Group Graph Convolutional Networks for 3D Human Pose Estimation

Zijian Zhang
zhangzj2015@bupt.edu.cn

Beijing University of Posts and Telecommunications
Beijing, China

Abstract

In skeleton-based 3D human pose estimation (HPE), graph convolutional networks (GCNs) have recently achieved encouraging performance. However, most previous GCNs are limited by coupling aggregation mechanism. To address this limitation, we introduce the decoupling aggregation mechanism in CNNs to GCNs and propose group graph convolutional networks (GroupGCN). It consists of two main components: group convolution and group interaction. Group convolution ensures that every group has its own spatial aggregation kernel: the adjacent matrix. Group interaction ensures that the features interact between groups. We consider four different forms of group interaction and four different types of spatial aggregation kernels, aiming to conduct a comprehensive and systematic study of decoupling aggregation mechanism in GCNs. The proposed approach achieves the state-of-the-art performance while using 70% fewer parameters.

1 Introduction

3D human pose estimation (HPE) aims to regress the 3D positions of body joints in the camera coordinate system from images or videos. It plays an essential role in numerous applications such as action recognition, computer animation, and human-computer interaction. Estimating 3D pose from 2D pose is an ill-posed problem and extremely challenging because of depth ambiguity and occluded joints.

In 2D-to-3D human pose lifting task, previous works treat human joints as a feature vector and use FCNs to model the relationship between joints [19, 22, 23, 27]. For example, Martinez *et al*. [19] construct a simple but effective fully connected network to regress 3D pose from 2D pose and yield promising 3D HPE performance. Both 2D and 3D pose can be naturally represented by a skeleton graph in the form of 2D or 3D joint coordinates, so that graph convolutional networks (GCNs) have been applied for 3D HPE recently [26, 28, 29, 33, 34]. Compared with FCN-based approaches, GCN-based approaches not only learn a compact representation defined on graph nodes but also explicitly capture their structural relationships.

Spatial aggregation is the key component of GCN. Most prior GCN approaches share a spatial aggregation kernel for all channels, which is called coupling GCN [4]. It is not an optimal choice since different channels represent different types of motion features and relationships between joints are not always the same, which limits the flexibility of feature

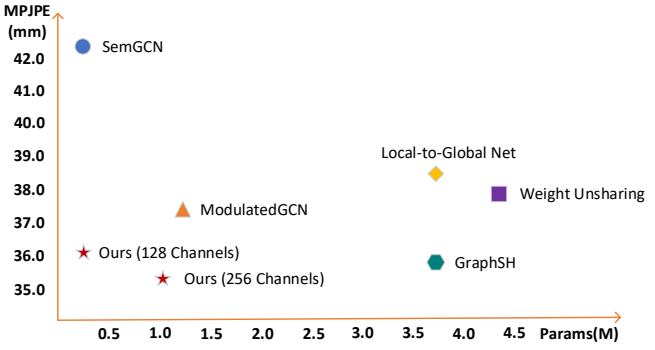

Figure 1: Comparison of the performance and model size between the proposed GroupGCN and state-of-the-art GCNs designed for 3D HPE, i.e., SemGCN [32], Local-to-Global Net [17], Weight Unsharing [16], Modulated GCN [34] and GraphSH [26]. A lower MPJPE value indicates better performance. All methods are evaluated on Human3.6M [11] with ground truth 2D joints as input.

extraction [3]. In contrast, every channel has an independent spatial aggregation kernel in CNNs, capturing different spatial information in different orientations, color, and frequencies. However, if we set channel-wise spatial aggregation kernels in GCNs, it makes the model too heavy and increases the difficulty of optimization.

In this paper, similar to group convolution in CNNs, we propose group graph convolutional networks (GroupGCN), a novel decoupling GCN for 3D HPE. It consists of group convolution and group interaction. Group convolution ensures that every group has its own spatial aggregation kernel and weight matrix. A drawback of group convolution is that the status of the other groups is completely unknown because they are independent. As a result, the set of independent group convolution may not be globally coherent, leading to poor performance. So we propose group interaction to account for global information by making the features interact between groups.

To addresses the dilemma between the accuracy and model complexity, we consider four different group interaction strategies: adding-interaction, connecting-interaction, shuffling-interaction, and interleaving-interaction. In addition, we consider four different spatial aggregation kernels: symmetric matrix, higher-order matrix, self-learning matrix, self-adaptive matrix. Specifically, we conduct a comprehensive and systematic study of the decoupling aggregation in GCNs for 3D HPE by controlling the number of channels, computational complexity and the number of groups.

In sum, the contribution of this paper can be summarized as follows:

- To our knowledge, we are the first to propose group graph convolutional network (GroupGCN) and have a comprehensive and systematic investigation of decoupling aggregation mechanism in GCNs.
- We make new conclusions that (1) decoupling aggregation can effectively improve the performance of graph convolution, (2) different group interaction strategies and spatial aggregation kernels have a significant impact on the performance of 3D HPE.
- The experimental results prove that GroupGCN can achieve state-of-the-art performance with fewer parameters, as shown in Fig 1.

