# Expressive Keypoints for Skeleton-Based Action Recognition via Progressive Skeleton Evolution

Yijie Yang, Jinlu Zhang, Jiaxu Zhang, Bo Du, *Senior Member, IEEE*, and Zhigang Tu, *Senior Member, IEEE*

***Abstract*—In the realm of skeleton-based human action recognition, the traditional methods which rely on coarse body keypoints fall short of capturing subtle human actions. In this work, we propose Expressive Keypoints that incorporates hand and foot details to form a fine-grained skeletal representation, to improve the discriminative ability for existing models in discerning intricate human actions. However, the increased computational cost from processing nearly three times more joints becomes a new challenge. To address this, we present the Progressive Skeleton Evolution strategy, which significantly improves efficiency while preserving the benefits of fine-grained keypoints. The core idea involves utilizing learnable mapping matrices, semantically initialized to progressively downsample keypoints and prioritize prominent joints by allocating importance weights. Additionally, a plug-and-play Instance Pooling module is exploited to extend our approach to multi-person scenarios without surging computation cost. Extensive experimental results over seven datasets demonstrate the superiority of our method compared to the state-of-the-arts for skeleton-based human action recognition. Code has been made available at https://github.com/YijieYang23/PSE-GCN**

***Index Terms*—Skeleton-based action recognition, graph convolutional network, fine-grained representation.**

## I. Introduction

SKELETON human action recognition has become a cornerstone for numerous vision applications such as video surveillance [1], [2], human-robot interaction [3], and sports analytics [4], due to its succinct representation and robustness to variations in lighting, scale, and viewpoint. Traditional methods primarily utilize simple body keypoints defined in NTU [5], [6] and COCO [7] formats to provide sparse representations of human motion. Despite their utility, the over concise representations are constrained by missing subtle but critical details involving hand and foot movements. Consequently, existing coarse skeletal representations are limited in effectively distinguishing intricate human actions.

Recently, some approaches [8], [9], [10] have resorted to the point cloud representation to capture the detailed spatial structure of human surface, thereby enhancing the ability to recognize complex movements. However, it comes with enormously increased computation cost, detracting from the efficiency of point-based representation. Moreover, several studies [11], [12], [13] have aimed to improve the recognition accuracy by introducing object points. However, the generalization of these methods is limited especially in the human-centric scenarios where no interacted object involved.

To solve the limitations of prior works, we incorporate richer limb keypoints into body keypoints to propose a fine-grained representation called Expressive Keypoints. It emphasizes nuanced hand interactions and foot movements which are crucial to discerning subtle actions. As shown in Fig. 1a, we present various data representations that are commonly utilized. Compared to the representations of RGB images, excessive point cloud data, and coarse body keypoints, the Expressive Keypoints representation stands out for its insensitivity to viewpoints, relatively small data footprint, and ability to represent fine-grained limb details. In practice, Expressive Keypoints can be easily estimated from RGB images based on COCO-WholeBody [14] annotations, without relying on obtaining depth information from multi-view data or lab-controlled motion capture system. Experimental results demonstrate that all three baseline methods [15], [16], [17] achieve significant improvement in accuracy (+ over 6%) when replacing coarse-grained keypoints with Expressive Keypoints. However, the computation cost of directly taking Expressive Keypoints as input also scales considerably, since nearly three times more joints need to be dealt with. To enhance the computationally efficiency, we propose the Progressive Skeleton Evolution (PSE) strategy to gradually downsample the skeletal representation of Expressive Keypoints across multiple stages. This novel strategy involves the learnable mapping matrices to refine skeleton features by re-weighting and downsampling the keypoints. These mapping matrices are initialized by semantic partitioning of human topology, and iteratively optimized during training. By further introducing variable group design for different skeletal scales, skeleton features are evenly split and evolved before concatenation. PSE strategy enables effective downsampling of keypoints and nuanced modeling in groups.

Received 14 August 2024; revised 10 June 2025 and 29 July 2025; accepted 7 November 2025. Date of publication 19 November 2025; date of current version 24 November 2025. This work was supported in part by the National Key Research and Development Program of China under Grant 2024YFC3015600 and in part by the Fundamental Research Funds for Central Universities under Grant 2042023KF0180 and Grant 2042025KF0053. The associate editor coordinating the review of this article and approving it for publication was Prof. Shang-Hong Lai. *(Yijie Yang and Jinlu Zhang are co-first authors.) (Corresponding author: Zhigang Tu.)*

Yijie Yang, Jiaxu Zhang, and Zhigang Tu are with the State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan 430072, China (e-mail: tuzhigang@whu.edu.cn).

Jinlu Zhang is with the Center on Frontiers of Computing Studies, School of Computer Science, Peking University, Beijing 100871, China.

Bo Du is with the School of Computer Science, Wuhan University, Wuhan 430072, China.

Digital Object Identifier 10.1109/TIP.2025.3632229

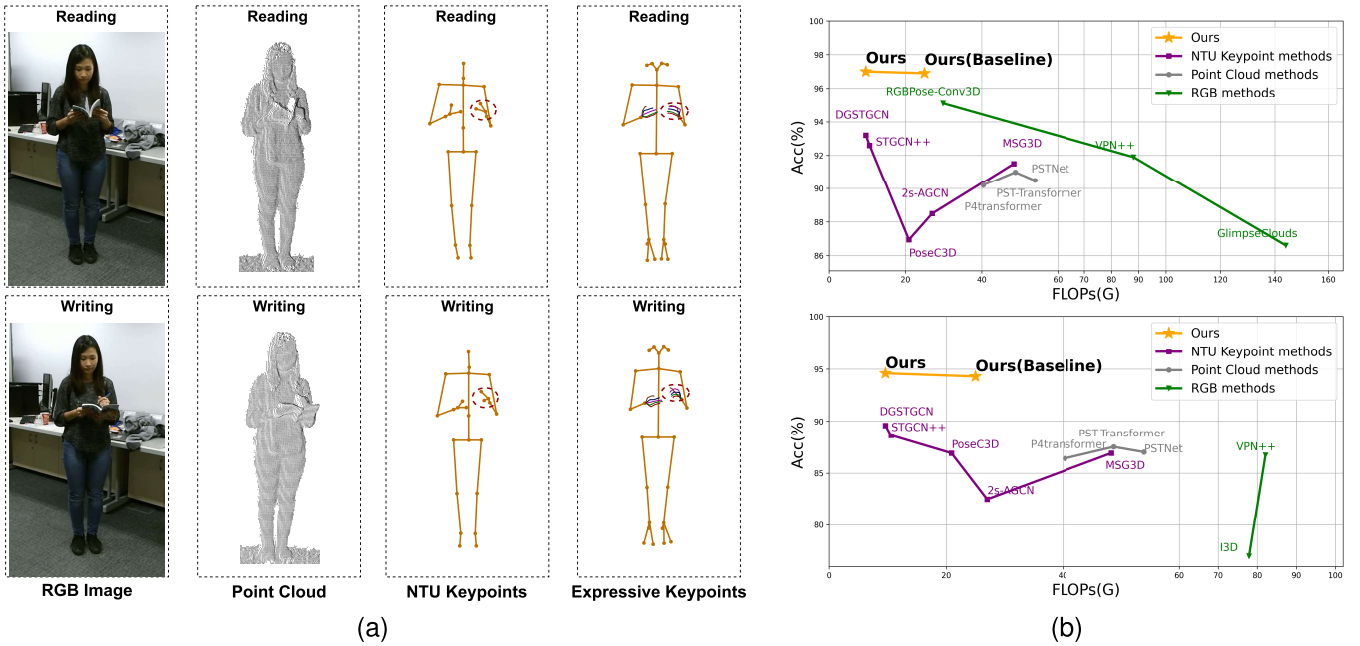



Fig. 1. **(a)** Various representations of the action *Reading* and *Writing*. **(b)** Accuracy and efficiency comparison of our method and the representative methods on NTU-60 [5] (Top) and NTU-120 [6] (Bottom).

It can be effortlessly integrated into most existing GCN-based skeleton action recognition methods, forming our PSE-GCN to efficiently process Expressive Keypoints. In experiments over four standard skeleton action recognition datasets [5], [6], [18], [19], PSE-GCN achieves comparable or even higher accuracy with much lower (less than half) FLOPs compared to its baseline GCN method.

To further validate the generalization ability of our method, we seek to evaluate our method on the general in-the-wild datasets [20], [21], [22] which include multi-person group activity scenarios. However, we find that traditional GCN methods perform feature modelling for each input person individually and conduct feature fusion in the late stage. Consequently, they have the limitation of exponentially increasing computation complexity as the number of individuals grows. Inspired by [13], we implement a lightweight Instance Pooling module before the GCN models. The key idea is to aggregate the features of multiple persons and projects them to a single skeletal representation in the early stage. By exploiting the plug-and-play Instance Pooling module, the classification of group activities can be supported without surging computation cost. This offers a viable solution for extending GCN-based skeleton action recognition methods (including our PSE-GCN) to multi-person scenarios.

In extensive experimental evaluations over the total of seven datasets [5], [6], [18], [19], [20], [21], [22], our pipeline consistently achieves the state-of-the-art across all the benchmarks (see Fig. 1b), demonstrating its superior performance and robust generalization. We find that strategically employing fine-grained keypoints enables recognizing intricate human actions with efficient computation complexity. In summary, the main contributions of our work are threefold:

- We introduce fine-grained limb details as the Expressive Keypoints representation for skeleton-based human action recognition, boosting the performance in identifying intricate cases.
- We propose the Progressive Skeleton Evolution strategy to highly promote the efficiency of the existing GCN-based skeleton action recognition methods meanwhile preserving their accuracy.
- We implement a plug-and-play Instance Pooling module to extend GCN methods to multi-person group activity scenarios without surging computation cost.

