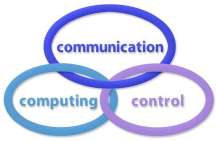
INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
Online ISSN 1841-9844, ISSN-L 1841-9836, Volume: 18, Issue: 3, Month: June, Year: 2023
Article Number: 5061, https://doi.org/10.15837/ijccc.2023.3.5061



CCC Publications

A Lightweight Architecture Attentional Shift Graph Convolutional Network for Skeleton-Based Action Recognition

Xianshan Li, Jingwen Kang, Yang Yang, Fengda Zhao

Xianshan Li
1. School of Information Science and Engineering, Yanshan University
Qinhuangdao 066004, China
2. Key Laboratory for Software Engineering of Hebei Province, Yanshan University
Qinhuangdao 066004, China
xjlxs@ysu.edu.cn

Jingwen Kang
1. School of Information Science and Engineering, Yanshan University
Qinhuangdao 066004, China
2. Key Laboratory for Software Engineering of Hebei Province, Yanshan University
Qinhuangdao 066004, China
k17303402859@163.com

Yang Yang
1. School of Information Science and Engineering, Yanshan University
Qinhuangdao 066004, China
2. Key Laboratory for Software Engineering of Hebei Province, Yanshan University
Qinhuangdao 066004, China
y857482666@163.com

Fengda Zhao*
1. School of Information Science and Engineering, Yanshan University
Qinhuangdao 066004, China
2. School of Information Science and Engineering, Xinjiang University of Science and Technology
Korla 841000, China
3. Key Laboratory for Software Engineering of Hebei Province, Yanshan University
Qinhuangdao 066004, China
*Corresponding author: zfd@ysu.edu.cn

Abstract

In the field of skeleton-based human behavior recognition, graph convolutional neural networks have made remarkable achievements. However, high precision networks are often accompanied by numerous parameters and computational cost, and their application in mobile devices has considerable limitations. Aiming at the problem of excessive spatiotemporal complexity of high-accuracy methods, this paper further analyzes the lightweight human action recognition model and proposes a lightweight architecture attentional shift graph convolutional network. There are three

main improvements in this model. Firstly, shift convolution is a lightweight convolution method that can be combined with graph convolution to effectively reduce its complexity. At the same time, a shallow architecture for multi-stream early fusion is designed to reduce the network scale by merging multi-stream networks and reducing the number of network layers. In addition, the efficient channel attention module is introduced into the model to capture the underlying characteristic information in the channel domain. Experiments are conducted on the three existing skeleton datasets, NTU RGB+D, NTU-120 RGB+D, and Northwestern-UCLA. Results demonstrate that the proposed model is not only competitive in accuracy, but also outperforms current mainstream methods in parameter count and computational cost, and supports running in some devices with limited computing and storage resources.

Keywords: action recognition, lightweight network, shift graph convolution, attention module.

1 Introduction

As a multidisciplinary research direction, human behavior recognition plays an increasingly important role in intelligent home [1], virtual reality [2], and video surveillance [3]. 3D skeleton data is widely used because it can better reflect human posture and motion trajectory. Human action recognition based on 3D skeleton data is mainly divided into two directions, the early methods are based on manual features, and the later methods are based on deep learning. In the method based on manual features, the features of each task need to be selected manually by researchers. When the amount of data is large, the manual extraction task will be extremely complicated and redundant. The method based on deep learning can independently learn effective features from massive data, adapt to a variety of complex scenarios, and greatly improve recognition accuracy compared with the learning method based on manual features. Therefore, researchers are more inclined to use deep learning methods for research recently.

The method based on Recurrent Neural Networks (RNNs) [4–9] usually converts the 3D skeleton coordinate vector into sequence information according to specific traversal rules and dynamically mines the temporal connections of the human skeleton. Nevertheless, such methods focus more on temporal information and have certain limitations in processing spatial information. The method based on Convolutional Neural Networks (CNNs) [10–15] usually models 3D skeleton information as pseudo-images, with spatial and temporal information encoded as rows and columns, respectively, to identify different actions by fully extracting the spatiotemporal feature information of the skeleton. However, such methods ignore the constraint relationships of the human skeleton, which are very important for skeleton recognition. The method based on the Graph Convolutional Neural Networks (GCNs) [16–35] constructs the topological map according to the physical connection of human joints along the spatial and temporal directions to fully extract the spatiotemporal characteristics of the human body, which currently has very broad application prospects. Yan et al. [16] pioneer the application of graph convolution to human behavior recognition, and propose the spatial-temporal graph convolutional network (ST-GCN). The model uses a graph convolutional network to learn spatial and temporal information of skeleton data respectively and has a strong generalization performance. The improved model based on ST-GCN is usually combined with other networks, such as Long Short-Term Memory (LSTM) and attention mechanism, to further improve the accuracy of the model. However, the existing high-precision models based on GCNs are often very complex. In other words, the excessively high number of parameters and computational cost make the network difficult to train, hindering the development of this field in mobile devices. Therefore, how to realize the lightweight of the network is a problem worthy of study.

There have been some studies on lightweight models. Cheng et al. [29] introduce shift convolution into graph convolution effectively reducing its complexity. However, their 4s Shift-GCN model divides the feature information into four streams and inputs them into the network successively, which expands the model complexity by four times. Based on Shift-GCN, Zang et al. [34] replace the shift graph convolution module with sparse shift graph convolution, which effectively reduces data redundancy. But this model, using the multi-stream network, has the same issue as 4s Shift-GCN. Therefore, reducing the network size by changing the network structure is a challenging problem in skeleton action recognition. Zhang et al. [28] embed multi-stream joint feature information into the same

high-dimensional space before the convolutional network, effectively reducing the model size. Song et al. [30] improve the multi-stream model by proposing an early fused multi-branch architecture and introducing the residual bottleneck structure, which significantly reduces the model complexity. Sun et al. [31] propose a SlowFast graph convolutional network in which the design of the Fast pathway is very lightweight. Jang et al.[32] design a simple and clear hierarchical feature extraction model with fewer parameters and less computations. However, the accuracy of these methods is insufficient.

