Topology-Aware Convolutional Neural Network for Efficient Skeleton-Based Action Recognition

Kailin Xu[1*†], **Fanfan Ye** [2†], **Qiaoyong Zhong** [2], **Di Xie** [2‡],

[1] Zhejiang University [2] Hikvision Research Institute
kailinxu@zju.edu.cn, {yefanfan, zhongqiaoyong, xiedi}@hikvision.com

Abstract

In the context of skeleton-based action recognition, graph convolutional networks (GCNs) have been rapidly developed, whereas convolutional neural networks (CNNs) have received less attention. One reason is that CNNs are considered poor in modeling the irregular skeleton topology. To alleviate this limitation, we propose a pure CNN architecture named Topology-aware CNN (Ta-CNN) in this paper. In particular, we develop a novel cross-channel feature augmentation module, which is a combo of map-attend-group-map operations. By applying the module to the coordinate level and the joint level subsequently, the topology feature is effectively enhanced. Notably, we theoretically prove that graph convolution is a special case of normal convolution when the joint dimension is treated as channels. This confirms that the topology modeling power of GCNs can also be implemented by using a CNN. Moreover, we creatively design a SkeletonMix strategy which mixes two persons in a unique manner and further boosts the performance. Extensive experiments are conducted on four widely used datasets, i.e. N-UCLA, SBU, NTU RGB+D and NTU RGB+D 120 to verify the effectiveness of Ta-CNN. We surpass existing CNN-based methods significantly. Compared with leading GCN-based methods, we achieve comparable performance with much less complexity in terms of the required GFLOPs and parameters.

1 Introduction

Skeleton-based action recognition has received widespread attention from the community, and many significant advances have been made in recent years. Unlike RGB-based image and video data, skeleton data are much more abstract and compact, reducing the parameters and computation resources required in the model design (Zhang et al. 2020). Looking back at the development of this field, methods that focus on hand-crafted features (Vemulapalli, Arrate, and Chellappa 2014; Zhang et al. 2019) have been surpassed by deep learning-based methods. With the emergence of graph convolutional networks (GCNs) (Yan, Xiong, and Lin 2018), the recognition performance is being improved rapidly. GCN is considered to be able to handle the irregular topological information in skeleton data properly, which has been verified in many previous works (Yan, Xiong, and Lin 2018; Ye et al. 2020; Chen et al. 2021a). One obvious limitation is that most of these pure GCN-based methods rely on heavy models (Liu et al. 2020; Shi et al. 2021). To attack the weakness, some works combine CNN and GCN into a compact model and get decent performance while balancing the model size (Zhang et al. 2020). However, few pure CNN-based methods (Zhang et al. 2019) were proposed to address the task and achieve competitive performance with GCN-based methods. It is widely recognized that CNN can not exploit the skeleton topology sufficiently. Inspired by this factor, a question came to our mind. Can we enhance the topology feature of a CNN model by borrowing the topology modeling insights from GCN-based methods?

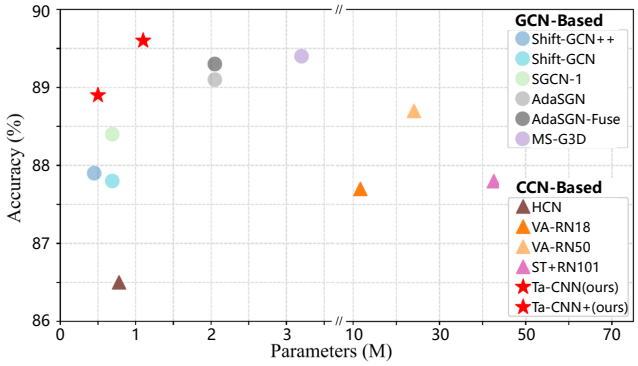


Figure 1: Comparison of various methods on NTU RGB+D with the cross-subject benchmark in terms of accuracy and number of parameters. The proposed Ta-CNN achieves state-of-the-art performance with a tiny model size.

Previous CNN-based methods mostly treat each dimension of the coordinate feature equally in convolutional layers. Also, none of them focuses on explicit modeling of the interaction among joints due to lack of a valid tool as the adjacency matrix for GCN. We argue that enhancing the coordinate and joint features is the key to effectively model the skeleton topology using CNN. To this end, we propose a novel Topology-aware CNN (Ta-CNN), which can fully mine the topological information of skeleton data and achieve remarkable performance. Based on CNN, it is

Copyright © 2022, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

*Internship at Hikvision Research Institute.

†Both authors contributed equally to this work.

‡Di Xie is the corresponding author.

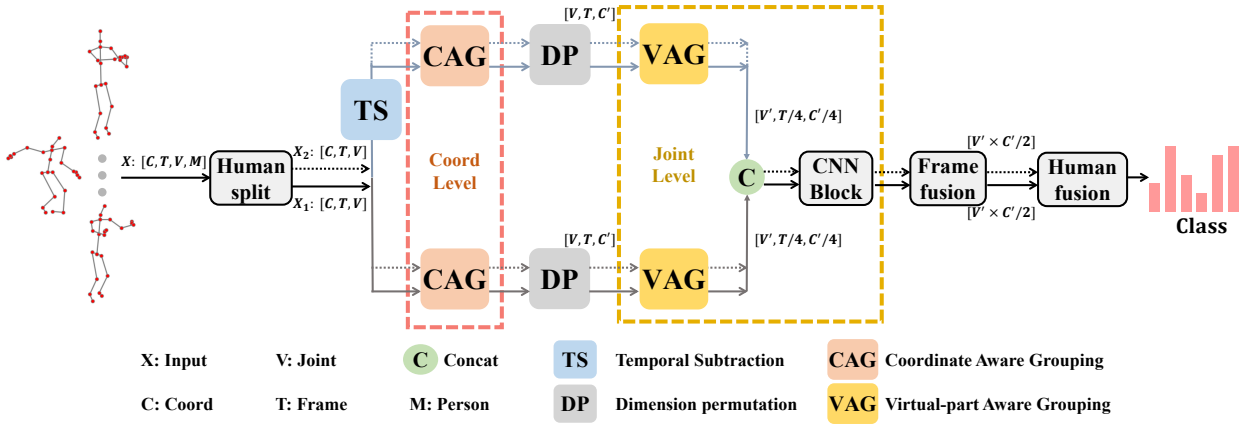


Figure 2: Pipeline of the proposed Ta-CNN framework. Two streams, *i.e.* the skeleton sequence and the motion information are fed into the network and fused in the middle. CAG and VAG are the key components to enhance the topology feature.

