Skeleton-Based ST-GCN for Human Action Recognition With Extended Skeleton Graph and Partitioning Strategy

QUANYU WANG, KAIXIANG ZHANG, AND MANJOTH ALI ASGHAR
School of Computer Science and Technology, Beijing Institute of Technology, Beijing 100081, China
Corresponding author: Kaixiang Zhang (zhangkaixiang015@gmail.com)
This work was supported by the Key Project of National Natural Science Foundation of China under Grant 71834001.

ABSTRACT

Skeleton-based Graph Convolutional Networks (GCN) for human action and interaction recognition have received considerable attention of researchers due to its compact and view-invariant nature of skeleton data. However, the static skeleton graph topology in conventional GCNs does not reflect the implicit relationships of non-adjacent joints, which contain vital latent information for a skeleton pose in an action sequence. Moreover, traditional tri-categorical node partitioning strategy discards much of the motion dependencies along temporal dimension for non-physically connected edges. We propose an extended skeleton graph topology along with extended partitioning strategy to extract much of the non-adjacent joint relational information in the model for robust discriminative features. Extended skeleton graph represents joints as vertices and weighted edges represent intrinsic and extrinsic relationships between physically connected and non-physically connected joints respectively. Furthermore, extended partitioning strategy divides the input graph for GCN as five-categorical fixed-length tensor to encompass maximal motion dependencies. Finally, the extended skeleton graph and partitioning strategy are realized by adopting Spatio-Temporal Graph Convolutional Network (ST-GCN). The experiments carried out over three large scale datasets NTU-RGB+D, NTU-RGB+D 120 and Kinetics-Skeleton show improved performance over conventional state-of-the-art ST-GCNs.

INDEX TERMS
Action recognition, deep learning, graph convolutional network, human skeleton, partition strategy.

I. INTRODUCTION

Skeleton data has been widely used in human action recognition in recent years due to view-invariant representation of pose structure. As opposed to RGB video data, human skeleton data as graphs are more robust and compact representation for human movements. Graph representation of skeleton data makes it insensitive to viewpoint variations, occlusions, background clutter, inter-class pose variations, lighting condition and clothing. Human skeleton data can be obtained through cameras (such as Kinect and motion sensors) or human pose estimation algorithms [1]–[3].

Earlier work in the field was motivated by image processing techniques to model shape and motion dependencies of skeleton joints using hand-crafted features. Some former studies proposed the utilization of Histograms of Oriented Optical Flow (HOF) [4], Histogram of Oriented Gradient (HOG) [5], Speeded Up Robust Feature (SURF) [6] and Scale-Invariant Feature Transform (SIFT) [7] as features to extract discriminative dependencies. Nevertheless, these features are not capable to extract spatio-temporal dependencies to encompass motion and trajectory features. An Improved Dense Trajectory (IDT) [8] method competently incorporates motion trajectories but is still unable to map strong temporal dependencies. Despite being capable of modelling the discriminative dependencies, hand-crafted features are sensitive to hyperparameters and require subtle approach during modeling. Recent studies utilize deep-learning techniques for automated feature engineering and spatio-temporal features extraction from video sequences. Recurrent Neural Networks (RNN) has achieved better performance in modeling temporal features [9]–[11] but inherently they do not model long-term dependencies efficiently. Long-Short Term Memory (LSTM) networks along with spatio-temporal cues have effectively modeled action recognition task by taking care of long-term dependencies [12]–[14]. [15], [16] use
vanilla Convolutional Neural Network (CNN) and 3D-CNN (C3D) for action classification. Deep Reinforcement Learning technique along with keyframe distillation network has also shown better performance on benchmark datasets [17]. Much of the promising study in the field is devoted to skeleton-based Graph Convolutional Networks (GCN). GCN based methods proposed in [18]–[21] use skeleton-based data graphs and graph convolutional operations for modeling spatio-temporal features. Sijie Yan et al. [18] first proposed to use the human skeleton to construct a graph and use graph convolutional to learn features for human actions recognition. Li Maosen et al. [19] extended the study and proposed Action Structure Graph Convolutional Network (AS-GCN), which uses actional-links and extended structural-links to recognize behavior. However, most of the GCNs only consider natural joint connections of human skeleton when constructing the input graph and discard much of the information implied in connections between distant non-connected joints.