The proposed method is based on 4s Shift-GCN [29], namely lightweight architecture attentional shift graph convolutional network (LA-SGCN). Inspired by the early fused multi-branch architecture [30], the goal of lessening the number parameters and reducing computational cost is achieved by improving the network architecture. A lightweight and efficient channel attention [36] module is added after the spatiotemporal graph convolution to effectively capture the key features of the model. The proposed model is evaluated on three datasets: NTU RGB+D [37], NTU-120 RGB+D [38], and Northwestern-UCLA [39]. Compared to previous methods, our model has fewer parameters and smaller computational cost while improving accuracy. Our main contributions are as follows:

(1) By integrating Shift convolution into graph convolution, parameter operations can be concentrated in point-wise convolution, which greatly reduces space and time complexity. In addition, the spatial shift graph convolution sets the skeleton as a complete graph, and each node can obtain the information of other nodes through the shift operation. The temporal shift graph convolution adaptively adjusts the receptive field of nodes through flexible shift operations, which effectively improves the model performance.

(2) A shallow architecture for multi-stream early fusion is designed, which not only meets the needs of multi-stream features in the early stage of the network, but also fuses all features into one stream in the later stages, thus reducing the model complexity. Moreover, considering that the shift graph convolution can adjust the receptive field autonomously, and does not require a very deep network to expand it, the depth of the network is compressed to further reduce the size of the model.

(3) Considering that the point-wise convolution in shift graph convolution is sensitive to channel information, a lightweight efficient channel attention module is added after each layer of shift temporal graph convolution, and key features are successfully extracted by adaptively learning the weight parameters of spatiotemporal information in each channel. This attention module not only effectively improves the accuracy of the model, but also barely places an additional burden on the network.

2 Related Work

With the development of artificial intelligence technology, the fields of smart home, public safety [40, 41], and smart transportation have put forward higher requirements for human-computer interaction [42]. How to identify and monitor people's behavior intelligently and provide a timely response to emergencies has become a research hotspot in these fields. Therefore, human behavior recognition has important research value and practical significance. Deep learning-based 3D skeleton human behavior recognition methods are mainly based on RNNs, CNNs, and GCNs.

RNNs-based approaches normally convert the 3D skeleton coordinate vector into sequence information according to specific traversal rules, and dynamically mine the temporal connection of the human skeleton. Du et al. [4] divide the human skeleton into five parts, input each part into a bidirectional RNN, and fuse them into higher-level features hierarchically. Liu et al. [5] extend skeleton behavior recognition based on RNN to spatial and temporal dimensions and optimize the network through trust gates. Liu et al. [6] introduce the global context memory cell into the LSTM network to improve the selective attention performance of the network. Zhu et al. [7] apply bidirectional LSTM to encode each skeleton sequence and introduce a regularization method to better constrain the relationship between skeleton nodes. Si et al. [8] propose a spatial inference algorithm and a temporal stacking learning algorithm to extract the spatial features and temporal dynamics of the human skeleton, which not only effectively extract the high-level features of the skeleton, but also the serial structure of multiple LSTMs can capture more long-term characteristic information.

CNNs-based approaches generally model 3D skeleton information as pseudo-images to extract spatiotemporal features of skeletons. Du et al. [10] propose an end-to-end convolutional neural

network-based action recognition method, applying CNNs to 3D human skeleton data for the first time. Compared with the RNNs-based method, not only the temporal information is utilized, but also the basic spatial information is retained. Wang et al. [11] use multiple joint trajectory maps and convolutional neural networks to form a trajectory convolutional mapping network, significantly improving the accuracy of network recognition. Considering the interpretability of the model, Kim et al. [12] propose a spatiotemporal representation method to enhance model features. Liu et al. [13] represent skeleton information as visual and motion-enhanced color images and use multi-stream CNN to explore the deep features of different types of color images. Li et al. [14] apply the motion information of adjacent frames to construct temporal information and utilize CNN to efficiently extract the spatial information in the original coordinates.

GCNs-based approaches construct topology maps according to the physical connection of human joints, and adequately extract the spatiotemporal features of the human body. Yan et al. [16] apply graph convolution to the skeleton-based human action recognition method for the first time and propose the model named ST-GCN. ST-GCN can learn spatiotemporal information autonomously and has a strong expressive ability. Li et al. [17] employ the encoder-decoder structure to autonomously learn some potentially connected nodes while expanding the skeleton graph to explore higher-order dependencies. Shi et al. [18] introduce bone orientation and length information into the network to adaptively adjust the graph structure, which is more flexible. Si et al. [19] fuse GCNs with LSTM to handle discriminative spatiotemporal features. Shi et al. [20] represent skeleton information as a directed graph and apply an adaptive network structure to train the model. Zhang et al. [21] integrate long-range dependencies among joints through context aware graph convolutions. Peng et al. [22] construct adjacency matrices dynamically through Neural Architecture Search (NAS). Plizzari et al. [23] utilize a self-attention algorithm to capture the potential relationship between joints and form a two-stream network with GCNs to extract the effective information in the joints. Huang et al. [24] design part relation and part attention blocks to learn part-level and joint-level information. Song et al. [25] apply class activation maps to optimize multi-stream graph convolutional networks. Wen et al. [26] use graph convolution to extract spatial features, and stack multiple different convolution kernels to extract temporal information. Shi et al. [27] adopt a gating mechanism to adaptively adjust the graph topology of different layers in graph convolution, and design three-dimensional attention modules of spatial, temporal, and channel to strengthen the corresponding features. Nevertheless, these models bring more parameters and incur higher computational cost. Therefore, reducing the complexity of the model remains a challenging issue. Zhang et al. [28] embed joint position and velocity information into the same high-dimensional space and incorporate high-level semantics like joint type and frame index to develop a lightweight strong baseline. Cheng et al. [29] combine shift convolutional neural networks (shift CNNs) with GCNs, which effectively reduces the computational complexity of the model. Song et al. [30] propose an early fused multi-branch architecture to reduce the complexity of the model at the network level and introduce the residual bottleneck module to decrease the computational cost of spatiotemporal graph convolution. Sun et al. [31] employ the SlowFast pathway to implement a spatiotemporal attention lightweight GCNs. Jiang et al. [32] use two 1×1 convolutions to embed the multi-stream network features separately, and then add the features for follow-up training, thereby reducing the number of parameters. Chen et al. [33] decrease model complexity by applying a strategy with low parameter burden such as fusion input and max pooling. Zang et al. [34] use a sparse shift module to improve the performance of shift graph convolution.