## II. Related Works

### A. Point-Based Action Recognition

Point-based action recognition methods are more robust against variations of lightning and view variation compared with RGB-based methods [23], [24], [25], [26], [27], [28]. Some works [8], [9], [10] take numerous unordered 3D point cloud as input. However, point cloud data introduces too much redundant information for learning action patterns, leading to high computation cost. Some works utilize 2D/3D keypoints [5], [7] to represent skeletal structure of human body, which commonly referred to as skeleton-based methods. Some CNN-based methods [29], [30], [31] attempt to project human body keypoints into multiple 2D pseudo-images to learn useful features, which can achieve notable performance. Among them, GCN-based methods [15], [16], [17], [32], [33], [34], [35] have been adopted frequently due to the effective representation for the graph structure [36]. More recently, transformer-based architectures [37], [38], [39], [40] have emerged as a competitive paradigm by leveraging self-attention mechanisms to capture global spatio-temporal dependencies. However, these methods often require high computational costs due to the usage of their large-sized attention maps. Nevertheless, existing skeleton-based methods use coarse-grained skeleton representation as input, leading to the challenge of discerning

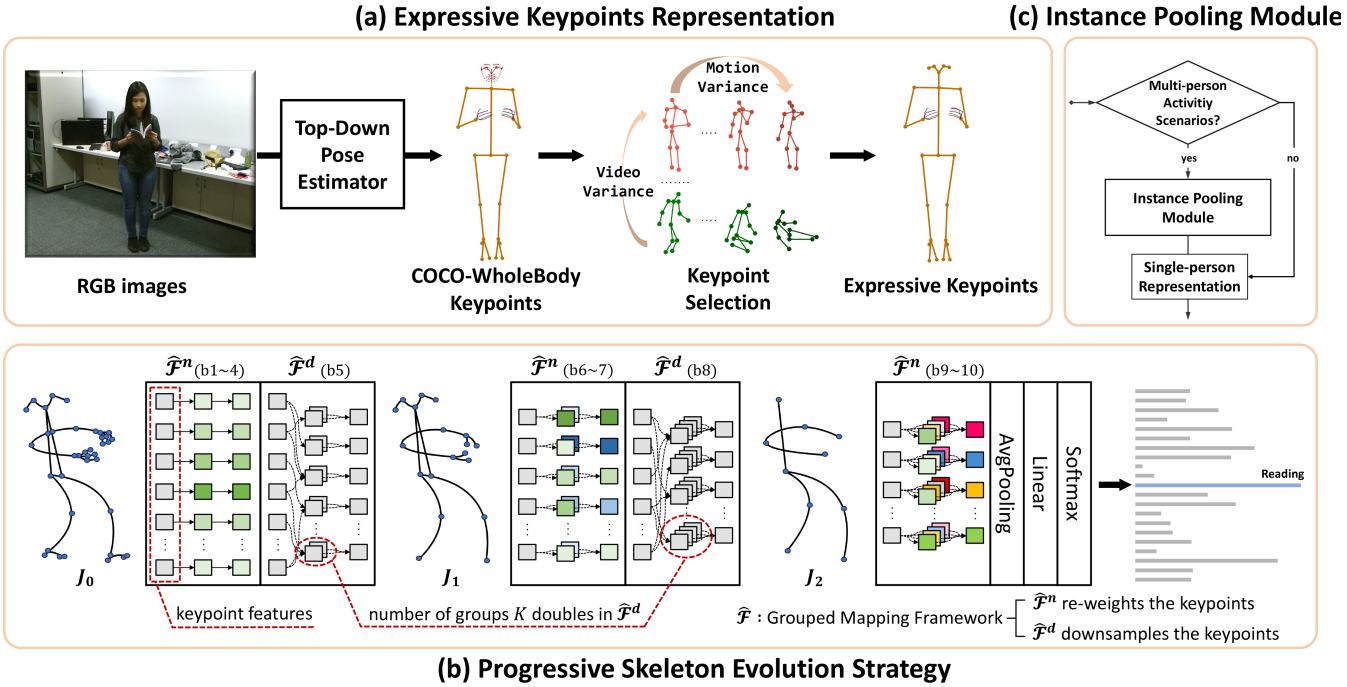



Fig. 2. **Overview of the proposed pipeline. (a)** A top-down estimator is used to extract COCO-WholeBody Keypoints from videos, and conduct keypoint selection based on statistical metrics to remove the redundant facial keypoints, forming our **Expressive Keypoints representation**. **(b)** A **Progressive Skeleton Evolution strategy** which can be integrated into most GCN methods is presented to efficiently process the Expressive Keypoints. It guides the network to progressively evolve skeletal structures in groups by re-weighting and downsampling the keypoints. **(c)** A **Instance Pooling module** is implemented to fuse the multiple instances in the early stage. We use it as an lightweight extension for evaluating our methods in general wild scenarios, which contains multi-person group activities.

complex actions, which results in limited performance. To this end, we propose to incorporate hand and foot keypoints into the body part, forming a fine-grained skeletal structure to better distinguish the intricate actions.

### B. GCNs for Skeleton-Based Action Recognition

STGCN [32] first utilized graph convolution to conduct skeleton action recognition, GCN-based methods soon became the mainstream. Different improvements have been made in recent works [15], [16], [17], [33]. AAGCN [33] proposes to adaptively learn the topology of graphs instead of setting it manually. CTRGCN [16] takes a shared topology matrix as the generic prior for network channels to improve performance. PYSKL [15] benchmarked representative GCN methods with good practices. DGSTGCN [17] proposes a lightweight yet powerful model without a predefined graph. However, traditional methods commonly face several limitations. First, they maintain a static skeleton structure with a fixed number of keypoints, which restricts their ability to capture multi-scale information. AdaSGN [41] has trained a policy network to adaptively select from multiple branch networks with different scales of skeletal structures, reducing the computational expenses but also suffering from extremely slow inference speed. In this work, we propose to dynamically downsample keypoints progressively in a single branch, achieving efficiency in both computation cost and inference speed. The second limitation is that previous GCN methods normally cropped input individuals to a maximum of two because the computation cost will linearly increase with each additional person. In this work, we introduce an Instance Pooling module to overcome the constraints of input individuals.

## III. Proposed Pipeline

The overview of our proposed pipeline is depicted in Fig. 2. In Sec. III-A, we incorporate detailed keypoints of limbs to coarse-grained body keypoints, forming the representation of Expressive Keypoints. We elaborate on the collection and preprocessing of these keypoints, highlighting the benefits of this approach. In Sec. III-B, we propose the Progressive Skeleton Evolution strategy to efficiently deal with more limb keypoints. We find that implicitly aggregating keypoint in latent space in the network processing can significantly reduce computational complexity while maintaining high accuracy. In Sec. III-C, we discover that individual modeling and late fusion of instance features in traditional methods limit their scalability in terms of input persons. Therefore, we exploit a plug-and-play Instance Pooling module for multiple instance inputs (in Sec. III-C), which supports the recognition of group activities without surging computational costs.

### A. Expressive Keypoints Representation

*1) Data Collection:* Benefiting from the dense landmarks provided by COCO-WholeBody [14], which encompasses 133 keypoints, including 17 keypoints for the body, 68 for the face, 42 for the hands, and 6 for the feet, we have a base representation for fine-grained skeleton. In practice, COCO-WholeBody can be extracted from a top-down estimator. We firstly extract human bounding boxes using the

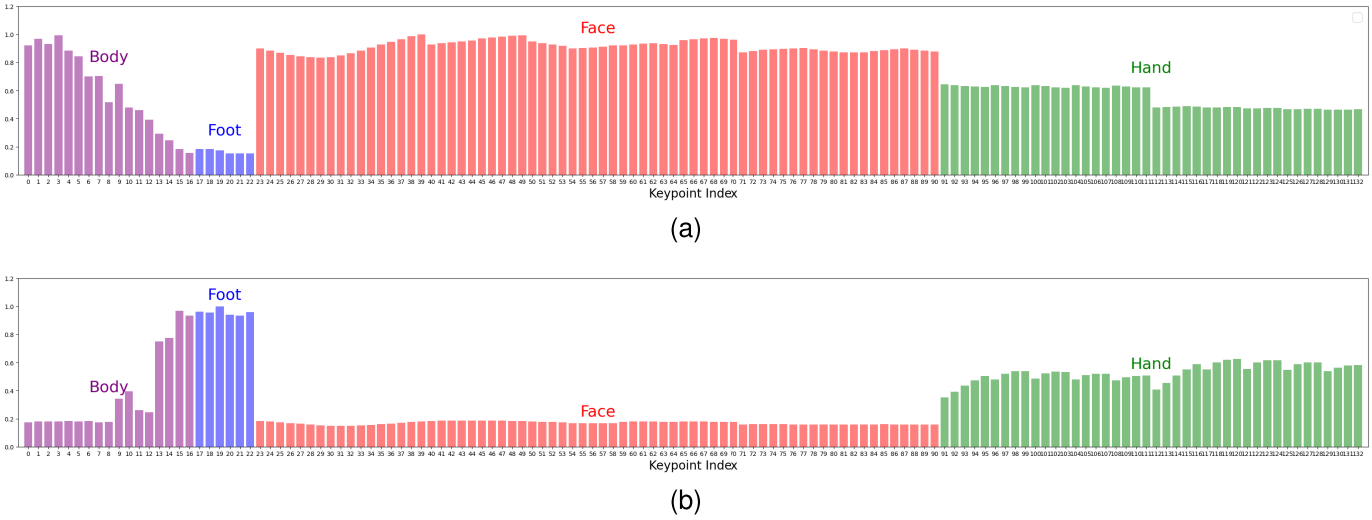



Fig. 3. Statistical results of whole-body keypoints on the NTU RGB+D dataset. (a) Video variance distribution. (b) Motion variance distribution.

ResNet50-based Faster-RCNN [42]. Subsequently, the COCO-WholeBody [14] keypoints within specified bounding boxes are obtained through the pre-trained human pose estimator [43].

*2) Keypoint Selection:* We observe directly using COCO-WholeBody as input not only incurred significant computational costs but also yielded lower performance, because there might be numerous redundant keypoints introducing substantial noise into the model. To alleviate this issue, we select the input 133 keypoints from two perspectives. First, COCO-WholeBody not only includes body and detailed hand keypoints, but also includes face landmarks, which are intuitively not related to the human action. Besides, we analyze two statistical metrics: *Video Variance* and *Motion variance* on the NTU-120 dataset, which calculate the variance of keypoints for each person and motion frequency of each keypoint between frames, respectively.

Specifically, **(i) Video Variance** $Var_i^v$, calculates the variance of keypoints for each person across all videos. A lower value of $Var_i^v$ is indicative of a keypoint distribution that is more consistent and, consequently, more amenable:

$$Var_i^v = \frac{1}{S}\sum_{s=1}^{S}(v_{i,s} - \overline{\mu}_{vi})^2, \tag{1}$$

where $S$ represents number of videos, $v_{i,s}$ is mean of $i$-th joint positions in each video $s$, and $\overline{\mu}_{vi}$ indicates mean of all $v_{i,s}$.

**(ii) Motion variance** $Var_i^m$, measures the motion frequency and the range of each keypoint between video frames, where higher $Var_i^m$ indicates more obvious movement for human action recognition.

$$Var_i^m = f_\sigma\left(\frac{1}{T-1}\sum_{t=1}^{T-1}\frac{\sqrt{(p_{i,t+1} - p_{i,t})^2}}{\epsilon_i}\right), \tag{2}$$

where $f_\sigma$ denotes the standard deviation function computed across videos, $p_{i,t+1}$ indicates $i$-th keypoint position in the $t$-th frame, and $\epsilon_i$ is area scale coefficient of different parts, which is used to normalize the motion variance.

As illustrated in Fig. 3, facial keypoints (23-90th) have higher video variance and lower motion frequency, which indicates low contribution for action recognition. This observation guides us to manually remove them.

### *B. Progressive Skeleton Evolution Strategy*

The representation of Expressive Keypoints provides abundant motion cues for skeleton action recognition. However, directly feeding Expressive Keypoints into existing GCN methods encounters several limitations. **(i) Low efficiency:** Handling with much more limb joints significantly increases computational complexity compared to the coarse-grained ones. **(ii) Sub-optimal accuracy:** The topology graph of Expressive Keypoints is more complex and has multi-hop connections which hinders the network from effectively exchange information among distant nodes. Consequently, it faces a more pronounced long-range dependency problem [34]. We claim that the key problem is that *traditional methods have a fixed skeleton structure during feed forward*.