### II. RELATED WORK

#### A. GRAPH CONVOLUTIONAL NETWORKS (GCN)

Graph convolutional networks (GCN) apply convolutional operations on graph data as opposed to image data in classical CNNs [18]. Many researchers have used GCN in vast array of applications due to their prominent results on graph data. There are two categories of GCN: spectral and spatial. Spectral GCN converts graph in spectral domain and apply graph fourier transform, while the spatial GCN extracts information from neighboring nodes. The method proposed in this study uses spectral GCN.

#### B. SPATIO-TEMPORAL GRAPH CONVOLUTIONAL NETWORKS (ST-GCN)

Skeleton-based human action recognition has been extensively studied using graph convolutional networks. Most of the recent and prominent work has been devoted towards spatio-temporal graph convolutional networks to capture motion and temporal dependencies in a video sequence. Traditional ST-GCN is composed of a set of ST-GCN blocks applying spatial graph convolutions and temporal graph convolutions alternatively over skeleton graph [18]. Finally, fully connected dense layers followed by SoftMax classifier predicts the action class.

Zheng Wanqiang and Punan Jing, propose skeleton-based spatio-temporal graph convolutional network (ST-GCN) [22]. Skeleton data is extracted from video using OpenPose human pose estimation algorithm and converted to skeleton graph. ST-GCN uses fixed static graph with tri-categorical partitioning scheme to extract spatial and temporal features for action classification.

Yang et al., use data-driven learning approach to generate dynamic skeleton graph along with three set partitioning. Spatial features are employed using attention-based adjacency matrix for each skeleton frame. While the temporal features are extracted from velocity semantic information. The network is implemented as attention-based generalized graph convolutional network (AG-GCN) [23].

Shi Lei et al., propose a novel skeleton-based two-stream adaptive graph convolutional network (2S-AGCN) for human action recognition from video [21]. Network processes first and second order information concurrently. First order information represents joint locations and second order information represents bone lengths and directions of human skeleton. Method achieves notable performance improvement on benchmark datasets.

Sijie Yan et al., propose a novel spatio-temporal graph convolutional network (ST-GCN) for human action classification by learning special and temporal features from data [18]. The model achieves substantial improvements on two large scale benchmark datasets.

Cheng K et al. propose a novel shift graph convolutional network (Shift-GCN) for human action classification with computationally friendly architecture [24]. Lightweight shift graph operations and lightweight pointwise convolutions makes the entire network lightweight and consume less computational operations. Moreover, shift graph operations make receptive fields adaptable during spatial and temporal convolutions.

Huang Z et al., propose spatio-temporal inception graph convolutional network (Inception ST-GCN) [25]. Method improves the performance by extracting and synthesizing scale and transformation information from different paths and levels.

All the mentioned approaches use skeleton graph based on natural human bone structure. Moreover, a three-set partitioning scheme is used to convert variable length graph to fixed sized tensor.

**FIGURE 1.** Overall system pipeline for skeleton-based human action recognition using ST-GCN.

### III. METHOD

Overall pipeline of the system is presented in Figure 1. Raw skeleton data is converted into an extended skeleton graph, details in section B. Thereafter, extended graph is fed to ST-GCN network for action classification as probability distribution over all action class space. ST-GCN is composed of multiple ST-GCN blocks, followed by a global average pooling layer and two fully connected dense layers, details in section D. Finally, a SoftMax layer outputs class scores for action classification.

