SiT-MLP: A Simple MLP With Point-Wise Topology Feature Learning for Skeleton-Based Action Recognition

Shaojie Zhang, Jianqin Yin, *Member, IEEE*, Yonghao Dang, and Jiajun Fu

***Abstract*— Graph convolution networks (GCNs) have achieved remarkable performance in skeleton-based action recognition. However, previous GCN-based methods rely on elaborate human priors excessively and construct complex feature aggregation mechanisms, which limits the generalizability and effectiveness of networks. To solve these problems, we propose a novel Spatial Topology Gating Unit (STGU), an MLP-based variant without extra priors, to capture the co-occurrence topology features that encode the spatial dependency across all joints. In STGU, to learn the point-wise topology features, a new gate-based feature interaction mechanism is introduced to activate the features point-to-point by the attention map generated from the input sample. Based on the STGU, we propose the first MLP-based model, SiT-MLP, for skeleton-based action recognition in this work. Compared with previous methods on three large-scale datasets, SiT-MLP achieves competitive performance. In addition, SiT-MLP reduces the parameters significantly with favorable results. The code will be available at https://github.com/BUPTSJZhang/SiT-MLP.**

***Index Terms*— Human action recognition, skeleton, MLP, attention, spatial-temporal optimization.**

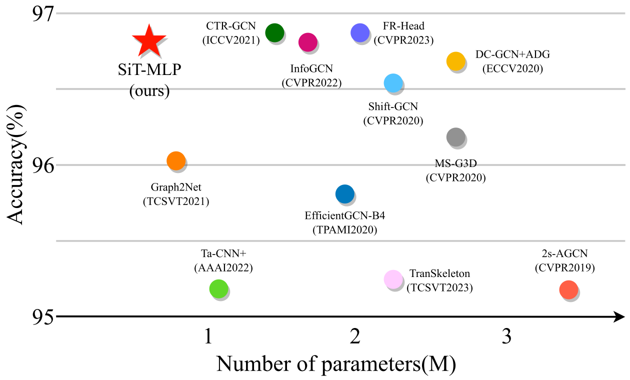


Fig. 1. Comparsion of performance and parameter size on X-sub benchmark of NTU RGB+D 60 dataset. We report the accuracy as performance on the vertical dimension. The closer to the top-left, the better. Our method (SiT-MLP, in red) archives the highest performance with the fewest parameters.

I. Introduction

HUMAN action recognition is an essential task in computer vision. It can enhance robot intelligence and improve the communication efficiency of human-computer interaction [1]. In recent years, thanks to the development of depth sensors [2] and human pose estimation algorithms [3], it is easy to capture the sequence of human skeletons. Due to the robustness of human skeletons to background clutter and illumination changes, skeleton-based action recognition has attracted much interest [4].

Yan et al. [5] first propose to build a human topology graph, treating joints and their explicit connections as nodes and edges respectively, and apply Graph Convolutional Network (GCN) [6] on such a predefined graph to capture spatial-temporal co-occurrence features. The predefined graph introduces priors of human natural connections, and GCN can aggregate the neighboring joints' information and update the current joint's features. Since then, GCN-based methods [5], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17], [18] have become a research paradigm for skeleton-based action recognition.

However, recent GCN-based approaches [9], [13], [15], [17], [19], [20], [21] tend to construct complex feature aggregation methods to improve performance. As shown in Fig. 1, the improved performance has also led to an increase in parameters. These approaches are always heavyweight and require additional elaborate priors. The vast number of parameters of the complex feature aggregations usually make the network not efficient. Moreover, the priors are related to the order of labeled joints and their physical connections. The introduction of the priors makes their network difficult to modify and generalize. Thus, a question naturally arises: "*Can we tackle the skeleton-based action recognition without any priors and complex aggregations?*"

To answer this question, we first review why GCN can achieve such success in skeleton-based action recognition. In most previous methods, the topology connections are set as learnable to break the limitations of the static topology attention and model the implicit connections. As shown in Fig. 2, the final optimized matrices in [15] tend to be the global

Manuscript received 23 February 2024; accepted 2 April 2024. Date of publication 9 April 2024; date of current version 30 September 2024. This work was supported in part by the National Natural Science Foundation of China under Grant 62173045, in part by the Natural Science Foundation of Hainan Province under Grant 622RC675, and in part by the Fundamental Research Funds for the Central Universities under Grant 2020XD-A04-3. This article was recommended by Associate Editor V. Stankovic. *(Corresponding author: Jianqin Yin.)*

The authors are with the School of Artificial Intelligence, Beijing University of Posts and Telecommunications, Beijing 100876, China (e-mail: zsj@bupt.edu.cn; jqyin@bupt.edu.cn; dyh2018@bupt.edu.cn; JaakkoFu@bupt.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCSVT.2024.3386553.

Digital Object Identifier 10.1109/TCSVT.2024.3386553

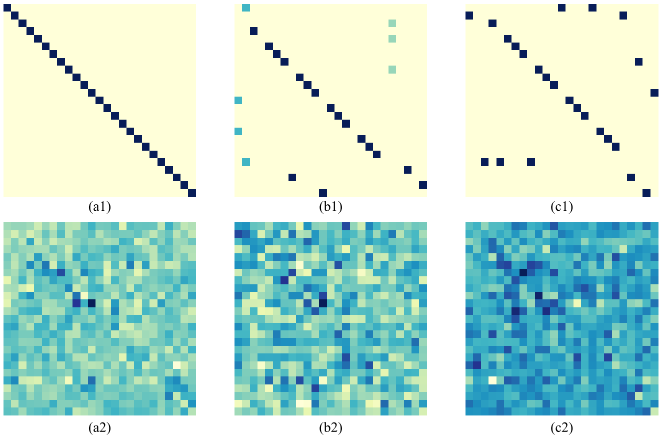


Fig. 2. The comparison between the initialized normalized adjacency matrices and the final optimized adjacency matrices in the previous method [15]. The letters a,b, and c denote the self-link matrix, the inward connections, and the outward connections matrix, respectively. The numbers 1 and 2 indicate the initialized adjacency matrix and the final adjacency matrix, respectively.

relationship between any two joints. In other words, the reason for the favorable performance of GCN-based approaches may not be the priors but the modeling of the relationships between any two joints. Thus, we think other structures, for example, **the simple MLP can also model such global relationships**.

Moreover, another reason we believe the SOTA GCN-based approaches have been able to achieve such success is the inclusion of sample-specific attention modeling based on feature interaction mechanisms. As demonstrated in Fig. 3 (a), in some previous SOTA methods [5], [14], the feature transformation module and the topology modeling module are independent. The learned topology attention is sample-generic. The static networks and shared topology connections limit the capability of the model. To break these limitations and learn sample-specific topology dependency, some feature interaction mechanism was adopted in other works [9], [15]. The interaction between features can allow the network to adjust the output according to the input samples dynamically. As shown in Fig. 3 (b), [15] model sample-specific attention according to the input features. The attention is denoted as affinity matrices, and this sample-specific topology attention is non-shared in the channel dimension. The adaptation of this channel-wise attention improves the performance of the network significantly.

Although the sample-specific modeling obtained good performance for action recognition, the temporal pooling operations adopted in [15] result in the channel-wise attention shared in the temporal dimension. Human action includes coupled spatial-temporal relationships. Therefore, the modeling of temporal-wise topology attention is necessary. For example, during the process of drinking coffee, the implicit connections between the head joints and the right-hand joints will gradually become stronger. The temporal-wise attention can model topological relationships over time dynamically and channel-wise attention can enhance the distinctiveness of features. The combination of these two kinds of attention can be denoted as point-wise attention, which is non-shared in the channel and temporal dimensions. **The modeling of point-wise sample-specific attention can encourage the network to fully explore the spatial-temporal relations of the input action**. The gating unit [22] can achieve such modeling without complex aggregations and huge parameters.

With the above findings, in this work, we present a simple yet effective topology feature learning model, SiT-MLP, the first model using MLP to address skeleton-based action recognition. As shown in Fig. 3 (c), in SiT-MLP, to model sample-specific spatial correlations, we propose the Spatial Topology Gating Unit(STGU), which is an MLP-baed structure without any priors and introduces a new gate-based feature interaction mechanism. On the one hand, in STGU, features can be activated point-to-point by the generated attention map. The point-wise sample-specific topologies provide independent relationships between any two joints in the temporal and channel dimensions. On the other hand, compared to the self-attention mechanism [23] containing up to 3rd-order interactions (e,g., $q_i k_j v_k$), the gate mechanism containing up to 2nd-order interactions (e.g., $z_i z_j$), reduces the computational consumption significantly.

We summarize our main contributions as follows:

- We show an insight that skeleton-based human action recognition can be modeled simply without elaborate priors. This prior-free characteristic can make the corresponding algorithm generalize easier, avoiding the complex design of the prior in different scenarios.
- We propose the Spatial Topology Gating Unit (STGU) to learn the point-wise sample-specific topology features with simple aggregation.
- Based on STGU, we propose SiT-MLP to tackle skeleton-based action recognition, which falls into the category of MLP-based methods. To the best of our knowledge, SiT-MLP is the first MLP-based model in skeleton-based action recognition.
- Extensive experiments on three large-scale datasets demonstrate the proposed SiT-MLP achieves favorable results against the previous schemes. Moreover, SiT-MLP also reduces the parameters and computing resources significantly.

