRF [62]	90.0	95.5	-	-	96.3
(P+C)Net [63]	86.1	93.5	-	-	-
LAGA-Net [28]	87.1	93.2	81.0	82.2	95.9
Graph2Net [18]	90.1	96.0	86.0	87.6	95.3
Sym-GNN [35]	90.1	96.4	-	-	-
MSIN [64]	91.5	96.5	88.2	89.4	-
CTR-GCN [19]	92.4	96.8	88.9	90.6	96.5
MCTM-Net (ours)	92.8	96.8	89.3	91.0	97.2

time, this also demonstrates that this way of extending to the channel dimension enhances the representation ability of the model.

5) *Visualization of the Confusion Matrix:* More insight about the action recognition results are presented in Figure 5. Different colors indicate the proportion of each correctly or incorrectly classified category. From (a) and (b), we can see that both models achieve a good performance overall. However, by more careful observation, we note that our

MCTM method significantly reduces the misclassification rate and improves the classification accuracy. This phenomenon is more evident in (c). Specifically, in the white solid line box, it can be observed that the correct classification rate has increased by more than 6%. At the same time, the results indicated by the black solid line box show that the probability of misclassification has been reduced significantly.

6) *Multi-Stream Fusion:* Finally, to mitigate the limitations of the model and to improve accuracy further, we use

TABLE VII
A COMPARISON OF OUR APPROACH WITH THE STATE OF THE ART METHODS ON UWA3D MULTIVIEW ACTIVITY II DATASET

Training views	$V_1 \& V_2$		$V_1 \& V_3$		$V_1 \& V_4$		$V_2 \& V_3$		$V_2 \& V_4$		$V_3 \& V_4$		Mean
Test view	V_3	V_4	V_2	V_4	V_2	V_3	V_1	V_4	V_1	V_3	V_1	V_2	
HOJ3D [65]	15.3	28.2	17.3	27.0	14.6	13.4	15.0	12.9	22.1	13.5	20.3	12.7	17.7
Actionlet ensemble [43]	45.0	40.4	35.1	36.9	34.7	36.0	49.5	29.3	57.1	35.4	49.0	29.3	39.8
Lie Group [23]	49.4	42.8	34.6	39.7	38.1	44.8	53.3	33.5	53.6	41.2	56.7	32.6	43.4
Enhanced skeleton visualization [46]	66.4	68.1	56.8	66.1	58.8	66.2	74.2	67.0	76.9	64.8	72.2	54.0	66.0
Ensemble TS-LSTM v2 [48]	72.1	79.1	74.0	77.6	75.6	70.1	79.6	79.9	83.9	66.1	79.2	69.7	75.6
ShiftGCN++ [66]	74.5	86.6	76.8	85.0	75.2	73.7	86.3	82.7	85.5	73.3	84.7	74.0	79.9
MCTM-Net (ours)	76.1	86.0	78.0	84.6	76.8	77.3	86.3	81.9	85.1	74.9	87.1	74.4	80.7

multi-stream features, including joints, bones, and their motions. Evidently, the time overhead for training and testing is bound to increase. For the evaluation, we adopt the method commonly used in action recognition [32], [33], which involves selecting half of the frames to train another set of parameters for the ensemble. The results show that this can achieve competitive performance. The final experimental results are shown in Table V. It can be seen that our model achieves a consistent improvement in multi-stream fusion. It is worth noting that, shifting the windows affecting the action modeling with the motion information used as a direct input helps to enhance the features with large motion differences.

D. A Comparison With the State of Art

Finally, we compare the proposed solution with state-of-the-art mainstream methods on NTU-RGB+D 60&120, Northwestern-UCLA, and UWA3D Multiview Activity II datasets. The experimental results are shown in Tables VI and Table VII. We have selected the most representative recent non-GCN methods. These methods often refer to some commonly used methods in RGB video-based action recognition tasks and are specific to the characteristics of the skeleton. The results are shown in the upper part of the table. Note, that our method has a significant lead over almost all the methods. Compared with the best method, GeomNet [56], our method exhibits a slight weakness on the NTU-RGB+D 60 dataset but has achieved a clear lead (89.3% vs 86.5%; 91% vs 87.6%) on the NTU-RGB+D 120 dataset.