Based on the above, a kind of skeletal feature information is proposed as another input, details in section G, and use the original network frame as the frame of the skeletal network branch, and finally combine the SoftMax values of the output of the two branch networks as the final network’s output predicted action label.
A. ST-GCN BLOCK STRUCTURE

The ST-GCN network comprises of two branch subnetworks with the same structures, which forms three cascaded ST-GCN blocks. The structure of ST-GCN block is shown in Figure 2. First ST-GCN block receives the extended graph as input, while the latter two blocks receive the extracted spatial and temporal features from former ST-GCN blocks. Two subnetworks Spatial Convolutional Network (Spatial-Conv) and Temporal Convolution Network (Temporal-Conv) extract the spatial dependencies from neighboring nodes and temporal dependencies from the consecutive frames, respectively. Batch Normalization (BN) layer performs the intermediate feature tensor normalization, while the ReLU layers serve as activation functions used after each feature domain type extraction. Moreover, a dropout regularization layer is used to overcome model overfitting while an additionally residual connection is provided for each block to stabilize the training process.

B. SKELETON EXTRACTION AND NOTATIONS

To construct skeleton graph, we use OpenPose [1] (human pose estimation algorithm) to detect skeleton from each video frame. Skeleton data is denoted as a set of joints, where each joint is a vector \((x, y, s)\). \(x\) and \(y\) represent spatial location of detected joint in the global image context and \(s\) denotes the confidence score. For a video sequence, skeleton data is denoted as set \(V = \{v_{ij}|i = 1, \ldots, N; j = 1, \ldots, T\}\), where \(v_{ij}\) represents \(i^{th}\) joint vector in \(j^{th}\) frame. Skeleton diagram along space-time in traditional ST-GCN is shown in Figure 3. The solid blue circles represent the body joints, hard black lines between circles represent natural body joint connections and the orange lines represent the temporal dependencies.

Skeleton data extraction using OpenPose is suitable for datasets containing different numbers of human joints. For example, in the Kinetic dataset, we use COCO model to obtain 18 joints, and in the NTU-RGB+D dataset, BODY-25 model extracts 25 joints. The study runs composition method on both datasets to achieve promising performance.

C. EXTENDED SKELETON GRAPH CONSTRUCTION

As the GCN only accepts graph at the input layer hence skeleton data is transformed into a skeleton graph structure. We represent skeleton graph as \(G = (V, E)\), where \(V\) is the set of vertices representing joints and \(E\) is the set of edges representing bones as joint pairs. GCN obtain the feature information by the adjacency matrix of the graph. In a set of human joints, it is necessary to manually construct a human joint map. However, the mainstream action recognition network only use the natural connections of the human body to build the graph. Moreover, all the edges in the skeleton graph are assigned a value of 1, while discarding all the implicit connections between non-adjacent joints. Realizing the importance of human body movement as linkage of all related nodes, in this study we connect all related nodes to construct skeleton graph. At the same time, considering there is stronger linkage between adjacent joints as to farther joints, this paper assigns values to the edges on the human skeleton graph according to the joint distances. The farther away the joints, the smaller the edge weight, as shown on the left in Figure 4. In Figure 4, green lines represent connection between a single joint and non-adjacent joint, and the number on line represents the weight of the corresponding connection, and the right subfigure shows the composition of a human joints in space and time.

The adjacency matrix for extended graph is represented as \(A \in \{x \in \mathbb{R}|0 \leq x \leq 1\}^{N \times N}\). While \(A_{x,y}\) represents the weighted connection between \(x^{th}\) joint and \(y^{th}\) joint. Weight for naturally connected joints is assigned as 1, while the weight value for non-physically connected joints is based on distance between the joints. Mathematical representation of final adjacency matrix is given as in equation (3).

\[
A_{x,y} = \frac{\sum_{z=0}^{N-1} D_{x,z}}{D_{x,y}} + I \tag{1}
\]

where, \(D_{x,y}\) represents the distance between the joint \(x\) and \(y\). \(I\) is the diagonal identity matrix for incorporating self-loop connections (with weight of 1) to each of the joint.