easy to design a lightweight architecture with a low budget for GFLOPs and parameters (see Figure 1). This makes our model more friendly than GCNs for deployment.

The pipeline of the proposed Ta-CNN framework is illustrated in Figure 2. Following HCN (Li et al. 2018), we first learn the per-joint coordinate feature, followed by per-person joint feature, and features of multiple persons are fused in a late stage. Specifically, we propose a novel cross-channel feature augmentation module, which is a combo of map-attend-group-map operations. It is adapted to enhance both coordinate feature and joint feature, resulting in the coordinate aware grouping (CAG) module and the virtual-part aware grouping (VAG) module. CAG is composed of three building blocks named feature mapping, channel attention and dual coordinate-wise convolution. This design is intuitive. In a multi-dimensional coordinate space, the importances of different coordinates are different for recognition of the underlying action. For example, it is easy to recognize the action of *walking* from the side view, but difficult from the front view. To model the interaction among joints, we transpose the joint dimension into channels as is done in HCN. The transposition trick is the key to make pure CNN models benefit from the topology of skeleton data. With a theoretical analysis of convolution with the joint dimension as channels, we conclude that graph convolution can be regarded as a special case of convolution. To fully explore the topology within different body parts, we append the virtual-part aware grouping module after the CAG module. CAG and VAG effectively enhance the topology feature and thus boost the performance without introducing heavy extra computational cost.

In addition, inspired by the mixup (Zhang et al. 2018) strategy for augmentation of image data, we creatively invent a SkeletonMix strategy for augmentation of skeleton data. Due to the difference in modality, the vanilla mixup does not apply to skeleton data. In the proposed SkeletonMix, we mix two samples by randomly combining the upper body and the lower body of two different persons. SkeletonMix significantly enriches the diversity of samples during training and brings robustness and accuracy gain.

Our contributions can be summarized as follows:

- We propose a novel pure CNN-based framework for skeleton-based action recognition. Equipped with the CAG and VAG modules, the topology feature is effectively enhanced without requiring a heavy network.
- We are the first to theoretically analyze and reveal the close relation between graph convolution and normal convolution, which may guide the design of CNN-based models.
- A novel SkeletonMix strategy is invented for augmentation of skeleton data, leading to robust learning and better performance.
- The proposed method surpasses all CNN-based methods and achieves comparable performance with GCN-based methods with significantly reduced parameters and GFLOPs.

2 Related Work

Skeleton-based Action Recognition. For this task, deep learning-based methods are the current state-of-the-arts and achieve significantly better performance than those using hand-crafted features (Vemulapalli, Arrate, and Chellappa 2014; Zhang et al. 2019). Here, we divide existing deep learning models into roughly three categories, namely RNNs, CNNs and GCNs.

RNN (Zaremba, Sutskever, and Vinyals 2014) and its variants (Cho et al. 2014) are a natural choice to model the temporal dynamics in skeleton data. In the early years, RNN has been incorporated for action recognition (Shahroudy et al. 2016; Li et al. 2017b). Later many CNN-based methods emerged, because CNN is found to be able to encode spatiotemporal feature at the same time. To make CNN work on skeleton data, some works chose to convert skeleton information into images (Du, Fu, and Wang 2015; Liu, Liu, and Chen 2017; Zhang et al. 2019), and others directly used the 3D skeleton data as input (Kim and Reiter 2017; Li et al. 2017a, 2018). Zhang et al. combined the view adaptive module with CNN, achieving the best result among CNN-based methods.

Owing to the advantages in dealing with irregular graphical structures, GCNs have taken an absolute leading position in the context of skeleton-based action recognition. The first successful work is ST-GCN (Yan, Xiong, and Lin 2018), which learn the spatial and temporal patterns in skeleton data with graph convolution. Based on ST-GCN, there are many follow-up works which mainly focus on improving the local and global perception of GCN (Liu et al. 2020; Zhang et al. 2020; Cheng et al. 2020b; Ye et al. 2020; Chen et al. 2021b) and reducing the model complexity (Zhang et al. 2020; Cheng et al. 2020b; Ye et al. 2020). The state-of-the-art performance is obtained by CTR-GCN (Chen et al. 2021a). This work proposed to relax the constraints of other graph convolutions and used the specific correlation to better model the channel topology. Although recent studies have mentioned the weakness of CNN in extracting skeleton topology information (Yan, Xiong, and Lin 2018; Ye et al. 2020; Chen et al. 2021a; Ye et al. 2019; Ye and Tang 2019), in this paper, we prove that the potential of CNN has been underestimated. With a carefully designed feature augmentation module and training strategy, we are able to achieve competitive performance using a pure CNN model.