II. Related Work

A. Skeleton-Based Action Recognition

In early studies, Recurrent neural network (RNN) based methods [24], [25], [26] and Convolution neural network (CNN) based methods [27], [28], [29], [30] are popular choices for solving the skeleton-based action recognition problem. However, the methods mentioned above overlook human topology and the spatial interactions between joints.

1) GCN-Based Methods: To effectively handle the skeleton sequence, GCN, which has a strong ability to extract features from non-Euclidean data, has become a more popular option in this field. ST-GCN [5] first applies GCN to model the joint correlations and significantly boosts the performance of skeleton-based action recognition. However, the inherent topology structure limits the GCN in capturing long-range relations. To break this limitation, MS-G3D [9] and STIGCN [31] adopt multi-scale graph topologies to GCNs

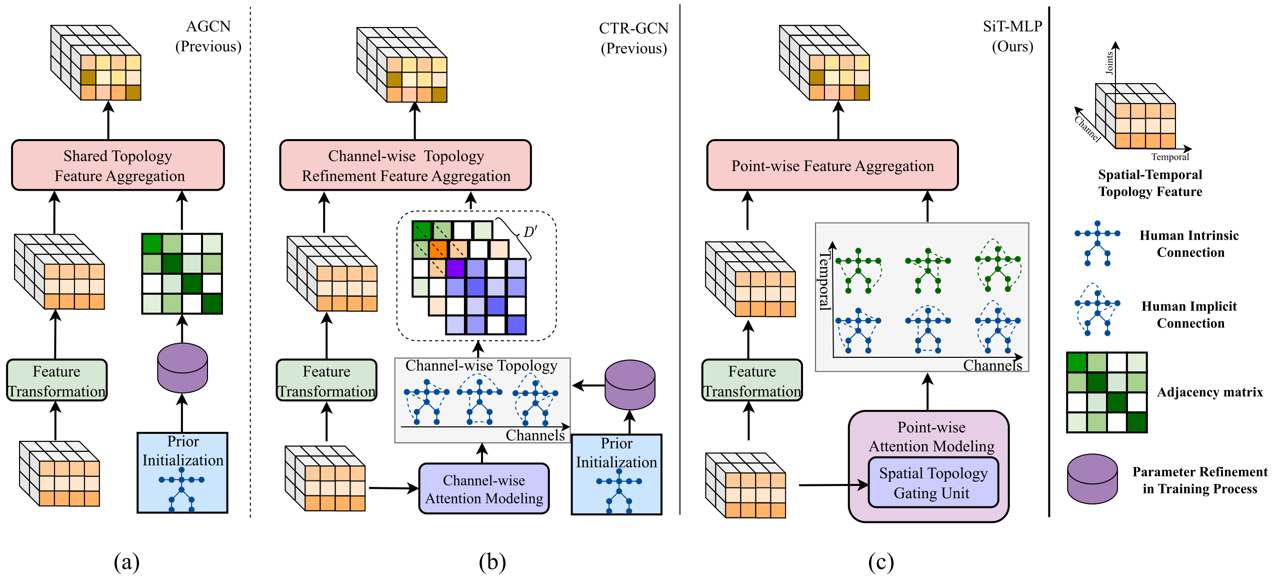

AGCN
(Previous)
Shared Topology Feature Aggregation
Feature Transformation
Prior Initialization
CTR-GCN
(Previous)
Channel-wise Topology Refinement Feature Aggregation
D'
Channel-wise Topology
Channels
Feature Transformation
Channel-wise Attention Modeling
Prior Initialization
SiT-MLP
(Ours)
Point-wise Feature Aggregation
Temporal
Channels
Feature Transformation
Point-wise Attention Modeling
Spatial Topology Gating Unit
Joints
Channel
Temporal
Spatial-Temporal Topology Feature
Human Intrinsic Connection
Human Implicit Connection
Adjacency matrix
Parameter Refinement in Training Process
(a)
(b)
(c)

Fig. 3. The spatial modeling structure of different approaches: (a) the normally sample-generic modeling module; (b) the channel-wise topology refinement modeling module; (c) the proposed Spatial Topology Gating Unit.

to model multi-range joint relations. To further aggregate the global spatial features, most subsequent works [12], [13], [14], [32] have adopted a learnable topology for action recognition.

Considering that the spatial relations that different actions depend on may be different, many works [12], [13], [14], [15], [33], [34] tend to construct sample-specific topology connections. SGN [12] enhances topology learning with the self-attention mechanism, which models correlation between two joints given corresponding features. CTR-GCN [15] learns different topologies dynamically and aggregates joint features in different channels effectively for skeleton-based action recognition. LKA-GCN [34] employs a skeleton large kernel attention operator, which can enlarge the receptive field and improve channel adaptability. DD-GCN [35] constructs the directed diffusion graph for action modeling and introduces the activity partition strategy. FR-Head [33] introduces contrastive learning to enhance the sample-specific features learning. CD-JBF-GCN [36] designs a novel correlation-driven joint-bone fusion graph convolutional network to fully explore the latent sample-specific correlation between joints and bones. HD-GCN [37] effectively constructs an HD-Graph by decomposing every joint node into several sets to extract major adjacent and distant edges.

Recently, some hypergraph-based methods [10], [38] have also achieved competitive results. Hyper-GNN [10] adopts a hypergraph to capture both spatial-temporal information and high-order dependencies for skeleton-based action recognition. SHGCN [38] adaptively chooses the optimal hypergraph structure to represent skeleton actions instead of using a fixed graph structure or predefined hypergraph structure.

2) Transformer-Based Methods: Attempts have been made recently to solve this problem using Transformers. They mainly focused on handling the challenge brought by the extra temporal dimension. ST-TR [39] adopts a two-stream model consisting of spatial and temporal self-attention for modeling intra- and inter-frame correlations, respectively. GAT [40] adapted transformer and graph-aware masks to model the spatial relationship between any joint. DSTA [32] employed a Transformer to model the spatial and temporal dimensions alternately. TranSkeleton [41] proposes a novel partition-aggregation temporal transformer, which works with hierarchical temporal partition and aggregation, and can capture both long-range dependencies and subtle temporal structures effectively. However, there is still a performance gap between GCN-based methods and Transformer-based methods. The reasons for this gap may be the two intrinsic factors: (1) transformer-based methods maintain the sequence length throughout the model, which leads to huge redundancy as the input sequences are generally long. The redundant information carried by the attention matrix, which increases with the length of the sequence, prevents the network from focusing on the key to the current action. (2) The absence of sample-generic modeling makes it different to capture the common features of input sequences, especially when trained on limited-scale skeleton datasets.

In summary, both GCN-based methods and Transformer-based methods improve the performance by contrasting complex aggregation mechanisms, leading to the heavyweight of the networks.

B. Vision MLP

Inspired by the successful adaptation of the Transformer [23] in the computer vision field [42], [43], many works have researched the intrinsic mechanism of self-attention. On the one hand, some works [44], [45], [46], [47], [48] explore the feasibility of MLP for the global modeling of the sequences. Mixer-MLP [44] and ResMLP [45] adopted an MLP architecture instead of the self-attention mechanism to achieve the cross-token communication. ViP [46] encodes the feature representations along the height and width dimensions with linear projections. SMLP [47] adds a feature extraction along channel dimension and a depthwise convolution in front of each block to enhance the local modeling to ViP. MorphMLP [48] achieves competitive performance with recent SOTA methods, demonstrating that the MLP-like backbone is also suitable for video recognition.

On the other hand, some other studies believe that the dynamic weights brought by the attention map provide a larger parameter space for the network. Thus, to break the limitation of the static parameters, the attention mechanism is introduced to the MLP-based network. GMLP [22] adopts the gating mechanism to the MLP-based architecture to improve the generalization of the network. GSwin [49] combines parameter efficiency and performance with locality and hierarchy in image recognition. VAN [50] indicates that the key characteristic of attention methods is adaptively adjusting output based on the input feature but not the normalized attention map.

Due to the absence of the adaptation of the skeleton data and the modeling of the complex spatiotemporal relations, these classical MLP-based approaches in computer vision can't obtain discriminative representations. Therefore, combining MLP and previous methods in the task of action recognition, we propose the SiT-MLP to tackle the skeleton-based action recognition without priors and complex aggregations.

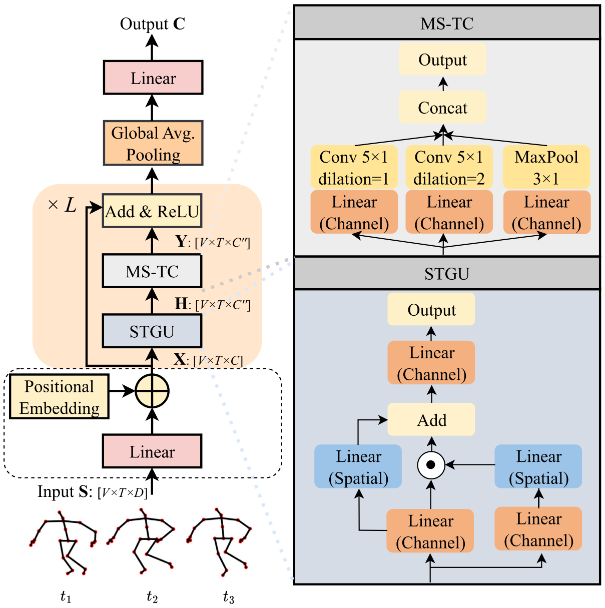


Fig. 4. Model architecture overview and illustration. The embedding block is adopted to retain the positional information. The STGU module captures the spatial dependency, and the MS-TC module aggregates the temporal information. The global average pooling layer is used to aggregate the global spatial-temporal joint information for the final linear classifier.