Second, we compare with the methods specifically related to spatiotemporal graph convolution modeling, which represents the mainstream feature modeling scheme in this field. As shown in the lower part of the table, our method outperforms all mainstream methods, except CTR-GCN [19], by a large margin. For CTR-GCN, we also have a particular advantage in accuracy, and as shown in Table V, our method's training and testing overhead is much less than that of CTR-GCN. Overall, the final experimental results prove that our method is superior to many mainstream methods based on different paradigms. Last but not least, our findings are also confirmed by the experimental results of our method on the UWA3D Multiview Activity II dataset. We evaluated our method in all possible combinations. The average performance figures presented in Table VII show that our method is the best in most combinations. The final average accuracy also surpasses other methods, which proves the merit of our strategy. Specifically, benefiting from the powerful representation ability of deep networks, the proposed method shows an improvement of more than 30% compared with traditional methods; Compared

with the methods based on deep network structures, our graph network also achieves a significant lead; Compared with the recent graph network method ShiftGCN++ [66], our method, which simultaneously models spatiotemporal features from multiple perspectives, produces experimental results which demonstrate its superiority.

V. CONCLUSION

We proposed a systematic modeling method for skeleton-based action recognition. Building on the mainstream graph convolution structure, we complement this basic framework with multiple innovative enhancements. Specifically, we replace the fixed adjacency representation with a parameterized representation for spatial modeling, which offers greater flexibility. More importantly, we complement the framewise spatial modeling with a motion relation model. For temporal modeling, we use an efficient multi-focus graph convolution structure. With the above innovations, we capitalize on the complementarity of informative clues and significantly improve the performance of the model. Thanks to these innovations, our method achieves state-of-the-art performance on multiple benchmarks.

REFERENCES

- [1] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2014, pp. 568–576.
- [2] Y. Song, J. Tang, F. Liu, and S. Yan, "Body surface context: A new robust feature for action recognition from depth videos," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 6, pp. 952–964, Jun. 2014.
- [3] J. Lu, D. Batra, D. Parikh, and S. Lee, "ViLBERT: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks," 2019, *arXiv:1908.02265*.
- [4] L. Shi, Y. Zhang, J. Cheng, and H. Lu, "Skeleton-based action recognition with multi-stream adaptive graph convolutional networks," *IEEE Trans. Image Process.*, vol. 29, pp. 9532–9545, 2020.
- [5] F. Xiao, Y. J. Lee, K. Grauman, J. Malik, and C. Feichtenhofer, "Audiovisual SlowFast networks for video recognition," 2020, *arXiv:2001.08740*.
- [6] H. Li, X.-J. Wu, and J. Kittler, "MDLatLRR: A novel decomposition method for infrared and visible image fusion," *IEEE Trans. Image Process.*, vol. 29, pp. 4733–4746, 2020.
- [7] J. Wang, X. Nie, Y. Xia, Y. Wu, and S.-C. Zhu, "Cross-view action modeling, learning, and recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 2649–2656.
- [8] A. Shahroudy, J. Liu, T.-T. Ng, and G. Wang, "NTU RGB+D: A large scale dataset for 3D human activity analysis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1010–1019.
- [9] J. Liu, A. Shahroudy, M. Perez, G. Wang, L.-Y. Duan, and A. C. Kot, "NTU RGB+D 120: A large-scale benchmark for 3D human activity understanding," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 10, pp. 2684–2701, Oct. 2019.
- [10] G. Papadopoulos, A. Axenopoulos, and P. Daras, "Real-time skeleton-tracking-based human action recognition using Kinect data," in *Proc. Int. Conf. Multimedia Model.* Cham, Switzerland: Springer, 2014, pp. 473–483.