To this end, we propose a novel Progressive Skeleton Evolution (PSE) strategy to progressively downsamples the Expressive Keypoints throughout the processing stages. The PSE strategy can be seamlessly integrated into most GCN methods to create our PSE-GCN (*e.g* baseline: DGSTGCN [17] → ours: PSE-DGSTGCN) without modifying the inner implementation of their graph convolutional and temporal convolutional layers or the high-level architectural design. What we do is to encapsulating the baseline graph convolutional layers within a proposed Grouped Mapping framework, where the input keypoint features are divided into groups and multiplied by the mapping matrices before being processed by the graph convolutional layers. By strategically exploit Expressive Keypoints, our PSE-GCN can achieve comparable or even higher accuracy with much lower GFLOPs compared with its baseline GCN method.

*1) Preliminary and Notations of GCN:* The skeleton sequence $\mathbf{X} \in \mathbb{R}^{J\times T\times C}$ is defined by $J$ joints with $C$ dimension channels at each joint in $T$ frames. For most existing

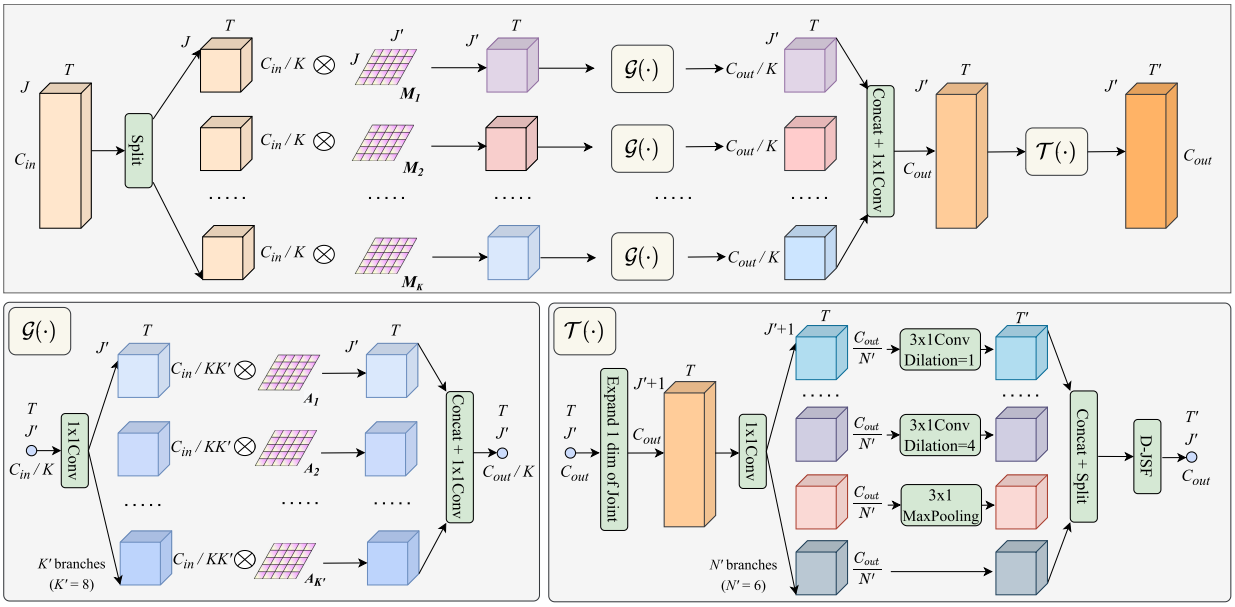



Fig. 4. The architecture of the proposed Grouped Mapping Framework $\hat{\mathcal{F}}$. To illustrate the internal structure $\mathcal{G}$ and $\mathcal{T}$ of the framework, we inherit the graph convolutional layer and the temporal convolutional layer from the baseline model DGSTGCN [17]. However, $\mathcal{G}$ and $\mathcal{T}$ are not limited to this implementation, most GCN-based methods' the graph convolutional layer and the temporal convolutional layer can be adopted as $\mathcal{G}$ and $\mathcal{T}$.

GCN-based methods, they share a same architecture design of $M$ spatial-temporal blocks, where each spatial-temporal block $\mathcal{F}$ contains a graph convolutional layer $\mathcal{G}$ and a temporal convolutional layer $\mathcal{T}$ to alternately model the spatial and temporal information. We use $\mathbb{B} = \{1, 2, .., M\}$ to denote the index set of spatial-temporal blocks, which has two subset $\mathbb{B}^n$ and $\mathbb{B}^d$, where $\mathbb{B}^d$ contains the indices of downsampling blocks $\mathcal{F}^d$ that downsample the temporal length and $\mathbb{B}^n$ contains the indices of other normal blocks $\mathcal{F}^n$. The adjacent martix $\mathbf{A} \in \mathbb{R}^{J\times J}$ defines the topology links of human skeleton, where $\mathbf{A}_{ij} = 1$ if $i$-th joint and $j$-th joint are physically connected and 0 otherwise. The computation of $\mathcal{F}$ can be summarized as:

$$\mathcal{F}(\mathbf{X}, \mathbf{A}) = \mathcal{T}(\mathcal{G}(\mathbf{X}, \widetilde{\mathbf{A}})) + \mathbf{X}, \tag{3}$$

where $\widetilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}$ is the skeletal topology graph with added self-link.

*2) Grouped Mapping Framework:* To achieve the PSE strategy for existing GCN methods, we propose the Grouped Mapping Framework to encapsulate original graph convolutional layers $\mathcal{G}$ and temporal convolutional layers $\mathcal{T}$ of any GCN methods without modifying their inner design. The same high-level architecture $\mathbb{B} = \mathbb{B}^n \cup \mathbb{B}^d$ is also inherited. We denote the Grouped Mapping Framework as $\hat{\mathcal{F}}$ and its detailed architecture is depicted in Fig. 4. Specifically, we split the channel dimension of the skeleton sequence $\mathbf{X}$ into $K$ groups, thereby reducing the channel width of each feature group to $C/K$. Subsequently, each feature group is independently multiplied by a corresponding mapping matrix $\mathbf{M}$ to adaptively alter the skeleton structure. Next, we parallelize $K$ baseline graph convolutional layers $\{\mathcal{G}_1, \ldots, \mathcal{G}_K\}$ to extract group-specified features that can greatly enrich the motion feature representations across diverse structures. Finally, $K$ group features are concatenated along the channel dimension and processed by the baseline temporal convolutional layer $\mathcal{T}$ to model the temporal dependency, generating the refined motion feature. To clearly illustrate the structure of $\mathcal{G}$ and $\mathcal{T}$ in Fig. 4, we instantiate them using the graph convolutional and temporal convolutional layers from our baseline model DGSTGCN. For more detailed elaboration of DGSTGCN's graph and temporal convolutional layers, readers are referred to the original paper [17]. However, it is important to note that $\mathcal{G}$ and $\mathcal{T}$ are flexible and can be substituted with corresponding modules from other GCN-based methods like [15], [16], and [32], as our framework is designed to be generally applicable. The whole processing of our Grouped Mapping Framework $\hat{\mathcal{F}}$ can be formulated as follows:

$$\hat{\mathcal{F}}(\mathbf{X}, \mathbf{A}, \mathbf{M}) = \mathcal{T}(\sigma(\mathcal{G}_k(\mathbf{M}_k\mathbf{X}_k, \widetilde{\mathbf{A}})\mathbf{W})) + \mathbf{M}\mathbf{X}, \tag{4}$$

where $k \in \{1, \ldots, K\}$ is the index of each group, $\mathbf{X}_k$ is the $k$-th split feature and $\mathbf{W}$ is a learnable weights. $\sigma(\cdot)$ is the activation function. We provide further elaborations of mapping matrix $\mathbf{M}$ subsequently.

**Mapping matrix.** The main idea of downsampling the keypoints is achieved by being multiplied with the mapping matrix $\mathbf{M}^d \in \mathbb{R}^{J_i \times J_{i+1}}$ to fuse correlated joints. It maps the original skeleton $\mathbf{X}$ with $J_i$ joints to a new skeleton $\mathbf{X}'$ with $J_{i+1}$ joints, which can be formulated as follows:

$$\mathbf{X}' = \mathbf{M}^d\mathbf{X}, \tag{5}$$

Once the skeleton structure is downsampled, the new adjacent matrix can be calculated as follows:

$$\mathbf{A}' = (\mathbf{M}^d)^T\mathbf{A}\mathbf{M}^d. \tag{6}$$

The downsampling operation is only conduct in the downsampling blocks with indices in $\mathbb{B}^d$. For the other normal blocks in $\mathbb{B}^n$, the mapping matrix $\mathbf{M}^n \in \mathbb{R}^{J_i \times J_i}$ is defined as a learnable diagonal matrix that does not downsample the

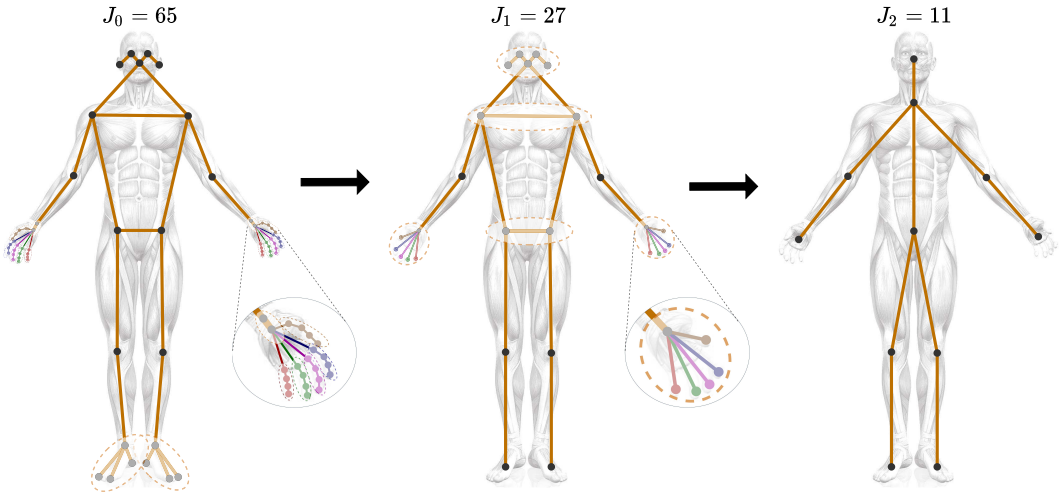



Fig. 5. Pre-defined keypoint partition. The joints are pooled from 65 to 27, then to 11. This partitioning is semantically guided, related joints like keypoints in the same finger are grouped as one part. It acts as an good semantical prior for initializing the downsampling mapping matrices to stabilize the training.