Mixup (Zhang et al. 2018) is a data augmentation strategy which enriches the samples by mixing two images and their labels. It successfully improves the performance of state-of-the-art image recognition models on many datasets such as ImageNet and CIFAR. Then many variants of mixup are proposed by performing other types of mixing and interpolation (Verma et al. 2019). Summers and Dinneen studied a broader scope of mixed-example data augmentation. To address the generation of unnatural samples in mixup, CutMix (Yun et al. 2019) replaces the image region with a patch cropped from another image. ResizeMix (Qin et al. 2020) further improves CutMix by replacing the cropping operation with resizing of the whole source image. The above strategies can not be adopted for skeleton data directly, as they will cause severe unreasonable deformation to the skeleton topology. Instead, we invent a novel SkeletonMix strategy, which is specialized for augmentation of skeleton data and distinct from the way to mix images.

3 Method

In this section, we first briefly review the task of CNN-based skeleton-based action recognition. Then we elaborate the details of the proposed Ta-CNN framework, including the coordinate aware grouping module and the virtual-part aware grouping module. After that, the SkeletonMix strategy for skeleton data augmentation will be presented.

3.1 Preliminaries

Skeleton-based action recognition is essentially a classification problem. The input is a sequence of skeleton data, which can be denoted as $\mathbf{X} \in R^{C \times T \times V}$. C, T and V represent the number of coordinates, time sequence length and the number of joints respectively. In this form, a skeleton sequence can be interpreted as a multi-spectral image. That is, the coordinate dimension C is analogous to the channels of an image, and the temporal and joint dimensions $T \times V$ can be regarded as height and width of the image. Following HCN (Li et al. 2018), the skeleton sequence is treated as a 3D tensor and fed into the network directly.

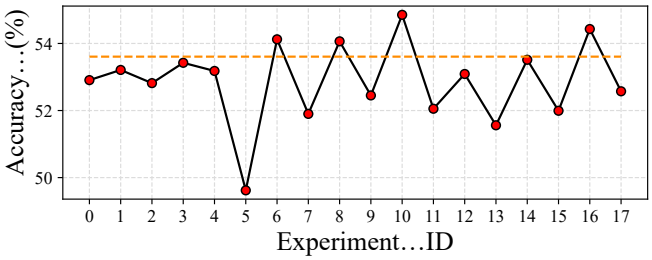


Figure 3: Results of the random coordinate scaling experiment, which is repeated for 18 times. The dotted line presents the accuracy without coordinate tweaking.

3.2 Enhancing Coordinate Feature via CAG

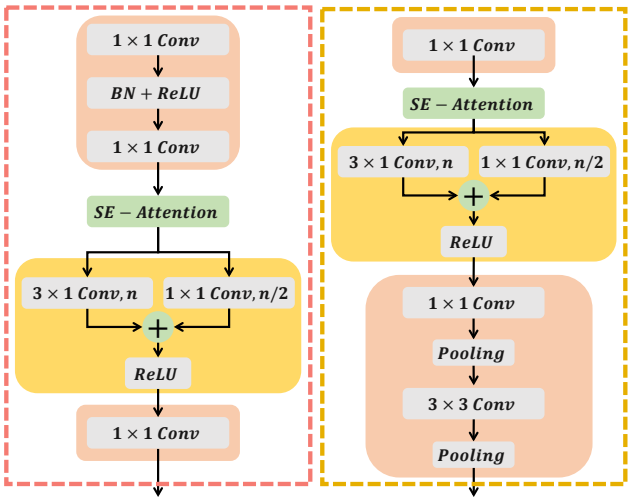
To validate our hypothesis on the different importances of different coordinates, we design a toy experiment. Specifically, we tweak the coordinate feature by multiplying a random scale factor from 0 to 1 to each of the three coordinate (x, y, z) and evaluate the accuracy. For simplicity, we use a tiny 5-layer network which is simplified from HCN by removing the convolutions following transposition operation. The model is trained on four difficult classes of NTU-RGB+D, namely reading, writing, playing with phone/tablet and typing on a keyboard. For each setting of scale factors, we train the model three times using different random seeds, and compute the mean accuracy. The results of coordinate scaling are shown in Figure 3. Unsurprisingly, the performance fluctuates greatly as different scale factors are applied to the coordinates and the accuracy without tweaking is not the best. This suggests that the coordinate feature deserves more attention.

The concept of cross-channel feature augmentation is first applied to the coordinate feature, instantiated as the coordinate aware grouping (CAG) module. As shown in Figure 4(a), CAG is composed of three main blocks, *i.e*. feature mapping, channel attention and dual coordinate-wise convolution.

Feature Mapping is mainly constructed with one or a few convolutional layers. It is used to map the coordinate feature into a high-dimensional latent space or reduce the dimension.

Channel Attention is implemented based on the squeeze-and-excitation attention mechanism (Hu, Shen, and Sun 2018). It learns channel-wise attention, which essentially enhances discriminative coordinate axes and suppresses unimportant and confusing axes. In other words, this operation helps the model dig out the most useful topology representation implicitly.

Dual Coordinate-wise Convolution is the key component of CAG. Two parallel grouped convolutions are leveraged to further strengthen the coordinate feature. One is a



(a) CAG (b) VAG

Figure 4: The detailed architectures of CAG and VAG. The blocks in pink, green and gold denote the feature mapping, channel attention and grouping operations respectively. The grouping operation is implemented as dual coordinate-wise convolution for CAG (a) and dual part-wise convolution for VAG (b).

3×1 convolution with n groups. The other is a 1×1 convolution with $n/2$ groups. The outputs of the two grouped convolutions are fused by element-wise summation. With grouped convolution, the entire high-dimensional feature space is divided into multiple low-dimensional subspaces. Then feature fusion occurs within each subspace separately. Besides, by using different numbers of groups in the two convolutional layers, the feature space is partitioned into subspaces of different granularities. This block effectively enhance the coordinate feature.