D. EXTENDED PARTITIONING STRATEGY

Current state-of-the-art skeleton-based ST-GCNs mostly rely on the designing of the mapping function by divided joints into three categories. The scheme works well but only considers the information contained in adjacent joint and discards the farther and physically disconnected joint parts. In fact, when humans make movements, even distant joints, such as hands and feet, will basically make a series of movements in conjunction. Therefore, we propose a new partitioning
strategy, that considers adjacent and distant joint information at the same time.

In conventional ST-GCN, three types of partitioning mapping functions are designed, namely: uni-labeling, distance, and spatial configuration partitioning methods. Experiments show that the third type of method works best. In most of the literature ST-GCNs divides nodes into three subsets: 1) center node/root node 2) concentric subset: the distance between the center point and the adjacent nodes from the center of gravity; 3) Eccentric subsets: the subsets of the neighboring nodes that are farther from the center of gravity than the center point. Three subset partitioning schemes is shown in Figure 5. ST-GCN takes the neck node as the center of gravity of the skeleton. Such partitioning scheme only considers few nodes closest to the center point and is disinclined to obtain information about farther nodes.

In this article we propose a better partitioning scheme by roughly dividing motion of body parts into concentric and eccentric motions. The average of coordinates for relevant node in the skeleton is considered as center of gravity. While the nodes around the center are partitioned into five categories. The partition groups are represented as partition set $\rho = \{\tau = 0, \xi = 1, \varphi = 2, \lambda = 3, \mu = 4\}$: 1) $\tau$: the center point itself; 2) $\xi$: nodes that are close to the center point and closer to the center of gravity than the center point; 3) $\varphi$: nodes that are near the center point and farther from the center of gravity than the center point; 4)$\lambda$: nodes that are far from the center point and closer to the center of gravity than the center point; 5)$\mu$: from the center A node that is far away and is farther from the center of gravity than the center of gravity. The partitioning scheme is repressed as:

$$l_i (v_j) = \begin{cases} \tau & \text{if } r_j = r_i \\ \xi & \text{if } d_{ij} = 1 \text{ and } r_j < r_i \\ \varphi & \text{if } d_{ij} = 1 \text{ and } r_j > r_i \\ \lambda & \text{if } d_{ij} = 2 \text{ and } r_j < r_i \\ \mu & \text{if } d_{ij} = 2 \text{ and } r_j > r_i \end{cases}$$

(2)

where $v_j$ is the adjacent node, $r_i$ is the distance from vertex $i$ to the center of gravity, $r_j$ is the distance from vertex $j$ to the center of gravity and $d_{ij}$ represents the distance from node $i$ to node $j$.

The extended partition scheme is shown in Figure 6. The red point represents $\tau$ joint, green points represent $\xi$ joints, orange points represent $\varphi$ joints, purple points represent $\lambda$ joints and yellow points represent $\mu$ joints. Different partitioning strategies are equivalent to changing the size of the convolution kernel. At the same time, the proposed partition strategy can cover most of the motion joint points in the human skeleton, such as arms and thighs, so it can no longer be limited to adjacent joint points, but can be effectively extended to other joints.

E. GRAPH CONVOLUTION OPERATIONS

Given the extended graph, a multi-layer convolution operation is used over the graph to extract high-dimensional information in spatial and temporal domains. Convolution operation over $v_i$ in spatial dimension is defined as:

$$f_{out} (v_i) = \sum_{v_j \in D_i} \frac{1}{Z_{ij}} f_{in} (v_j) \cdot w (l_i (v_j))$$

(3)

where $f$ represents the feature matrix obtained from the human body joint map, $v$ represents the vertex map in the human skeleton, and $Z_{ij}$ is a normalization factor. $D_i$ represents the sampling area of the vertex graph, which is the nearest point of the vertex in ST-GCN. $w$ represents a weighting function, which is similar to the original convolution operation. Note that the matrix is fixed but the number of vertices is variable length. $l_i$ represents a mapping function.
to map each vertex to a different weight matrix, which is a partition strategy. Specifically, this strategy sets the size of convolution kernel to 5 based on experience and divides it into 5 types of subsets: \( \rho = \{\tau, \xi, \psi, \lambda, \mu\} \).

