Supplementary Materials of
Hierarchically Decomposed GCNs for Skeleton-Based Action Recognition

1. Additional Details of HD-Graph

Universality of HD-Graph. It is easier to construct our HD-Graph than existing handcrafted graph [8] even if ours is composed of more edges than the existing one. [8]'s graph requires every physically adjacent edges for human joints as shown in Algorithm 1. On the other hand, our HD-Graph requires only the hierarchy-wise node sets as shown in Algorithm 2. It verifies that our HD-Graph is more universal than the existing graph in that the requirements of the HD-Graph are fewer than those of the existing one.

Algorithm 1: Physically Adjacent Graph

Input: Physically adjacent inward edge set
$\mathcal{E} = \{e_1, e_2, e_3, \cdots, e_{N_\mathcal{E}}\}$
NTU-RGB+D : $\mathcal{E} =$
$\{(1,2),(2,21),(3,21),(4,3),(5,21),(6,5),(7,6),$
$(8,7),(9,21),(10,9),(11,10),(12,11),(13,1),$
$(14,13),(15,14),(16,15),(17,1),(18,17),(19,18),$
$(20,19),(22,23),(23,8),(24,25),(25,12)\}$
Initialize Adjacency matrix $\mathbf{A} \in \mathbb{R}^{3\times N\times N}$ to $\mathbf{0}$
Assign value of 1 to all diagonal components of $\mathbf{A}^{id}$ to get identity nodes.
for e *to* $\mathcal{E}$ **do**
  Centripetal edges: $\mathbf{A}^{cp}[e] \leftarrow 1$
  Centrifugal edges: $\mathbf{A}^{cf}[reverse(e)] \leftarrow 1$
Initialize degree matrix $\mathbf{\Lambda} \in \mathbb{R}^{3\times N\times N}$ to $\mathbf{0}$
for $n = 1$ *to* N **do**
  $\mathbf{\Lambda}[n,n] \leftarrow$ the number of non-zero elements in column n of $\mathbf{A}$
Normalize adj. matrix with degree matrix:
$\mathbf{A} \leftarrow \mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{A}\mathbf{\Lambda}^{-\frac{1}{2}}$
return $\mathbf{A}$

Algorithm 2: Hierarchically Decomposed Graph

Input: Hierarchy-wise node sets
$\mathbf{H} = \{H_1, H_2, \cdots, H_L\}$
NTU-RGB+D :
$H_1 = \{2\}$,
$H_2 = \{1, 21\}$,
$H_3 = \{13, 17, 3, 5, 9\}$,
$H_4 = \{14, 18, 4, 6, 10\}$,
$H_5 = \{15, 19, 7, 11\}$,
$H_6 = \{16, 20, 8, 12\}$,
$H_7 = \{22, 23, 24, 25\}$
Initialize Adjacency matrix $\mathbf{A} \in \mathbb{R}^{(L-1)\times 3\times N\times N}$ to $\mathbf{0}$
for $l = 1$ *to* $L-1$ **do**
  For H_l and H_{l+1}, include all nodes of those subsets in the diagonal components of the adjacency matrix to get identity nodes: $\mathbf{A}_l^{id}[H_l, H_l] \leftarrow 1$, $\mathbf{A}_l^{id}[H_{l+1}, H_{l+1}] \leftarrow 1$
  for $i = 1$ *to* $length(H_l)$ **do**
    for $j = 1$ *to* $length(H_{l+1})$ **do**
      Centripetal edges: $\mathbf{A}_l^{cp}[H_{l+1}(j), H_l(i)] \leftarrow 1$
      Centrifugal edges: $\mathbf{A}_l^{cf}[H_l(i), H_{l+1}(j)] \leftarrow 1$
  Initialize degree matrix $\mathbf{\Lambda}_l \in \mathbb{R}^{3\times N\times N}$ to $\mathbf{0}$
  for $n = 1$ *to* N **do**
    $\mathbf{\Lambda}_l[n,n] \leftarrow$ the number of non-zero elements in column n of $\mathbf{A}_l$
  Normalize adj. matrix with degree matrix:
  $\mathbf{A}_l \leftarrow \mathbf{\Lambda}_l^{-\frac{1}{2}}\mathbf{A}_l\mathbf{\Lambda}_l^{-\frac{1}{2}}$
return $\mathbf{A}$