The overall process of CAG can be formulated as:

$$\mathbf{Z} = \mathcal{M}_2(\mathcal{G}(\mathcal{A}(\mathcal{M}_1(\mathbf{X})), \dot{\theta}) + \mathcal{G}(\mathcal{A}(\mathcal{M}_1(\mathbf{X})), \ddot{\theta}))), \quad (1)$$

where $\mathbf{X} \in R^{C\times T\times V}$ and $\mathbf{Z} \in R^{C'\times T\times V}$ are the input and output of the CAG module respectively.

$\mathcal{M}(\cdot)$ and $\mathcal{A}(\cdot)$ mean feature mapping and the channel attention individually. $\mathcal{G}(\cdot, \dot{\theta})$ and $\mathcal{G}(\cdot, \ddot{\theta})$ are the two grouped convolutions of the dual coordinate-wise convolution module, which can be expressed as

$$\mathcal{G}(\widetilde{\mathbf{X}}, \theta) = \left[f(\widetilde{X}^1, \theta^1), f(\widetilde{X}^2, \theta^2), \ldots, f(\widetilde{X}^n, \theta^n)\right], \quad (2)$$

where $\widetilde{\mathbf{X}} = [\widetilde{X}^1, \widetilde{X}^2, \ldots, \widetilde{X}^n]$ represents the input feature map partitioned into n groups, and $\theta = [\theta^1, \theta^2, \ldots, \theta^n]$ denotes the weights of the convolutional layer f. $[\cdot]$ is the operation of concatenation.

3.3 Graph Convolution Is a Special Convolution

Here we illustrate a tight relation between graph convolution and normal convolution. Given a skeleton sequence $\mathbf{X} \in R^{C\times T\times V}$, the graph convolution can be formulated as

$$\mathbf{Y} = \mathbf{WXA}, \quad (3)$$

where $\mathbf{A} \in R^{V\times V}$ denotes the adjacency matrix representing the skeleton topological information, and $\mathbf{W} \in R^{C'\times C}$ is the weighting matrix for feature transformation. For each element in $\mathbf{Y} \in R^{C'\times T\times V}$, if we ignore the feature transformation, the core calculation could be shown as

$$Y_{c,t,v} = \sum_{u=0}^{V-1} X_{c,t,u} A_{u,v} \quad (4)$$

The essence of GCN is to aggregate the temporal-spatial features over all joints by weighted average guided by the adjacency matrix.

When the joint dimension is transposed as channels, *i.e.* $\mathbf{X} \in R^{V\times T\times C}$, a normal convolution with a kernel size of $P \times Q$ could be formulated as

$$Y_{v,t,c} = \sum_{u=0}^{V-1} \sum_{p=0}^{P-1} \sum_{q=0}^{Q-1} w_{v,u,p,q} X_{u,t+p-P/2,c+q-Q/2}, \quad (5)$$

where p and q represent the indices of the two-dimensional kernel. When the kernel size is 1×1, Eq. (5) is simplified to

$$Y_{v,t,c} = \sum_{u=0}^{V-1} w_{v,u} X_{u,t,c} \quad (6)$$

If the number of output channels is equal to the input channels, and the adjacency matrix is employed as the weights, then the convolution is equivalent to the graph convolution. On the other hand, graph convolution can be considered as a special case of convolution. Notably, with a larger kernel size (*e.g.* 3×3), the convolutional network can perceive not only the topology of joints, but also the temporal feature and coordinate feature jointly. By varying the number of output channels, we are able to expand or reduce the joint dimension, which makes the CNN-based model more flexible and expressive than GCN.

3.4 Enhancing Joint Feature via VAG

From the relation between graph convolution and normal convolution, we can conclude that CNN is capable of modeling the topology of joints implicitly. However, we argue that this capability can be further improved. Huang et al. split the skeleton into multiple parts explicitly and aggregate features of each part hierarchically. Inspired by this work, we introduce the virtual-part aware grouping module (VAG). It follows the same map-attend-group-map paradigm as CAG does. As shown in Figure 4(b), VAG mainly consists of three blocks, namely feature mapping, channel attention and dual part-wise convolution. Note that although the three blocks are analogous to those of CAG, their architectures have been adapted. Here the two grouped convolutions essentially divide the virtual joints into multiple virtual parts, which helps to learn more discriminative joint feature.

3.5 Ta-CNN Framework

Figure 2 illustrates the architecture of our Ta-CNN framework. Like HCN, there are also two input streams, *i.e.* the skeleton sequence and the skeleton motion. Accordingly

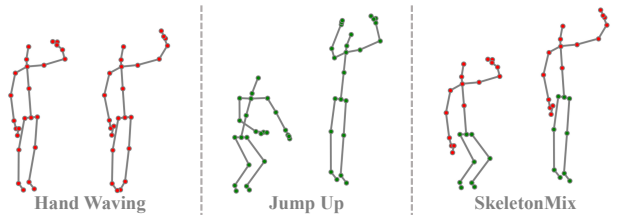


Figure 5: SkeletonMix augments skeleton data by combining the upper body and the lower body of two persons performing different actions.

there two sub-networks, both of which are equipped with CAG and VAG. Features of multiple persons are fused by element-wise maximum in a late stage. We tweak the architecture of HCN by adding batch normalization after the first convolutional layer and replacing the first fully-connected layer with a mean operation, which slightly improves the performance with fewer parameters. The modified HCN is adopted as a strong baseline. The detailed configuration of the network will be given in the supplementary material.

3.6 SkeletonMix

Mixup (Zhang et al. 2018) was originally designed for augmentation of image data. It works by mixing two images and their labels linearly by

$$\begin{cases} \hat{x} = \lambda x_i + (1-\lambda)x_j \\ \hat{y} = \lambda y_i + (1-\lambda)y_j \end{cases}, \tag{7}$$

where (x_i, y_i) and (x_j, y_j) are two image-label pairs sampled from the training dataset and $(\hat{x}, \hat{y})$ is the output sample. $\lambda \in [0, 1]$ is used to control the bias between the two input samples. If we apply the mixup strategy directly to skeleton data, we observe severe deformation to the skeleton topology, leading to poor performance.