- [11] Y. Hbali, S. Hbali, L. Ballihi, and M. Sadgal, "Skeleton-based human activity recognition for elderly monitoring systems," *IET Comput. Vis.*, vol. 12, no. 1, pp. 16–26, Feb. 2018.
- [12] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," 2016, *arXiv:1609.02907*.
- [13] S. Yan, Y. Xiong, and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition," in *Proc. AAAI Conf. Artif. Intell. (AAAI)*, 2018, pp. 7444–7452.
- [14] L. Shi, Y. Zhang, J. Cheng, and H. Lu, "Two-stream adaptive graph convolutional networks for skeleton-based action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 12026–12035.
- [15] L. Shi, Y. Zhang, J. Cheng, and H. Lu, "Skeleton-based action recognition with directed graph neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 7912–7921.
- [16] K. Cheng, Y. Zhang, X. He, W. Chen, J. Cheng, and H. Lu, "Skeleton-based action recognition with shift graph convolutional network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 183–192.
- [17] Z. Liu, H. Zhang, Z. Chen, Z. Wang, and W. Ouyang, "Disentangling and unifying graph convolutions for skeleton-based action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 143–152.
- [18] C. Wu, X.-J. Wu, and J. Kittler, "Graph2Net: Perceptually-enriched graph learning for skeleton-based action recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 4, pp. 2120–2132, Apr. 2022.
- [19] Y. Chen, Z. Zhang, C. Yuan, B. Li, Y. Deng, and W. Hu, "Channel-wise topology refinement graph convolution for skeleton-based action recognition," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 13359–13368.
- [20] H. Rahmani, A. Mahmood, D. Huynh, and A. Mian, "Histogram of oriented principal components for cross-view action recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 12, pp. 2430–2443, Dec. 2016.
- [21] S. Zhang, H. Tong, J. Xu, and R. Maciejewski, "Graph convolutional networks: A comprehensive review," *Comput. Social Netw.*, vol. 6, no. 1, pp. 1–23, Dec. 2019.
- [22] H.-S. Chen, H.-T. Chen, Y.-W. Chen, and S.-Y. Lee, "Human action recognition using star skeleton," in *Proc. 4th ACM Int. Workshop Video Survill. Sensor Netw.*, Oct. 2006, pp. 171–178.
- [23] R. Vemulapalli, F. Arrate, and R. Chellappa, "Human action recognition by representing 3D skeletons as points in a lie group," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 588–595.
- [24] Y. Du, Y. Fu, and L. Wang, "Skeleton based action recognition with convolutional neural network," in *Proc. 3rd IAPR Asian Conf. Pattern Recognit. (ACPR)*, Nov. 2015, pp. 579–583.
- [25] T. Mikolov, M. Karafiat, L. Burget, J. Černocký, and S. Khudanpur, "Recurrent neural network based language model," in *Proc. 11th Annu. Conf. Int. Speech Commun. Assoc.*, vol. 2, 2010, pp. 1045–1048.
- [26] S. Venugopalan, H. Xu, J. Donahue, M. Rohrbach, R. Mooney, and K. Saenko, "Translating videos to natural language using deep recurrent neural networks," 2014, *arXiv:1412.4729*.
- [27] Y. Du, W. Wang, and L. Wang, "Hierarchical recurrent neural network for skeleton based action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1110–1118.
- [28] R. Xia, Y. Li, and W. Luo, "LAGA-Net: Local-and-global attention network for skeleton based action recognition," *IEEE Trans. Multimedia*, vol. 24, pp. 2648–2661, 2022.
- [29] J. Lee, I. Lee, and J. Kang, "Self-attention graph pooling," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 3734–3743.
- [30] L. Shen, M. Sun, Q. Li, B. Li, Z. Pan, and J. Lei, "Multiscale temporal self-attention and dynamical graph convolution hybrid network for EEG-based stereogram recognition," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 30, pp. 1191–1202, 2022.
- [31] I. Sutskever, J. Martens, G. Dahl, and G. Hinton, "On the importance of initialization and momentum in deep learning," in *Proc. Int. Conf. Mach. Learn.*, 2013, pp. 1139–1147.
- [32] M. Zolfaghari, K. Singh, and T. Brox, "ECO: Efficient convolutional network for online video understanding," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*. Cham, Switzerland: Springer, 2018, pp. 695–712.
- [33] J. Lin, C. Gan, and S. Han, "TSM: Temporal shift module for efficient video understanding," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 7083–7093.