2 Related Work

Group Convolution. AlexNet [13] is the first to use group convolution to handle the memory issue by distributing the model over two GPUs. The channel-wise separable convolutions proposed in Xception [6], is an extreme case of group convolutions. ShuffleNet [31] introduces channel shuffle operation to make feature exchange information between groups. IGCV [30] proposes the concept of interleaved group convolutions, which is efficient in parameter and computation. Our work generalizes group convolution to GCNs in a novel form.

Graph Convolutional Networks. GCNs generalize CNNs to inputs with graph structured. The principle of constructing GCNs on the graph can be divided into two streams: the spectral-based approaches [7, 14, 24] and the spatial-based approaches [1, 8, 12, 25]. Our approaches fall into the second stream. We briefly review the vanilla GCN as proposed in [12]. A graph based convolutional propagation contains two steps: XW and $\widetilde{A}X$. First, input features are transformed by a learnable weight matrix $W \in \mathbb{R}^{C\times C^{'}}$. Second, these transformed input features are gathered by a spatial aggregation kernel $\widetilde{A} \in \mathbb{R}^{N\times N}$. The convolution operation can be written as:

$$X^{'} = \sigma\left(\widetilde{A}(XW)\right) \tag{1}$$

Where σ is the activation function, i.e., ReLU [21]. $X \in \mathbb{R}^{N\times C}$ and $X^{'} \in \mathbb{R}^{N\times C^{'}}$ are the collection of features of all nodes before and after the convolution respectively. $A \in [0,1]^{N\times N}$ is the adjacent matrix of $\mathcal{G}$. If the joint j is depending on the joint i, then $a_{ij} = 1$. $\widetilde{A}$ is symmetrically normalized from A.

Lifting based 3D human pose estimation. Early attempts [19] simply use fully-connected networks (FCNs) to lift 2D keypoints into 3D space. Then some works [16, 26, 32, 34] utilize graph convolutional networks (GCNs) to learn a compact representation defined on graph nodes and explicitly capture their structural relationships. Zhao *et al.* [32] propose a semantic GCN by multiplying a learnable mask to the skeleton-based affinity matrix. Liu *et al.* [16] have a comprehensive investigation of weight sharing in a GCN.

3 Our Approach

We first introduce group convolution in Sec. 3.1. Then, group interaction is introduced in Sec. 3.2. Finally, we present the network architecture in Sec. 3.3.

3.1 Group convolution

For clarity, we first compare the convolution kernel in CNNs and GCNs. The size of convolution kernel is expressed as:

$$CNNs : d \times d \times C^{'} \times C \tag{2}$$

$$GCNs : n \times n + C^{'} \times C \tag{3}$$

In CNNs, "$\times$" means that every channel has an independent convolutional kernel, i.e., Eqs. (2). However, "$+$" means that the spatial aggregation kernel is shared by all channels in GCNs, i.e., Eqs. (3). Coupling aggregation forces GCN to aggregate features with the same topology in different channels.

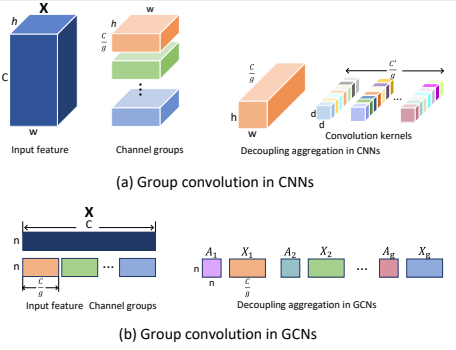

Figure 2: Illustration of group convolution. (a) Group convolution employ channel grouping and decoupling aggregation in CNNs. (b) We introduce the group convolution into GCNs and propose GroupGCN.

To resolve this issue, we introduce group convolution to GCNs. As shown in Fig. 2, similar to the group convolution in CNNs, we also divide the channels into g groups. Every group has an independent adjacent matrix in GroupGCN. Channels in a group share one adjacent matrix to reduce the redundancy of the adjacent matrices. In our experiments, $2 \sim 16$ groups are enough.

Inspired by [13], we consider four different adjacent matrices to learn the relationship beyond the natural connections of body joints.