Here, we invent a novel mixup strategy specialized for skeleton data. SkeletonMix works by randomly combining the upper body and the lower body of two different persons, which can be formulated as

$$\begin{cases} \hat{x} = Concat(x_{i,V_u}, x_{j,V_l}) \\ \hat{y} = \lambda y_i + (1-\lambda)y_j \end{cases}, \tag{8}$$

where V_u and V_l are the joint indices for the upper body and the lower body respectively. Figure 5 shows an example, in which two actions, *i.e.* hand waving and jump up are mixed into a new multi-label action. In our experiments, we randomly select a part of samples in a batch with a certain ratio $\alpha \in [0, 1]$ to perform the SkeletonMix augmentation, and the rest retain unchanged.

4 Experiments

Our method is evaluated on four widely used datasets, *i.e.* NTU RGB+D, NTU RGB+D 120, Northwestern-UCLA and SBU Kinect Interaction. Extensive ablation studies are conducted to verify the impact of different components of the framework. Finally, performance comparisons with state-of-the-art methods are reported.

Method	Param. (M)	GFLOPs	Acc. (%)
HCN (Li et al. 2018)	0.78	0.06	86.5
Baseline	0.54	0.11	87.4(↑ 0.9)
+VAG (6)	0.53	0.09	87.8 (↑ 1.3)
+VAG (10)	0.53	0.09	87.8 (↑ 1.3)
+VAG (10)+CAG (10)	0.53	0.08	88.7 (↑ 2.2)
+VAG (6)+CAG (10)	0.53	0.08	88.8 (↑ 2.3)

Table 1: Accuracy gains of VAG and CAG on NTU RGB+D.

Method	Param. (M)	GFLOPs	Acc. (%)
Baseline	0.54	0.11	87.4
+CA	0.56	0.11	87.9 (↑ 0.5)
+CA+FM	0.54	0.12	88.3 (↑ 0.9)
+CA+FM+SGC	0.53	0.08	88.1 (↑ 0.7)
+CA+FM+DGC	0.53	0.08	88.8 (↑ 1.4)

Table 2: Accuracy gains by channel attention (CA), feature mapping (FM) and dual grouped convolution (DGC) on NTU RGB+D. SGC represents single grouped convolution.

4.1 Datasets

NTU RGB+D contains 56,880 samples of 60 classes performed by 40 distinct subjects. There are two recommended benchmarks, *i.e.* cross-subject (X-sub) and cross-view (X-view).

NTU RGB+D 120 is an extension of NTU RGB+D containing 114,480 samples. The numbers of classes and subjects are expanded to 120 and 106 respectively. There are two benchmarks, *i.e.* cross-subject (X-sub) and cross-setup (X-setup).

Northwestern-UCLA (N-UCLA) is captured by using three Kinect cameras. It contains 1494 samples of 10 classes. The same evaluation protocol in Wang et al. is adopted.

SBU Kinect Interaction (SBU) depicts human interaction captured by Kinect. It contains 282 sequences covering 8 action classes. Subject-independent 5-fold cross-validation is performed following the same evaluation protocol as previous works (Yun et al. 2012).

4.2 Implementation Details

The proposed model is implemented in PyTorch (Paszke et al. 2017). The model is trained for 800 epochs with the Adam optimizer. The learning rate is set to 0.001 initially and decayed by a factor of 0.1 at 650, 730 and 770 epochs for all datasets. The weight decay is set to 0.0002 for SBU and 0.0001 for the rest datasets. The batch sizes for NTU RGB+D, NTU RGB+D 120, N-UCLA and SBU are 64, 64, 16 and 8 respectively. As for data pre-processing method, we follow previous works (Shi et al. 2019; Cheng et al. 2020b; Li et al. 2018). For SBU, a warmup strategy is used during the first 30 epochs. The λ in Eq. (8) is set to 0.6, and the mixing ratio α for SkeletonMix is set to $1/16$.

Method	w/o SkeletonMix		w/ SkeletonMix	
	X-sub	X-setup	X-sub	X-setup
HCN (2018)	76.5	75.1	76.5	78.2
2s-AGCN (2019)	76.4	78.4	76.9	80.0

Table 3: Performance of SkeletonMix combined with HCN and 2s-AGCN on NTU RGB+D 120.

Modality	w/o SkeletonMix		w/ SkeletonMix	
	X-sub	X-setup	X-sub	X-setup
Joint	82.1	83.5	82.4	84.0
Bone	82.3	84.0	82.6	84.4
Joint Motion	77.2	78.7	77.5	79.0
Bone Motion	76.9	78.6	77.1	78.9

Table 4: Performance of SkeletonMix when trained using various data modalities on NTU RGB+D 120.

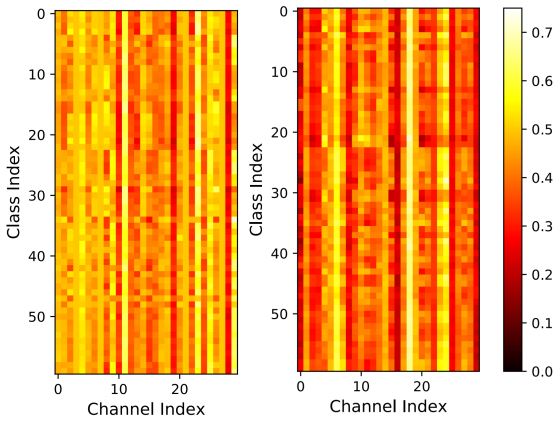
4.3 Ablation Studies

Effectiveness of CAG and VAG. Table 1 shows the performance gains brought by CAG and VAG with the cross-subject benchmark on the NTU RGB+D dataset. We first evaluate the strong baseline model which is derived from HCN. It improves the accuracy of HCN by 0.9% with fewer parameters. Starting from the strong baseline, VAG improves the accuracy slightly with GFLOPs reduced from 0.11 to 0.9. VAG with $n = 6$ and $n = 10$ achieve similar accuracy. CAG further boosts the accuracy by 1% without increasing the parameters and GFLOPs. In total we achieve an accuracy of 88.8%, which improves the accuracy reported in HCN by 2.3% with a more efficient model.