III. Methods

In this section, we first introduce the architecture of our SiT-MLP. Then we elaborate on our Spatial Topology Gating Unit (STGU), which is the main contribution of our SiT-MLP. Finally, we compare our STGU with other GCN-based methods and analyze why our method works.

A. Model Architecture

We construct a novel but simple network SiT-MLP for skeleton-based human action recognition. As shown in Fig. 4, the entire network consists of an embedding block and five basic blocks, followed by a global average pooling and a linear classifier to predict action categories. It is more difficult for joint spatial-temporal optimization [48]. Therefore, following previous works [5], [14], [15], we place the spatial STGU and temporal MS-TC module in the sequential style in the basic block. Particularly, SiT-MLP is built by stacking several of these basic blocks. The global average pooling layer is used to aggregate the spatial-temporal information for the final linear classifier. Although there are some convolutions in the temporal modeling module, our SiT-MLP is mainly composed of liner layers, falling into the category of MLP-based method.

1) Embedding Block: Our STGU is an MLP-based structure, and MLPs are insensitive to the permutation of positions. To break the permutation equivariance in our SiT-MLP, an embedding block is used to retain positional information.

As shown in Fig. 4, the input sequence $\boldsymbol{S}$ is processed by a learnable embedding, which linearly projects body joint $s_t \in \mathbb{R}^{V\times D}$ through a linear layer to the hidden dimension C. To identify the positional information of the joints, we add pose embeddings (PE) to the projected features. The learnable parameters PE are shared across times. We refer to the output of the learnable linear projection $\boldsymbol{X}^{(0)} = \left\{x_1^{(0)}, \ldots, x_t^{(0)}, \ldots, x_T^{(0)}\right\} \in \mathbb{R}^{V\times T\times C}$ as:

$$x_t^{(0)} = s_t\mathbf{W} + PE, \tag{1}$$

where $x_t^{(0)}, PE \in \mathbb{R}^{V\times C}; \mathbf{W} \in \mathbb{R}^{D\times C}$; t is the time index. Moreover, $\boldsymbol{X}^{(0)}$ is also the input of the first stacked basic block.

2) Spatial Modeling: As shown in Fig 4, in a spatial modeling module, we just use an STGU block to extract correlations between human joints. The details of STGU are demonstrated in section III-B. To improve the diversity of feature space and avoid the risk of overfitting, we follow the previous approaches [49] adopting the multi-head mechanism in [23] in our STGU. The multi-head mechanism enables the network to compute simultaneously without additional consumption. Note that we omit the multi-head operation following for simplicity. Different from previous approaches [5], [9], [14], [15] that required three parallel branches to introduce different human priors, we only adopt one branch for the spatial information extraction without priors. As the above-mentioned decomposition, SiT-MLP can reduce parameters and computational overhead by one-third in the process of spatial modeling.

3) Temporl Modeling: As shown in the grey block in Fig. 4, we adopt the Multi-Scale Temporal Convolution (MS-TC) [9] to aggregate the temporal dependency of the human pose. This module contains three parallel branches with a combination of different kernel sizes and dilation rates, and max-pooling to capture multi-scale temporal information. The extracted features of different branches are concatenated. The MS-TC can capture action characteristics over varying time lengths, which allows the model to understand both short-term rapid movements and long-term, slowly evolving actions.

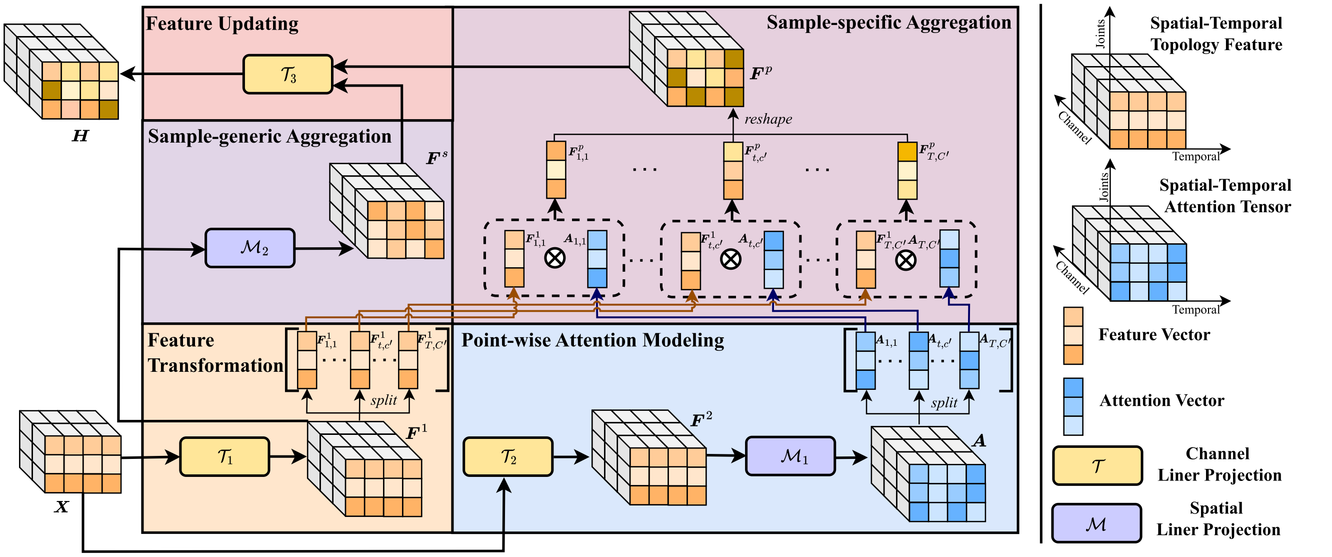


Fig. 5. Framework of the proposed spatial topology gating unit. Feature transformation aims at transforming input features into latent high-dimensional feature space. Point-wise attention modeling builds the entire independent topology attention. Sample-specific aggregation aims to select dynamic features for the current sample. Sample-generic aggregation is for capturing the common feature between all samples. Feature updating aims at fusing and updating the feature after aggregation.

SiT-MLP is constructed by stacking STGU and Temporal Convolution layers alternately as follows:

$$H^{(l)} = \mathrm{STGU}(X^{(l)}), \tag{2}$$

$$Y^{(l)} = \text{MS-TC}(H^{(l)}), \tag{3}$$

$$X^{(l+1)} = \mathrm{ReLU}(Y^{(l)} + X^{(l)}). \tag{4}$$

where l is the index of the stacked basic block. As shown in Fig. 4, for training stability, the standard residual connections are used in each basic block. Moreover, to add non-linearity, a ReLU activation layer is adopted after each basic block after spatial and temporal modeling modules. To manage the temporal dynamics inherent in action recognition, the temporal dimension is halved at the 2-nd and 4-th blocks by strided temporal convolution for sequential frame feature fusion.

B. Spatial Topology Gating Unit

The general framework of our STGU is shown in Fig. 5. Specifically, Our STGU contains five parts; (1) Feature transformation aims at transforming input features into latent high-dimensional feature space; (2) Point-wise attention modeling builds the entire independent topology attention; (3) Sample-specific aggregation aims to select dynamic features for the current sample; (4) Sample-generic aggregation is for capturing the common feature between all samples. (5) The feature updating aims at fusing and updating the feature after aggregation.

Here, we introduce the details of the projection $\mathcal{T}(\cdot)$ and $\mathcal{M}(\cdot)$ in our STGU, which are shown in Fig. 6. Channel linear layer $\mathcal{T}(\cdot)$ acts on channels of feature, allowing the aggregation of the spatial features between different channels and operating on each joint and frame independently. Spatial linear layer $\mathcal{M}(\cdot)$ acts on joints of feature, capturing the spatial across all joints and operating on each channel and frame independently. Next, we will describe each block in detail.

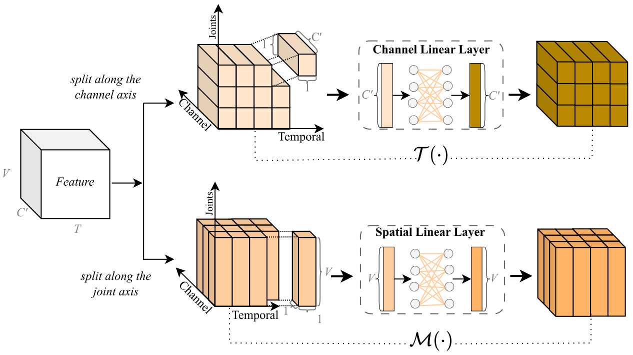


Fig. 6. The details of the projection $\mathcal{T}(.)$ and $\mathcal{M}(.)$ in our STGU. Here, for ease of understanding, we define the input feature $\in \mathbb{R}^{V\times T\times C'}$.