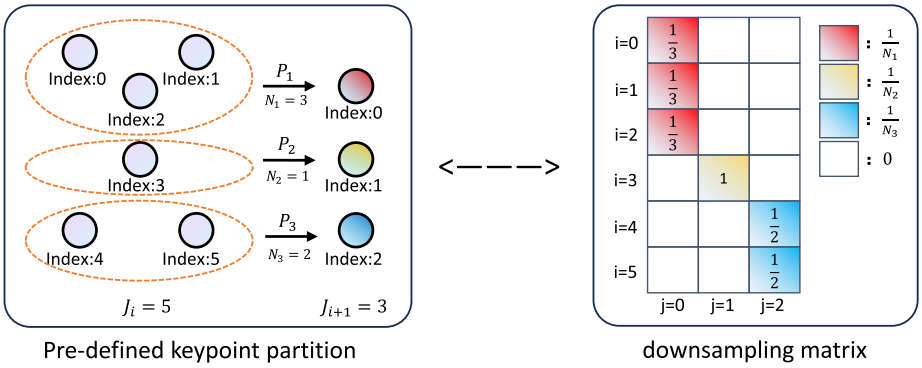



Fig. 6. The correspondence between pre-defined keypoint partition and initialized downsampling mapping matrix.

keypoints. It serves to re-weight the skeleton joints, enabling the network to prioritize important joints by allocating weights on the diagonal. Considering the index of $\hat{\mathcal{F}}$ and the type of mapping matrix, Eq.(4) can be detailed as follows:

$$\hat{\mathcal{F}}_{(i)} = \begin{cases} \mathcal{T}(\sigma(\{[\mathcal{G}_k(\mathbf{M}_k^n \mathbf{X}_k, \widetilde{\mathbf{A}})]\}\mathbf{W}) + \mathbf{X}, & i \in \mathbb{B}^n, \\ \mathcal{T}(\sigma(\{[\mathcal{G}_k(\mathbf{M}_k^d \mathbf{X}_k, \widetilde{\mathbf{A}})]\}\mathbf{W}) + \mathbf{M}^d\mathbf{X}, & i \in \mathbb{B}^d. \end{cases} \tag{7}$$

**Pre-defined keypoint partition.** The downsampling process of skeleton structures follows the pre-defined keypoint partitioning, as shown in Fig. 5. The joints are downsampled from 65 to 27, then to 11. Inspired by the idea that adjacent areas have similar semantics for human action, We initially define the keypoint partitioning based on the hierarchical skeletal structure of Expressive Keypoints. It guides the initialization of the downsampling mapping matrix $\mathbf{M}^d$. To facilitate understanding, Fig. 6 uses a simulated example to demonstrate the pre-defined keypoint partition from $V_i$ to $V_{i+1}$ and its corresponding downsampling matrix. As shown in the figure, the original $V_i = 5$ joints are partitioned into three parts $\{P_j\}_{j\in\{1,2,\ldots,V_{i+1}\}}$ and merge $N_j$ nodes within each part $P_j$ to yield a new skeletal structure with $V_{i+1} = 3$ joints. Once the partitioning is determined, the initialized downsampling matrix $\mathbf{M}^d$ can be formulated as follows:

$$\mathbf{M}_{ij}^d = \begin{cases} \dfrac{1}{N_j}, & i \in P_j, \\ 0, & \text{otherwise.} \end{cases} \tag{8}$$

The partition operations are semantically guided, ensuring that the partitioned nodes and their corresponding parts share the same semantic meaning (the semantically related nodes like big toe, little toe, and heel are grouped as one part).

*3) Overall Architecture of Our PSE-GCN:* Three representative GCN methods are adopted to be our baseline model, which are STGCN++, CTRGCN, DGSTGCN. All these models share the same high-level design. We apply the PSE strategy to form our corresponding PSE-GCN, which are PSE-STGCN++, PSE-CTRGCN, and PSE-DGSTGCN, respectively.

The integration of the PSE strategy is seamless, thus the same overall architecture is inherited. Which includes 10 spatial-temporal blocks (b1~b10), and the output channels (number of features) for each block are configured as 64, 64, 64, 64, 128, 128, 128, 256, 256, and 256, respectively. The 5th and 8th blocks (b5, b8) are downsampling blocks, while the other blocks are normal blocks. In each downsampling block, the groups expand at a factor of 2, and the number of joints

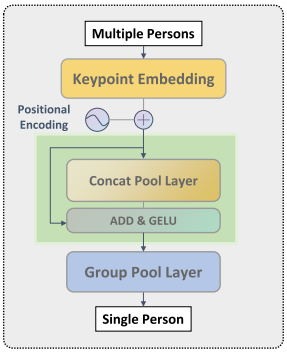



Fig. 7. Instance Pooling (IP) module.

is downsampled from 65 to 27 to 11. Through a 2D Avg-Pooling, the temporal and joint dimensions are eliminated and the output is used by the classifier to predict a score vector for video-level action recognition.

### C. Instance Pooling Module

The computation of previous GCN-based works scale linearly with the increasing number of persons in the video, making it less efficient for group activity recognition. The key problem is that traditional methods *independently model each person's skeleton sequence* and then *perform feature fusion at the late stage*.

To tackle this problem, we implement an plug-and-play Instance Pooling (IP) module which performs early feature fusion of the multiple input skeletons before feeding them to GCN. As shown in Fig. 7, we obtain the keypoint embedding via utilizing a fully connected layer and a keypoint positional encoding from the multi-person skeletal sequences. Subsequently, the Concat Pool Layer $\mathcal{P}_c(\cdot)$ and the Group Pool Layer $\mathcal{P}_g(\cdot)$ exploited by [13] are adopted to aggregate $I$ instance-wise feature vectors. Where this process can be formulated as:

$$\mathbf{Y}' = \mathcal{P}_g(\sigma(\mathcal{P}_c(\mathbf{Y}) + \mathbf{Y})), \tag{9}$$

where $\mathbf{Y} = emb(\{\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_I\}) \in \mathbb{R}^{I\times J\times T\times C}$ is multi-person skeletal features, the $emb(\cdot)$ represents a positional embedding applied to the joint dimension. These positional embeddings are initialized as random values sampled from a normal distribution and will be optimized during training. $\mathbf{Y}' \in \mathbb{R}^{J\times T\times C}$ is the aggregated single-person representation where the dimension of instance $I$ has been eliminated. Through early fusion in the lightweight IP module, the computationally burdensome spatial-temporal modeling will be conducted only once in the subsequent GCN, regardless of the number of input instances. The IP module serves as a flexible and lightweight extension for any GCN-based methods (including our PSE-GCN). It offers a practical and efficient solution for extending GCN-based skeleton action recognition to multi-person group activity scenarios without surging computation cost.

## IV. Experiments

We conduct comprehensive experiments to evaluate our proposed pipeline over seven datasets, including NTU-60 [5], NTU-120 [6], PKU-MMD [18], N-UCLA [19], Volleyball [44], Kinetics-400 [20], UCF-101 [21], and HMDB-51 [22]. We report Top-1 accuracy to evaluate model's recognition performance, and report floating point operations (FLOPs) and number of parameters (Params.) to evaluate model's efficiency in terms of computation cost and model size.

### A. Datasets

**NTU-60** and **NTU-120** can be can be collectively referred to as NTU RGB+D, which is currently the largest dataset for skeleton human action recognition. The NTU-60 dataset contains 56,880 videos of 60 human actions. The authors of this dataset recommend two split protocols: CS and cross-view (CV). The NTU-120 dataset is a superset of NTU-60 and contains a total of 113,945 samples over 120 classes. The authors of this dataset recommend two split protocols: cross-subject (CS) and cross-set (CX). We conduct experiments on NTU-60 and NTU-120 following those recommended protocols.

**PKU-MMD** dataset is originally proposed for action detection. For the action recognition task, we crop long videos to get short clips based on the temporal annotations following [45]. The PKU-MMD has two versions: PKU-MMD I (PKU-I) dataset and PKU-MMD II (PKU-II) dataset, with nearly 20,000 action instances over 51 classes and 7,000 action instances over 41 classes, respectively. We follow the recommended CS split protocol for training and testing.

**Volleyball** is a group activity recognition dataset with 4830 videos of 8 group activity classes. Each frame contains approximately 12 persons, while only the center frame has annotations for GT person boxes. We use tracking boxes from [46] for pose extraction.

**N-UCLA** contains 1494 video clips covering 10 action categories, which are performed by 10 different subjects. It has the most various significant variations in viewpoint and severe occlusions. We follow the same evaluation protocol in [16].

**Kinetics-400**, **UCF-101**, and **HMDB-51** are general action recognition datasets collect from web. With the incorporation of the Instance Pooling module, we have extended our pipeline to these in-the-wild datasets. The Kinetics-400 is a large-scale video dataset with 300,000 videos and 400 action classes. The UCF-101 dataset comprises approximately 13,000 videos sourced from YouTube, categorized into 101 action labels. The HMDB-51 consists of around 6,700 videos with 51 actions.

### B. Implementation Details

*1) Hyperparameters:* Following the good practices of PYSKL [15], we use the same hyperparameter setting for all GCN models to ensure fair comparison. Specifically, we employ the Stochastic Gradient Descent with a Nesterov momentum of 0.9 and weight decay of 0.0005. When training from scratch, the initial learning rate is set to 0.1, and we train all models for 120 epochs with the Cosine Annealing LR scheduler. On the UCF-101 and HMDB-51 datasets, we fine-tune all models based on the Kinetics-400 pretrained weights for 120 epochs with a initial learning rate of 0.01, which will decay with a factor 0.1 at epoch 90 and 110. The

TABLE I
HYPERPARAMETERS AND AUGMENTATION OF EACH DATASET DURING TRAINING

| | NTU-RGB+D | PKU-MMD | N-UCLA | Volleyball | Kinetics-400 | UCF-101 | HMDB-51 |
|---|---|---|---|---|---|---|---|
| Optimizer | Stochastic Gradient Descent | | | | | | |
| Number of epochs | 120 | | | | | | |
| Number of persons | 2 | 2 | 1 | 12 | 10 | 10 | 10 |
| Temporal length | 100 | 100 | 50 | 100 | 100 | 100 | 100 |
| Batch size | 128 | 64 | 16 | 64 | 128 | 64 | 64 |
| Pretraining dataset | None | None | None | None | None | Kinetics-400 | Kinetics-400 |
| Learning rate | 0.1 | | | | | 0.01 | |
| Scheduler | Cosine Annealing | | | | | Step [90, 110] | |
| Weight decay | 0.0005 | | | | | | |
| Momentum | Nesterov, 0.9 | | | | | | |
| Random scaling | None | | | | [0.85, 1.15] | | |
| Random cropping | None | | | | [0.56, 1.00] | | |
| Random flipping | None | | | | 0.5 | | |
| Temporal sampling | Uniform Sampling | | | | | | |

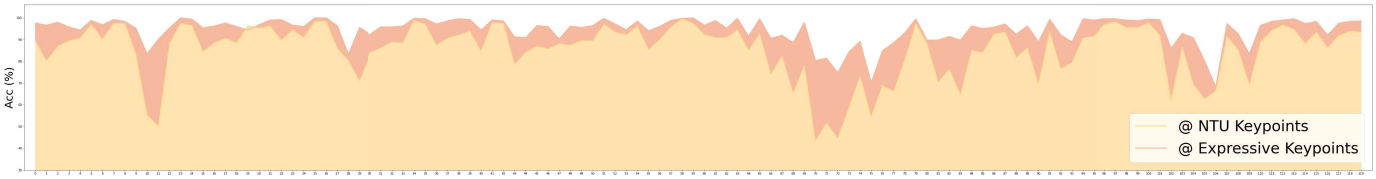



Fig. 8. The classification accuracy distribution of 120 actions in NTU RGB+D datasets. The orange volume represents the accuracy when trained on the NTU Keypoints, the red volume represents the accuracy when trained on our Expressive Keypoints.