- Symmetric matrix $\widetilde{A}_s$, which encodes the human skeleton symmetrical structure for joints that have a symmetrical counterpart, i.e. leg joints. ρ is Softmax nonlinearity, $M_s \in \mathbb{R}^{N\times N}$ is a learnable mask matrix, and $\odot$ is an element-wise mutlipication operation. Formally, the $\widetilde{A}_s$ is defined as
$$\widetilde{A}_s = \rho\left(M_s \odot A_s\right) \tag{4}$$

- High-order matrix $\widetilde{A}_k$, which explicitly encodes first-order and second-order kinematic connections for joints, i.e. shoulder-elbow, shoulder-wrist. $M_k \in \mathbb{R}^{N\times N}$ is a learnable mask matrix. Formally,
$$\widetilde{A}_k = \rho\left(M_k \odot A_k\right) \tag{5}$$

- Self-learning matrix A_c, which can create new connections and the existence and strength of connections are updated during the training process. $C \in \mathbb{R}^{N\times N}$ is a learnable matrix. Formally,
$$A_c = A + C \tag{6}$$

- Self-adaptive matrix A_b, which expresses a data-dependent matrix to determine whether a connection exists between nodes and how strong the connection is. Given two node features x_i and x_j, we first use two embedding functions θ and ϕ to reduce feature dimension before sending input features into correlation modeling function $\mathcal{M}(\cdot)$. $[\cdot\|\cdot]$ denotes concatenation and σ is the LeakyReLU nonlinearity with negative input slope $\alpha = 0.2$. The operation is formulated as
$$\alpha_{ij} = \frac{e^{\sigma\left(\mathcal{M}\left([\theta(x_i)\|\phi(x_j)]\right)\right)}}{\sum_{k=1}^{N} e^{\sigma(\mathcal{M}([\theta(x_i)\|\phi(x_k)]))}} \tag{7}$$

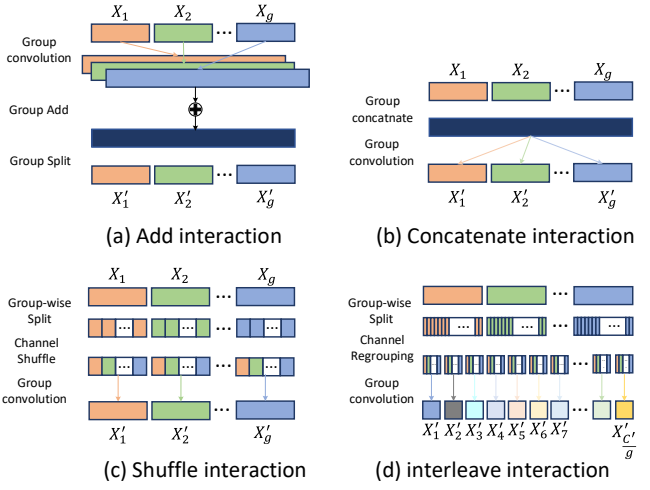

Figure 3: Illustration of four group interaction approaches.

3.2 Group interaction

The channel groups are independent of each other, i.e., Eqs. (3), hindering the exchange of information between different groups of different nodes and limiting the generalization ability of GroupGCN. To overcome the side effects brought by group convolutions, we introduce group interaction to account for the information of different groups. In addition, we consider four group interaction methods, illustrated in Fig. 3.

The four group interaction methods are easy to implement:

- Adding Interaction (AI). The results of individual convolution of each group are added together to realize the interaction of information between different groups. Formally,

$$X^{'} = \sigma\left(\sum_{i=1}^{g} A_i \left(X_i W_i\right)\right), W_i \in \mathbb{R}^{\frac{C}{g} \times C^{'}} \tag{8}$$

- Connecting Interaction (CI). First, we concatenate the channels of each group. Then we copy it g times for group convolution to generalize the features of each group independently. Formally,

$$X_g^{'} = \sigma\left(A_g \left(Concatenate\left(X_1, ..., X_g\right) W_g\right)\right), W_g \in \mathbb{R}^{C \times \frac{C^{'}}{g}} \tag{9}$$

- Shuffling Interaction (SI). This structure is similar to that used in ShuffleNet [31], which splits the channels in each group into g sub-groups and feed each group with different subgroups. For example, suppose a GroupGCN layer with g group whose input shape is (g,d), we first transpose it into (d,g), flattening and then reshape it back as the input shape. The channel shuffle operation is efficient and elegant.

- Interleaving Interaction (II). We introduce interleaved group convolutions proposed by IGCNets [30] into GCN. It can be simple and efficiently implemented by a channel permute operation which permutes input shape of the GroupGCN layer from (g,d) to

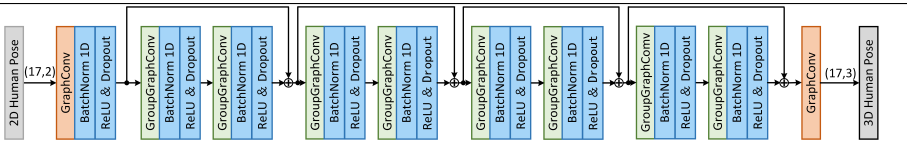

Figure 4: The network architectures of the proposed GroupGCN for 3D HPE.

(d, g). The number of channels group and the channels of each group are interchanged that means the channels in each group of output come from different groups of input.

3.3 Network Architecture

As illustrated in Fig. 4, we use the network architecture for 3D HPE and compare different decoupling aggregation approaches in our experiments. Following Martinez et al [19], we use two group graph convolutional layers as a building block with residual connections [10]. Each group graph convolutional layer is followed by a batch normalization and a ReLU [21] activation function except for the last one.

