Dynamic Multiscale Graph Neural Networks for 3D Skeleton-Based Human Motion Prediction - Supplementary Material

Maosen Li[1], Siheng Chen[2] ✉, Yangheng Zhao[1], Ya Zhang[1] ✉, Yanfeng Wang[1], and Qi Tian[3]

[1] Cooperative Medianet Innovation Center, Shanghai Jiao Tong University
[2] Mitsubishi Electric Research Laboratories
[3] Huawei Noah's Ark Lab

{maosen_li, zhaoyangheng-sjtu, ya_zhang, wangyanfeng} @sjtu.edu.cn, schen@merl.com, tian.qi1@huawei.com

1. Detailed Architecture

Here we show the detailed structure of the proposed DMGNN. We first show the structure of the encoder, including the single-scale graph convolution block (SS-GCB) and cross-scale fusion block (CS-FB). We then show the structure of the decoder, including the graph-based gated recurrent unit (G-GRU).

1.1. Encoder

Single-scale graph convolution block (SS-GCB). SS-GCB consists of a graph convolution and a temporal convolution. Table 1 presents the structures of four cascaded SS-GCB at scale s in the encoder of DMGNN. We see that we use four SS-GCBs to extract spatio-temporal motion features. In each SS-GCB, we employ ReLU, batch normalization, and dropout operations. We use stride 2 to downsample along the temporal dimension.

Table 1. The structure of four SS-GCBs at scale s in the encoder.

Idx	Shape & Operations	Feature	Remarks
1	$[32, 3, 1, 1] \times 2$ -bn-relu	$[32, 32, M_s, 49]$	graph conv
	$[32, 32, 5, 1]$, stride=1 bn-dropout-relu	$[32, 32, M_s, 49]$	temporal conv
2	$[64, 32, 1, 1] \times 2$ -bn-relu	$[32, 64, M_s, 49]$	graph conv
	$[64, 64, 5, 1]$, stride=2 bn-dropout-relu	$[32, 64, M_s, 25]$	temporal conv
3	$[128, 64, 1, 1] \times 2$ -bn-relu	$[32, 128, M_s, 25]$	graph conv
	$[128, 128, 5, 1]$, stride=2 bn-dropout-relu	$[32, 128, M_s, 13]$	temporal conv
4	$[256, 128, 1, 1] \times 2$ -bn-relu	$[32, 256, M_s, 13]$	graph conv
	$[256, 256, 5, 1]$, stride=2 bn-dropout-relu	$[32, 256, M_s, 7]$	temporal conv

Cross-scale fusion block (CS-FB) We use CS-FB to fuse multiscale features. Table 2 presents the structure of the first CS-FB to fuse the feature from s_1 to s_2. We first use a temporal convolution to shrink the temporal dimension and obtain a compact feature vector for each body-component; we then use four MLPs to learn the feature embeddings for two body-scales, respectively; we finally calculate the inner product of these two embeddings and employ a softmax to calculate the corresponding edge weight in a cross-scale graph.

Table 2. The structure of the first CS-FB from s_1 to s_2.

Step	Shape & Operations
1	temporal conv: $[32, 32, 5, 1]$, stride=2; vectorize
2	for both f_{s_1} and f_{s_2}: 800-256-relu -dropout-256-relu-bn; Sum
	for both g_{s_1} and g_{s_2}: 512-256-relu -dropout-256-relu-bn
3	Computing (2e) in paper

Total architecture In summary, we show the total architecture of the encoder, which combine SS-GCBs at multiple scales and CS-FB across scales. Table 3 presents the structure of the encoder. We see that we use four MGCUs, where the first two MGCUs use SS-GCBs and CS-FBs to learn the features from multiscale bodies and the last two MGCUs only use SS-GCB to extract features.

1.2. Decoder

Graph-based Gated Recurrent Unit (G-GRU) G-GRU is one of the key components in the proposed decoder for synthesizing precise and reasonable future poses. Table 4 presents the structure of the G-GRU at time stamp t.

Table 3. The structure of the encoder.

MGCU	Initialize three scales		
1	SS-GCB 1 at s_1	SS-GCB 1 at s_2	SS-GCB 1 at s_3
	CS-FB 1 between s_1&s_2 and s_2&s_3		
2	SS-GCB 2 at s_1	SS-GCB 2 at s_2	SS-GCB 2 at s_3
	CS-FB 2 between s_1&s_2 and s_2&s_3		
3	SS-GCB 3 at s_1	SS-GCB 3 at s_2	SS-GCB 3 at s_3
4	SS-GCB 4 at s_1	SS-GCB 4 at s_2	SS-GCB 4 at s_3
	Weighted sum		
	A final SS-GCB at s_1		
	Temporal average pooling		

We see that we take the historical motion state and the on-

Table 4. The structure of the G-GRU in the decoder at time t.