3 Method

This section introduces the details of the proposed LA-SGCN model. The overall process is illustrated in Figure 1. Four streams are input to the network, each through 3 layers of Shift-Block. The four streams are then spliced in the channel dimension to form one stream, passing through 5 layers of Shift-Block followed by a fully connected layer that outputs the classification result. The Shift-Block module includes three important modules: spatial shift graph convolution (Shift-GCN), temporal shift graph convolution (Shift-TCN), and efficient channel attention (ECA).

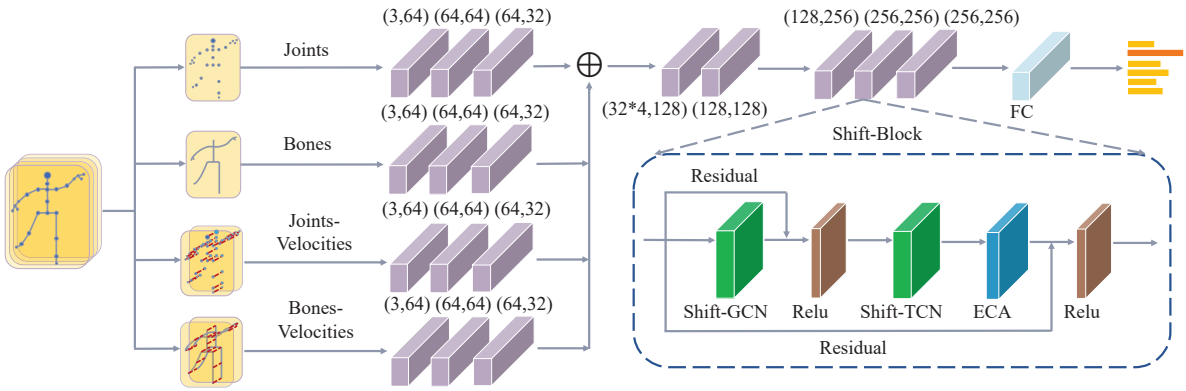


Figure 1: Overall framework of the network

3.1 Shift graph convolution

Spatial shift graph convolution. In a previous work [16], the connection between non-adjacent nodes is ignored because graph convolution of spatial dimensions is completed with the help of adjacency matrices. Consequently, the convolution of the spatial shift graph transforms the skeleton graph into a complete graph, and there is a direct correlation between each node and every other node. As shown in Figure 2, the convolution is carried out in three steps: shift transformation, point-wise convolution, and shift transformation. Assuming that there are $N = 10$ nodes, and each node has $C = 6$ channels, the shift transformation connects the channels of node 1 and node 10 end to end to perform a shift operation. The translation step size of the i^{th} channel is $i \, mod \, N$. Point-wise convolution can deeply correlate the offset feature graph and output the specified channel dimension. The first shift transformation shifts the channel of each node upward to realize the information exchange between nodes. The second shift transformation shifts the channel of each node downward, making the channel information return to the original node, avoiding the confusion of node feature information. The spatial skeleton feature map is denoted $X \in \mathbb{R}^{N \times C}$, the matrix H is added to capture the effective links between nodes:

$$\tilde{X}_H = \tilde{X} \cdot (tanh(H) + 1) \tag{1}$$

where $\tilde{X}$ is the feature after shift graph convolution. Since the shift transformation operation can be implemented in memory while the number of parameters and the computational cost are concentrated in point-wise convolution, the complexity of graph convolution can be significantly reduced.

Temporal shift graph convolution. In the time dimension, shift graph convolution can also be divided into three steps: shift transformation, point-wise convolution, and shift transformation. The shift transformation is used to shift the channel information between frames. Since different layers require different time receptive fields, setting the same shift value for each channel will not achieve the optimal effect. Therefore, a learnable shift parameter $\alpha_i, i = 1, 2, ..., C$ is set up to adaptively calculate the optimal shift value for each channel in each layer. If the shift parameter is an integer, it is not differentiable, and the gradient cannot be passed. Therefore, the shift parameter is expanded from integer range to real number range. The skeleton sequence feature map is denoted $X \in \mathbb{R}^{N \times T \times C}$, the shift parameter can be calculated by linear interpolation:

$$\tilde{X}_{(v,t,i)} = (1 - \mu) \cdot X_{(v,\lfloor t+\alpha_i \rfloor,i)} + \mu \cdot X_{(v,\lfloor t+\alpha_i \rfloor+1,i)} \tag{2}$$

where $\mu = \alpha_i - \lfloor \alpha_i \rfloor$, while v and t determine the node and frame where channel i is located, respectively.