Tree Structures for HD-Graph. Tree structures for skeletal modality has been already proposed in [6], which applies depth first search (DFS) algorithm to identify the kinematic dependency relations between the joints. It traverses every joint nodes from the root node to the leaf nodes to model the spatial dependency of the joints. Nevertheless, as [6]'s tree identifies only the adjacent connections of human joints, it cannot discover direct relationships between structurally distant nodes. Moreover, it is dependent to much on the fixed joint visiting order, which makes the model reflect only the topologically fixed edge features. On the other hand, our HD-Graph is free from those drawbacks. Although we also uses the tree structure to construct the HD-Graph, direct relationships of the structurally distant edges are identified by connecting every nodes for adjacent hierarchy node sets. In addition, because there is no fixed node visiting order in the process of constructing HD-Graph, HD-GCN leverages various edge features via FC-edges and adaptively highlights significant edge sets by A-

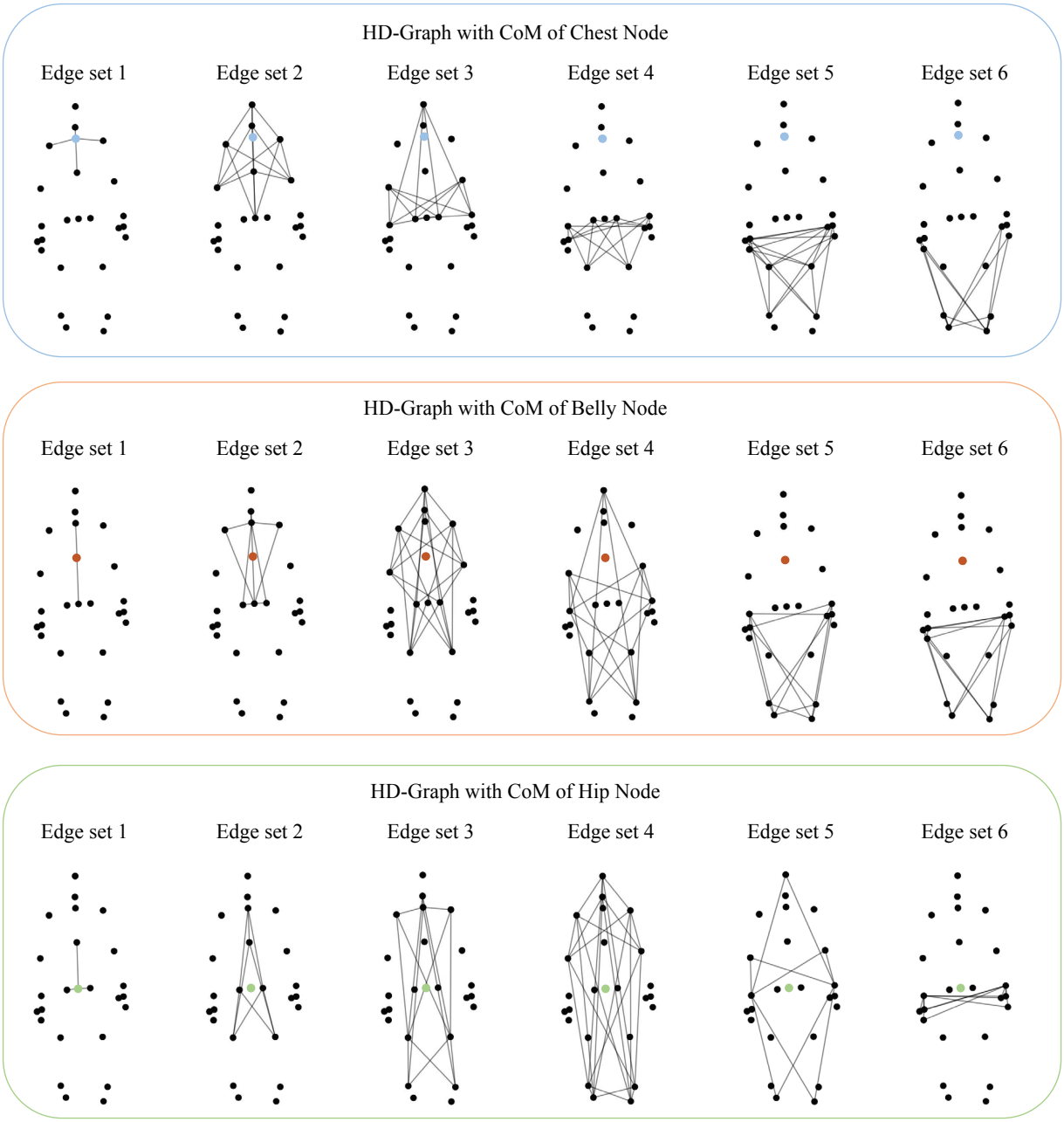


Figure 1. Different HD-Graphs are composed of different edge sets. Colored nodes denote CoM nodes of the graphs.

HA module.

2. Effectiveness of Six-way Ensemble