Variables	Operations
$\mathbf{r}^{(t)}$	input: $\mathbf{H}^{(t)}, \mathbf{I}^{(t)}$; r_{in}: $9 \to 256$ graph conv: $256 \to 256$; r_{hid}: $256 \to 256$ sum and sigmoid
$\mathbf{u}^{(t)}$	input: $\mathbf{H}^{(t)}, \mathbf{I}^{(t)}$; u_{in}: $9 \to 256$ graph conv: $256 \to 256$; u_{hid}: $256 \to 256$ sum and sigmoid
$\mathbf{c}^{(t)}$	input: $\mathbf{H}^{(t)}, \mathbf{I}^{(t)}$; c_{in}: $9 \to 256$ graph conv: $256 \to 256$; c_{hid}: $256 \to 256$ element-wise product of c_{hid} and $\mathbf{r}^{(t)}$ sum and tanh
$\mathbf{H}^{(t+1)}$	$\mathbf{u}^{(t)} \odot \mathbf{H}^{(t)} + (1 - \mathbf{u}^{(t)}) \odot \mathbf{c}^{(t)}$

line 3D skeleton-based information as inputs and introduce the graph convolution to propagate the motion information to produce the motion state at the next frame. The hidden dimension of the G-GRU is 256.

Total architecture Here, we show the total architecture of the decoder, which combines the proposed G-GRU and an MLP-formed output function. Table 5 presents the structure the decoder at time stamp t. We see that, given

Table 5. The structure of the decoder at time t.

	Operations
Inputs	$\mathbf{H}^{(t)}, \mathbf{I}^{(t)} = [\widehat{\mathbf{X}}^{(t)}, \Delta^1\widehat{\mathbf{X}}^{(t)}, \Delta^2\widehat{\mathbf{X}}^{(t)}]$
G-GRU	$\mathbf{H}^{(t+1)} = \text{G-GRU}(\mathbf{I}^{(t)}, \mathbf{H}^{(t)}), 9, 256 \to 256$
f_{pred}	$f_{\text{pred}}(\mathbf{H}^{(t+1)}), 256 \to 256 \to 3$
$\widehat{\mathbf{X}}^{(t+1)}$	$\widehat{\mathbf{X}}^{(t+1)} = \widehat{\mathbf{X}}^{(t)} + f_{\text{pred}}(\mathbf{H}^{(t+1)})$

the hidden motion state and current input information, we use a G-GRU and an MLP-formed output function f_{pred} to model the displacement of motions between two consecutive frames, and we emply residual connections to obtain the estimated poses. The hidden dimensions are 256.

2. Quantitative Comparison with more Baselines

In our paper submission, we only compare DMGNN to several state-of-the-art works, while many other methods has been developed. Here we compare DMGNN to as many previous methods as possible. Table 6 presents the MAE of many methods for short-term motion prediction on 4 representative actions of Human 3.6M We see that, the proposed DMGNN outperforms the state-of-the-art methods on most actions. Notably, we have cited all of baselines presented in Table 6 in our paper submission.

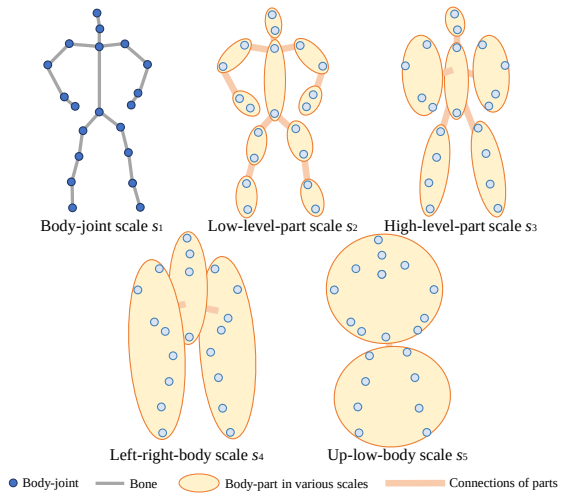

Figure 1. Three body scales on Human 3.6M. In body-joint scale, we consider 20 joints with non-zero exponential maps [7]; In s_4 and s_5, we consider 3 and 2 parts, respectively.

3. Coarser Body-scales in Ablation Studies

In the first experiment of ablation studies ('effects of multiple scales'), we initialize two coarser body-scales (s_4 and s_5) besides the effective three scales (s_1, s_2 and s_3) that used in our DMGNN. Here we present s_4 and s_5 in details.