1) Feature Transformation: As in previous methods, the input spatial-temporal feature of our STGU is $X \in \mathbb{R}^{V\times T\times C}$. Feature transformation aims at transforming input features into latent high-dimensional feature space via $\mathcal{T}_1(\cdot)$. We adopt a simple linear transformation here as a linear layer, which can be formulated as

$$F^1 = \mathcal{T}_1(X) = \sigma(X\mathbf{W}_1), \tag{5}$$

where $F^1 \in \mathbb{R}^{V\times T\times C'}$ is the transformed feature and $\mathbf{W}_1 \in \mathbb{R}^{C\times C'}$ weight matrix of the linear channel projection. The details of transformation $\mathcal{T}(\cdot)$ are shown in Fig. 6. The channel liner layer $\mathcal{M}(\cdot)$ can project the features and treat each joint in each frame individually. The linear channel projection can be represented as a 1×1 convolution layer. And σ is an activation function such as GeLU [51].

2) Point-Wise Attention Modeling: As shown in the blue block in Fig. 5, we aim at constructing dynamic parameters, which can provide point-wise attention for the transformed feature. First, we adopt the same operation in feature transformation to project the input to the latent feature space. Then we use a spatial linear layer to encourage communication of each joint. This cross-joint communication can provide a global receptive field of the current pose for the attention map. The attention modeling process can be formulated as

$$\boldsymbol{A} = \mathcal{M}_1(\mathcal{T}_2(\boldsymbol{X})) = \mathbf{W}_A(\sigma(\boldsymbol{X}\mathbf{W}_2)), \tag{6}$$

Here, $\boldsymbol{A} \in \mathbb{R}^{V\times T\times C'}$ is the attention map, which is dynamically parameterized based on the input feature. The details of the spatial linear layer $\mathcal{M}(\cdot)$ are shown in Fig. 6, and $\mathbf{W}_A \in \mathbb{R}^{V\times V}$ is the learnable parameters of the spatial linear layer. There is not any dimension pooling in this process, so our attention map still retains the diversity of features in the channel dimension and the seriality in the temporal dimension. In this way, the point in the attention $\boldsymbol{A}$ can carry the global spatial-temporal information of all the input sequences.

3) Sample-Specific Aggregation: The transformed feature $\boldsymbol{F}^1$ and defined attention map $\boldsymbol{A}$ can be split across the channel dimension and the temporal dimension. In other words, both the spatial-temporal topology feature and spatial-temporal attention can be regarded as a set of vectors, which can be formulated as follows

$$\begin{aligned} \boldsymbol{F}^1 &= \{\boldsymbol{F}^1_{1,1}, \cdots, \boldsymbol{F}^1_{t,c'}, \cdots, \boldsymbol{F}^1_{T,C'}\}, \\ \boldsymbol{A} &= \{\boldsymbol{A}_{1,1}, \cdots, \boldsymbol{A}_{t,c'}, \cdots, \boldsymbol{A}_{T,C'}\}, \end{aligned} \tag{7}$$

Here, t and c' are the indexes of features in the temporal dimension and channel dimension respectively. $\boldsymbol{F}^1_{t,c'}, \in \mathbb{R}^V$ is the feature vector, and $\boldsymbol{A}_{t,c'} \in \mathbb{R}^V$ is the attention vector. Since reshaping was not performed in the previous modeling process, $\boldsymbol{F}$ and $\boldsymbol{A}$ are perfectly aligned in both the channel dimension and temporal dimension. Each topology vector is able to find the corresponding attention vector. Then we can adopt the attention vector to aggregate the feature vector point-to-point.

$$\boldsymbol{F}^p_{t,c'} = \boldsymbol{F}^1_{t,c'} \odot \boldsymbol{A}_{t,c'}, \tag{8}$$

where, $\boldsymbol{F}^p_{t,c'} \in \mathbb{R}^V$ is the topology feature after aggregation. Since the attention $\boldsymbol{A}_{t,c'}$ is generated from the input feature, which will change dynamically depending on different inputs. This gating unit, which is denoted as element-wise multiplication, allows the network to select features of interest based on the generated attention map. In this way, our STGU can adaptively select the discriminative features and ignore noisy responses automatically. Then we reshape the set of feature vectors after sample-specific aggregation into the topology feature.

$$\{\boldsymbol{F}^p_{1,1}, \cdots, \boldsymbol{F}^p_{t,c'}, \cdots, \boldsymbol{F}^p_{T,C'}\} = \boldsymbol{F}^p, \tag{9}$$

Here, $\boldsymbol{F}^p \in \mathbb{R}^{V\times T\times C'}$ is the feature after sample-specific aggregation. This aggregation is capable of capturing complex spatial interactions across joints. In $\boldsymbol{F}^p$, each vector $\boldsymbol{F}^p_{t,c'}$ is obtained by activating $\boldsymbol{F}^1_{t,c'}$ and $\boldsymbol{A}_{t,c'}$ at the corresponding positions. And these corresponding positions denote the same temporal and channel indexes. Thus, our sample-specific aggregation is point-wise aggregation. In this way, we successfully model the temporal-wise attention dynamically while retaining the channel-wise topology attention. Moreover, the entire point-wise aggregation can be computed in parallel without much computational overhead.

4) Sample-Generic Aggregation: To encourage our STGU to model the connectivity between joints and learn the common features of all samples, we use another spatial liner layer to perform shared-topology feature modeling.

$$\boldsymbol{F}^s = \mathcal{M}_2(\boldsymbol{F}^1) = \mathbf{W}_S\boldsymbol{F}^1, \tag{10}$$

where $\boldsymbol{F}^s \in \mathbb{R}^{V\times T\times C'}$ is the feature after sample-generic aggregation. Although the generation process of $\boldsymbol{F}^s$ and $\boldsymbol{A}$ is similar, the modeling is different. We adopt the attention $\boldsymbol{A}$ to select the discriminative features through the gating unit. And $\boldsymbol{F}^s$ without any special definitions and feature interactions denotes the feature after sample-generic aggregation, which models the common dependency across all samples.

5) Feature Updating: Feature updating aims at fusing $\boldsymbol{F}^p$ after sample-specific aggregation and $\boldsymbol{F}^s$ after sample-generic aggregation and updating the feature. The total modeling process can be formulated as:

$$\boldsymbol{H} = \mathcal{T}_3(\boldsymbol{F}^p, \boldsymbol{F}^s) = \sigma((\boldsymbol{F}^p + \boldsymbol{F}^s)\mathbf{W}_O) \tag{11}$$

Here, $\mathbf{W}_O \in \mathbb{R}^{C'\times C''}$ is the weight of the channel projection layer we adopt to update the feature. And $\boldsymbol{H} \in \mathbb{R}^{V\times T\times C''}$ is the output of our STGU.

The static sample-generic modeling pays more attention to modeling human connectivity, and the point-wise attention balances the joint-level and implicit connectivity modeling well. By fusing the features after two aggregations, the output $\boldsymbol{H}$ of our STGU updates the information of the input $\boldsymbol{X}$ successfully without complex aggregations.

The pseudo-code of our STFU is shown in Algorithm 1. For training stability, we initialize $\mathbf{W}_A$ in equation 6 as near-zero values, meaning that $\boldsymbol{A} \approx \boldsymbol{F}^p \approx \mathbf{0}$ at the beginning of training. In this way, the equation 11 can be written as

$$\boldsymbol{H} = \sigma(\boldsymbol{F}^s\mathbf{W}_O) = \sigma(\mathbf{W}_S\boldsymbol{F}^1\mathbf{W}_O), \tag{12}$$

This initialization ensures each STGU block behaves like a regular GCN at the early stage of training. $\mathbf{W}_S$ is the aggregation weight matrix and $\mathbf{W}_O$ is the updating weight matrix. We find that such initialization plays an important role in model convergence. With constant training and parameter updating, the STGU gradually injects specific spatial information related to the current sample.

C. Comparison of Modeling Processes With GCN-Based Approaches

The most significant difference between SiT-MLP and GCN-based approaches in the modeling processes is the spatial modeling module. In previous methods [5], [7], [15], [17], [39], whether the adjacency matrices in the GCN blocks or the normalized affinity matrix in the self-attention [52], it is an intuitive representation of human connections. Each element in the normalized 2D matrix can denote the intensity of this connection. The connection may be explicit or implicit. The network can aggregate the information of related joints in the affinity matrix.

However, as shown in Fig. 7, in the sample-specific aggregation module, each element in our attention map indicates the importance of the corresponding joints. This attention

Algorithm 1 Pseudo-code for the STGU (Pytorch-like)

```
# x: input tensor of shape (B, T, V,C)
# d_model: out-channel of our STGU
# num_joints: number of the joints

def STGU(x, d_model, num_joints):
  shortcut = x
  x = norm(x, axis="channel")
  x = proj(x, 2*d_model,axis="channel")
  F_1, F_2 = split(x, axis="channel")
  A = proj(F_2, num_joints, axis="spatial", init_weight = 0)
  F_p = F_1 * A
  F_s = proj(F_1, num_joints, axis="spatial")
  H = proj((F_s + F_p), d_model, axis="channel")
  return shortcut + H
```

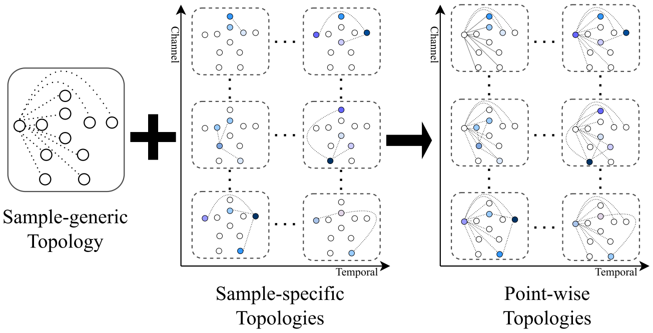


Fig. 7. The topology modeling process of our STGU. The dotted line represents implicit connections. The darker color of the joints indicates more attention to the joints.

shows diversity in the channel dimension and adaptability in the temporal dimension. Moreover, since the attention is modeled through a spatial liner layer $\mathcal{M}(\cdot)$, there is a spatial communication between joints. Every element can carry information about other joints, which can be represented as an implicit connection. Thus, the sample-specific aggregation can be viewed as a joint representation of the importance of joints and implicit connections.