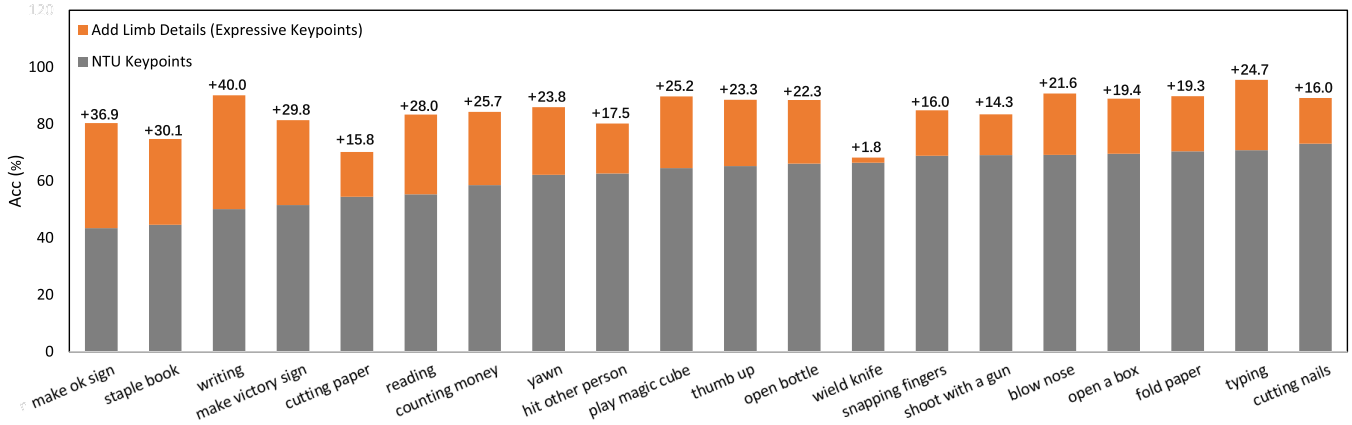



Fig. 9. Comparison of Top-20 hard cases. The gray bars denote the accuracy on the coarse-grained NTU Keypoints, the orange bars denote the improved accuracy when incorporating limb details to form our fine-grained Expressive Keypoints representation.

hyperparameters of batch size, temporal length, and number of input persons employed for each datasets are listed in Tab. I. We use zero-padding or cropping for each video to satisfy the fixed number of input persons. Our models are implemented with the PyTorch deep learning framework. All the experiments are conduct on a single Linux server with four RTX 3090 GPUs for distributed training and testing.

*2) Data Augmentation:* Uniform Sampling [29] is adopted as a strong temporal augmentation strategy, which evenly partitions the original skeleton sequence into $T$ splits and randomly extracts one frame from each split to form a clip of length $T$. On the NTU RGB+D, PKU-MMD, and N-UCLA datasets, no spatial augmentation is utilized for processing 2D Expressive Keypoints. On the Kinetics-400, UCF-101, and HMDB-51 datasets, we employ substantial spatial data augmentations, *e.g.* random scaling, cropping, and flipping the keypoints. Detailed augmentation for each datasets are listed in Tab. I.

### C. Effectiveness of Proposed Components

We conduct evaluations for testing the effectiveness of every component in our proposed pipeline, which including the Expressive Keypoints representation, the PSE strategy, and the IP module.

*1) Expressive Keypoints Representation:* On NTU-120, we directly feed Expressive Keypoints into three representative GCN methods, which are STGCN++ [15], CTRGCN [16], and DGSTGCN [17]. As shown in Tab. II, the Expressive Keypoints representation significantly enhances action recognition performance on all three baseline networks (+ 7.8%, + 8.6%, + 6.5%, respectively). To provide more granular

TABLE II
EFFECTIVENESS OF PSE STRATEGY ON EXPRESSIVE KEYPOINTS

| Method | Format | CS(%) | CX(%) | FLOPs |
|---|---|---|---|---|
| STGCN++ | NTU Keypoints | 84.3 | 86.7 | 2.7G |
| STGCN++ | Expressive Keypoints | 92.6 +8.3 | **94.5** +7.8 | 6.9G |
| PSE-STGCN++ | Expressive Keypoints | **92.7** | **94.5** | **2.6G** -4.3 |
| CTRGCN | NTU Keypoints | 84.0 | 85.9 | 2.7G |
| CTRGCN | Expressive Keypoints | **92.8** +8.8 | 94.5 +8.6 | 7.5G |
| PSE-CTRGCN | Expressive Keypoints | **92.8** | **94.7** | **2.5G** -5.0 |
| DGSTGCN | NTU Keypoints | 85.7 | 87.9 | **2.4G** |
| DGSTGCN | Expressive Keypoints | 92.6 +6.9 | 94.4 +6.5 | 6.3G |
| PSE-DGSTGCN | Expressive Keypoints | **93.1** | **94.8** | **2.4G** -3.9 |

TABLE III
EFFECTIVENESS OF PSE STRATEGY ON NTU KEYPOINTS

| Format | Method | CS(%) | CX(%) | FLOPs |
|---|---|---|---|---|
| NTU Keypoints | STGCN++ | 84.3 | **86.7** | 2.7G |
| | PSE-STGCN++ | **84.9** | **86.7** | 1.5G -1.2 |
| | CTRGCN | 84.0 | 85.9 | 2.7G |
| | PSE-CTRGCN | **84.1** | **86.4** | 1.5G -1.2 |
| | DGSTGCN | **85.7** | **87.9** | 2.4G |
| | PSE-DGSTGCN | **85.7** | 87.8 | 1.5G -0.9 |

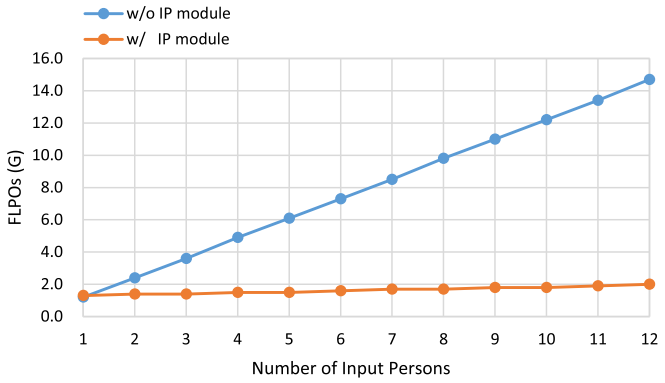



Fig. 10. Ablation study on the IP module with respect to the input person numbers. The FLOPs increases linearly with person number increasing without the IP module. While the FLOPs hardly increases with the IP module.

TABLE IV
EFFECTIVENESS OF IP MODULE

| Dataset | Method | Input persons | Accuracy(%) | FLOPs |
|---|---|---|---|---|
| HMDB-51 | w/o IP module | 2 | 44.1 | 2.4G |
| | w/o IP module | 5 | 49.6 | 6.1G |
| | w/o IP module | 10 | OOM | 12.2G |
| | w/ IP module | 10 | **51.6** | 1.8G |
| Volleyball | w/o IP module | 2 | 37.2 | 2.4G |
| | w/o IP module | 5 | 63.7 | 6.1G |
| | w/o IP module | 12 | OOM | 14.7G |
| | w/ IP module | 12 | **88.3** | 2.0G |

TABLE V
ABLATION STUDY ON INPUT KEYPOINT SELECTION

| Protocol | Config of $V$ | $N$ | Accuracy(%) | FLOPs |
|---|---|---|---|---|
| #1 | COCO-WholeBody | 133 | 93.4 | 12.8G |
| #2 | #1+w/o face | 65 | **94.4** | **6.3G** |
| #3 | #2+w/o feet | 59 | 94.1 | 5.8G |
| #4 | #2+simple fingers | 35 | 90.6 | 3.4G |
| #5 | #2+w/o hands | 23 | 88.0 | 2.4G |
| #6 | #2+**Ours: PSE** | 65 | **94.8** | **2.4G** |

TABLE VI
ABLATION STUDY ON THE GROUP CONFIGURATION

| $K_0$ | $c$ | Config of $K$ | Accuracy(%) |
|---|---|---|---|
| 1 | 1 | [1, 1, 1] | 93.1 |
| 2 | 1 | [2, 2, 2] | 93.8 |
| 4 | 1 | [4, 4, 4] | 93.5 |
| 1 | 2 | [1, 2, 4] | **94.8** |
| 2 | 2 | [2, 4, 8] | 94.1 |
| 1 | 4 | [1, 4,16] | 93.9 |

insights into the performance improvements, Fig. 8 illustrates the classification accuracy distribution across 120 action labels using both NTU Keypoints representation and Expressive Keypoints representation on the NTU RGB+D dataset, demonstrating that our Expressive Keypoints representation consistently outperforms the coarse-grained representation. Furthermore, from these 120 categories, we extract the Top-20 hard cases (ranked by accuracy from low to high), which primarily involve fine-grained actions requiring subtle limb movement discrimination—such as *Reading* vs. *Writing* and *make ok sign* vs. *make victory sign*. When incorporating detailed limb keypoints by Expressive Keypoints, the action recognition performance substantially improves, particularly for those challenging actions with nuanced limb movements, as demonstrated in Fig. 9.

*2) PSE Strategy:* Building upon the strong baseline performance achieved with Expressive Keypoints, we further integrate the proposed PSE strategy into the previous baseline GCN methods to form our PSE-GCN models, which are PSE-STGCN++, PSE-CTRGCN, and PSE-DGSTGCN. By progressively downsampling the Expressive Keypoints, three baseline models applying PSE strategy significantly reduce more than half of the computation cost (−4.3G, −5.0G, −3.9G) while achieving comparable or even higher accuracy, as shown in Tab. II. To provide a more complete evaluation, we also evaluate the effectiveness of PSE strategy with NTU Keypoints input. This additional evaluation serves to verify whether the benefits of PSE are specific to Expressive Keypoints or can be generalized to traditional coarse-grained skeletal representations. As shown in Tab. III, PSE strategy can greatly reduce the computation cost (from 2.4G~2.7G to 1.5G) of processing coarse-grained skeletal data while preserving accuracy. It can be observed that a slight accuracy drop occurs in one of the six settings. We consider this is because the coarse-grained skeletal representation is already very concise, and further downsampling might result in under-represented features. Nevertheless, the overall results confirm that PSE strategy maintains competitive performance while offering substantial