3.2 Shallow architecture for multi-stream early fusion

In previous experiments, most high-precision models adopt a multi-flow mode [17–20, 22–25, 29]. The advantage of multi-stream architecture is that each input feature can be fully convolved to maxi-

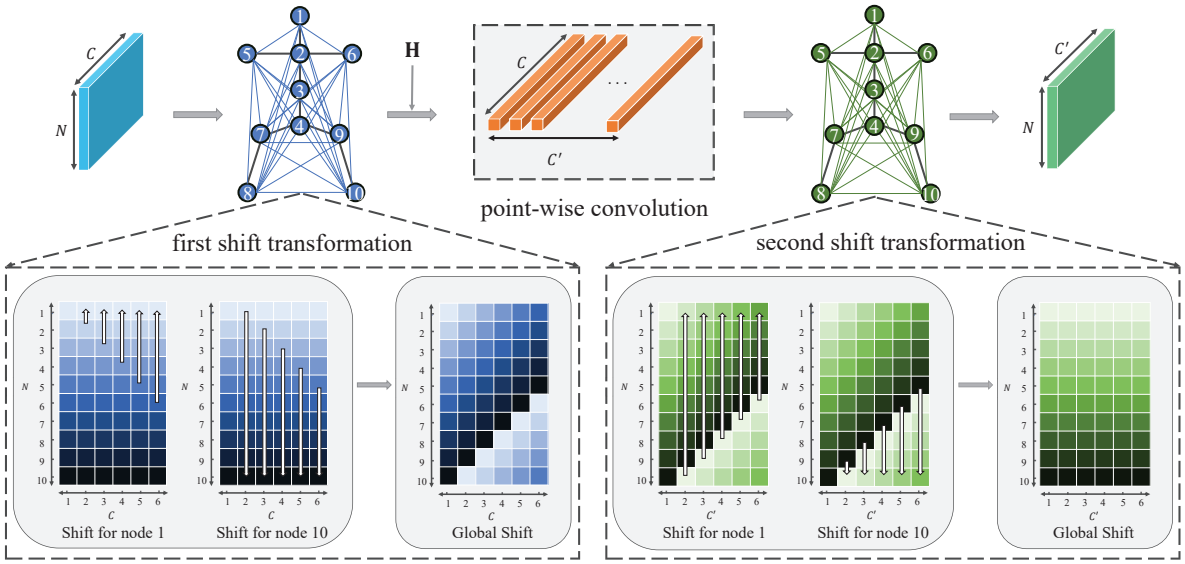


Figure 2: Spatial shift graph convolution

mize the utilization of information. The disadvantage is that the same network is run multiple times, resulting in multiplied computing costs. In the proposed multi-stream early fusion model, feature information is divided into four input streams: Joints, Bones, Joints-Velocities, and Bones-Velocities. The Joints stream is the 3D coordinate of the human body junction. The Bones stream is the length and direction of the human skeleton. The Joints-Velocities and Bones-Velocities streams are the differences between two adjacent frames of the Joints and Bones streams, respectively. In the early stage of the network, the four streams are trained separately, which can make the network get enough features. In the later stage of the network, the feature information of the four streams is spliced in the channel dimension and fused into one stream to complete the subsequent training, which can effectively reduce the model size. This strategy can substantially lower the computational cost while ensuring that the number of network input features is not decreased.

Graph convolution is the aggregation of information. Taking the nodes of the spatial dimension as an example, the first graph convolution aggregates the information of the first-order neighbors, the second graph convolution aggregates the information of the second-order neighbors, and so on. Multi-layer convolution is required to complete the aggregation of the entire skeleton and form an effective connection with all the remaining nodes. In other words, as the number of convolutional layers deepens, the farther the nodes can aggregate the features, the broader the receptive field becomes. In shift graph convolution, the skeleton graph is a complete graph where the node receptive field is extended to the maximum. Consequently, there is no need for too many layers to complete the aggregation of information. Based on the above theory, the original 10 layers of the 4s Shift-GCN are reduced to 8 layers. Specifically, the four-stream network first trains 3 layers respectively for each stream, and then trains 5 layers after fusion, which effectively decreases the number of parameters and computational cost.

3.3 Efficient channel attention mechanism

The ECA module judges the importance of each channel through the local cross-channel interaction strategy without dimensionality reduction and assigns corresponding weights to them. Embedding this module behind the spatiotemporal shift graph convolution can effectively enhance the weights of key nodes. The specific process is shown in Figure 3.

Given the skeleton feature $X \in \mathbb{R}^{N\times T\times C}$, ECA maps the global spatiotemporal information to each channel through global average pooling (GAP), captures the information of channel direction, and outputs an attention vector through the nonlinear layer. Finally, each channel of the input feature is weighted by multiplying the attention vector with the corresponding element in the input feature.

The channel direction information is captured by obtaining the information of k neighbors of each channel, and all channels share learning parameters. This strategy can be implemented using a 1D convolution with kernel size k:

$$\tilde{X}_A = X \cdot \sigma(Conv1D_k(GAP(X))) \tag{3}$$

Considering that the value of channel C will change after shift graph convolution, there is a reasonable proportional mapping between the size of channel interaction range k and channel C theoretically, and assuming there is a simple linear relationship between C and k:

$$C = \phi(k) = a \times k - b \tag{4}$$

But the linear relationship expression is too limited, and channel dimension C is usually set to a power of 2. Thus, it is assumed that there is a nonlinear mapping between C and k:

$$C = \phi(k) = 2^{(a \times k - b)} \tag{5}$$

Then, k can be calculated adaptively according to C:

$$k = \psi(C) = |\frac{log_2(C)}{a} + \frac{b}{a}|_{odd} \tag{6}$$

where a and b are both hyperparameters and are set to 2 and 1, respectively. $|x|_{odd}$ denotes the nearest odd number to x. Through mapping φ, each channel can adaptively select the size of the 1D convolution kernel, carry out information exchange in a suitable range, and have better interactive performance.