To initialize s_4, we average the input features of three body-components: left-body, head-and-torso, and right-body as the nodes of corresponding body-graph. We build two initial edges to respectively connect head-and-torso with left-body and right-body. To initialize s_5, we average the input features of two body-components: upper-body and lower-body as the graph nodes. We build an edge between these two body-components. Figure 1 illustrates the two coarser body-scales as well as the body-joint scale on Human 3.6M [8]. We name s_4 as 'Left-right-body scale' and name s_5 as 'Up-low-body scale'.

4. Effects of Numbers and Positions of CS-FBs

In our DMGNN, we employ CS-FBs with aggregating relative features at different MGCUs to fuse various levels of motion features across different scales; see Equation (2a) in the submission. Here we further investigate the effects of numbers and positions of CS-FBs at cascaded MGCUs. In the four MGCUs, we use one to four CS-FBs at different

Table 6. Mean angle errors (MAE) of different methods for short-term prediction on 4 representative actions of H3.6M.

Motion	Walking				Eating				Smoking				Discussion			
milliseconds	80	160	320	400	80	160	320	400	80	160	320	400	80	160	320	400
ZeroV [14]	0.39	0.68	0.99	1.15	0.27	0.48	0.73	0.86	0.26	0.48	0.97	0.95	0.31	0.67	0.94	1.04
ERD [2]	0.93	1.18	1.59	1.78	1.27	1.45	1.66	1.80	1.66	1.95	2.35	2.42	0.31	0.67	0.94	1.04
LSTM-3R [2]	0.977	1.00	1.29	1.47	0.89	1.09	1.35	1.46	1.34	1.65	2.04	2.16	1.88	2.12	2.25	2.23
SRNN [9]	0.81	0.94	1.16	1.30	0.97	1.14	1.35	1.46	1.45	1.68	1.94	2.08	1.22	1.49	1.83	1.93
DropAE [3]	1.00	1.11	1.39	/	1.31	1.49	1.86	/	0.92	1.03	1.15	/	1.11	1.20	1.38	/
Res-sup. [14]	0.27	0.46	0.67	0.75	0.23	0.37	0.59	0.73	0.32	0.59	1.01	1.10	0.30	0.67	0.98	1.06
CSM [11]	0.33	0.54	0.68	0.73	0.22	0.36	0.58	0.71	0.26	0.49	0.96	0.92	0.32	0.67	0.94	1.01
TP-RNN [1]	0.25	0.41	0.58	0.65	0.20	0.33	0.53	0.67	0.26	0.47	0.88	0.90	0.30	0.66	0.96	1.04
QuaterNet [15]	0.21	0.34	0.56	0.62	0.20	0.35	0.58	0.70	0.25	0.47	0.93	0.90	0.26	0.60	0.85	0.93
AGED [5]	0.21	0.35	0.55	0.64	0.18	0.28	0.50	0.63	0.27	0.43	0.81	0.83	0.26	0.56	0.77	0.84
Skel-TNet [6]	0.31	0.50	0.69	0.76	0.20	0.31	0.53	0.69	0.25	0.50	0.93	0.89	0.30	0.64	0.89	0.98
BiHMP-GAN [10]	0.33	0.52	0.63	0.67	0.20	0.33	0.54	0.70	0.26	0.50	0.91	0.86	0.33	0.65	0.91	0.95
VGRU-r1 [4]	0.34	0.47	0.64	0.72	0.27	0.40	0.64	0.79	0.36	0.61	0.85	0.92	0.46	0.82	0.95	1.21
HMR [12]	0.23	0.35	0.56	0.65	0.21	0.32	0.55	0.67	0.26	0.47	0.90	0.89	0.29	0.55	0.83	0.94
Imit-L [16]	0.21	0.34	0.53	0.59	0.17	0.30	0.52	0.65	0.23	0.44	0.87	0.85	0.23	0.56	0.82	0.91
Traj-GCN [13]	**0.18**	0.32	**0.49**	**0.56**	**0.17**	0.31	0.52	0.62	0.22	0.41	0.84	0.79	**0.20**	**0.51**	**0.79**	**0.87**
DMGNN	**0.18**	**0.31**	**0.49**	0.58	**0.17**	**0.30**	**0.49**	**0.59**	**0.21**	**0.39**	**0.81**	**0.77**	0.26	0.65	0.92	0.99

MGCUs, and we obtain the average prediction MAEs of different model variants.

Table 7 presents the average MAEs of DMGNN with different numbers of CS-FBs at different MGCUs on H3.6M for short-term motion prediction. We also compare the performance of CS-FBs with or without aggregating relative information from all the body-components ('with relative' or 'without relative'). We denote the numbers of CS-FBs at the column 'Number' and denote the CS-FB positions as MGCU indices at column 'Position'. We see that 1) when we aggregate global relative information to in the CS-FB, we obtain lower MAEs than the module without relative information aggregation; 2) when we use two CS-FBs with relative information aggregation at the 1st and 2nd MGCUs, DMGNN produces the most precise predictions across different model variants; 3) fusing multiscale features at first few MGCUs outperforms fusing at last ones. The reason behind could be, if we use only one CS-FB, we cannot fuse rich features for comprehensive pattern learning; if we use too many CS-FBs, the capacity of the network become much larger, leading to overfitting.