But the feature map is \( C \times T \times N \) tensor, where \( N \) represents the number of vertices, \( T \) represents the number of frames in an action video, and \( C \) represents the number of channels. To apply graph convolution over human skeleton graph, equation (1) can be transformed as:

\[
    f_{out} = \sum_k K_v \times W_k (f_{in} A_k) \odot M_k
\]

where \( K_v \) is the kernel size set to 5. Among them, \( A_k = \Lambda_k^{-\frac{1}{2}} \widetilde{A} \Lambda_k^{-\frac{1}{2}} = \Lambda_k^{-\frac{1}{2}} (\Lambda_k - A_k) \Lambda_k^{-\frac{1}{2}} = I_n - \Lambda_k^{-\frac{1}{2}} A_k \Lambda_k^{-\frac{1}{2}} \) is a Laplacian matrix, \( \widetilde{A}_k \) is the addition of the \( N \times N \) adjacency and identity matrix, \( A_k \) indicates whether the vertex \( v_j \) is in the subset \( S_k \) of vertex \( v_i \), \( M_k \) is \( N \times N \) attention map. Based on the above partitioning strategy of dividing joint points into 5 categories, \( M_k \) is set to 5. \( G_k \) is an attention mechanism to indicate the importance of each vertex in the calculation, and \( W_k \) is a weight vector of \( C_{out} \times C_{in} \times 1 \times 1 \) including \( 1 \times 1 \) convolution operation, \( \odot \) represents the dot product operation.

Since the partition strategy remains same in temporal dimension, the vertex neighborhood also remains fixed. Hence graph convolution operation in time domain is similar to the classic graph \( K_v \times 1 \) convolution, where \( K_v \) is the size of the convolution kernel.

**F. ST-GCN NETWORK IMPLEMENTATION**

This paper adopts the graph convolution proposed by Kipf and Welling paper [26]. The skeleton graph for a single frame is represented by the adjacency matrix \( A \) and the identity matrix \( I \). The network can be realized by the following formula:

\[
    f_{out}^{(l+1)} = \sigma \left( \tilde{D}^{-\frac{1}{2}} (A + I) \tilde{D}^{-\frac{1}{2}} f_{in}^{(l)} W^{(l)} \right)
\]

where, \( \tilde{D}^{-\frac{1}{2}} = \sum_j (A^{jk} + I) \) denotes the degree matrix. Combining spatio-temporal dimension information, the input feature map can be expressed as \( (C, V, T) \) dimensions. For partitioning strategies with multiple subsets, such as the extended partitioning strategy described above, the adjacency matrix can be decomposed into multiple matrices, that is, \( A + I = \sum_j A_j \), so the degree matrix will also become \( \tilde{D} = \sum_k A_{jk} \). Therefore, the above formula can be transformed into:

\[
    f_{out}^{(l+1)} = \sigma \left( \sum_j D_j^{-\frac{1}{2}} A_j D_j^{-\frac{1}{2}} f_{in}^{(l)} W^{(l)} \right)
\]

In accordance with the extended partition scheme, \( j \) is set to 5, while weight vectors of multiple output channels are superimposed to form a weight matrix \( W \).

**G. SKELETAL BRANCH NETWORK**

In ST-GCN, only the coordinate information of the joint point is used for each vertex, but the bone feature information is also important. 2S-AGCN [21] added the skeleton information to the action recognition based on graph convolution for the first time. They defined the human joint points closer to the center of gravity of the bones as the starting point of the bone and the farthest joint point as the end point. For the joint source point \( v_1 = (x_1, y_1, z_1) \) and the joint end point \( v_2 = (x_2, y_2, z_2) \), define the bone information as:

\[
    e(v_1, v_2) = (x_2 - x_1, y_2 - y_1, z_2 - z_1)
\]

But this expression is too simplistic.