- [34] C. Feichtenhofer, H. Fan, J. Malik, and K. He, "SlowFast networks for video recognition," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6202–6211.
- [35] M. Li, S. Chen, X. Chen, Y. Zhang, Y. Wang, and Q. Tian, "Symbiotic graph neural networks for 3D skeleton-based human action recognition and motion prediction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 6, pp. 3316–3333, Jun. 2022.
- [36] C. Si, W. Chen, W. Wang, L. Wang, and T. Tan, "An attention enhanced graph convolutional LSTM network for skeleton-based action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1227–1236.
- [37] A. Paszke et al., "Automatic differentiation in PyTorch," in *Proc. Adv. Neural Inf. Process. Syst. Workshops (NeurIPS)*, 2017, pp. 1–4.
- [38] P.-T. de Boer, D. P. Kroese, S. Mannor, and R. Y. Rubinstein, "A tutorial on the cross-entropy method," *Ann. Oper. Res.*, vol. 134, no. 1, pp. 19–67, 2005.
- [39] N. Qian, "On the momentum term in gradient descent learning algorithms," *Neural Netw.*, vol. 12, no. 1, pp. 145–151, 1999.
- [40] P. Zhang, C. Lan, W. Zeng, J. Xing, J. Xue, and N. Zheng, "Semantics-guided neural networks for efficient skeleton-based human action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 1112–1121.
- [41] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2818–2826.
- [42] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, "ShuffleNet V2: Practical guidelines for efficient CNN architecture design," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 116–131.
- [43] J. Wang, Z. Liu, Y. Wu, and J. Yuan, "Learning actionlet ensemble for 3D human action recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 5, pp. 914–927, May 2014.
- [44] J. Liu, G. Wang, P. Hu, L.-Y. Duan, and A. C. Kot, "Global context-aware attention LSTM networks for 3D action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1647–1656.
- [45] S. Song, C. Lan, J. Xing, W. Zeng, and J. Liu, "An end-to-end spatio-temporal attention model for human action recognition from skeleton data," in *Proc. AAAI Conf. Artif. Intell. (AAAI)*, 2017, pp. 4263–4270.
- [46] M. Liu, H. Liu, and C. Chen, "Enhanced skeleton visualization for view invariant human action recognition," *Pattern Recognit.*, vol. 68, pp. 346–362, Aug. 2017.
- [47] P. Zhang, C. Lan, J. Xing, W. Zeng, J. Xue, and N. Zheng, "View adaptive recurrent neural networks for high performance human action recognition from skeleton data," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2117–2126.
- [48] I. Lee, D. Kim, S. Kang, and S. Lee, "Ensemble deep learning for skeleton-based action recognition using temporal sliding LSTM networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 1012–1020.
- [49] S. Li, W. Li, C. Cook, C. Zhu, and Y. Gao, "Independently recurrent neural network (IndRNN): Building a longer and deeper RNN," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5457–5466.
- [50] Y. Tang, Y. Tian, J. Lu, P. Li, and J. Zhou, "Deep progressive reinforcement learning for skeleton-based action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 5323–5332.
- [51] C. Li, Q. Zhong, D. Xie, and S. Pu, "Co-occurrence feature learning from skeleton data for action recognition and detection with hierarchical aggregation," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, Jul. 2018, pp. 786–792.
- [52] Q. Ke, M. Bennamoun, S. An, F. Sohel, and F. Boussaid, "Learning clip representations for skeleton-based 3D action recognition," *IEEE Trans. Image Process.*, vol. 27, no. 6, pp. 2842–2855, Jun. 2018.
- [53] C. Caetano, F. Brémont, and W. R. Schwartz, "Skeleton image representation for 3D action recognition based on tree structure and reference joints," in *Proc. 32nd SIBGRAPI Conf. Graph., Patterns Images (SIBGRAPI)*, Oct. 2019, pp. 16–23.
- [54] P. Zhang, C. Lan, J. Xing, W. Zeng, J. Xue, and N. Zheng, "View adaptive neural networks for high performance skeleton-based human action recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 8, pp. 1963–1978, Aug. 2019.
- [55] C. Dhiman and D. K. Vishwakarma, "View-invariant deep architecture for human action recognition using two-stream motion and shape temporal dynamics," *IEEE Trans. Image Process.*, vol. 29, pp. 3835–3844, 2020.