4 Experiments

4.1 Datasets and Evaluation Protocols

We perform our experiments on Human3.6M [11] and MPI-INF-3DHP [20] dataset and follow the standard evaluation procedure.

Dataset. The Human 3.6M dataset consists of 3.6 million video frames which are captured from 4 camera viewpoints at 50 Hz. There are 11 subjects and 15 daily activities like eating, discussion, sitting, greeting, walking and so on. For data preprocessing, we follow previous work [16, 18, 23, 34] to apply standard normalization to 2D and 3D keypoints for fair and effective comparison. We use five subjects (S1, S5, S6, S7, S8) for training and two subjects (S9, S11) for testing.

The MPI-INF-3DHP dataset is also a recently popular large-scale 3D human pose dataset. It contains both constrained indoor scenes and complex outdoor scenes, covering a greater diversity of poses and actions, where it is usually taken as a cross-dataset setting to verify the generalization ability of the proposed methods.

Evaluation protocols. It is commonly evaluated by two standard protocols: Protocol #1 and Protocol #2. Protocol #1 calculates the mean per joint positioning error (MPJPE) between the prediction and the ground truth after aligning the root joint. Protocol #2 employs a rigid alignment with the ground truth to relieve the inherent rotation, translation and scale problems before calculating the mean per joint positioning error (P-MPJPE).

4.2 Ablation Study

We conduct comprehensive ablation study to compare the different decoupling aggregation methods in controlled settings. Note that the 2D ground truth is taken as input for all model to avoid the influence of 2D human pose detector. Our model is implemented in Pytorch and all experiments are conducted on a single Nvidia RTX 2080Ti GPU.

Channels	Group Num	Params	MPJPE	P-MPJPE
128	1	0.27M	37.65	28.88
128	2	0.27M	37.34	28.63
128	4	0.28M	**37.01**	**28.33**
128	8	0.29M	37.56	28.94
256	1	1.06M	36.72	28.40
256	2	1.06M	36.02	27.85
256	4	1.07M	35.80	**27.53**
256	8	1.08M	**35.33**	27.68
256	16	1.10M	35.55	28.06

Table 1: Ablation study on Adding-interaction (AI).

Channels	Group Num	Params	MPJPE	P-MPJPE
128	1	0.27M	37.65	28.88
128	2	0.27M	37.64	29.40
128	4	0.28M	36.83	28.50
128	8	0.29M	**35.93**	**28.29**
256	1	1.06M	36.72	28.40
256	2	1.06M	36.14	27.89
256	4	1.07M	**35.28**	**27.36**
256	8	1.08M	36.45	**27.36**
256	16	1.10M	37.01	27.85

Table 2: Ablation study on Connecting-interaction (CI).

Implementation details. We set the initial learning rate 0.001, the decay factor 0.95 per epoch and adopt Adam [9] as optimization method. Following previous work [23], we initialize the weights in GroupGCN using the technique described in [9]. We use a batch size 256 and train each model for 50 epochs.

Effect of the GroupGCN. We compare the different group interaction methods *AI*,*CI*,*SI*,*II*, described in Section. 3.2. By default, decoupling graph convolution degenerates into coupling graph convolution ($g = 1$), which serves as baseline in our experiments. We fix the spatial aggregation kernel of each group to be self-learning matrix A_c for fair comparison. From Table 1 $\sim$ 4, we can draw the following conclusions:

- Compared to coupling graph convolution network ($g = 1$), group graph convolution network achieves higher performance. It is found that both *AI* and *CI* are superior to *SI* and *II*. *AI* and *CI* improve upon baseline by a large margin while only increase few parameters. Specifically, *AI* improves upon baseline by 1.39mm (MPJPE) and 0.87mm (P-MPJPE) when $C = 256$ and $g = 8$. *CI* improve upon baseline by 1.44mm (MPJPE) and 1.04mm (P-MPJPE) when $C = 256$ and $g = 4$. However, *SI* and *II* performs worse than baseline. *AI* and *CI* are used as our default setting in the following discussion.

- The number of groups has a significant impact on the performance of 3D HPE. We do not need to decouple the adjacent matrix of every channel. It is shown that the performance first improves with more groups. However, with more than 8 groups, the performance drops. 4$\sim$8 groups are enough for 3D HPE.

Effect of the spatial aggregation kernel. Then we study the impact of different spatial aggregation kernels on the 3D HPE performance. We fix the number of groups to be 4. We fix channels of each group graph convolutional layer to be 256. The result is shown in Table 5 and Table 6. A_c plays an important role in the performance of the GroupGCN because it allows the graph to include extra edges beyond the predefined. However, A_b perform worse than A_c indicates that too much freedom can lead to overfitting and harm the generalization ability of GroupGCN. $\widetilde{A}_s$, $\widetilde{A}_k$ and A_b have little effect on the performance of models.