As we mentioned in our main paper, we propose the ensemble method with joint and bone streams without motion streams. Model with each stream is trained with three different HD-Graphs, which have different CoM nodes; chest, belly, and hip. In other words, training ways for our ensemble methods are as follows: (1) joint stream with CoM of chest node, (2) bone stream with CoM of chest node, (3) joint stream with CoM of belly node, (4) bone stream with CoM of belly node, (5) joint stream with CoM of hip node, (6) bone stream with CoM of hip node. As shown in Fig. 1, corresponding edge sets for all graphs are different from each others. We compare the performance of those three graphs for several labels on NTU-RGB+D 120 joint dataset

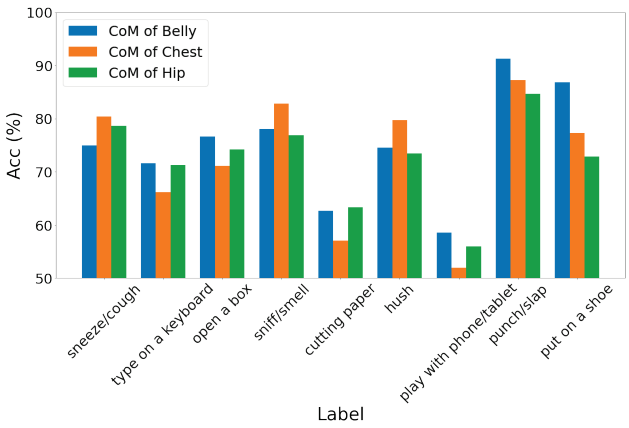


Figure 2. Classification accuracy for several classes with different HD-Graphs.

CoM	Stream	NTU-RGB+D 60 X-Sub (%)	NTU-RGB+D 60 X-View (%)	NTU-RGB+D 120 X-Sub (%)	NTU-RGB+D 120 X-Set (%)
Chest	Joint	90.4	95.3	85.2	87.0
	Bone	90.7	95.2	86.7	88.1
Belly	Joint	90.5	95.6	85.7	87.3
	Bone	90.8	95.0	86.3	88.4
Hip	Joint	90.6	95.7	85.2	87.2
	Bone	90.9	95.1	86.4	88.4
Ensemble		**93.4**	**97.2**	**90.1**	**91.6**

Table 1. Experimental results according to CoMs and data streams.

as shown in Fig. 2. The differences between the maximum and minimum accuracy for those three graphs range from 4% to 13%. It heuristically proves that the models trained on each of the three graphs have different learning patterns.