There are some similarities between our method and previous methods [5], [14]. The sample-generic aggregation module in STGU plays the same role as conventional GCNs. The spatial liner layer can model the relations between any two joints and the weight can be denoted as a joint-to-joint affinity matrix. The proposed SiT-MLP can aggregate the global spatial dependency through the spatial liner layer. The sample-generic aggregation models the common connections across all samples.

Finally, after point-wise feature aggregation and global connectivity feature aggregation, the topology features learned in our SiT-MLP contain the spatial-temporal relations of the input sample.

IV. Experiment

To demonstrate the effectiveness of our proposed SiT-MLP, we conduct skeleton-based action recognition on three large-scale datasets. We compare our model with powerful baselines on both performance and complexity. Moreover, we conduct ablation studies in order to examine the effect of individual components. To further evaluate the SiT-MLP, a more extensive failure case analysis is provided. In the end, some visualization will be shown to evidence our motivation and contribution.

A. Datasets

1) NTU RGB+D 60: NTU RGB+D 60 [55] is a large-scale dataset for skeleton-based human action recognition. It contains 56880 videos performed by 40 volunteers. The action sequences can be categorized into 60 classes. Following the authors of this dataset recommendation, we process this dataset into two benchmarks: cross-subject(X-sub) and cross-view(X-view). In the cross-subject setting, sequences of 20 subjects are for training, and the sequences of the rest 20 subjects are for validation. In the cross-view setting, skeleton sequences are split by camera views. Samples from two camera views are used for training, and the rest are used for evaluation.

2) NTU RGB+D 120: NTU RGB+D 120 [56] dataset adds 57367 new skeleton sequences and 60 new action classes to the original NTU RGB+D 60 dataset. There are 32 various video configurations in it, each of which depicts a different location and background. The authors offered the cross-subject(X-sub) and cross-setup(X-set) as two benchmark evaluations. In the cross-subject setting, sequences from 53 subjects are for training, and sequences from the other 53 subjects are for testing. In the cross-setup setting, skeleton sequences are split by setup ID. Samples from even set-up IDs are used for training, and the odd setup IDs are used for evaluation.

3) Northwestern-UCLA: Northwestern-UCLA dataset [57] contains 1494 video clips of 10 different actions captured from three Kinect cameras. We follow the same evaluation protocol in [57]: the first two cameras for training and the other for testing.

B. Implementation Details

Following the previous experimental setting [15], we train our model with SGD with momentum 0.9, weight decay 0.0004, and training batch size 64. The total training epoch is set to 90, and a warmup strategy is used in the first five 5 epochs to make the training process more stable. We set the learning rate to decay with cosine annealing [58], with a base learning rate of 0.1 and an end learning rate of 0.0001. The standard Cross-Entropy loss is adopted to optimize our model. For NTU RGB+D 60 and NTU RGB+D 120 datasets, we adopt the data preprocessing method from [15]. For the Northwestern-UCLA dataset, the batch size is 16, and we process the data followed by [7]. Our project is based on Pytorch [59], and the training and testing experiments are conducted on four NVIDIA GTX 1080Ti GPUs.

TABLE I

ACTION CLASSIFICATION PERFORMANCE COMPARISON AGAINST SOTA METHODS ON THE NTU RGB+D 60 AND NTU RGB+D 120 DATASET. INFOGCN [17] ADOPTED ADDITIONAL MMD LOSSES FOR BETTER RECOGNITION PERFORMANCE. * INDICATES THAT WE RETRAIN THE MODELS USING THEIR OFFICIALLY RELEASED CODE

Type	Methods	Parameters (M)	FLOPs (G)	NTU RGB+D 60		NTU RGB+D 120	
				X-Sub(%)	X-View(%)	X-Sub(%)	X-Set(%)
RNN	VA-LSTM [26]	-	-	79.4	87.6	-	-
	AGC-LSTM [52]	22.9 (↓ 97.3%)	-	89.2	95.0	-	-
CNN	DWNet [29]	-	-	84.1	89.8	-	-
	HCN [27]	-	-	86.5	91.1	-	-
	VA-CNN [53]	24.1 (↓ 97.5%)	-	88.7	94.3	-	-
	Ta-CNN+ [54]	1.1 (↓ 45.5%)	-	90.7	95.1	85.7	87.3
GCN	ST-GCN [5]	3.1 (↓ 80.6%)	-	81.5	88.3	70.7	73.2
	Shift-GCN [7]	0.7 (↓ 14.3%)	2.5	90.7	96.5	85.9	87.6
	Graph2Net [19]	0.9 (↓ 22.2%)	-	90.1	96.0	86.0	87.6
	DC-GCN+ADG [8]	2.5 (↓ 96.0%)	2.8	90.8	96.6	86.5	88.1
	MS-G3D [9]	2.8 (↓ 78.5%)	5.2	91.5	96.2	86.9	88.4
	Hyper-GNN [10]	-	-	89.5	95.7	-	-
	MST-GCN [11]	2.8 (↓ 78.5%)	-	91.5	96.6	87.5	88.8
GCN+Att	SGN [12]	0.7 (↓14.3%)	0.8	89.0	94.5	79.2	81.5
	2s-AGCN [14]	3.5 (↓ 82.9%)	3.9	88.5	95.1	82.9	84.9
	CD-JBF-GCN ([36]	-	-	89.0	95.4	-	-
	LKA-GCN [34]	-	-	90.7	96.1	86.3	87.8
	Dynamic GCN [13]	14.4 (↓ 95.8%)	-	91.5	96.0	87.3	88.6
	EfficientGCN-B4 [16]	2.0 (↓ 70.0%)	2.0	91.7	95.7	88.3	89.1
	CTR-GCN* [15]	1.4 (↓ 60.0%)	1.8	92.4	96.8	88.9	90.5
	InfoGCN* [17]	1.5 (↓ 62.5%)	1.7	92.3	96.7	89.2	90.7
	DD-GCN [35]	-	-	92.6	96.9	88.9	90.2
	FR-Head [33]	2.0 (↓ 70.0%)	-	92.8	96.8	89.5	90.9
	HD-GCN [37]	1.7 (↓ 64.7%)	1.6	**93.0**	**97.2**	**89.8**	**91.2**
Transformer	GAT [40]	5.9 (↓ 89.8%)	-	89.0	95.2	84.0	86.1
	ST-TR [39]	12.1 (↓ 95.0%)	259.4	89.9	96.1	82.7	84.7
	DSTA [32]	4.1 (↓ 85.3%)	64.7	91.5	96.4	86.6	89.0
	TranSkeleton [41]	2.2 (↓ 72.7%)	9.2	92.8	97.0	89.4	90.5
MLP	SiT-MLP (Joint Only)	**0.6**	0.7	90.0	95.0	83.9	85.7
	SiT-MLP (4 ensemble)	**0.6**	**0.7**	92.3	96.8	89.0	90.5

C. Comparison With the State-of-the-Art

For a fair comparison of network performance, we follow the most recent SOTA approaches [8], [15] to train models on 4 skeleton modalities(**joint**, **bone**, **joint motion**, **bone motion**) and report the recognition performance of the ensemble. As shown in the TABLE I, the comparisons include the RNN/CNN-based methods [26], [27], [29], [52], [53], [54], GCN-based methods [5], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17], [19],Transformer-based methods [32], [39], [40].

We compare our results with previous SOTA approaches on NTU RGB+D 60 in TABLE I. The X-sub protocol accuracy and X-view protocol accuracy of our SiT-MLP are 92.3% and 96.8% respectively. The comparison of NTU RGB+D 120 is also shown in the TABLE I. The X-sub protocol accuracy and X-set protocol accuracy of our SiT-MLP are 89.0% and 90.5% respectively, which are also favorable results. The comparison of the Northwestern-UCLA dataset is shown in TABLE II, the accuracy on the validation set is 96.5%. SiT-MLP achieves competitive results, which surpass most of the existing methods. For instance, we largely outperform the lightweight SGN [12] model by 9.8% and 9.0% on NTU120 X-sub and X-view benchmark respectively. Although the performance of our SiT-MLP falls behind the recent state-of-the-art GCN-based methods to some extent, HD-GCN [37] effectively constructs an HD-Graph by decomposing every joint node into several sets to model high-order relations for better performance. FR-Head [33] introduces contrastive learning to enhance sample-specific feature learning. These methods all sacrifice computational efficiency in exchange for higher accuracy. Without complex aggregations and elaborate priors, the proposed SiT-MLP significantly reduces parameters and consumption with competitive performance on various benchmarks.