Channels	Group Num	Params	MPJPE	P-MPJPE
128	1	0.27M	37.65	28.88
128	2	0.14M	38.37	29.51
256	1	1.06M	36.72	28.40
256	2	0.54M	37.18	28.64
256	4	0.28M	37.80	28.97
256	8	0.16M	39.38	31.08

Table 3: Ablation study on Shuffling-interaction (SI).

Channels	Group Num	Params	MPJPE	P-MPJPE
128	1	0.27M	37.65	28.88
128	2	0.13M	42.13	33.03
256	1	1.06M	36.72	28.40
256	2	0.43M	40.97	31.71
256	4	0.23M	42.11	32.23
256	8	0.14M	41.12	32.75

Table 4: Ablation study on Interleaving-interaction (II).

Methods	adjacent matrixs	Params	MPJPE	P-MPJPE
AI	$\widetilde{A}$,$\widetilde{A_k}$,A_c,A_b	1.13M	36.14	27.77
AI	$\widetilde{A_s}$,$\widetilde{A}$,A_c,A_b	1.13M	37.93	28.85
AI	$\widetilde{A_s}$,$\widetilde{A_k}$,$\widetilde{A}$,A_b	1.13M	36.21	27.87
AI	$\widetilde{A_s}$,$\widetilde{A_k}$,A_c,$\widetilde{A}$	1.06M	36.63	27.83
AI	$\widetilde{A_s}$,$\widetilde{A_k}$,A_c,A_b	1.13M	37.41	28.29
AI	A_c,A_c,A_c,A_c	1.07M	**35.80**	**27.53**
AI	A_b,A_b,A_b,A_b	1.33M	38.30	29.41

Table 5: Ablation study on different spatial aggregation kernels.

Methods	adjacent matrixs	Params	MPJPE	P-MPJPE
CI	$\widetilde{A}$,$\widetilde{A_k}$,A_c,A_b	2.12M	36.06	28.07
CI	$\widetilde{A_s}$,$\widetilde{A}$,A_c,A_b	2.12M	35.64	27.73
CI	$\widetilde{A_s}$,$\widetilde{A_k}$,$\widetilde{A}$,A_b	2.12M	36.72	28.40
CI	$\widetilde{A_s}$,$\widetilde{A_k}$,A_c,$\widetilde{A}$	1.06M	36.44	28.16
CI	$\widetilde{A_s}$,$\widetilde{A_k}$,A_c,A_b	2.12M	36.59	27.99
CI	A_c,A_c,A_c,A_c	1.07M	**35.28**	**27.36**
CI	A_b,A_b,A_b,A_b	1.33M	37.84	28.52

Table 6: Ablation study on different spatial aggregation kernels.

4.3 Comparison with the State-of-The-Art Methods

First, we compare the GroupGCN with some state-of-the-art methods on Human3.6M under both Protocol #1 and Protocol #2. We use two types of 2D joint detection data for evaluation: Cascaded Pyramid Network (CPN) [2] detections and ground truth 2D keypoints.

Table 7 and Table 8 show the results under two protocols respectively. Compared with other baselines, our methods achieve the state-of-the-art performance with either 2D detected or 2D ground truth as input. Note that our methods only have around one forth parameters 1.07M of [26] (3.70M).

Then, we compare the GroupGCN with some state-of-the-art methods on MPI-INF-3DHP, as shown in Table 9. Although we train the model using only the Human3.6M, GroupGCN outperforms others on MPI-INF-3DHP, indicating that our approach has strong generalization capabilities to unseen datasets.

4.4 Qualitative Results

Figure 5 and 6 shows that the performance of the proposed GroupGCN model on the Human3.6M dataset and in-the-wild images. It can accurately predict 3D poses of different persons for various actions. It indicating the effectiveness of our proposed approach in tackling the 2D-to-3D pose estimation problem.

5 Conclusions

In this paper, we propose GroupGCN and have a comprehensive and systematic study of decoupling aggregation mechanism in GCNs. With our unique group interaction methods, together with the group convolution strategy which has different types of spatial aggregation