Ensemble Coefficients. For most recent skeleton-based action recognition models [1, 2, 4], optimal coefficients for their ensemble methods should be chosen, which have different values depending on their model. For example, [1] suggests ensemble coefficients of [1.0, 1.0, 0.6, 0.6], which represent joint, bone, joint motion, bone motion streams. Moreover, [2] presents [0.7, 0.7, 0.3, 0.3] and [3] suggests [0.6, 0.6, 0.4, 0.4]. It reduces the universality of the model in that the coefficients should be manually determined. However, there is no need to set those coefficients for six-way ensemble because our method does not require motion streams. Instead of using low-performance motion streams, we use only joint and bone streams and apply the ensemble to all six models with equal contribution. In other words, our ensemble method does not require any ensemble coefficients that determine how much each stream contributes to the model. Applying our ensemble method, our HD-GCN outperforms state-of-the-art methods without the motion streams and manually fixed ensemble coefficients.

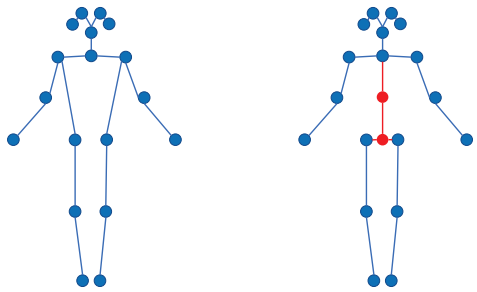
	X-Sub (%)	X-Set (%)	GFLOPs	# Param. (M)
DC-GCN [2]	84.0⋆	86.1⋆	2.74	3.45
MS-G3D [7]	84.9⋆	86.8⋆	5.22	3.22
CTR-GCN [1]	84.9	86.5⋆	1.97	**1.46**
InfoGCN [4]	85.1	86.3	1.68	1.57
HD-GCN	**85.7**	**87.3**	**1.60**	1.68

Table 2. **Comparison of computational and model complexity of the state-of-the-arts.** Each experiment is based on NTU-RGB+D 120 joint stream dataset. ⋆ results are implemented based on the released codes.

Additional Experimental Results. Tab. 1 shows every single experimental result for our six-way ensemble method. Comparing the results of ours in Tab. 1 and other models shown in Tab. 2, it shows that our model outperforms the others even on single-stream experiments by a large margin.

3. Architectures for Kinetics-Skeleton

We modify the original graph of Kinetics-Skeleton [5] to apply our HD-Graph. The original architecture of the dataset contains 18 nodes, which does not have hip and belly nodes for CoM, so we manually set those nodes by using existing nodes. Firstly, we set the CoM hip node, which is the middle point of left and right hip nodes. In addition, the belly CoM node is the middle point of chest and hip nodes. The modified skeleton architecture contains 20 nodes due to the generated CoM hip and belly nodes. The original and modified versions of the skeleton are shown in Fig. 3.



(a) Original Architecture (b) Modified Architecture

Figure 3. The original and modified version of the Kinetics-Skeleton architecture. The red lines and red circles denote newly generated edges and nodes, respectively.

References

[1] 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. 3

[2] Ke Cheng, Yifan Zhang, Congqi Cao, Lei Shi, Jian Cheng, and Hanqing Lu. Decoupling gcn with dropgraph module for skeleton-based action recognition. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 3

[3] Ke Cheng, Yifan Zhang, Xiangyu He, Weihan Chen, Jian Cheng, and Hanqing Lu. Skeleton-based action recognition with shift graph convolutional network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 183–192, 2020. 3

[4] Hyung-gun Chi, Myoung Hoon Ha, Seunggeun Chi, Sang Wan Lee, Qixing Huang, and Karthik Ramani. Infogcn: Representation learning for human skeleton-based action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20186–20196, 2022. 3

[5] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017. 3

[6] Jun Liu, Amir Shahroudy, Dong Xu, and Gang Wang. Spatio-temporal lstm with trust gates for 3d human action recognition. In *European conference on computer vision*, pages 816–833. Springer, 2016. 1

[7] Ziyu Liu, Hongwen Zhang, Zhenghao Chen, Zhiyong Wang, and Wanli Ouyang. Disentangling and unifying graph convolutions for skeleton-based action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 143–152, 2020. 3

[8] Sijie Yan, Yuanjun Xiong, and Dahua Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *Thirty-second AAAI Conference on Artificial Intelligence*, 2018. 1