TABLE VII

WE BENCHMARK GCN SKELETON-BASED ACTION RECOGNITION METHODS ON THE NTU-60 DATASET. THE WEIGHTS ASSIGNED TO COMPONENTS OF 2S-FUSION AND 4S-FUSION ARE [1:1] AND [3:3:2:2], RESPECTIVELY

| Method | NTU-60 CS | | | | NTU-60 CV | | | | Efficiency | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Joint(%) | Bone(%) | 2s(%) | 4s(%) | Joint(%) | Bone(%) | 2s(%) | 4s(%) | FLOPs | Params |
| STGCN++ | 95.6 | 95.8 | 96.5 | 96.8 | **99.1** | 98.9 | **99.4** | **99.5** | 6.9G | 1.4M |
| PSE-STGCN++ | **95.7** | **95.9** | **96.6** | **97.0** | **99.1** | **99.0** | **99.4** | **99.5** | **2.6G** | **1.2M** |
| CTRGCN | 95.8 | **96.2** | 96.7 | 96.9 | 99.0 | **99.0** | 99.4 | **99.5** | 7.5G | 1.4M |
| PSE-CTRGCN | **96.0** | **96.2** | **97.0** | **97.1** | **99.2** | **99.0** | **99.5** | **99.5** | **2.5G** | **1.1M** |
| DGSTGCN | 95.1 | 95.8 | 96.6 | 96.9 | **99.3** | **99.1** | **99.5** | **99.6** | 6.3G | 1.6M |
| PSE-DGSTGCN | **95.8** | **96.0** | **96.7** | **97.0** | **99.3** | **99.1** | **99.5** | **99.6** | **2.4G** | **1.3M** |

TABLE VIII

WE BENCHMARK GCN SKELETON-BASED ACTION RECOGNITION METHODS ON THE NTU-120 DATASET. THE WEIGHTS ASSIGNED TO COMPONENTS OF 2S-FUSION AND 4S-FUSION ARE [1:1] AND [3:3:2:2], RESPECTIVELY

| Method | NTU-120 CS | | | | NTU-120 CX | | | | Efficiency | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Joint(%) | Bone(%) | 2s(%) | 4s(%) | Joint(%) | Bone(%) | 2s(%) | 4s(%) | FLOPs | Params |
| STGCN++ | 92.6 | **92.6** | 94.0 | 94.3 | **94.5** | 94.6 | 95.8 | 96.1 | 6.9G | 1.4M |
| PSE-STGCN++ | **92.7** | **92.6** | **94.1** | **94.5** | **94.5** | **94.9** | **95.9** | **96.3** | **2.6G** | **1.2M** |
| CTRGCN | **92.8** | 92.7 | 94.0 | 94.3 | 94.5 | 94.8 | **95.9** | **96.3** | 7.5G | 1.4M |
| PSE-CTRGCN | **92.8** | **92.9** | **94.1** | **94.5** | **94.7** | **94.9** | **95.9** | **96.3** | **2.5G** | **1.1M** |
| DGSTGCN | 92.6 | **92.8** | 94.1 | 94.3 | 94.4 | **95.1** | **96.0** | 96.1 | 6.3G | 1.6M |
| PSE-DGSTGCN | **93.1** | **92.8** | **94.3** | **94.6** | **94.8** | **95.1** | **96.0** | **96.4** | **2.4G** | **1.3M** |

computational savings across different input representations and model architectures.

*3) IP Module:* To validate the effectiveness of the IP module, we conducted comprehensive ablation studies on both HMDB-51 and Volleyball datasets using PSE-DGSTGCN to evaluate computation costs and accuracy with/without the IP module. The results are presented in Tab. IV. HMDB-51 dataset primarily contains single-person actions, but some categories could involve multiple human interactions in crowded scenes. Without the IP module, we tested with 2, 5, and 10 input persons. The results show that recognition accuracy gradually improved with more input persons (from 44.1% to 49.6%), but at the cost of exponentially increasing FLOPs (from 2.4G to 12.2G), eventually causing Out-Of-Memory (OOM) errors at 10 persons. After implementing the IP module, it achieves the highest accuracy of 51.6% while dramatically reducing computational costs to just 1.8G FLOPs for 10-person inputs. To further verify the module's robustness, we evaluate on the specialized group activity dataset Volleyball, which contains 12-person volleyball matches. Without IP, using only 2 or 5 persons yields poor accuracy (37.2% and 63.7% respectively) as critical interacting players are often cropped out. While full 12-person inputs could capture all interactions, they cause OOM errors due to excessive computation (14.7G FLOPs). Incorporating IP module successfully resolved this limitation, enabling 12-person processing at merely 2.0G FLOPs while achieving the best accuracy of 88.3%. Moreover, Fig. 10 illustrates the variation in FLOPs with the number of input presons. Without the IP module, the computation cost escalates rapidly as the number of individuals increases due to the substantial feature modeling required for each individual in the traditional GCN pipeline. However, with the inclusion of the IP module, the increase in FLOPs is minimal since the features of multiple individuals are aggregated into a single representation by the lightweight IP module before fed into the subsequent GCN model.

## D. Configuration Exploration

*1) Input Keypoints Selection:* We extensively explore the selection of initial input keypoints. As shown in Tab. V, experimental results demonstrate that removing facial keypoints from the COCO-WholeBody Keypoints (protocol #1) to form our Expressive Keypoints (protocol #2) is reasonable and aligns with the statistical analysis. Removing redundant points reduces the impact of introduced noise, resulting in higher accuracy with lower computation cost. Based on the Expressive Keypoints, we try to further prune some keypoints. It is noticeable that removing the keypoints of limbs in a explicit way can achieve a decrease in FLOPs, but also incurs an equivalent drop in accuracy (protocol #3~#5, *simple fingers* mean only one keypoint is retained for each finger). We argue that it is not applicable for explicitly selecting detailed limb keypoints in various actions of large-scale datasets. That is why we adopt a learning-based method, i.e. the PSE strategy, for the implicit selection from Expressive Keypoints (protocol #6), achieving great saving in FLOPs meanwhile maintaining high accuracy.

*2) Group Design:* Tab. VI presents six different configurations based on the initial number of groups $K_0$ and the group expansion factor $c$. It is noticeable that the static group

TABLE IX
ACCURACY AND EFFICIENCY COMPARISON WITH OTHER STATE-OF-THE-ART METHODS ON NTU-60 AND NTU-120 DATASETS

| Method | Modality | NTU-60 | | NTU-120 | | Efficiency | |
|---|---|---|---|---|---|---|---|
| | | CS(%) | CV(%) | CS(%) | CX(%) | FLOPs | Params. |
| I3D [24] | RGB | — | — | 77.0 | 80.1 | 107.9G | 12.1M |
| GlimpseClouds [47] | RGB | 86.6 | 93.2 | — | — | 168.0G | 46.8M |
| VPN++ [48] | RGB | 91.9 | 94.9 | 86.7 | 89.3 | 112.1G | 14.0M |
| RGBPose-Conv3D (RGB) [29] | RGB | 95.1 | — | — | — | 41.8G | 31.6M |
| P4Transformer [8] | Point cloud | 90.2 | 96.4 | 86.4 | 93.5 | 40.4G | 44.1M |
| PSTNet [9] | Point cloud | 90.5 | 96.5 | 87.0 | 93.8 | 54.1G | 8.4M |
| PST-Transformer [10] | Point cloud | 91.0 | 96.4 | 87.5 | 94.0 | 48.8G | 44.2M |
| PoseConv3D [29] | Skeleton | 94.1 | 96.9 | 86.9 | 90.3 | 20.9G | 4.0M |
| DSTA-Net [37] | Skeleton | 91.5 | 96.4 | 86.6 | 89.0 | 64.7G | 13.8M |
| Hyperformer [38] | Skeleton | 92.9 | 96.5 | 89.9 | 91.3 | 38.6G | 10.8M |
| Skateformer [39] | Skeleton | 93.5 | 97.8 | 89.8 | 91.4 | 14.5G | 8.1M |
| STGCN [32] | Skeleton | 90.7 | 96.5 | 86.2 | 88.4 | 21.4G | 12.3M |
| AAGCN [33] | Skeleton | 91.5 | 96.7 | 86.9 | 88.8 | 24.3G | 15.1M |
| ShiftGCN [49] | Skeleton | 90.7 | 96.5 | 85.9 | 87.6 | 10.0G | 2.8M |
| MSG3D [50] | Skeleton | 91.7 | 96.9 | 87.9 | 89.6 | 41.1G | 12.7M |
| STGCN++ [15] | Skeleton | 92.1 | 97.0 | 87.5 | 89.8 | 10.6G | 5.5M |
| CTRGCN [16] | Skeleton | 92.1 | 97.0 | 88.1 | 89.9 | 10.8G | 5.6M |
| DGSTGCN [17] | Skeleton | 93.2 | 97.5 | 89.6 | 91.4 | 9.6G | 6.6M |
| SelfGCN [51] | Skeleton | 93.1 | 96.6 | 89.4 | 91.0 | 10.0G | 10.7M |
| InfoGCN [52] | Skeleton | 93.0 | 97.1 | 89.8 | 91.2 | 10.0G | 9.4M |
| HDGCN [34] | Skeleton | 93.4 | 97.2 | 90.1 | 91.6 | 9.6G | 10.1M |
| DeGCN [53] | Skeleton | 93.6 | 97.4 | 91.7 | 92.1 | 9.7G | 5.6M |
| **Ours: Baseline*** | Skeleton(+limb details) | 96.9 | **99.6** | 94.3 | 96.1 | 25.0G | 6.6M |
| **Ours: PSE*** | Skeleton(+limb details) | **97.0** | **99.6** | **94.6** | **96.4** | 9.6G | 5.2M |

designs ($c = 1$) yield sub-optimal performances across the tested settings. In contrast, the expanding group designs, where the number of groups increases layer by layer, demonstrate a clear performance advantage. Among these, the configuration with a group sequence of [1, 2, 4] achieves the best accuracy, indicating that a moderate and progressive increase in group granularity is beneficial. We hypothesize that this is because excessive grouping leads to a reduction in the number of channels per group, which in turn weakens the feature representation capability of each group. Thus, a balanced group expansion strategy is essential to ensure both diversity and sufficient representation capacity.

### *E. Benchmarking GCN Methods on Expressive Keypoints*

With the fine-grained human body representations provided by Expressive Keypoints, most GCN methods can significantly enhance accuracy by simply adjusting the input keypoints. Our proposed Progressive Skeleton Evolution (PSE) strategy can be applied to these methods, forming our PSE-GCN models, which achieves comparable or even higher accuracy with substantially lower computation cost. We conduct a comprehensive benchmark on the NTU-60 and NTU-120 datasets for three representative GCN methods: STGCN++ [15], CTRGCN [16], and DGSTGCN [17] with Expressive Keypoints as input, as well as their PSE-GCN counterparts: PSE-STGCN++, PSE-CTRGCN, and PSE-DGSTGCN. We measure the Top-1 accuracy of joint-stream (Joint), bone-stream (Bone), two-stream fusion (2s) [60], and four-stream fusion (4s) [33]. As shown in Tab. VII and Tab. VIII, our PSE methods obtain better performance and efficiency than baselines in terms of Top-1 accuracy, FLOPs, and number of parameters.

### *F. Comparison With the State-of-the-Art*

According to Sec. IV-E, among three representative graph convolutional networks [15], [16], [17], we choose the best performed DGSTGCN [17] with Expressive Keypoints input as the strongest baseline method (denoted as **Ours: Baseline**), and apply PSE strategy to form our PSE-DGSTGCN (denoted as **Ours: PSE**), to make comparison with the state-of-the-art (SOTA). In experiments, we report the four-stream fusion results similar to the previous works [15], [16], [17], [33]. The marker * indicates using Expressive Keypoints.