Method		Direct	Discuss	Eat	Greet	Phone	Photo	Pose	Purcha.	Sit	SitD	Smoke	Wait	WalkD	Walk	WalkT	Avg.
Martinez et al. [19]	ICCV'17	51.8	56.2	58.1	59.0	69.5	78.4	55.2	58.1	74.0	94.6	62.3	59.1	65.1	49.5	52.4	62.9
Park et al. [20]	BMVC'18	49.4	54.3	51.6	55.0	61.0	73.3	53.7	50.0	68.5	88.7	58.6	56.8	57.8	46.2	48.6	58.6
Zhao et al. [27]†	CVPR'19	47.3	60.7	51.4	60.5	61.1	49.9	**47.3**	68.1	86.2	**55.0**	67.8	61.0	**42.1**	60.6	45.3	57.6
liu et al. [16]†	ECCV'20	46.3	52.2	47.3	50.7	55.5	67.1	49.2	**46.0**	60.4	71.1	51.5	50.1	54.5	40.3	43.7	52.4
Xu et al. [26]†	CVPR'21	45.2	49.9	47.5	50.9	54.9	66.1	48.5	46.3	**59.7**	71.5	51.4	48.6	53.9	39.9	44.1	51.9
Ours-AI †		45.4	51.4	49.8	50.3	55.0	**60.8**	47.9	48.4	61.0	70.7	52.7	48.9	55.2	40.1	41.9	52.0
Ours-CI †		**45.0**	**50.9**	49.0	**49.8**	**52.2**	60.9	49.1	46.8	61.2	70.2	51.8	**48.6**	54.6	**39.6**	**41.2**	**51.6**
Martinez et al. [19]	ICCV'17	37.7	44.4	40.3	42.1	48.2	54.9	44.4	42.1	54.6	58.0	45.1	46.4	47.6	36.4	40.4	45.5
Zhao et al. [27]†	CVPR'19	37.8	49.4	37.6	40.9	45.1	41.4	40.1	48.3	50.1	**42.2**	53.5	44.3	40.5	47.3	39.0	43.8
liu et al. [16]†	ECCV'20	36.8	40.3	33.0	36.3	37.5	45.0	39.7	34.9	40.3	47.7	37.4	38.5	38.6	29.6	32.0	37.8
Zeng et al. [28]	ECCV'20	35.9	36.7	29.3	34.5	36.0	42.8	37.7	**31.7**	40.1	44.3	35.8	37.2	36.2	33.7	34.0	36.4
Ci et al. [6]†	ICCV'19	36.3	38.8	29.7	37.8	**34.6**	42.5	39.8	32.5	36.2	39.5	34.4	38.4	38.2	31.3	34.2	36.3
Xu et al. [26]†	CVPR'21	35.8	38.1	31.0	35.3	35.8	43.2	**37.3**	31.7	38.4	45.5	35.4	36.7	36.8	27.9	30.7	35.8
Ours-AI †		**31.1**	37.3	**29.9**	**32.7**	35.0	40.5	38.3	32.7	39.4	48.4	33.6	37.0	35.7	**27.8**	29.5	35.3
Ours-CI †		32.5	**36.4**	30.7	33.2	34.9	**40.0**	37.8	33.1	**38.3**	47.8	**34.4**	**36.2**	**35.1**	28.4	**29.2**	**35.2**

Table 7: Comparison of single-frame 3D pose estimation in terms of MPJPE on Human3.6M. Works above the double line show results from detected 2D poses, and the below results from 2d groupd truth inputs to explore the upper bound of these methods. We highlight the graph-based methods by †. Best results in bold.

Method		Direct	Discuss	Eat	Greet	Phone	Photo	Pose	Purcha.	Sit	SitD	Smoke	Wait	WalkD	Walk	WalkT	Avg.
Martinez et al. [19]	ICCV'17	39.5	43.2	46.4	47.0	51.0	56.0	41.4	40.6	56.5	69.4	49.2	45.0	49.5	38.0	43.1	47.7
Park et al. [20]	BMVC'18	38.3	42.5	41.5	43.3	47.5	53.0	39.3	37.1	54.1	64.3	46.0	42.0	44.8	34.7	38.7	45.0
Ci et al. [6]†	ICCV'19	36.9	41.6	**38.0**	41.0	41.9	51.1	38.2	37.6	49.1	62.1	43.1	39.9	43.5	32.2	37.0	42.2
liu et al. [16]†	ECCV'20	35.9	40.0	38.0	41.5	42.5	51.4	37.8	36.0	**48.6**	56.6	41.8	38.3	42.7	31.7	36.2	41.2
Ours-AI †		35.7	39.7	38.5	40.9	41.6	46.1	**36.3**	36.5	49.1	55.2	41.6	36.6	43.5	31.2	34.4	40.49
Ours-CI †		**35.3**	**39.3**	38.4	**40.8**	**41.4**	**45.7**	36.9	**35.1**	48.9	**55.2**	**41.2**	**36.3**	**42.6**	**30.9**	**33.7**	**40.14**
Martinez et al. [19]	ICCV'17	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	35.25
Zhao et al. [27]†	CVPR'19	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	31.46
liu et al. [16]†	ECCV'20	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	30.09
Zeng et al. [28]	ECCV'20	26.0	28.9	**23.7**	26.9	27.4	33.1	**27.9**	25.0	32.4	40.9	28.8	29.2	29.3	23.3	24.5	28.5
Ours-AI †		**23.4**	29.1	24.7	26.2	26.3	31.1	28.8	**24.3**	**30.7**	37.7	**27.2**	28.6	29.1	23.4	**24.0**	27.6
Ours-CI †		23.4	**27.9**	24.9	**26.0**	**26.0**	**30.1**	28.3	24.7	31.0	**37.3**	27.2	**27.8**	**28.6**	**23.1**	26.0	**27.3**

Table 8: Comparison of single-frame 3D pose estimation in terms of P-MPJPE on Human3.6M. Works above the double line show results from detected 2D poses, and the below results from 2d groupd truth inputs to explore the upper bound of these methods. We highlight the graph-based methods by †. Best results in bold.