Impact of the Blocks in CAG & VAG. Both CAG and VAG are composed of the feature mapping (FM), channel attention (CA) and dual grouped convolution (DGC) blocks. Table 2 shows their individual contribution to the final performance. CA, FM and DGC improve the accuracy by 0.5%, 0.4% and 0.5% respectively. It is worth noting that single grouped convolution (SGC) slightly harms the performance as the GFLOPs gets reduced. This indicates that grouped convolution alone does not suffice to enhance the topology feature, and the design of two convolutions with different numbers of groups is necessary.

Visualization of Channel Attention. Figure 6 visualizes the learned channel attention on the NTU RGB+D dataset. The two heat maps show the average attention response for each channel and each class in the test dataset in CAG and VAG of the skeleton sequence branch. We can see for some channels, the attention values are highly correlated between different classes. While for others, the response varies for different classes. This confirms our analysis that the dimensions of coordinate and joint features are not equally important for recognition of the action.

Effectiveness of SkeletonMix. To validate the proposed SkeletonMix strategy, we conduct experiments on the more challenging NTU RGB+D 120 dataset using both cross-subject and cross-setup benchmarks. Apart from the proposed Ta-CNN, we also apply it to two existing methods, *i.e.* HCN and 2s-AGCN. As compared in Table 3, both HCN and 2s-AGCN benefit from the strategy a lot, especially in the cross-setup setting. This proves the generalizability of SkeletonMix. As shown in Table 4, for Ta-CNN, no matter what data modality is used as input, it can bring considerable improvement. Note that as compared in Table 5, ap-



(a) Coordinate attention (b) Joint attention

Figure 6: Visualization of the learned (a) coordinate attention in CAG and (b) joint attention in VAG.

Method	Baseline	Mixup	SMix ($\alpha : 1$)	SMix ($\alpha : 1/16$)
Acc	82.1	79.8	75.7	82.4

Table 5: Comparison of the original mixup and SkeletonMix (SMix) with two mixing ratios (α) on NTU RGB+D 120.

Type	Method	Param. (M)	GFLOPs	Acc
CNN	HCN (2018)	0.78	0.06	86.5
	SResNet101 (2019)	42.62	-	87.8
	SResNet152 (2019)	58.27	-	88.2
	VResNet10 (2019)	5.39	-	83.0
	VResNet18 (2019)	11.66	-	87.7
	VResNet50 (2019)	24.09	-	88.7
GCN	Shift-GCN (2020b)	0.69	2.50	87.8
	Shift-GCN++ (2021)	0.45	0.40	87.9
	SGN-5 (2020)	0.69	0.80	89.0
	SGN-1 (2020)	0.69	0.16	88.4
	AdaSGN (2021)	2.05	0.07	89.1
	AdaSGN-Fuse (2021)	2.05	0.32	89.3
	MS-G3D (2020)	3.20	24.4	89.4
Ours	Ta-CNN	0.53	0.08	88.8
	Ta-CNN+	1.06	0.16	89.6

Table 6: The effectiveness and efficiency of Ta-CNN compared with the state-of-the-art methods on NTU RGB+D. SResNet and VResNet mean S-trans+ResNet and VA-ResNet respectively.

Dataset	Method	Acc. (%)
N-UCLA	Lie Group (2014)	74.2
	Actionlet ensemble (2013)	76.0
	HBRNN-L (2015)	78.5
	Ensemble TS-LSTM (2017)	89.2
	VA-CNN (aug.) (2019)	90.7
	VA-RNN (aug.) (2019)	93.8
	VA-fusion (aug.) (2019)	95.3
	AGC-LSTM (2019)	93.3
	Shift-GCN (2020b)	94.6
	Shift-GCN++ (2021)	95.0
	DC-GCN+ADG (2020a)	95.3
	CTR-GCN (2021a)	96.5
	Ta-CNN	96.1
	Ta-CNN+	97.2
SBU	HBRNN-L (2015)	80.4
	Co-occurrence RNN (2016)	90.4
	STA-LSTM (2017)	91.5
	ST-LSTM + Trust Gate (2016)	93.3
	GCA-LSTM (2017)	94.1
	Clips+CNN+MTLN (2017)	93.6
	VA-RNN (aug.) (2019)	97.5
	VA-CNN (aug.) (2019)	95.7
	VA-fusion (aug.) (2019)	98.3
	HCN (2018)	98.6
	Ta-CNN	98.5
	Ta-CNN+	98.9

Table 7: Comparisons of the top-1 accuracy (%) with the state-of-the-art methods on N-UCLA and SBU.

plying the original mixup strategy directly to skeleton data leads to performance degradation. And the mixing ratio (α) of SkeletonMix is also important.

Efficiency of Ta-CNN. As a pure CNN-based method, we compare the proposed Ta-CNN with previous CNN-based and GCN-based methods in terms of the number of parameters and GFLOPs. As summarized in Table 6, we achieve state-of-the-art performance while balancing the efficiency. Ta-CNN+ is an ensemble of two Ta-CNNs with different n (6 and 10) in VAG. Although the computational complexity is doubled, it is still competitive considering the improved accuracy. This characteristic makes it very suitable for deployment, especially on edge devices where computational and memory resources are limited.