Furthermore, we compare the throughput and memory usage of the SiT-MLP and previous methods. Throughput indicates the number of input sequences that can be processed per second, and memory usage indicates the resource required in the inference phase. As shown in Table III, the proposed SiT-MLP can process the skeleton sequences more efficiently with less memory usage. For instance, our SiT-MLP can process 395.5 more sequences per second with

TABLE II
CLASSIFICATION PERFORMANCE ON THE NORTHWESTERN-UCLA DATASET

Type	Methods	UCLA Acc(%)
RNN	TS-LSTM [60]	89.2
	2s-AGC-LSTM [52]	93.3
CNN	VA-CNN [53]	90.7
	Ta-CNN [54]	96.1
GCN	Shift-GCN	94.5
	Graph2Net [19]	95.3
	CTR-GCN [15]	96.5
	InfoGCN w/ MMD losses [17]	**97.0**
	DD-GCN [35]	96.7
	FR-Head [33]	96.8
	HD-GCN [37]	96.9
MLP	SiT-MLP (Joint Only)	95.7
	SiT-MLP (4 ensemble)	96.5

TABLE III
THE COMPARISONS WITH THROUGHPUT AND MEMORY USAGE BETWEEN THE PROPOSED SIT-MLP WITH PREVIOUS METHODS

Method	Throughput (sequence/s) ↑	Memory Usage (G) ↓
2S-AGCN [14]	397.7	1.9
MS-G3D [9]	249.2	1.9
CTR-GCN [15]	296.1	1.8
InfoGCN [17]	307.4	1.9
HD-GCN [37]	97.5	1.8
SiT-MLP (ours)	457.0	1.6

TABLE IV
CONSTRUCTING SIT-MLP FROM THE VANILLA BASELINE

Model	Parameters	Acc(%)
MH-STGU Only	0.5M	76.9
MS-TC Only	0.3M	86.4
SiT-MLP (STGU + MS-TC)	0.6M	**90.0**

competitive performance. Therefore, SiT-MLP is more promising in terms of real-time processing and its deployment in resource-constrained environments.

All the experimental results proved that our SiT-MLP can extract spatial-temporal co-occurrence features more effectively.

D. Ablation Studies

To examine the effect of individual components of SiT-MLP, we compare the classification accuracy of different configurations of our model. All experimental ablation studies are conducted on NTU RGD+D 60 dataset cross-subject benchmarks with joint modal information.

1) The Design of SiT-MLP: As shown in Fig. 4, the basic block of our SiT-MLP is the combination of STGU and MS-TC. To verify that our SiT-MLP can achieve spatial-temporal optimization and extract the spatial-temporal co-occurrence feature, we explore the ablation with only the STGU module and the MS-TC module. As shown in TABLE IV, the temporal modeling module MS-TC improves accuracy by 10.1 %, and our STGU contributes to the final performance, with a significant improvement of 3.6% accuracy. Our SiT-MLP, the combination of STGU and MS-TC, can capture the spatial-temporal co-occurrence features well.

TABLE V
THE EFFECTIVENESS OF THE MH-STGU COMPONENTS

Method	Acc(%)
STGU	**90.0**
STGU w/o Sample-generic Aggregation	88.5
STGU w/o Sample-specific Topology Aggregation	89.0
STGU w/o Temporal-wise Modeling	89.8
STGU w/o Channel-wise Modeling	89.3
STGU w Prior Initialization	**90.0**

2) The Effectiveness of the STGU Components: The innovation of our SiT-MLP mainly focuses on the STGU module. In this section, we verify the effectiveness of the STGU in individual components. The experimental details are shown in TABLE V.

(1) Evaluation of the Sample-generic Aggregation Module: This module is for modeling common features across samples. If we prune the module, the network just focuses on sample-specific modeling and ignores sample-generic modeling. Fully dynamic networks lack modeling of common features, and we can observe that the recognition accuracy of the network drops by 1.5%. It shows that the shared topology modeling can capture rich common feature information across all samples.

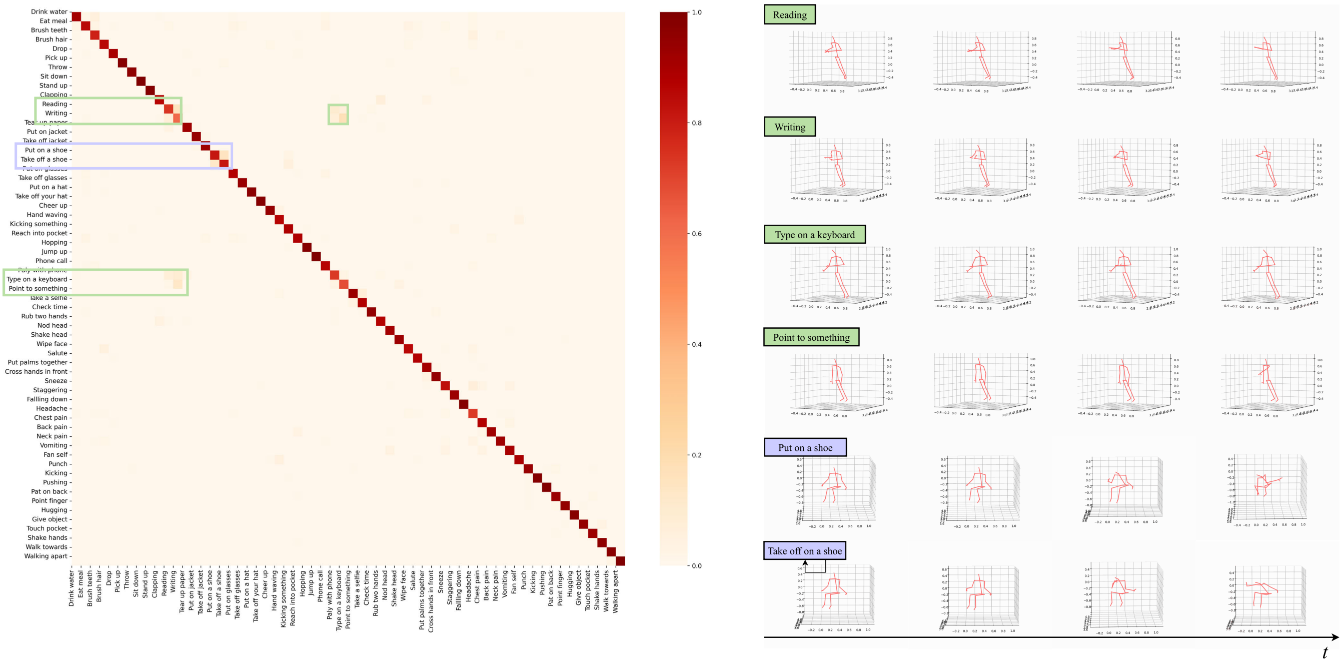
(2) Evaluation of the Sample-specific Aggregation Module: Contrary to what was said before, if we prune the dynamic modeling module, the whole network is statically parameterized, which can't adjust the parameter of the network based on the input sample. From TABLE IV, we can find that compared with the original STGU, the performance has dropped approximately by 1%. It demonstrates that sample-specific modeling and sample-generic modeling play an important role in recognition performance.

(3) Evaluation of Temporal-wise Modeling: To further verify the effectiveness of our point-wise attention modeling We follow the recent approach [15] and adopt the temporal pooling operation in attention modeling. In this way, we force the network to model temporal-shared topologies. From TABLE IV, the performance has decreased by 0.2%. Thus, the temporal-wise topology modeling is working for skeleton-based action recognition.

(4) Evaluation of Channel-wise Modeling: Similar to the above-mentioned, we adopt the channel pooling operation in attention modeling, which forces the network to model channel-shared topology. In this way, the performance has decreased by 0.7%. The channel-shared topology reduces the distinctiveness of features. Thus, modeling of channel-wise topology is vital for learning distinctive topology.

From a combination of results *(3)* and *(4)*, either temporal-wise or channel-wise topology modeling has an impact on the performance of the network. Therefore, the point-wise sample-specific topology modeling, which not only achieves adaptability in the channel dimension but also adaptability in the temporal dimension, achieves great performance.

(5) Evaluation of Prior Initialization: To verify our SiT-MLP is insensitive to the initialization, we initialize the



(a)

(b)

Fig. 8. (a) The confusion matrix on the X-sub benchmark of NTU RGB+D 60. The vertical coordinate is the true label, and the horizontal is the prediction. (b) The visualization of 3D skeletons of the failure case.

TABLE VI

THE COMPARISONS BETWEEN THE 3D SKELETONS CAPTURED FROM THREE MICROSOFT KINECT V2 CAMERAS AND SKELETONS EXTRACTED FROM THE RGB VIDEOS OF JOINT MODALITY

Method	NTU120/Xsub (Kinect)	NTU120/Xsub (RGB)
ST-GCN [5]	82.1	80.1 (↓ 2.0)
2S-AGCN [14]	82.8	80.2 (↓ 2.6)
CTR-GCN [14]	84.0	82.2 (↓ 1.8)
SiT-MLP (ours)	83.9	83.2 (↓ 0.7)

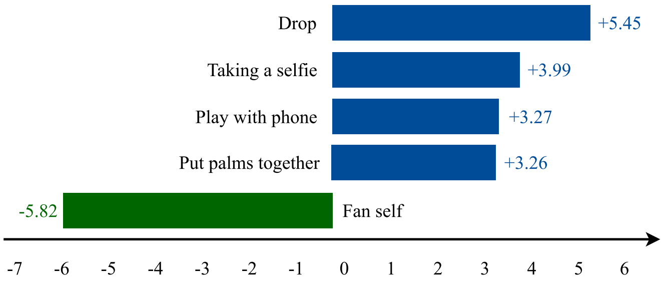


Fig. 9. Action classes with accuracy differences higher than 3% between CTR-GCN and our method on NTU60 X-sub benchmark.