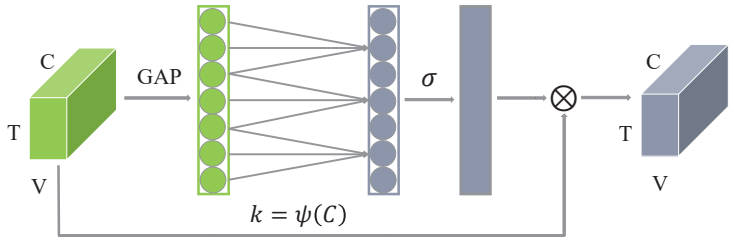


Figure 3: Efficient channel attention (ECA) module(Wang [36])

4 Experiments

4.1 Datasets and implementation details

NTU RGB+D. NTU RGB+D dataset [37] contains 56880 data samples, a total of 60 types of actions, taken by 3 Microsoft Kinect V2 cameras from different angles. This dataset is divided into training and testing sets using two benchmarks: (1) cross-subject (X-sub) is divided according to character ID, and (2) cross-view (X-View) is divided according to the camera ID.

NTU-120 RGB+D. NTU-120 RGB+D dataset [38] is an expanded version of NTU RGB+D, with 57600 additional video samples, and 60 classes of actions. This dataset uses two benchmarks when dividing the training and testing sets: (1) cross-subject (X-sub) is divided according to character ID, and (2) cross-setup (X-setup) is divided according to the distance and height of the camera.

Northwestern-UCLA. The Northwestern-UCLA dataset [39] contains 1494 video samples, and 10 types of actions, captured by three Kinect cameras. The evaluation benchmark is the same as [24]. The training and testing sets are divided according to camera ID.

Training. The batch size of the three datasets in the experiments is 16, and the maximum number of training times is 140. The stochastic gradient decent (SGD) initial learning rate is set to 0.1 and

divided by 10 at the 60-th, 80-th and 100-th epochs. All experiments are performed on two GTX 1080 GPUs.

Evaluation indicators. The model complexity is measured by parameters and FLOPs. The parameters are the memory resources consumed by the model and usually represent its space complexity. The FLOPs are the number of floating-point operations, i.e., the number of multiplication and addition operations in the model, usually representing its time complexity.

4.2 Ablation Study

Shallow architecture for multi-stream early fusion. In order to determine the specific structure of the network, the control variable method is adopted to optimize the network gradually. As shown in Figure 4, the convolutional network layers are divided into two parts, the upper half is four-stream, and the lower half is one-stream. The convolutional layers with the same number of input and output channels are denoted as S_1, S_2, and S_3, respectively. The number of layers of the network is gradually determined by changing the number of layers of $S_j, j = 1, 2, 3$. The experiment details are shown in Table 1. The results demonstrate that the network performance is optimal when $S_1 = 1$, $S_2 = 1$ and $S_3 = 2$. As shown in Table 2, compared with 4s Shift-GCN, our method effectively lowers the number of parameters and reduces computational cost.

Table 1: The best values of S_1, S_2, and S_3 are determined step by step using the control variable method on NTU RGB+D X-sub task

Const.	Var.	X-sub (%)
$S_2 = 2, S_3 = 2$	$S_1 = 0$	88.5
	$S_1 = 1$	**88.9**
	$S_1 = 2$	88.5
$S_1 = 1, S_3 = 2$	$S_2 = 0$	88.8
	$S_2 = 1$	**90.1**
	$S_2 = 2$	88.9
$S_1 = 1, S_2 = 1$	$S_3 = 1$	89.9
	$S_3 = 2$	**90.1**
	$S_3 = 3$	89.7

Table 2: Comparisons with 4s Shift-GCN on NTU RGB+D X-sub task

Methods	Layers (layer)	X-sub(%)	Param.(M)	FLOPs(G)
4s Shift-GCN [29]	10	**90.7**	2.76	10.0
L-SGCN (ours)	**8**	90.1	**0.46**	**2.8**

Efficient channel attention mechanism. As shown in Table 3, the control variable method is used to obtain the optimal position and number of attention mechanisms in the network. There are three cases where ECA is added: after the spatial shift graph convolution, after the temporal shift graph convolution, and after both the spatial and the temporal shift graph convolution. The results show that adding ECA after temporal shift graph convolution exhibits the best performance.

Table 3: The best position of ECA on NTU RGB+D X-sub task

ECA position	X-sub(%)
After spatial shift graph convolution	90.1
After temporal shift graph convolution	**90.4**
After spatial & temporal shift graph convolution	90.2

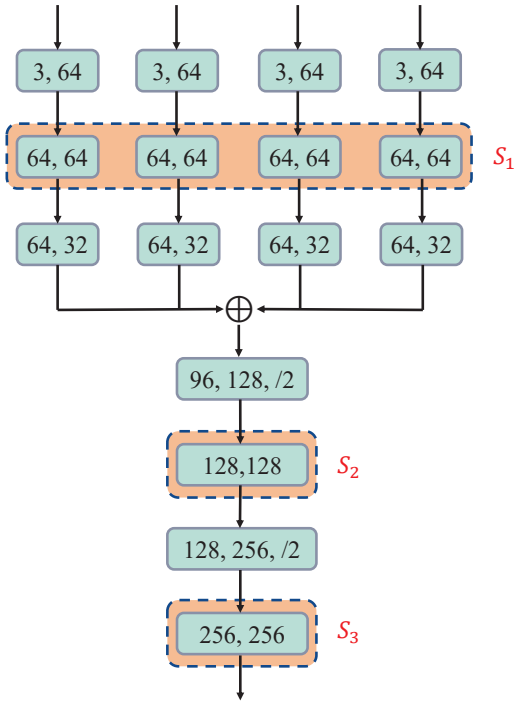


Figure 4: Network structure division

Results in Table 4 demonstrate that ECA improves our model to varying degrees without increasing the parameter number or computational cost.