This paper proposes the characteristics of bone weight, bone direction and bone length.

1) **BONE LENGTH**

For the bone vector formed by the joint source point \( v_1 = (x_1, y_1, z_1) \) and the joint end point \( v_1 = (x_1, y_1, z_1) \), the length of the bone can be represented by the modulus of the vector:

\[
    m = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}
\]

2) **BONE DIRECTION**

The direction of the bone can be represented by the direction of the vector connected by the joint source point and the joint end point:

\[
    \cos \alpha, \cos \beta, \cos \gamma = \left( \frac{x_2 - x_1}{m}, \frac{y_2 - y_1}{m}, \frac{z_2 - z_1}{m} \right)
\]

3) **BONE WEIGHT**

The distance between the center of gravity of the bone and the center of gravity of the human is used to represent the weight of the bone. The center of gravity of the bone \( v_i \) can be represented by the center point of the joint source point and the joint end point, that is, \( v_i = (x_1, y_1, z_1) = ((x_2 - x_1)/2, (y_2 - y_1)/2, (z_2 - z_1)/2) \), so the bone weight can be calculated as:

\[
    v_{wi} = \sqrt{(x_i - x_k)^2 + (y_i - y_k)^2 + (z_i - z_k)^2}
\]

where \( v_{wi} \) represents the weight of the i-th bone, \( (x_k, y_k, z_k) \) represents the coordinate point of the human center of gravity.

Since there are no rings in a single human skeleton diagram, each bone can have a unique joint source point. However, the source joint points will not be assigned to any bone, so the human joint points will have one more input than the bones. In order to input the two networks uniformly, add a section of empty bones to the source joint points. The characteristics of the empty bones are all 0. Because the input dimensions of the original network and the branch of the skeletal network are the same, the original network frame is used as the frame of the skeletal network branch, thereby constructing a skeletal network branch whose input is the skeletal feature.
IV. DATASETS AND EVALUATION INDICATORS

We test the network on two large scale benchmark datasets, NTU-RGB+D and Kinetics-Skeleton.

A. NTU-RGB+D [27]

It is currently the most widely used dataset in the field of human action recognition consisted of 56,000 action clips, divided into 60 categories. The clips were taken in the laboratory with three cameras. Each joint in the segment is represented by 3D coordinate (x,y,z). There are 25 joints in each frame for maximum of two persons. The study performs training and testing on the two benchmarks (X-Subject and X-View) of the dataset. X-Subject uses 40,320 segments as the training set, and the remaining segments are reserved for the test set. While in X-View, 37,920 clips collected from camera 2 and 3 are used as the training set, while the remaining clips collected from camera 1 are used as the test set.

B. NTU-RGB+D 120 [32]

It is an extension dataset of NTU-RGB+D dataset, which is collected from 106 distinct subjects and contains more than 114 thousand video samples and 8 million frames, divided into 120 categories.

C. SKELETON-KINETICS [28]

Contains 300,000 video clips, with annotated actions divided into 400 categories. Each video segment lasts 10s. Each human joint is a two-dimensional coordinate information along with confidence score for each person. Single skeleton is represented with 18 different joints. The study divides the dataset into training set and test set with 240,000 and 20,000 instances respectively.

V. EXPERIMENTS AND RESULTS

The network is realized on standalone system with 1080Ti graphics card using PyTorch [29] framework. The initial learning for the training is set to 0.025, thereafter the rate is reduced by 0.1 coefficient in the 10\textsuperscript{th} and 50\textsuperscript{th} training stages. The total number of training epochs are 80.

In this section, four sets of control experiments are designed and tested on the above two large scale datasets. This article proposes two different division strategies to test according to the author’s suggestion. Among them, the improvement of the extended graph, the extension of the partition strategy and the skeletal bone network will be compared with the original ST-GCN to verify the effectiveness of the proposed method for network improvement. Finally, the holistic network performance is compared with the current mainstream action recognition networks to verify the effectiveness and accuracy of extended graph convolutional network.