On NTU-60 and NTU-120, as shown in Tab. IX, Expressive Keypoints greatly improves the accuracy for skeleton-based action recognition, even surpassing the SOTA point cloud-based [10] and RGB-based methods [29]. Upon applying the PSE strategy, our method achieves significant savings in the computation cost (25.0G → 9.6G), with comparable or even higher accuracy.

On N-UCLA, as showed in Tab. X, our method achieves 97.6% Top-1 accuracy, which also surpasses the previous

TABLE X
PERFORMANCE COMPARISON ON THE N-UCLA DATASETS

| Method | N-UCLA(%) |
|---|---|
| ShiftGCN [49] | 94.6 |
| CTRGCN [16] | 96.5 |
| InfoGCN [52] | 97.0 |
| HDGCN [34] | 97.2 |
| DeGCN [53] | 97.2 |
| **Ours: PSE*** | **97.6** |

TABLE XI
PERFORMANCE COMPARISON ON THE PKU-I AND PKU-II DATASETS

| Method | PKU-I(%) | PKU-II(%) |
|---|---|---|
| ISC [45] | 80.9 | 36.0 |
| CPM [54] | 88.8 | 48.3 |
| Eq-Contrast [55] | 91.7 | —— |
| MAMP [56] | 92.2 | 53.8 |
| SRNet [57] | 93.1 | —— |
| **Ours: PSE*** | **98.4** | **83.8** |

TABLE XII
PERFORMANCE COMPARISON ON THE KINETICS-400 DATASET

| Method | Kinetics-400(%) |
|---|---|
| STGCN [32] | 30.7 |
| MSG3D [50] | 38.0 |
| HDGCN [34] | 40.9 |
| PoseConv3D [29] | 47.7 |
| SKP [13] (w/o objects) | 50.3 |
| SKP [13] (w/ objects) | 52.3 |
| **Ours: PSE +IP*** | **53.1** |

best method [34]. Notably, among the standard skeleton-based datasets, N-UCLA has the most significant variations in viewpoint and severe occlusions. Despite being limited by the estimated 2D representation that is unable to leverage the depth information and 3D spatial augmentations (*e.g.* 3D random rotation), our approach still obtains a promising performance.

On PKU-MMD, Tab. XI shows that our method surpasses the existing skeleton-based methods by a noticeable margin, achieving the state-of-the-art performance on both the datasets of PKU-I and PKU-II, with the Top-1 accuracy achieves to 98.4% and 83.8%, respectively.

We further extending PSE-DGSTGCN with the IP module (denoted as **Ours: PSE +IP**), which allows for evaluating our method on the more general in-the-wild action recognition datasets [20], [21], [22]. For Kinetic-400 that encompass many human-object interaction scenarios, such as *peeling apples* and *peeling potatoes*, the accuracy of pure skeleton-based methods on the Kinetic-400 is far below than other datasets since they lack of capturing object information. As a result, SKP [13] resorts to incorporating object contours

TABLE XIII
PERFORMANCE COMPARISON ON THE UCF-101 AND HMDB-51 DATASETS

| Method | Kinetics-400 Pretraining | UCF-101 (%) | HMDB-51 (%) |
|---|---|---|---|
| Potion [58] | × | 65.2 | 43.7 |
| PA3D [59] | × | —— | 55.3 |
| PoseConv3D [29] | × | 79.1 | 58.6 |
| **Ours: PSE +IP*** | × | **82.5** | **60.1** |
| PoseConv3D [29] | ✓ | 87.0 | 69.3 |
| SKP [13] | ✓ | 87.8 | 70.9 |
| **Ours: PSE +IP*** | ✓ | **88.7** | **74.6** |

and improves the accuracy of keypoint-based benchmark to 52.3%. However, as showed in Tab. XII, by strategically utilizing Expressive Keypoints, our method achieves the SOTA performance (53.1%) on the Kinetics-400 dataset even without the object information. This is made possible through our expressive skeletal representation and effective PSE strategy, demonstrating the effectiveness of our pipeline even under these challenging conditions.

Moreover, we provide an apple-to-apple comparison on UCF-101 and HMDB-51. As demonstrated in Tab. XIII, our method consistently surpasses the previous skeleton-based SOTA methods [13], [29] regardless of whether pre-training is conducted on the Kinetics-400 dataset or not.

**Further Comparison with the Multi-Modality Methods.** Across three standard skeleton action recognition datasets, including NTU RGB+D [5], [6], PKU-MMD [18], and N-UCLA [19], our method not only surpasses all the skeleton-based action recognition methods but also achieves the best performance among all the single-modality methods (RGB-based, point cloud-based). To further demonstrate the superiority of strategically employing Expressive Keypoints, we compare our method with the SOTA multi-modality methods. It can be observed that on the NTU-60 and NTU-120 datasets (Tab. XIV), we achieve comparable performance to the SOTA multi-modality method RGBPose-Conv3D [29] in three out of four evaluation protocols. On the PKU-MMD dataset (Tab. XV) and the N-UCLA dataset (Tab. XVI), our method outperforms the SOTA multi-modality method [63].

The experimental results demonstrate that our method, despite being based on a single-modality skeleton input, achieves comparable or even higher performance with a lightweight computation cost than multi-modality methods. This remarkable result primarily stems from introducing fine-grained limb details to the skeleton and employing a PSE strategy for effective feature modeling, providing a promising solution for the community.

## V. LIMITATION

In brief, our method mainly has two limitations. **(i)** Although we extend our method to in-the-wild scenarios by using the Instance Pooling module, it still struggles to distinguish certain scene-based human actions or human-object interactions due to the lack of capturing objects and scenes. Overcoming this limitation likely necessitates moving

TABLE XIV
PERFORMANCE COMPARISON WITH THE SOTA MULTI-MODALITY METHODS ON THE NTU-60 AND NTU-120 DATASETS. S, R, AND D DENOTE SKELETON, RGB, AND DEPTH, RESPECTIVELY

| Method | Modalities | NTU-60 | | NTU-120 | |
|---|---|---|---|---|---|
| | | CS(%) | CV(%) | CS(%) | CX(%) |
| STAR-Transformer [61] | S + R | 92.0 | 96.5 | 90.3 | 92.7 |
| VPN++ [48] (w/ 3D Poses) | S + R | 94.9 | 98.1 | 90.7 | 92.5 |
| HCMFN [62] | S + R + D | 95.2 | 98.0 | 89.9 | 92.7 |
| MMNet [63] | S + R | 96.0 | 98.8 | 92.9 | 94.4 |
| RGBPose-Conv3D [29] | S + R | **97.0** | **99.6** | **95.3** | **96.4** |
| **Ours: PSE*** | S | **97.0** | **99.6** | 94.6 | **96.4** |

TABLE XV
PERFORMANCE COMPARISON WITH THE SOTA MULTI-MODALITY METHODS ON PKU-MMD. S AND R DENOTE SKELETON AND RGB

| Method | Modalities | PKU-MMD(%) |
|---|---|---|
| TSMF [64] | S + R | 95.8 |
| MMNet [63] | S + R | 97.4 |
| **Ours: PSE*** | S | **98.4** |

TABLE XVI
PERFORMANCE COMPARISON WITH THE SOTA MULTI-MODALITY METHODS ON N-UCLA. S AND R DENOTE SKELETON AND RGB

| Method | Modalities | N-UCLA(%) |
|---|---|---|
| VPN++ [48] | S + R | 93.5 |
| MMNet [63] | S + R | 93.7 |
| **Ours: PSE*** | S | **97.6** |

towards multi-modal approaches. Promising directions include fusing skeleton data with RGB or object features, potentially leveraging techniques for employing cross-modality correlation [65] or distillation [66], [67] to transfer knowledge from richer modalities to the skeleton stream. **(ii)** Acquiring high-quality annotations, especially for fine-grained keypoints under occlusion or for complex interactions, remains challenging. Exploring semi-supervised [68] or weakly-supervised [69] paradigms, could help mitigate the annotation cost barrier.

## VI. CONCLUSION

In this work, we propose the Progressive Skeleton Evolution strategy using the Expressive Keypoints representation to obtain high performance in discriminating detailed actions while maintaining the high efficiency. Furthermore, we explore an Instance Pooling module, expanding the applicability of GCN-based methods to multi-person scenarios. Comprehensive experiments over seven datasets demonstrate our pipeline has superior performance and robust generalization.

## ACKNOWLEDGMENT

The numerical calculation is supported by a super-computing system at the Super-computing Center, Wuhan University.

## REFERENCES

[1] W. Lin, M.-T. Sun, R. Poovandran, and Z. Zhang, "Human activity recognition for video surveillance," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2008, pp. 2737–2740.
[2] W. Sultani, C. Chen, and M. Shah, "Real-world anomaly detection in surveillance videos," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6479–6488.
[3] I. Rodomagoulakis et al., "Multimodal human action recognition in assistive human–robot interaction," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2016, pp. 2702–2706.
[4] M. Qi, J. Qin, A. Li, Y. Wang, J. Luo, and L. Van Gool, "StagNet: An attentive semantic RNN for group activity recognition," in *Proc. Eur. Conf. Comput. Vis.*, Sep. 2018, pp. 101–117.
[5] 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.
[6] 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.
[7] T. Lin et al., "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 740–755.
[8] H. Fan, Y. Yang, and M. Kankanhalli, "Point 4D transformer networks for spatio-temporal modeling in point cloud videos," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 14204–14213.
[9] H. Fan, X. Yu, Y. Ding, Y. Yang, and M. Kankanhalli, "PSTNet: Point spatio-temporal convolution on point cloud sequences," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2022, pp. 1–8.
[10] H. Fan, Y. Yang, and M. Kankanhalli, "Point spatio-temporal transformer networks for point cloud video modeling," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 2, pp. 2181–2192, Feb. 2023.
[11] S. Kim, K. Yun, J. Park, and J. Y. Choi, "Skeleton-based action recognition of people handling objects," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2019, pp. 61–70.
[12] L. Xu, C. Lan, W. Zeng, and C. Lu, "Skeleton-based mutually assisted interacted object localization and human action recognition," *IEEE Trans. Multimedia*, vol. 25, pp. 4415–4425, 2023.
[13] R. Hachiuma, F. Sato, and T. Sekii, "Unified keypoint-based action recognition framework via structured keypoint pooling," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 22962–22971.
[14] S. Jin et al., "Whole-body human pose estimation in the wild," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 196–214.
[15] H. Duan, J. Wang, K. Chen, and D. Lin, "PYSKL: Towards good practices for skeleton action recognition," in *Proc. ACM Int. Conf. Multimedia (ACMMM)*, 2022, pp. 7351–7354.
[16] 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.*, Oct. 2021, pp. 13359–13368.
[17] H. Duan, J. Wang, K. Chen, and D. Lin, "DG-STGCN: Dynamic spatial–temporal modeling for skeleton-based action recognition," 2022, *arXiv:2210.05895*.
[18] C. Liu, Y. Hu, Y. Li, S. Song, and J. Liu, "PKU-MMD: A large scale benchmark for skeleton-based human action understanding," in *Proc. Workshop Vis. Anal. Smart Connected Communities*, Oct. 2017, pp. 1–8.

[19] 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.