Table 4: Comparisons of LA-SGCN with and without ECA module on NTU RGB+D and NTU-120 RGB+D

Methods	**X-sub** (%)	**X-view** (%)	**X-sub120** (%)	**X-set120** (%)	**Param.** (M)	**FLOPs** (G)
LA-SGCN w/o ECA	90.1	95.6	85.6	85.8	0.46	2.8
LA-SGCN	**90.4**	**96.0**	**86.8**	**87.3**	**0.46**	**2.8**

4.3 Comparison with previous models

To verify the generalizability and superiority of the proposed LA-SGCN model, we conduct experiments on NTU RGB+D, NTU-120 RGB+D, and Northwestern-UCLA datasets, all of which are compared with RNNs, CNNs and GCNs-based methods, as shown in Table 5, Table 6, and Table 7, respectively. The evaluation indicators are accuracy, parameters, and FLOPs. On the NTU RGB+D dataset, 90.5% accuracy is achieved on the X-sub benchmark and 96% accuracy is achieved on the X-view benchmark. On the NTU-120 RGB+D dataset, 86.8% accuracy is achieved on the X-sub benchmark and 87.3% accuracy is achieved on the X-set benchmark. On the NTU RGB+D and NTU-120 RGB+D datasets, 0.46M parameters and 2.8G FLOPs are realized. On the Northwestern-UCLA dataset, 95.7% accuracy, 0.43M parameters, and 0.2G FLOPs are achieved.

In terms of accuracy, LA-SGCN exhibits a considerable improvement compared with previous mainstream models. Compared to 4s Shift-GCN [29], the accuracy is very close on the NTU RGB+D and NTU-120 RGB+D datasets, and it is improved by 1.1% on the Northwestern-UCLA dataset. Compared with other lightweight models such as SGN [28], ResGCN-N51 [30], MSSF-GCN [31], FLAGCN [32], NLB-ACSE [33], and 2s SparseShift-GCN [34], higher accuracy is achieved on the three datasets.

In terms of model complexity, LA-SGCN achieves lower complexity compared with previous mod-

Table 5: Comparisons with different models on NTU RGB+D dataset

Methods	**X-sub (%)**	**X-view(%)**	**Param.(M)**	**FLOPs(G)**
HBRNN [4]	59.1	64.0	-	-
TCN [12]	74.3	83.1	-	-
Synthesized CNN [13]	80.0	87.2	-	-
3scale ResNet 152 [14]	84.6	90.9	-	-
ST-GCN [16]	81.5	88.3	3.10	-
CA-GCN [21]	83.5	91.4	-	-
2s AS-GCN [17]	86.8	94.2	6.99	27.0
3s RA-GCN [25]	87.3	93.6	6.21	-
2s AGCN [18]	88.5	95.1	6.94	35.8
SGN [28]	89.0	94.5	0.69	-
ResGCN-N51 [30]	89.1	93.5	0.77	-
2s AGC-LSTM [19]	89.2	95.0	22.89	54.4
PL-GCN [24]	89.2	95.0	-	-
FLAGCN [32]	89.4	94.8	0.83	4.1
NAS-GCN [22]	89.4	95.7	6.57	-
MSSF-GCN [31]	89.5	96.2	-	-
ST-TR [23]	89.9	96.1	-	-
4s DGNN [20]	89.9	96.1	24.83	126.8
4s Shift-GCN [29]	**90.7**	**96.5**	2.76	10.0
LA-SGCN (ours)	90.5	96.0	**0.46**	**2.8**

Table 6: Comparisons with different models on NTU-120 RGB+D dataset

Methods	**X-sub (%)**	**X-set(%)**	**Param.(M)**	**FLOPs(G)**
ST-LSTM [5]	55.7	57.9	-	-
GCA-LSTM [6]	58.3	59.2	-	-
RotClips+MTCNN [15]	62.2	61.8	-	-
SGN [28]	79.2	81.5	0.69	-
3s RA-GCN [25]	81.1	82.7	6.25	-
FLAGCN [32]	81.6	82.9	0.83	4.1
ST-TR-agcn [23]	82.7	84.7	-	-
ResGCN-N51 [30]	84.0	84.2	0.77	-
2s AGCN [18]	84.2	85.5	6.94	35.8
MSSF-GCN [31]	84.4	86.1	-	-
4s Shift-GCN [29]	85.9	**87.6**	2.76	10.0
2s SparseShift-GCN [34]	86.1	87.5	-	7.7
LA-SGCN (ours)	**86.8**	87.3	**0.46**	**2.8**

Table 7: Comparisons with different models on Northwestern-UCLA dataset

Methods	**Top-1 (%)**	**Param.(M)**	**FLOPs(G)**
HBRNN-L [4]	78.5	-	-
Ensemble TS-LSTM [9]	89.2	-	-
2s AGC-LSTM [19]	93.3	-	10.9
4s Shift-GCN [29]	94.6	2.56	0.7
NLB-ACSE [33]	95.3	1.21	-
LA-SGCN (ours)	**95.7**	**0.43**	**0.2**

els. Compared with 4s Shift-GCN, the number of parameters is 6× less on all three datasets, whereas the computational cost is 3.6× less on the NTU RGB+D and NTU-120 RGB+D datasets, and 3.5× less on the Northwestern-UCLA dataset. Especially compared with GCN-based lightweight models [28, 30, 32–34], fewer parameters and lower computational cost are achieved on the three datasets.

In summary, the proposed LA-SGCN model exhibits obvious improvements in the three evaluation indicators of accuracy, space complexity, and time complexity, which proves that our method is lightweight. The low parameter count and the low computational cost improve the training efficiency of the model, so that the model can be trained in some small devices. Lower model complexity fosters the development of behavior recognition in mobile devices, enabling them to perform tasks like real-time monitoring.