		GS	noGS	Outdoor	All (PCK)	All (AUC)
Martinez et al. [19]	ICCV'17	49.8	42.5	31.2	42.5	17.0
Ci et al. [6]	ICCV'19	74.8	70.8	77.3	74.0	36.7
Zeng et al. [27]	ECCV'20	-	-	80.3	77.6	43.8
liu et al. [16]	ECCV'20	77.6	80.5	80.1	79.3	47.6
Xu et al. [26]	CVPR'21	**81.5**	81.7	75.2	80.1	45.8
Zeng [28]	ICCV'21	-	-	**84.6**	**82.1**	46.2
Ours-AI †		81.1	84.0	77.6	81.3	49.7
Ours-CI †		80.4	**84.5**	77.2	81.1	**49.9**

Table 9: Results on the MPI-INF-3DHP test set.

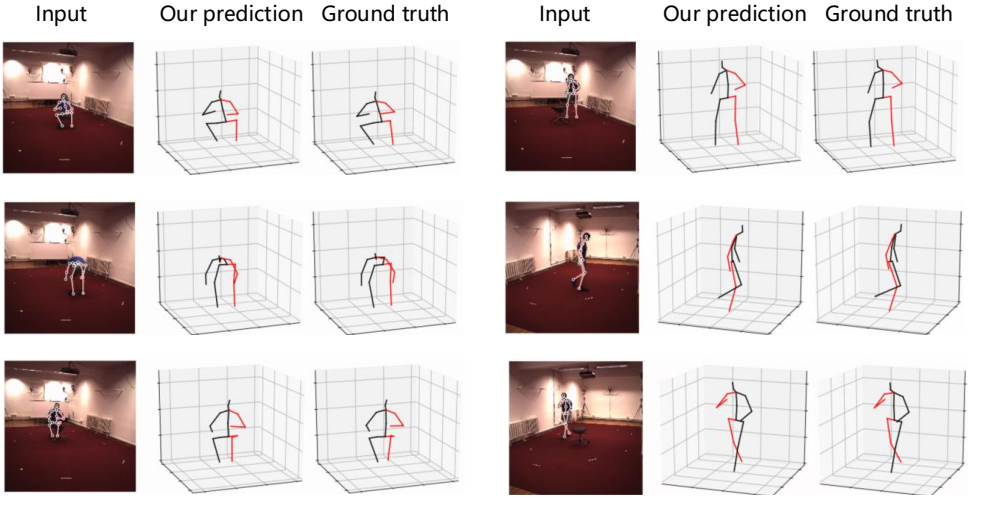

Figure 5: Qualitative results of our method on Human3.6M [].

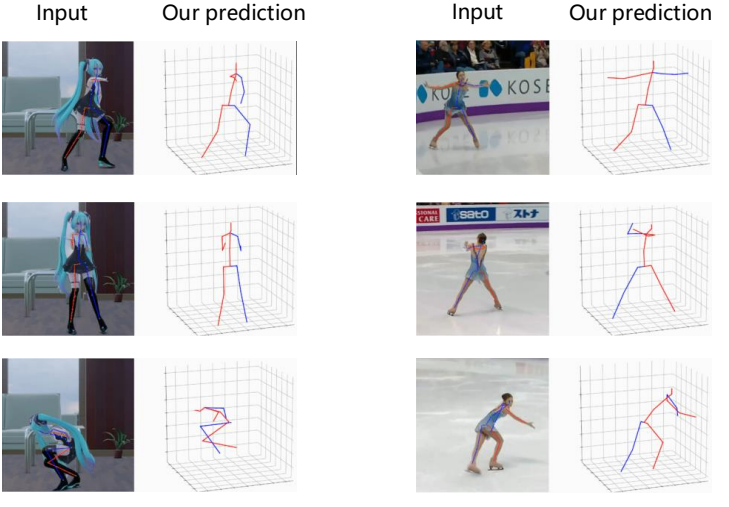

Figure 6: Qualitative results of our method on in-the-wild images.

kernels, our methods achieve accurate 2D-to-3D human pose estimation outperforming the start-of-the-art. We hope that our methods would inspire the field of skeleton-based 3D HPE.

References

[1] Peter W Battaglia, Jessica B Hamrick, Victor Bapst, Alvaro Sanchez-Gonzalez, Vinicius Zambaldi, Mateusz Malinowski, Andrea Tacchetti, David Raposo, Adam Santoro, Ryan Faulkner, et al. Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261*, 2018.

[2] Yilun Chen, Zhicheng Wang, Yuxiang Peng, Zhiqiang Zhang, Gang Yu, and Jian Sun. Cascaded pyramid network for multi-person pose estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7103–7112, 2018.

[3] Yuxin Chen, Ziqi Zhang, Chunfeng Yuan, Bing Li, Ying Deng, and Weiming Hu. Channel-wise topology refinement graph convolution for skeleton-based action recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13359–13368, 2021.