The first set of experiments test the effectiveness of the extended graph using the original space partition strategy. Among them, the accuracy of top-1% and top-5% are compared on the two datasets. On the NTU-RGB+D dataset, this article uses the X-Subject and X-View divisions for experiments. Table 1 and Table 2 show the experimental results on two datasets.

From Table 1 and Table 2, the accuracy of the extended graph in this article on the two datasets, including top-1 and top-5 accuracy, has a certain improvement compared to the original graph convolutional network. In Kinetics dataset, the accuracy of using the extended map is increased by 1.2%. While in the NTU-RGB+D dataset, the accuracy of using the X-Subject classification method is increased by 0.9%, and the accuracy of using the X-View classification method is increased by 1.3%.

The second set of experiments uses the original method of constructing a human skeleton graph, and the extended partition strategy proposed in this article to conduct experiments on two datasets to verify the effectiveness of the extended partition strategy. The accuracy of top-1 and top-5 is compared on both datasets. Table 3 and Table 4 show the experimental results on two datasets.

It is evident from Table 3 and Table 4 that the accuracy of extended partitioning strategy has improved on both datasets. In Kinetics dataset, extended partition strategy has increased the accuracy of top-1 by 2.1%, and the accuracy of top-5 by 2.8%. While in the NTU-RGB+D dataset, it has increased the accuracy by 4.9% and 3.5% using X-Subject and X-View divisions.
The third set of experiments combines the proposed skeletal branch network, to carry out tests on both datasets. Table 5 shows the experimental results.

Among them, 2S-GCN represents the experimental results of only joining the skeletal branch network, and 2S-EGCN represents the experimental results of applying the extended human skeleton graph and the extended partition strategy to the dual-stream action recognition network. When only the skeleton branch network is added, the Kinetics-Skeleton dataset and the NTU-RGB+D dataset are greatly improved compared to the original ST-GCN network. Specifically, it is improved on the Kinetics-Skeleton dataset. In the X-Subject subset, it increased by 4.8%, and it increased by 3.9% in the X-View subset. At the same time, the adding extended human skeleton graph and extended partition strategy can also effectively improve the recognition accuracy of the dual-stream network, of which the highest is increased by 3.3% in the X-View subset.

Figure 7 is a test result of the extended proposed method and the original ST-GCN network. Specifically, the golf short-cut video in the Kinetics-Skeleton test set is selected, and the test result of the 50th frame in the video is selected. Figure 7(a) is the test result of the original network. Figure 7(b) is the test result of the extended network. The circles in the figure represent the features of the joints involved in action recognition, and the size of the circles indicates the importance of the joints in action recognition. Compared with the original ST-GCN, the extended proposed method can extract more global action information, which proves the effectiveness of the extended human skeleton map and extended partition strategy.

![Figure 7](image_url)

**FIGURE 7.** Some test results of the extended network proposed in this chapter and the original network are shown.

In Table 6, the top-1 and top-5 accuracy of this network and other action recognition networks are compared in the Kinetics dataset, and the top-1 accuracy of two different division strategies are compared in the NTU-RGB+D dataset. It can be seen that the test accuracy of the proposed network in the two datasets has greatly improved compared with conventional ST-GCNs. In Kinetics dataset, the accuracy of the proposed model has increased by 6.4%. While in NTU-RGB+D dataset, it has increased the accuracy by 7.6% and 7.2% using X-Subject and X-View divisions. At the same time, the model achieves the highest accuracy among all models in the X-Subject classification. Besides, we also evaluate the 2S-EGCN(ours) on the NTU RGB+D 120 dataset[32], the top-1 accuracy of the cross-subject evaluation criteria is 66.01%, and the top-1 accuracy of the cross-setup evaluation criteria is 76.26%. The accuracy of the provided method has decreased greatly in the data set, but it still has good generalization ability.