5 Conclusions

This paper designs a lightweight attention shift graph convolutional network, which achieves fast and efficient human action recognition and addresses the issue of the high precision network's large size. First, shift convolution is introduced into graph convolution to enlarge the receptive field of the nodes, realizing the lightness of graph convolution. Meanwhile, the shallow architecture for multi-stream early fusion not only makes full use of feature information but also distinctly decreases the complexity of the model. Moreover, the efficient channel attention mechanism is suitable for shift graph convolution and can effectively capture crucial spatiotemporal features in the channel dimension. Experimental results indicate that our model achieves the desired effect and has great advantages over other models of the same type. However, this study still has some limitations. The process of using the control variable method to find the optimal result is too cumbersome. In a subsequent work, we will improve the shallow architecture for multi-stream early fusion so that it can adaptively determine the depth and width of the network. At the same time, the datasets used are complete skeleton data, but in real life, human body parts may be obscured. In the future, we will further explore the realization of lightweight human action recognition on incomplete skeleton data to improve the generalization ability of the model, so that it can be better applied in real scenes.

Funding

This study was supported in part by the Xinjiang Uygur Autonomous Region University Scientific Research Project (Key Natural Science Project) (No. XJEDU2021I029), the Natural Science Foundation of Xinjiang Uygur Autonomous Region (No. 2022D01A59), the National Natural Science Foundation of China (No. U20A20167), and Project of Hebei Key Laboratory of Software Engineering (No. 22567637H).

Author contributions

The authors contributed equally to this work.

Conflict of interest

The authors declare no conflict of interest.

References

[1] R. Y. Lee, T. Y. Chai, S. Y. Chua, Y. L. Lai, Y. W. Sim, and S. C. Haw, "Cashierless checkout vision system for smart retail using deep learning," *Journal of System and Management Sciences*, vol. 12, no. 4, pp. 232–250, 2022.

[2] D. Lai, S. L. Lew, and S. Y. Ooi, "Mobile interactive system in virtual classroom based on tpack: A study from students' perspectives," *Journal of Logistics, Informatics and Service Science*, vol. 9, no. 3, pp. 159–171, 2022.

[3] Z. J. Khow, M. K. O. Goh, C. Tee, and C. Y. Law, "A yovo5 based real-time helmet and mask detection system," *Journal of Logistics, Informatics and Service Science*, vol. 9, no. 3, pp. 97–111, 2022.

[4] Du, Yong and Wang, Wei and Wang, Liang (2015). Hierarchical recurrent neural network for skeleton based action recognition, *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1110–1118, 2015.

[5] Liu, Jun and Shahroudy, Amir and Xu, Dong and Wang, Gang (2016). Spatio-temporal lstm with trust gates for 3d human action recognition, *European conference on computer vision*, 816–833, 2016.

[6] Liu, Jun and Wang, Gang and Hu, Ping and Duan, Ling-Yu and Kot, Alex C (2017). Global context-aware attention lstm networks for 3d action recognition, *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1647–1656, 2017.

[7] W. Zhu, C. Lan, J. Xing, W. Zeng, Y. Li, L. Shen, and X. Xie, "Co-occurrence feature learning for skeleton based action recognition using regularized deep lstm networks," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 30, no. 1, 2016.

[8] C. Si, Y. Jing, W. Wang, L. Wang, and T. Tan, "Skeleton-based action recognition with spatial reasoning and temporal stack learning," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 103–118.

[9] Lee, Inwoong and Kim, Doyoung and Kang, Seoungyoon and Lee, Sanghoon (2017). Ensemble deep learning for skeleton-based action recognition using temporal sliding lstm networks, *Proceedings of the IEEE international conference on computer vision*, 1012–1020, 2017.

[10] Y. Du, Y. Fu, and L. Wang, "Skeleton based action recognition with convolutional neural network," in *2015 3rd IAPR Asian conference on pattern recognition (ACPR)*. IEEE, 2015, pp. 579–583.

[11] P. Wang, Z. Li, Y. Hou, and W. Li, "Action recognition based on joint trajectory maps using convolutional neural networks," in *Proceedings of the 24th ACM international conference on Multimedia*, 2016, pp. 102–106.

[12] Soo Kim, Tae and Reiter, Austin (2017). Interpretable 3d human action analysis with temporal convolutional networks, *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 20–28, 2017.

[13] Liu, Mengyuan and Liu, Hong and Chen, Chen (2017). Enhanced skeleton visualization for view invariant human action recognition, *Pattern Recognition*, 68, 346–362, 2017.

[14] Li, Chao and Zhong, Qiaoyong and Xie, Di and Pu, Shiliang (2017). Skeleton-based action recognition with convolutional neural networks, *2017 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, 597–600, 2017.

[15] Ke, Qiuhong and Bennamoun, Mohammed and An, Senjian and Sohel, Ferdous and Boussaid, Farid (2018). Learning clip representations for skeleton-based 3d action recognition, *IEEE Transactions on Image Processing*, 27(6), 2842–2855, 2018.

[16] Yan, Sijie and Xiong, Yuanjun and Lin, Dahua (2018). Spatial temporal graph convolutional networks for skeleton-based action recognition, *Thirty-second AAAI conference on artificial intelligence*.

[17] Li, Maosen and Chen, Siheng and Chen, Xu and Zhang, Ya and Wang, Yanfeng and Tian, Qi (2019). Actional-structural graph convolutional networks for skeleton-based action recognition, *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 3595–3603, 2019.

[18] Shi, Lei and Zhang, Yifan and Cheng, Jian and Lu, Hanqing (2019). Two-stream adaptive graph convolutional networks for skeleton-based action recognition, *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 12026–12035, 2019.

[19] Si, Chenyang and Chen, Wentao and Wang, Wei and Wang, Liang and Tan, Tieniu (2019). An attention enhanced graph convolutional lstm network for skeleton-based action recognition, *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 1227–1236, 2019.

[20] Shi, Lei and Zhang, Yifan and Cheng, Jian and Lu, Hanqing (2019). Skeleton-based action recognition with directed graph neural networks, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7912–7921, 2019.

[21] Zhang, Xikun and Xu, Chang and Tao, Dacheng (2020). Context aware graph convolution for skeleton-based action recognition, *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 14333–14342, 2020.

[22] Peng, Wei and Hong, Xiaopeng and Chen, Haoyu and Zhao, Guoying (2020). Learning graph convolutional network for skeleton-based human action recognition by neural searching, *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(3), 2669–2676, 2020.