[4] Ke Cheng, Yifan Zhang, Congqi Cao, Lei Shi, Jian Cheng, and Hanqing Lu. Decoupling gcn with dropgraph module for skeleton-based action recognition. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIV 16*, pages 536–553. Springer, 2020.

[5] François Chollet. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1251–1258, 2017.

[6] Hai Ci, Chunyu Wang, Xiaoxuan Ma, and Yizhou Wang. Optimizing network structure for 3d human pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2262–2271, 2019.

[7] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. *Advances in neural information processing systems*, 29:3844–3852, 2016.

[8] Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. In *International conference on machine learning*, pages 1263–1272. PMLR, 2017.

[9] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256. JMLR Workshop and Conference Proceedings, 2010.

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[11] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3. 6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE transactions on pattern analysis and machine intelligence*, 36(7): 1325–1339, 2013.

[12] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

[13] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25:1097–1105, 2012.

[14] Ron Levie, Federico Monti, Xavier Bresson, and Michael M Bronstein. Cayleynets: Graph convolutional neural networks with complex rational spectral filters. *IEEE Transactions on Signal Processing*, 67(1):97–109, 2018.

[15] Junfa Liu, Juan Rojas, Yihui Li, Zhijun Liang, Yisheng Guan, Ning Xi, and Haifei Zhu. A graph attention spatio-temporal convolutional network for 3d human pose estimation in video. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3374–3380. IEEE, 2021.

[16] Kenkun Liu, Rongqi Ding, Zhiming Zou, Le Wang, and Wei Tang. A comprehensive study of weight sharing in graph networks for 3d human pose estimation. In *European Conference on Computer Vision*, pages 318–334. Springer, 2020.

[17] Kenkun Liu, Zhiming Zou, and Wei Tang. Learning global pose features in graph convolutional networks for 3d human pose estimation. In *Proceedings of the Asian Conference on Computer Vision*, 2020.

[18] Ruixu Liu, Ju Shen, He Wang, Chen Chen, Sen-ching Cheung, and Vijayan Asari. Attention mechanism exploits temporal contexts: Real-time 3d human pose reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5064–5073, 2020.

[19] Julieta Martinez, Rayat Hossain, Javier Romero, and James J Little. A simple yet effective baseline for 3d human pose estimation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2640–2649, 2017.

[20] Dushyant Mehta, Helge Rhodin, Dan Casas, Pascal Fua, Oleksandr Sotnychenko, Weipeng Xu, and Christian Theobalt. Monocular 3d human pose estimation in the wild using improved cnn supervision. In *2017 international conference on 3D vision (3DV)*, pages 506–516. IEEE, 2017.

[21] Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *Icml*, 2010.

[22] Sungheon Park and Nojun Kwak. 3d human pose estimation with relational networks. *arXiv preprint arXiv:1805.08961*, 2018.

[23] Dario Pavllo, Christoph Feichtenhofer, David Grangier, and Michael Auli. 3d human pose estimation in video with temporal convolutions and semi-supervised training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7753–7762, 2019.

[24] David I Shuman, Sunil K Narang, Pascal Frossard, Antonio Ortega, and Pierre Vandergheynst. The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. *IEEE signal processing magazine*, 30(3):83–98, 2013.

[25] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017.

[26] Tianhan Xu and Wataru Takano. Graph stacked hourglass networks for 3d human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16105–16114, 2021.

[27] Ailing Zeng, Xiao Sun, Fuyang Huang, Minhao Liu, Qiang Xu, and Stephen Lin. Srnet: Improving generalization in 3d human pose estimation with a split-and-recombine approach. In *European Conference on Computer Vision*, pages 507–523. Springer, 2020.

[28] Ailing Zeng, Xiao Sun, Lei Yang, Nanxuan Zhao, Minhao Liu, and Qiang Xu. Learning skeletal graph neural networks for hard 3d pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11436–11445, 2021.

[29] Junhao Zhang, Yali Wang, Zhipeng Zhou, Tianyu Luan, Zhe Wang, and Yu Qiao. Learning dynamical human-joint affinity for 3d pose estimation in videos. *IEEE Transactions on Image Processing*, 30:7914–7925, 2021.

[30] Ting Zhang, Guo-Jun Qi, Bin Xiao, and Jingdong Wang. Interleaved group convolutions. In *Proceedings of the IEEE international conference on computer vision*, pages 4373–4382, 2017.

[31] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, and Jian Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6848–6856, 2018.

[32] Long Zhao, Xi Peng, Yu Tian, Mubbasir Kapadia, and Dimitris N Metaxas. Semantic graph convolutional networks for 3d human pose regression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3425–3435, 2019.

[33] Jie Zhou, Ganqu Cui, Shengding Hu, Zhengyan Zhang, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun. Graph neural networks: A review of methods and applications. *AI Open*, 1:57–81, 2020.

[34] Zhiming Zou and Wei Tang. Modulated graph convolutional network for 3d human pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11477–11487, 2021.