**VI. CONCLUSION**

Conventional ST-GCN methods construct the skeleton graph representing only the natural human bone structure. We propose a different method to construct extended skeleton graph, by considering the role of adjacent and non-adjacent joints within an action sequence. Further, a five-subset partitioning scheme is proposed to overcome limitations of three partition scheme by extracting maximum spatial and temporal dependencies. Extended skeleton graph and extended partition scheme are integrated with ST-GCN for human action recognition. The proposed network has achieved significant improvements on two large scale datasets. The research results in this article will serve as the basis for the follow-up study of skeleton-based action recognition.

**REFERENCES**

[1] Z. Cao, G. Hidalgo, T. Simon, S. E. Wei, and Y. Sheikh, “OpenPose: Realtime multi-person 2D pose estimation using part affinity fields,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 43, no. 1, pp. 172–186, Jan. 2021, doi: 10.1109/TPAMI.2019.2929257.

[2] Y. Chen, Z. Wang, Y. Peng, Z. Zhang, G. Yu, and J. Sun, “Cascaded pyramid network for multi-person pose estimation,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 7103–7112, doi: 10.1109/CVPR.2018.00742.

[3] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh, “Convolutional pose machines,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 4724–4732, doi: 10.1109/CVPR.2016.511.
K. Yang, X. Ding, and W. Chen, “Attention-based generative graph con-

W. Zheng, P. Jing, and Q. Xu, “Action recognition based on spatial

L. Shi, Y. Zhang, J. Cheng, and H. Lu, “Two-stream adaptive graph

S. Ji, W. Xu, M. Yang, and K. Yu, “3D convolutional neural net-

Z. Wu, X. Wang, Y.-G. Jiang, H. Ye, and X. Xue, “Modeling spatial-

V. Veeriah, N. Zhuang, and G.-J. Qi, “Differential recurrent neural net-

J. Yue-Hei Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals,

V. Veeriah, N. Zhuang, and G.-J. Qi, “Differential recurrent neural net-

Y. Du, W. Wang, and L. Wang, “Hierarchical recurrent neural net-

V. Veeriah, N. Zhuang, and G.-J. Qi, “Differential recurrent neural net-

J. Yue-Hei Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals,

I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld, “Learning realistic

N. Dalal and B. Triggs, “Histograms of oriented gradients for human detec-

I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld, “Learning realistic

P. Scovanner, S. Ali, and M. Shah, “A 3-dimensional sift descriptor and

V. Veeriah, N. Zhuang, and G.-J. Qi, “Differential recurrent neural net-

Y. Du, W. Wang, and L. Wang, “Hierarchical recurrent neural net-

Y. Du, W. Wang, and L. Wang, “Hierarchical recurrent neural net-

T. N. Kipf and M. Welling, “Semi-supervised classification with graph

A. Shahroudy, J. Liu, T.-T. Ng, and G. Wang, “NTU RGB+D: A large
data set for 3D human activity analysis,” in Proc. IEEE Conf. 

W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev, M. Suleyman, and A. Zisserman, “The kinetics human video action dataset,” 2017, arXiv:1705.06950.

A. Paszke, S. Gross, S. Chintala, G. Chanran, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, “Automatic Differentiation in PyTorch,” in Proc. NIPS Autodiff Workshop, 2017, pp. 1–4.

T. S. Kim and A. Reiter, “Interpretable 3D human action analysis with temporal convolutional networks,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW), Jul. 2017, pp. 1623–1631, doi: 10.1109/CVPRW.2017.207.

B. Li, X. Li, Z. Zhang, and F. Wu, “Spatio-temporal graph routing for skeleton-based action recognition,” in Proc. 33rd AAAI Conf. Artif. Intell., 31st Innov. Appl. Artif. Intell. Conf. 9th AAAI Symp. Educ. Adv. Artif. Intell. (AAAI/IAAI/EAAI), 2010, pp. 8561–8568, doi: 10.1609/aaai.v33i01.33018561.

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, doi: 10.1109/TPAMI.2019.2916873.