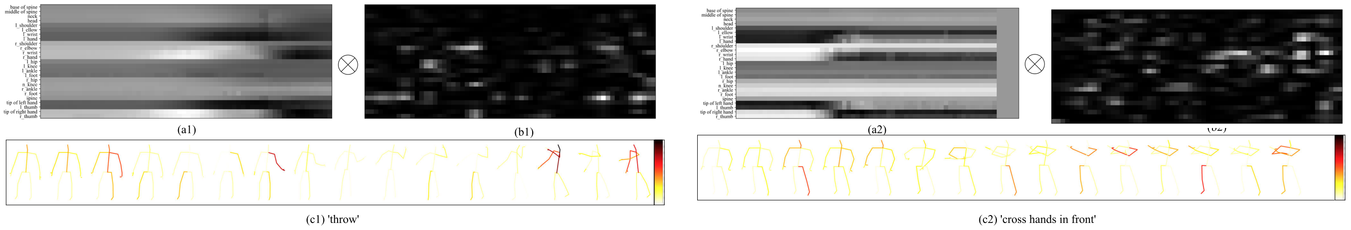
spatial linear layer in equation 10 with the human body's natural connections. We initialize the weight of the linear layer with the binary outward connection matrix. As shown in TABLE IV, compared with our SiT-MLP, whose spatial mixing linear layers are initialized as diagonal matrices, the classification accuracy of SiT-MLP with prior initialization is flat. The results confirm that our SiT-MLP can learn spatial information without any well-designed initialization.

3) Evaluation of the Generalization: To quantitatively analyze the model's robustness and generalization capabilities, we measure the performance of SiT-MLP in real-world scenarios. We extract human skeletons from the RGB-based videos, where actions occur in complex environments with background noise. We use Faster-RCNN [61] for human detection and HRNet [62] for human pose estimation. As shown in TABLE VI, the performance of SiT-MLP is decreased by 0.7%. Since there is no need to introduce priors of the human body connections to the network, the proposed SiT-MLP demonstrates greater generality. Compared with the existing GCN-based method, the accuracy drop of our SiT-MLP is relatively small, which proves that our method is more general and robust.

E. Discussion

To further analyze the results of our SiT-MLP, we visualize the confusion matrix on the X-sub benchmark of NTU RGB+D 60 in Fig. 8(a). The main confusion in our approach lies in two parts: actions of "*Reading*", "*Writing*", "*Type on a keyboard*", "*Point to something*", and actions of "*Put on a shoe*", "*Take off a shoe*". There are two main reasons for our confusion. First, as shown in Fig. 8(b), due to the absence of information omitted in skeletonization (hand skeleton, head, related objects, etc.), some actions have been difficult to classify. Second, our SiT-MLP is designed to pay attention to capturing global relations. Because of the absence of local modeling of the human body, SiT-MLP failed to classify the fine-grained actions well. The main difference in actions of "*Reading*" and "*Writing*" may focus on the details of the hand skeleton. This lack of modeling of fine-grained local information leads our SiT-MLP to confuse these two actions.

Moreover, to further analyze the failure cases, we compare the classification results of SiT-MLP (90.0) with CTR-GCN (89.6), which is a powerful GCN-based method. As shown in


(a1) (b1) (a2) (b2)

(c1) 'throw' (c2) 'cross hands in front'

Fig. 10. Heatmap visualization of the proposed SiT-MLP: (a) motion time series images, (b) Grad-CAM images, and (c) skeleton motion highlighted with color for recognition reasons. The darker the color, the more attention the area receives.

Fig. 9, there is a decrease in the accuracy of actions "*Fan self*". We argue the reason for this decrease is that the proposed SiT-MLP can't tell the repetitive actions well, like "*Fan self*". The SiT-MLP can model the frame-wise attention. However, because of the repetition within the action, the frame-wise attention modeling makes it hard to capture the general feature of the periodicity of actions, leading to a performance drop. Conversely, thanks to point-wise topology feature learning, the proposed SiT-MLP can adjust the spatial relations between joints dynamically over time. Actions "*Type on a keyboard*", "*Drop*", "*Taking a selfie*", '*Play with phone*", and "*Put palms together*"s are prolonged actions, which are not repetitive and contain more complex spatial relations. The reason for the increase in these actions is that frame-wise attention provides independent spatial relations in the spatial dimension.

F. Qualitative Analysis

In this section, we make some visualizations and analyses to support our work.

1) Grad-CAM: As shown in Fig. 10, we select two classical actions 'throw' and 'cross hands in front', and visualize both the original images in Fig. 10(a) and corresponding Grad-CAM [63] images in Fig. 10(b). To show the ROI of the SiT-MLP more directly, we visualize the results of the Grad-CAM on the input skeleton sequence in Fig. 10(c). It can be observed that the SiT-MLP always relies on the key area of the input sequence to predict the categories. For the action of 'throw', the activation areas are more focused on the movements of upper body joints in the keyframe where the action occurs. Furthermore, for the action of 'cross hand in front', the activation areas are focused on the joints of the hands and arms, which can be viewed as a more semantically informative region.

2) Sample-Specific Topology Learning: We illustrate the sample-specific topology attention of four typical action samples in Fig 11. The darker color and larger size of the joints indicate that the more attention it receives. During the jumping, it is intuitive that our SiT-MLP pays more attention to the tendency of the legs and knees to move. Furthermore, in an action like kicking, it is kinematically correct to focus on the knee and the opposite foot at the same time. For actions like sitting, our model pays more attention to the hip. This attention slowly diminishes as the human begins to rise. During the hugging, Our SiT-MLP can also focus equally on the trajectories of the limbs. Above all, our SiT-MLP realizes not only spatial but also temporal topology awareness.

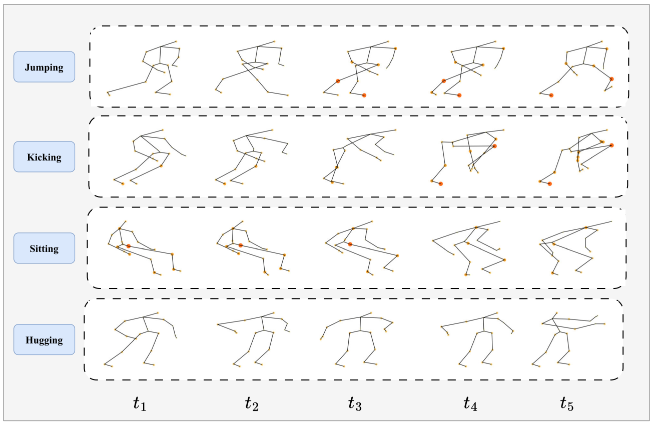


Fig. 11. Examples of sample-specific topology attention modeling of STGU. Lines indicate the skeletal connections between human bodies. The darker color and larger size of the joints indicate that the more attention it receives.

V. Limitations

Although the proposed SiT-MLP achieves inspiring performance on various benchmarks, however, according to TABLE I and II, SiT-MLP falls behind HD-GCN. This gap may be caused by poor performance in the recognition of fine-grained and repetitive actions. Specifically, our SiT-MLP is designed to capture global spatial relations but reduce the ability to model the local information in the body part, leading to a performance drop for fine-grained action recognition. In this case, the possible solution is to introduce the complementary modeling of local and global information to assist network learning. In addition, because frame-wise attention modeling hinders the capture of the general feature of the periodicity of action to some extent, the SiT-MLP can't recognize repetitive action well. It is necessary to design between frame-wise topology attention to recognize periodic actions. In further work, we will continue to study and solve these two drawbacks.

VI. Conclusion

In this paper, we try to use MLPs to address skeleton-based action recognition. We propose STGU, the first MLP-based structure for spatial dependency modeling without extra priors and complex aggregations. In STGU, a new feature interaction mechanism, the gate mechanism, is introduced to this task, which can implement sample-specific and point-wise attention modeling. Without the elaborate priors, our SiT-MLP shows great generalizability, which can be generalized to other fields of action recognition. Without complex aggregations, the proposed SiT-MLP significantly reduces parameters with competitive performance on various benchmarks. Moreover, the SiT-MLP is more promising in terms of real-time processing and its deployment in resource-constrained environments. We hope the proposed SiT-MLP will help the community with some inspiration for the task of skeleton-based action recognition.

References

[1] B. Ren, M. Liu, R. Ding, and H. Liu, "A survey on 3D skeleton-based action recognition using learning method," 2020, *arXiv:2002.05907*.

[2] Z. Zhang, "Microsoft Kinect sensor and its effect," *IEEE Multimedia Mag.*, vol. 19, no. 2, pp. 4–10, Feb. 2012.

[3] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2D pose estimation using part affinity fields," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 7291–7299.

[4] C. Wang and J. Yan, "A comprehensive survey of RGB-based and skeleton-based human action recognition," *IEEE Access*, vol. 11, pp. 53880–53898, 2023.

[5] S. Yan, Y. Xiong, and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition," in *Proc. AAAI Conf. Artif. Intell.*, vol. 32, no. 1, 2018, pp. 1–10.

[6] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," 2016, *arXiv:1609.02907*.

[7] 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.