- [56] X. S. Nguyen, "GeomNet: A neural network based on Riemannian geometries of SPD matrix space and Cholesky space for 3D skeleton-based interaction recognition," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 13379–13389.
- [57] M. Li, S. Chen, X. Chen, Y. Zhang, Y. Wang, and Q. Tian, "Actional-structural graph convolutional networks for skeleton-based action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3595–3603.
- [58] W. Peng, X. Hong, H. Chen, and G. Zhao, "Learning graph convolutional network for skeleton-based human action recognition by neural searching," in *Proc. AAAI Conf. Artif. Intell. (AAAI)*, 2020, pp. 2669–2676.
- [59] M. Wang, B. Ni, and X. Yang, "Learning multi-view interactional skeleton graph for action recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Oct. 21, 2020, doi: [10.1109/TPAMI.2020.3032738](https://doi.org/10.1109/TPAMI.2020.3032738).
- [60] K. Cheng, Y. Zhang, C. Cao, L. Shi, J. Cheng, and H. Lu, "Decoupling GCN with dropgraph module for skeleton-based action recognition," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2020, pp. 536–553.
- [61] F. Ye, S. Pu, Q. Zhong, C. Li, D. Xie, and H. Tang, "Dynamic GCN: Context-enriched topology learning for skeleton-based action recognition," in *Proc. 28th ACM Int. Conf. Multimedia*, Oct. 2020, pp. 55–63.
- [62] K. Liu, L. Gao, N. M. Khan, L. Qi, and L. Guan, "A multi-stream graph convolutional networks-hidden conditional random field model for skeleton-based action recognition," *IEEE Trans. Multimedia*, vol. 23, pp. 64–76, 2021.
- [63] S. Miao, Y. Hou, Z. Gao, M. Xu, and W. Li, "A central difference graph convolutional operator for skeleton-based action recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 7, pp. 4893–4899, Jul. 2022.
- [64] H. Wang, B. Yu, J. Li, L. Zhang, and D. Chen, "Multi-stream interaction networks for human action recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 5, pp. 3050–3060, May 2022.
- [65] L. Xia, C.-C. Chen, and J. K. Aggarwal, "View invariant human action recognition using histograms of 3D joints," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Workshops*, Jul. 2012, pp. 20–27.
- [66] K. Cheng, Y. Zhang, X. He, J. Cheng, and H. Lu, "Extremely lightweight skeleton-based action recognition with ShiftGCN++," *IEEE Trans. Image Process.*, vol. 30, pp. 7333–7348, 2021.