[23] Plizzari, Chiara and Cannici, Marco and Matteucci, Matteo (2021). Skeleton-based action recognition via spatial and temporal transformer networks, *Computer Vision and Image Understanding*, 208, 103219, 2021.

[24] Huang, Linjiang and Huang, Yan and Ouyang, Wanli and Wang, Liang (2020). Part-level graph convolutional network for skeleton-based action recognition, *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(7), 11045–11052, 2020.

[25] Song, Yi-Fan and Zhang, Zhang and Shan, Caifeng and Wang, Liang (2020). Richly activated graph convolutional network for robust skeleton-based action recognition, *IEEE Transactions on Circuits and Systems for Video Technology*, 31(5), 1915–1925, 2020.

[26] Y.-H. Wen, L. Gao, H. Fu, F.-L. Zhang, and S. Xia, "Graph cnns with motif and variable temporal block for skeleton-based action recognition," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, no. 01, 2019, pp. 8989–8996.

[27] L. Shi, Y. Zhang, J. Cheng, and H. Lu, "Skeleton-based action recognition with multi-stream adaptive graph convolutional networks," *IEEE Transactions on Image Processing*, vol. 29, pp. 9532–9545, 2020.

[28] Zhang, Pengfei and Lan, Cuiling and Zeng, Wenjun and Xing, Junliang and Xue, Jianru and Zheng, Nanning (2020). Semantics-guided neural networks for efficient skeleton-based human action recognition, *proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 1112–1121, 2020.

[29] Cheng, Ke and Zhang, Yifan and He, Xiangyu and Chen, Weihan and Cheng, Jian and Lu, Hanqing (2020). Skeleton-based action recognition with shift graph convolutional network, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 183–192, 2020.

[30] Song, Yi-Fan and Zhang, Zhang and Shan, Caifeng and Wang, Liang (2020). Stronger, faster and more explainable: A graph convolutional baseline for skeleton-based action recognition, *proceedings of the 28th ACM international conference on multimedia*, 1625–1633, 2020.

[31] Sun, Ning and Leng, Ling and Liu, Jixin and Han, Guang (2021). Multi-stream slowFast graph convolutional networks for skeleton-based action recognition, *Image and Vision Computing*, 109, 104141, 2021.

[32] Jiang, Yujian and Yang, Xue and Liu, Jingyu and Zhang, Junming (2021). A Lightweight Hierarchical Model with Frame-Level Joints Adaptive Graph Convolution for Skeleton-Based Action Recognition, *Security and Communication Networks*, 2021.

[33] Chen, Hongbo and Li, Menglei and Jing, Lei and Cheng, Zixue (2021). Lightweight Long and Short-Range Spatial-Temporal Graph Convolutional Network for Skeleton-Based Action Recognition, *IEEE Access*, 9, 161374–161382, 2021.

[34] Zang, Ying and Yang, Dongsheng and Liu, Tianjiao and Li, Hui and Zhao, Shuguang and Liu, Qingshan (2022). SparseShift-GCN: High precision skeleton-based action recognition, *Pattern Recognition Letters*, 153, 136–143, 2022.

[35] Feng, Liqi and Zhao, Yaqin and Zhao, Wenxuan and Tang, Jiaxi (2021). A comparative review of graph convolutional networks for human skeleton-based action recognition, *Artificial Intelligence Review*, 1–31, 2021.

[36] Wang, Qilong and Wu, Banggu and Zhu, Pengfei and Li, Peihua and Zuo, Wangmeng and Hu, Qinghua (2020). Supplementary material for 'ECA-Net: Efficient channel attention for deep convolutional neural networks, *Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, IEEE, Seattle, WA, USA*, 13–19, 2020.

[37] Shahroudy, Amir and Liu, Jun and Ng, Tian-Tsong and Wang, Gang (2016). Ntu rgb+ d: A large scale dataset for 3d human activity analysis, *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1010–1019, 2016.

[38] Liu, Jun and Shahroudy, Amir and Perez, Mauricio and Wang, Gang and Duan, Ling-Yu and Kot, Alex C (2019). Ntu rgb+ d 120: A large-scale benchmark for 3d human activity understanding, *IEEE transactions on pattern analysis and machine intelligence*, 42(10), 2684–2701, 2019.

[39] Wang, Jiang and Nie, Xiaohan and Xia, Yin and Wu, Ying and Zhu, Song-Chun (2014). Cross-view action modeling, learning and recognition, *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2649–2656, 2014.

[40] Y. Wang, S. Guo, F. Zhou *et al.*, "Monte carlo method-based behavioral reliability analysis of fully-mechanized coal mining operators in underground noise environment," *Tehnički vjesnik*, vol. 28, no. 1, pp. 178–184, 2021.

[41] M. Ozkahraman and H. Livatyali, "Artificial intelligence in foreign object classification in fenceless robotic work cells using 2-d safety cameras," *Tehnički vjesnik*, vol. 29, no. 5, pp. 1491–1498, 2022.

[42] J.-S. Han, C.-I. Lee, Y.-H. Youn, and S.-J. Kim, "A study on real-time hand gesture recognition technology by machine learning-based mediapipe," *Journal of System and Management Sciences*, vol. 12, no. 2, pp. 462–476, 2022.

Journal's webpage: http://univagora.ro/jour/index.php/ijccc/

This journal is a member of, and subscribes to the principles of,
the Committee on Publication Ethics (COPE).
https://publicationethics.org/members/international-journal-computers-communications-and-control

Cite this paper as:

Li, X.; Kang, J.; Yang, Y.; Zhao, F. (2023). A Lightweight Architecture Attentional Shift Graph Convolutional Network for Skeleton-Based Action Recognition, *International Journal of Computers Communications & Control*, 18(3), 5061, 2023.
https://doi.org/10.15837/ijccc.2023.3.5061