4.4 Comparison with the State-of-the-arts

Here, we compare our final performance with existing methods on the aforementioned four datasets. On the N-UCLA and CBU datasets, we obtain an accuracy of 97.2% and 98.9% respectively (see Table 7), setting the new state-of-the-art. On NTU RGB+D (Table 8) and NTU RGB+D 120 (Table 9), we outperform all existing CNN-based methods by a large margin with much fewer parameters and calculation amounts. Compared with the well developed GCN-based methods, we achieve comparable performance with the superiority in terms of model size.

Type	Method	X-sub	X-view
Heavy GCN	2s-AGCN (2019)	88.5	95.1
	PL-GCN (2020)	89.2	95.0
	AGC-LSTM (2019)	89.2	95.0
	Dynamic GCN (2020)	91.5	96.0
	MST-GCN (2021b)	91.5	96.6
	MS-G3D (2020)	91.5	96.2
	CTR-GCN (2021a)	92.4	96.8
Light GCN	SGN (2020)	89.0	94.5
	Shift-GCN (2020b)	90.7	96.5
	Shift-GCN++ (2021)	90.5	96.3
	3s-AdaSGN (2021)	90.5	95.3
CNN	Res-TCN (2017)	74.3	83.1
	Two-stream CNN (2017a)	83.2	89.3
	HCN (2018)	86.5	91.1
	VA-CNN (aug.) (2019)	88.7	94.3
	VA-fusion (aug.) (2019)	89.4	95.0
Ours	Ta-CNN	90.4	94.8
	Ta-CNN+	90.7	95.1

Table 8: Comparisons of the top-1 accuracy (%) with the state-of-the-art methods on NTU RGB+D.

Type	Method	X-sub	X-setup
Heavy GCN	2s-AGCN (2019)	82.5	84.2
	Dynamic GCN (2020)	87.3	88.6
	MST-GCN (2021b)	87.5	88.8
	MS-G3D (2020)	86.9	88.4
	CTR-GCN (2021a)	88.9	90.6
Light GCN	SGN (2020)	79.2	81.5
	Shift-GCN (2020b)	85.9	87.6
	Shift-GCN++ (2021)	85.6	87.2
	3s-AdaSGN (2021)	85.9	86.8
Ours	Ta-CNN	85.4	86.8
	Ta-CNN+	85.7	87.3

Table 9: Comparisons of the top-1 accuracy (%) with the state-of-the-art methods on NTU RGB+D 120.

5 Conclusion

This paper proposes a novel pure CNN model for skeleton-based action recognition. By rethinking the weakness of CNN in encoding the irregular skeleton topology, we are committed to enhance the topology feature. With a carefully designed cross-channel feature augmentation module and a mixup strategy specialized for skeleton data, we achieve state-of-the-art performance with a tiny model. In addition, we theoretically prove that graph convolution is essentially a special case of normal convolution when the joint dimension is treated as channels. The conclusion is consistent with our model design. We argue that the potential of CNN for modeling of irregular graph data beyond skeleton data has not been fully exploited and deserves more attention. The excellent performance and tiny model size make our method well suitable for real-world deployment.

References

Chen, Y.; Zhang, Z.; Yuan, C.; Li, B.; Deng, Y.; and Hu, W. 2021a. Channel-wise topology refinement graph convolution for skeleton-based action recognition. In *Proceedings of the IEEE international conference on computer vision*, 13359–13368.

Chen, Z.; Li, S.; Yang, B.; Li, Q.; and Liu, H. 2021b. Multi-Scale Spatial Temporal Graph Convolutional Network for Skeleton-Based Action Recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 1113–1122.

Cheng, K.; Zhang, Y.; Cao, C.; Shi, L.; Cheng, J.; and Lu, H. 2020a. Decoupling GCN with DropGraph Module for Skeleton-Based Action Recognition. In *Proceedings of the European Conference on Computer Vision*.

Cheng, K.; Zhang, Y.; He, X.; Chen, W.; Cheng, J.; and Lu, H. 2020b. Skeleton-based action recognition with shift graph convolutional network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 183–192.

Cheng, K.; Zhang, Y.; He, X.; Cheng, J.; and Lu, H. 2021. Extremely Lightweight Skeleton-Based Action Recognition With ShiftGCN++. *IEEE Transactions on Image Processing*, 30: 7333–7348.

Cho, K.; van Merriënboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; and Bengio, Y. 2014. Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, 1724–1734.

Du, Y.; Fu, Y.; and Wang, L. 2015. Skeleton based action recognition with convolutional neural network. In *2015 3rd IAPR Asian Conference on Pattern Recognition*, 579–583. IEEE.

Du, Y.; Wang, W.; and Wang, L. 2015. Hierarchical recurrent neural network for skeleton based action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1110–1118.

Hu, J.; Shen, L.; and Sun, G. 2018. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 7132–7141.

Huang, L.; Huang, Y.; Ouyang, W.; and Wang, L. 2020. Part-level graph convolutional network for skeleton-based action recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 11045–11052.

Ke, Q.; Bennamoun, M.; An, S.; Sohel, F.; and Boussaid, F. 2017. A new representation of skeleton sequences for 3d action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3288–3297.

Kim, T. S.; and Reiter, A. 2017. Interpretable 3d human action analysis with temporal convolutional networks. In *2017 IEEE conference on computer vision and pattern recognition workshops*, 1623–1631. IEEE.

Lee, I.; Kim, D.; Kang, S.; and Lee, S. 2017. Ensemble deep learning for skeleton-based action recognition using temporal sliding lstm networks. In *Proceedings of the IEEE international conference on computer vision*, 1012–1020.

Li, C.; Zhong, Q.; Xie, D.; and Pu, S. 2017a. Skeleton-based action recognition with convolutional neural networks. In *2017 IEEE International Conference on Multimedia & Expo Workshops*, 597–600. IEEE.