Cong Wu received the B.Sc. degree from the School of Science, Jiangnan University, China, in 2018, where he is currently pursuing the Ph.D. degree with the Jiangsu Provincial Engineering Laboratory of Pattern Recognition and Computational Intelligence. His research interests include video analysis, action recognition, and human pose estimation.

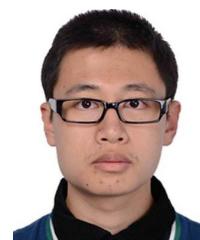


Xiao-Jun Wu received the B.S. degree in mathematics from Nanjing Normal University, Nanjing, China, in 1991, and the M.S. and Ph.D. degrees in pattern recognition and intelligent system from the Nanjing University of Science and Technology, Nanjing, in 1996 and 2002, respectively. He was a fellow of the International Institute for Software Technology (UNU/IIST), United Nations University, from 1999 to 2000. From 1996 to 2006, he taught at the School of Electronics and Information, Jiangsu University of Science and Technology, where he was an exceptionally promoted as a Professor. He joined Jiangnan University, in 2006, where he is a Distinguished Professor at the School of Artificial Intelligence and Computer Science. He was a Visiting Post-Doctoral Researcher at the Centre for Vision, Speech, and Signal Processing (CVSSP), University of Surrey, U.K., from 2003 to 2004, under the supervision of Prof. Josef Kittler. He has published more than 400 papers in his fields of research. His current research interests include pattern recognition, computer vision, fuzzy systems, neural networks, and intelligent systems. He is a fellow of IAPR. He is in

charge of Wuxi IEEE Smart Cities Pioneering Program and IEEE Smart Cities Initiative. He won the Most Outstanding Postgraduate Award by the Nanjing University of Science and Technology. He also won different awards, including the international award, the national award, and the provincial award for his research achievements. He was an Associate Editor of *International Journal of Computer Mathematics*. He is currently a Review Editor of *Frontiers in Neurorobotics*; an Editor of *Journal of Algorithm and Computational Technology*; an Associate Editor of *Computers in Biology and Medicine*, *SN Computer Science*, and *Pattern Recognition Letters*; and the Editor-in-Chief of *Artificial Intelligence Advances*.



Tianyang Xu received the B.Sc. degree in electronic science and engineering from Nanjing University, Nanjing, China, in 2011, and the Ph.D. degree from the School of Artificial Intelligence and Computer Science, Jiangnan University, Wuxi, China, in 2019. He is currently an Associate Professor at the School of Artificial Intelligence and Computer Science, Jiangnan University. He has published several scientific papers, including *IJCV*, *ICCV*, *IEEE TRANSACTIONS ON IMAGE PROCESSING*, *IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY*, *IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING*, *IEEE TRANSACTIONS ON MULTIMEDIA*, and *IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY*. His research interests include visual tracking and deep learning. He achieved top one performance in competitions, including the VOT2018 public dataset (ECCV18), VOT2020 RGBT challenge (ECCV20), Anti-UAV challenge (CVPR20), and Multi-Modal Video Reasoning and Analyzing Competition (ICCV21).



Zhongwei Shen received the B.Eng. degree in computer science from the Jiangsu University of Science and Technology, Zhenjiang, China, in 2013, and the M.S. degree in image processing and multi-media from INP-ENSEEIHT, Toulouse, France, in 2015. He is currently pursuing the Ph.D. with the School of Internet of Things Engineering, Jiangnan University, Wuxi, China. His research interests include computer vision and machine learning.



Josef Kittler (Life Member, IEEE) received the B.A., Ph.D., and D.Sc. degrees from the University of Cambridge, in 1971, 1974, and 1991, respectively. He is currently a Distinguished Professor of machine intelligence at the Centre for Vision, Speech and Signal Processing, University of Surrey, Guildford, U.K. He conducts research on biometrics, video and image database retrieval, medical image analysis, and cognitive vision. He published the text book *Pattern Recognition: A Statistical Approach* and over 700 scientific papers. His publications have been cited over 60,000 times (Google Scholar). He is a Series Editor of *Lecture Notes in Computer Science* (Springer). He currently serves on the Editorial Boards for *Pattern Recognition Letters*, *International Journal of Pattern Recognition and Artificial Intelligence*, and *Pattern Analysis and Applications*. He has also served as a member of the Editorial Board for the *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE* from 1982 to 1985. He served on the Governing Board for the International Association for Pattern Recognition (IAPR), as one of the two British representatives, from 1982 to 2005, and the President of the IAPR, from 1994 to 1996.