[20] W. Kay et al., "The kinetics human action video dataset," 2017, *arXiv:1705.06950*.

[21] K. Soomro, A. R. Zamir, and M. Shah, "UCF101: A dataset of 101 human actions classes from videos in the wild," 2012, *arXiv:1212.0402*.

[22] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre, "HMDB: A large video database for human motion recognition," in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 2556–2563.

[23] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Proc. Adv. Neural Inf. Process. Syst.*, 2022, pp. 1–7.

[24] J. Carreira and A. Zisserman, "Quo vadis, action recognition? A new model and the kinetics dataset," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6299–6308.

[25] C. Feichtenhofer, H. Fan, J. Malik, and K. He, "SlowFast networks for video recognition," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6202–6211.

[26] Z. Tu et al., "Multi-stream CNN: Learning representations based on human-related regions for action recognition," *Pattern Recognit.*, vol. 79, pp. 32–43, Jul. 2018.

[27] Z. Tu, H. Li, D. Zhang, J. Dauwels, B. Li, and J. Yuan, "Action-stage emphasized spatiotemporal VLAD for video action recognition," *IEEE Trans. Image Process.*, vol. 28, no. 6, pp. 2799–2812, Jun. 2019.

[28] Y. Liu, J. Yuan, and Z. Tu, "Motion-driven visual tempo learning for video-based action recognition," *IEEE Trans. Image Process.*, vol. 31, pp. 4104–4116, 2022.

[29] H. Duan, Y. Zhao, K. Chen, D. Lin, and B. Dai, "Revisiting skeleton-based action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2022, pp. 2969–2978.

[30] C. Caetano, J. Sena, F. Brémond, J. A. Dos Santos, and W. R. Schwartz, "SkeleMotion: A new representation of skeleton joint sequences based on motion information for 3D action recognition," in *Proc. 16th IEEE Int. Conf. Adv. Video Signal Based Surveill. (AVSS)*, Sep. 2019, pp. 1–8.

[31] C. Caetano, F. Brémond, and W. R. Schwartz, "Skeleton image representation for 3D action recognition based on tree structure and reference joints," in *Proc. 32nd SIBGRAPI Conf. Graph., Patterns Images*, Oct. 2019, pp. 16–23.

[32] S. Yan, Y. Xiong, and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition," in *Proc. AAAI Conf. Artif. Intell.*, 2018, vol. 32, no. 1, pp. 1–9.

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

[34] 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. 10410–10419.

[35] J. Zhang et al., "A spatial attentive and temporal dilated (SATD) GCN for skeleton-based action recognition," *CAAI Trans. Intell. Technol.*, vol. 7, no. 1, pp. 46–55, Mar. 2022.

[36] B. Ren, M. Liu, R. Ding, and H. Liu, "A survey on 3D skeleton-based action recognition using learning method," *Cyborg Bionic Syst.*, vol. 5, p. 0100, Jan. 2024.

[37] L. Shi, Y. Zhang, J. Cheng, and H. Lu, "Decoupled spatial–temporal attention network for skeleton-based action-gesture recognition," in *Proc. Asian Conf. Comput. Vis.*, 2021, pp. 38–53.

[38] Y. Zhou et al., "Hypergraph transformer for skeleton-based action recognition," 2022, *arXiv:2211.09590*.

[39] J. Do and M. Kim, "SkateFormer: Skeletal-temporal transformer for human action recognition," in *Proc. Eur. Conf. Comput. Vis.*, 2024, pp. 401–420.

[40] J. Zhang, Y. Jia, W. Xie, and Z. Tu, "Zoom transformer for skeleton-based group activity recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 12, pp. 8646–8659, Dec. 2022.

[41] L. Shi, Y. Zhang, J. Cheng, and H. Lu, "AdaSGN: Adapting joint number and model size for efficient skeleton-based action recognition," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 13393–13402.

[42] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.

[43] 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. (CVPR)*, Jun. 2019, pp. 5686–5696.

[44] M. S. Ibrahim, S. Muralidharan, Z. Deng, A. Vahdat, and G. Mori, "A hierarchical deep temporal model for group activity recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1971–1980.

[45] F. M. Thoker, H. Doughty, and C. G. M. Snoek, "Skeleton-contrastive 3D action representation learning," in *Proc. 29th ACM Int. Conf. Multimedia*, Oct. 2021, pp. 1655–1663.

[46] K. Sendo and N. Ukita, "Heatmapping of people involved in group activities," in *Proc. 16th Int. Conf. Mach. Vis. Appl. (MVA)*, May 2019, pp. 1–6.

[47] F. Baradel, C. Wolf, J. Mille, and G. W. Taylor, "Glimpse clouds: Human activity recognition from unstructured feature points," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 469–478.

[48] S. Das, R. Dai, D. Yang, and F. Bremond, "VPN+: Rethinking video-pose embeddings for understanding activities of daily living," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 12, pp. 9703–9717, Dec. 2022.

[49] 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.*, Jun. 2020, pp. 183–192.

[50] 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.

[51] Z. Wu et al., "SelfGCN: Graph convolution network with self-attention for skeleton-based action recognition," *IEEE Trans. Image Process.*, vol. 33, pp. 4391–4403, 2024.

[52] 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.

[53] W. Myung, N. Su, J.-H. Xue, and G. Wang, "DeGCN: Deformable graph convolutional networks for skeleton-based action recognition," *IEEE Trans. Image Process.*, vol. 33, pp. 2477–2490, 2024.

[54] H. Zhang, Y. Hou, W. Zhang, and W. Li, "Contrastive positive mining for unsupervised 3D action representation learning," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 36–51.

[55] L. Lin, J. Zhang, and J. Liu, "Mutual information driven equivariant contrastive learning for 3D action representation learning," *IEEE Trans. Image Process.*, vol. 33, pp. 1883–1897, 2024.

[56] Y. Mao, J. Deng, W. Zhou, Y. Fang, W. Ouyang, and H. Li, "Masked motion predictors are strong 3D action representation learners," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 10147–10157.

[57] W. Nie, W. Wang, and X. Huang, "SRNet: Structured relevance feature learning network from skeleton data for human action recognition," *IEEE Access*, vol. 7, pp. 132161–132172, 2019.

[58] V. Choutas, P. Weinzaepfel, J. Revaud, and C. Schmid, "PoTion: Pose MoTion representation for action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7024–7033.

[59] A. Yan, Y. Wang, Z. Li, and Y. Qiao, "PA3D: Pose-action 3D machine for video recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Long Beach, CA, USA, Jun. 2019, pp. 7922–7931.

[60] 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.

[61] D. Ahn, S. Kim, H. Hong, and B. Chul Ko, "STAR-transformer: A spatio-temporal cross attention transformer for human action recognition," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2023, pp. 3330–3339.

[62] Z. Hu, J. Xiao, L. Li, C. Liu, and G. Ji, "Human-centric multimodal fusion network for robust action recognition," *Expert Syst. Appl.*, vol. 239, Apr. 2024, Art. no. 122314.

[63] B. X. B. Yu, Y. Liu, X. Zhang, S.-H. Zhong, and K. C. C. Chan, "MMNet: A model-based multimodal network for human action recognition in RGB-D videos," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 3, pp. 3522–3538, Mar. 2023.

[64] B. X. B. Yu, Y. Liu, and K. C. C. Chan, "Multimodal fusion via teacher–student network for indoor action recognition," in *Proc. AAAI Conf. Artif. Intell.*, vol. 35, 2021, pp. 3199–3207.

[65] Y. Liu, C. Li, S. Xu, and J. Han, "Part-whole relational fusion towards multi-modal scene understanding," *Int. J. Comput. Vis.*, vol. 133, no. 7, pp. 4483–4503, Jul. 2025.

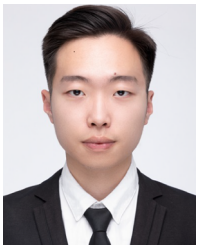
[66] T. Zhang, Q. Zhang, K. Debattista, and J. Han, "Cross-modality distillation for multi-modal tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 47, no. 7, pp. 5847–5865, Jul. 2025.

[67] Z. Li, M. Hu, Z. Qian, and X. Jiang, "Connecting consistency distillation to score distillation for text-to-3D generation," in *Proc. Eur. Conf. Comput. Vis.*, 2024, pp. 274–291.

[68] C. Chen, J. Han, and K. Debattista, "Virtual category learning: A semi-supervised learning method for dense prediction with extremely limited labels," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 8, pp. 5595–5611, Aug. 2024.

[69] J. Chen, H. Duan, X. Zhang, B. Gao, V. Grau, and J. Han, "From gaze to insight: Bridging human visual attention and vision language model explanation for weakly-supervised medical image segmentation," 2025, *arXiv:2504.11368*.

**Yijie Yang** received the B.S. degree from Northeastern University, Shenyang, China, in 2022, and the M.S. degree from the State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing (LIESMARS), Wuhan University, China, in 2025. He is currently pursuing the Ph.D. degree in computer science and technology with Fudan University, China. His research interests include video understanding, video generation, and computer vision.



**Jinlu Zhang** received the bachelor's degree in computer science from Shandong University in 2020 and the master's degree in computer technology from Wuhan University in 2023. He is currently pursuing the Ph.D. degree in computer science with Peking University. His current research interests include computer vision and machine learning. Specifically, he focused on human-centric artificial intelligence that can interact with different scenes and agents in the real world.

**Jiaxu Zhang** received the B.S. degree from Southeast University, Nanjing, China, in 2020. He is currently pursuing the Ph.D. degree with the State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing (LIESMARS), Wuhan University, China. His research interests include computer vision, computer graphics, and motion synthesis.

**Bo Du** (Senior Member, IEEE) received the Ph.D. degree in photogrammetry and remote sensing from the State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan, China, in 2010. He is currently a Professor with the School of Computer Science, Wuhan University. He has over 80 research articles published in the journals of IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, IEEE TRANSACTIONS ON IMAGE PROCESSING, IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, and *ISPRS Journal of Photogrammetry and Remote Sensing*. More than 30 of them are ESI hot articles or highly cited articles. His research interests include pattern recognition, hyperspectral image processing, machine learning, and signal processing.

**Zhigang Tu** (Senior Member, IEEE) received the Ph.D. degree from Wuhan University, China, in 2013, and the Ph.D. degree from Utrecht University, The Netherlands, in 2015.

From 2015 to 2016, he was a Postdoctoral Researcher with Arizona State University, USA. From 2016 to 2018, he was a Research Fellow with Nanyang Technological University, Singapore. He is currently a Professor with Wuhan University. He has co-/authored more than 80 papers in international SCI-indexed journals and conferences. His research interests include computer vision, image processing, and video analytics, with a focus on motion estimation/ retargeting, human behavior (action, pose, and gesture) recognition, reconstruction, and generation. He is the Area Chair of AAAI2023/2024/2025, the Associate Editor of the SCI-indexed journals of *CAAI Transactions on Intelligent Technology* (IF=7.3), *Journal of Visual Communication and Image Representation* (IF=3.1), and *Visual Computer* (IF=2.9). He received the Best Student Paper Award at the 4th Asian Conference on Artificial Intelligence Technology and one of the three best reviewers awards for IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY in 2022.