[8] 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.* Glasgow, U.K.: Springer, Aug. 2020, pp. 536–553.

[9] 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.

[10] X. Hao, J. Li, Y. Guo, T. Jiang, and M. Yu, "Hypergraph neural network for skeleton-based action recognition," *IEEE Trans. Image Process.*, vol. 30, pp. 2263–2275, 2021.

[11] Z. Chen, S. Li, B. Yang, Q. Li, and H. Liu, "Multi-scale spatial temporal graph convolutional network for skeleton-based action recognition," in *Proc. AAAI Conf. Artif. Intell.*, vol. 35, no. 2, 2021, pp. 1113–1122.

[12] 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.

[13] 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.

[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] 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.*, May 2021, pp. 13359–13368.

[16] Y.-F. Song, Z. Zhang, C. Shan, and L. Wang, "Constructing stronger and faster baselines for skeleton-based action recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 2, pp. 1474–1488, Feb. 2023.

[17] H.-G. Chi, M. H. Ha, S. Chi, S. W. Lee, Q. Huang, and K. Ramani, "InfoGCN: Representation learning for human skeleton-based action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 20154–20164.

[18] Q. Pan, Z. Zhao, X. Xie, J. Li, Y. Cao, and G. Shi, "View-normalized and subject-independent skeleton generation for action recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 12, pp. 7398–7412, Nov. 2022.

[19] 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. 2021.

[20] X. Xiong, W. Min, Q. Wang, and C. Zha, "Human skeleton feature optimizer and adaptive structure enhancement graph convolution network for action recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 1, pp. 342–353, Jan. 2022.

[21] 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.

[22] H. Liu, Z. Dai, D. So, and Q. V. Le, "Pay attention to MLPs," in *Proc. Conf. Neural Inf. Proces. Syst. (NeurIPS)*, 2021, pp. 9204–9215.

[23] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–13.

[24] 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.

[25] 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," *Proc. AAAI Conf. Artif. Intell.*, vol. 31, no. 1, pp. 4263–4270, Feb. 2017.

[26] 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.

[27] 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. (IJCAI)*, 2018, pp. 786–792.

[28] 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.

[29] Y. Dang, F. Yang, and J. Yin, "DWNet: Deep-wide network for 3D action recognition," *Robot. Auto. Syst.*, vol. 126, Apr. 2020, Art. no. 103441.

[30] A. Banerjee, P. K. Singh, and R. Sarkar, "Fuzzy integral-based CNN classifier fusion for 3D skeleton action recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 6, pp. 2206–2216, Jun. 2021.

[31] Z. Huang, X. Shen, X. Tian, H. Li, J. Huang, and X.-S. Hua, "Spatio-temporal inception graph convolutional networks for skeleton-based action recognition," in *Proc. 28th ACM Int. Conf. Multimedia*, Oct. 2020, pp. 2122–2130.

[32] L. Shi, Y. Zhang, J. Cheng, and H. Lu, "Decoupled spatial–temporal attention network for skeleton-based action recognition," 2020, *arXiv:2007.03263*.

[33] H. Zhou, Q. Liu, and Y. Wang, "Learning discriminative representations for skeleton based action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2023, pp. 10608–10617.

[34] Y. Liu, H. Zhang, Y. Li, K. He, and D. Xu, "Skeleton-based human action recognition via large-kernel attention graph convolutional network," *IEEE Trans. Vis. Comput. Graphics*, vol. 29, no. 5, pp. 2575–2585, May 2023.

[35] C. Li, Q. Huang, and Y. Mao, "DD-GCN: Directed diffusion graph convolutional network for skeleton-based human action recognition," in *Proc. IEEE Int. Conf. Multimedia Expo. (ICME)*, Jul. 2023, pp. 786–791.

[36] Z. Tu, J. Zhang, H. Li, Y. Chen, and J. Yuan, "Joint-bone fusion graph convolutional network for semi-supervised skeleton action recognition," *IEEE Trans. Multimedia*, vol. 25, pp. 1819–1831, 2022.

[37] J. Lee, M. Lee, D. Lee, and S. Lee, "Hierarchically decomposed graph convolutional networks for skeleton-based action recognition," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 10444–10453.

[38] Y. Zhu, G. Huang, X. Xu, Y. Ji, and F. Shen, "Selective hypergraph convolutional networks for skeleton-based action recognition," in *Proc. Int. Conf. Multimedia Retr.*, Jun. 2022, pp. 518–526.

[39] C. Plizzari, M. Cannici, and M. Matteucci, "Skeleton-based action recognition via spatial and temporal transformer networks," *Comput. Vis. Image Understand.*, vols. 208–209, Jul. 2021, Art. no. 103219.

[40] J. Zhang, W. Xie, C. Wang, R. Tu, and Z. Tu, "Graph-aware transformer for skeleton-based action recognition," *Vis. Comput.*, vol. 39, no. 10, pp. 4501–4512, Oct. 2023.

[41] H. Liu, Y. Liu, Y. Chen, C. Yuan, B. Li, and W. Hu, "TranSkeleton: Hierarchical spatial–temporal transformer for skeleton-based action recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 8, pp. 4137–4148, Aug. 2023.

[42] A. Dosovitskiy et al., "An image is worth 16x16 words: Transformers for image recognition at scale," 2020, *arXiv:2010.11929*.

[43] Z. Liu et al., "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 10012–10022.

[44] I. Tolstikhin et al., "MLP-mixer: An all-MLP architecture for vision," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 24261–24272.

[45] H. Touvron et al., "ResMLP: Feedforward networks for image classification with data-efficient training," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 4, pp. 5314–5321, Apr. 2023.

[46] Q. Hou, Z. Jiang, L. Yuan, M.-M. Cheng, S. Yan, and J. Feng, "Vision permutator: A permutable MLP-like architecture for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 1, pp. 1328–1334, Jan. 2023.

[47] C. Tang, Y. Zhao, G. Wang, C. Luo, W. Xie, and W. Zeng, "Sparse MLP for image recognition: Is self-attention really necessary?" in *Proc. AAAI Conf. Artif. Intell.*, vol. 36, no. 2, 2022, pp. 2344–2351.

[48] D. J. Zhang et al., "MorphMLP: An efficient MLP-like backbone for spatial–temporal representation learning," 2021, *arXiv:2111.12527*.

[49] M. Go and H. Tachibana, "GSWIN: Gated MLP vision model with hierarchical structure of shifted window," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Jun. 2023, pp. 1–5.

[50] M.-H. Guo, C.-Z. Lu, Z.-N. Liu, M.-M. Cheng, and S.-M. Hu, "Visual attention network," 2022, *arXiv:2202.09741*.

[51] D. Hendrycks and K. Gimpel, "Gaussian error linear units (GELUs)," 2016, *arXiv:1606.08415*.

[52] 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.

[53] 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.

[54] K. Xu, F. Ye, Q. Zhong, and D. Xie, "Topology-aware convolutional neural network for efficient skeleton-based action recognition," in *Proc. AAAI Conf. Artif. Intell. (AAAI)*, vol. 36, no. 3, 2022, pp. 2866–2874.

[55] 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.

[56] 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. 2020.

[57] 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. (CVPR)*, Jun. 2014, pp. 2649–2656.

[58] I. Loshchilov and F. Hutter, "SGDR: Stochastic gradient descent with warm restarts," 2016, *arXiv:1608.03983*.

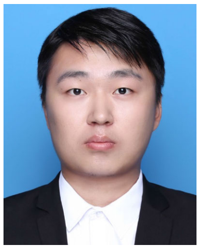
[59] A. Paszke et al., "PyTorch: An imperative style, high-performance deep learning library," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 1–7.

[60] 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.

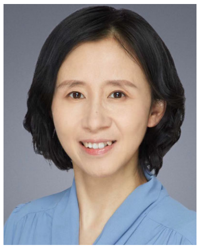
[61] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

[62] K. Sun, B. Xiao, D. Liu, and J. Wang, "Deep high-resolution representation learning for human pose estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 5693–5703.

[63] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 618–626.

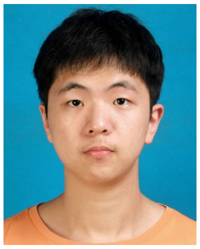


Shaojie Zhang received the bachelor's degree from the School of Artificial Intelligence, Beijing University of Posts and Telecommunications, Beijing, China, in 2022, where he is currently pursuing the M.S. degree with the School of Artificial Intelligence. His research interests include computer vision, deep learning, and human action recognition.



Jianqin Yin (Member, IEEE) received the Ph.D. degree from Shandong University, Jinan, China, in 2013. She is currently a Professor with the School of Artificial Intelligence, Beijing University of Posts and Telecommunications, Beijing, China. Her research interests include service robots, pattern recognition, machine learning, and image processing.

Yonghao Dang received the doctor's degree in control science and engineering from the Beijing University of Posts and Telecommunications, Beijing, China, in 2023. He is currently a post-doctor with the School of Artificial Intelligence, Beijing University of Posts and Telecommunications. His research interests include computer vision, embodied intelligence, and deep learning.



Jiajun Fu received the B.E. degree in computer science and technology from Beijing University of Posts and Telecommunications, Beijing, China, in 2020, where he is currently pursuing the M.S. degree with the School of Artificial Intelligence. His research interests include spatio-temporal multimedia applications, such as human motion prediction and traffic flow prediction.