Li, C.; Zhong, Q.; Xie, D.; and Pu, S. 2018. Co-occurrence feature learning from skeleton data for action recognition and detection with hierarchical aggregation. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, 786–792.

Li, W.; Wen, L.; Chang, M.-C.; Nam Lim, S.; and Lyu, S. 2017b. Adaptive RNN tree for large-scale human action recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, 1444–1452.

Liu, J.; Shahroudy, A.; Xu, D.; and Wang, G. 2016. Spatio-temporal lstm with trust gates for 3d human action recognition. In *European conference on computer vision*, 816–833. Springer.

Liu, J.; Wang, G.; Hu, P.; Duan, L.-Y.; and Kot, A. C. 2017. Global context-aware attention lstm networks for 3d action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1647–1656.

Liu, M.; Liu, H.; and Chen, C. 2017. Enhanced skeleton visualization for view invariant human action recognition. *Pattern Recognition*, 68: 346–362.

Liu, Z.; Zhang, H.; Chen, Z.; Wang, Z.; and Ouyang, W. 2020. Disentangling and unifying graph convolutions for skeleton-based action recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 143–152.

Paszke, A.; Gross, S.; Chintala, S.; Chanan, G.; Yang, E.; DeVito, Z.; Lin, Z.; Desmaison, A.; Antiga, L.; and Lerer, A. 2017. Automatic differentiation in pytorch.

Qin, J.; Fang, J.; Zhang, Q.; Liu, W.; Wang, X.; and Wang, X. 2020. Resizemix: Mixing data with preserved object information and true labels. *arXiv preprint arXiv:2012.11101*.

Shahroudy, A.; Liu, J.; Ng, T.-T.; and Wang, G. 2016. Ntu rgb+ d: A large scale dataset for 3d human activity analysis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1010–1019.

Shi, L.; Zhang, Y.; Cheng, J.; and Lu, H. 2019. Two-stream adaptive graph convolutional networks for skeleton-based action recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 12026–12035.

Shi, L.; Zhang, Y.; Cheng, J.; and Lu, H. 2021. AdaSGN: Adapting Joint Number and Model Size for Efficient Skeleton-Based Action Recognition. In *Proceedings of the IEEE International Conference on Computer Vision*.

Si, C.; Chen, W.; Wang, W.; Wang, L.; and Tan, T. 2019. An attention enhanced graph convolutional lstm network for skeleton-based action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 1227–1236.

Song, S.; Lan, C.; Xing, J.; Zeng, W.; and Liu, J. 2017. An end-to-end spatio-temporal attention model for human action recognition from skeleton data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31.

Summers, C.; and Dinneen, M. J. 2019. Improved mixed-example data augmentation. In *2019 IEEE Winter Conference on Applications of Computer Vision*, 1262–1270. IEEE.

Vemulapalli, R.; Arrate, F.; and Chellappa, R. 2014. Human action recognition by representing 3d skeletons as points in a lie group. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 588–595.

Verma, V.; Lamb, A.; Beckham, C.; Najafi, A.; Mitliagkas, I.; Lopez-Paz, D.; and Bengio, Y. 2019. Manifold mixup: Better representations by interpolating hidden states. In *International Conference on Machine Learning*, 6438–6447. PMLR.

Wang, J.; Liu, Z.; Wu, Y.; and Yuan, J. 2013. Learning actionlet ensemble for 3D human action recognition. *IEEE transactions on pattern analysis and machine intelligence*, 36(5): 914–927.

Wang, J.; Nie, X.; Xia, Y.; Wu, Y.; and Zhu, S.-C. 2014. Cross-view action modeling, learning and recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2649–2656.

Yan, S.; Xiong, Y.; and Lin, D. 2018. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Ye, F.; Pu, S.; Zhong, Q.; Li, C.; Xie, D.; and Tang, H. 2020. Dynamic GCN: Context-enriched topology learning for skeleton-based action recognition. In *Proceedings of the 28th ACM International Conference on Multimedia*, 55–63.

Ye, F.; and Tang, H. 2019. Skeleton-based action recognition with JRR-GCN. *Electronics Letters*, 55(17): 933–935.

Ye, F.; Tang, H.; Wang, X.; and Liang, X. 2019. Joints relation inference network for skeleton-based action recognition. In *2019 IEEE International Conference on Image Processing*, 16–20. IEEE.

Yun, K.; Honorio, J.; Chattopadhyay, D.; Berg, T. L.; and Samaras, D. 2012. Two-person interaction detection using body-pose features and multiple instance learning. In *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 28–35. IEEE.

Yun, S.; Han, D.; Oh, S. J.; Chun, S.; Choe, J.; and Yoo, Y. 2019. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE international conference on computer vision*, 6023–6032.

Zaremba, W.; Sutskever, I.; and Vinyals, O. 2014. Recurrent neural network regularization. *arXiv preprint arXiv:1409.2329*.

Zhang, H.; Cisse, M.; Dauphin, Y. N.; and Lopez-Paz, D. 2018. mixup: Beyond Empirical Risk Minimization. In *International Conference on Learning Representations*.

Zhang, P.; Lan, C.; Xing, J.; Zeng, W.; Xue, J.; and Zheng, N. 2019. View adaptive neural networks for high performance skeleton-based human action recognition. *IEEE transactions on pattern analysis and machine intelligence*, 41(8): 1963–1978.

Zhang, P.; Lan, C.; Zeng, W.; Xing, J.; Xue, J.; and Zheng, N. 2020. Semantics-guided neural networks for efficient skeleton-based human action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 1112–1121.

Zhu, W.; Lan, C.; Xing, J.; Zeng, W.; Li, Y.; Shen, L.; and Xie, X. 2016. Co-occurrence feature learning for skeleton based action recognition using regularized deep LSTM networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 30.