Table 7. Average MAEs of DMGNN with different numbers of CS-FBs at different MGCUs on H3.6M across 400 ms.

Number	Position	MAE (without relative)	MAE (with relative)
1	1	0.621	0.621
	2	0.620	0.618
	3	0.620	0.616
	4	0.622	0.619
2	1,2	0.620	**0.613**
	1,3	0.619	0.614
	1,4	0.621	0.615
	2,3	0.622	0.616
	2,4	0.622	0.617
	3,4	0.625	0.620
3	1,2,3	0.622	0.616
	1,2,4	0.623	0.619
	1,3,4	0.624	0.622
	2,3,4	0.625	0.622
4	1,2,3,4	0.622	0.619
0	/	0.630	

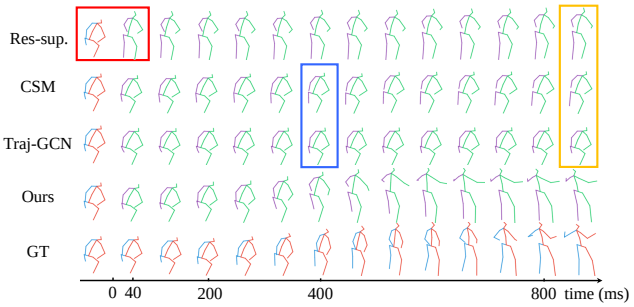

Figure 2. Predicted samples of the action of 'Posing' in Human 3.6M dataset from four models in a long term.

5. More Generated Motion Samples

To further demonstrate the effectiveness of the AS-GNN, we illustrate more predicted samples on both Human 3.6M [8] and CMU Mocap [1] dataset.

5.1. Human 3.6M Dataset

We first illustrate two generated motions of the actions of 'Posing' and 'Waiting' on Human 3.6 dataset (H3.6). We compare the DMGNN with three models: Res-sup. [14], CSM [11] and Traj-GCN [13].

Figure 2 illustrates the predicted poses of 'Posing' in Human 3.6M in 1000 ms. We see that the proposed DMGNN could well model the posture, such as stretched bodies and arms; however, Res-sup predicts the motion with large discontinuity between the last observed pose the first predicted

[1] http://mocap.cs.cmu.edu/

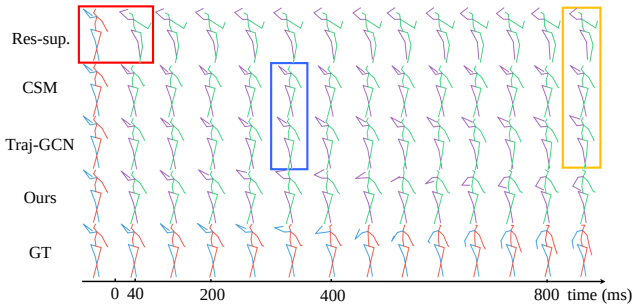

Figure 3. Predicted samples of the action of 'Waiting' in Human 3.6M dataset from four models in a long term.

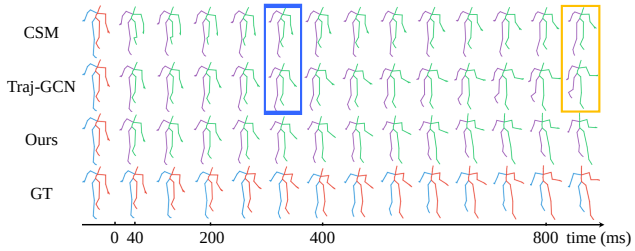

Figure 4. Predicted samples of the action of 'Basketball' in CMU Mocap dataset from three models in a long term.

one (red box); CSM and Traj-GCN tends to have large errors after the 400th ms (blue box); all the baselines produce unreasonable poses at the 1000th ms (yellow box), which are far from the ground truth.

We also predict the action of 'Waiting' in Human 3.6M in a long term with different methods. The results are illustrated in Figure 3. We see that, for baselines, the motion predicted by res-sup has large discontinuity between the last observed pose the first predicted one (red box) and loses the movements, which is far from the ground truths. CSM and Traj-GCN suffer from large errors after the 320th ms; all the baselines predict unreasonable poses at the 1000th ms (yellow box); but the predictions from DMGNN could complete the action reasonably.

5.2. CMU Mocap Dataset

We then test DMGNN on the two actions of 'Basketball' and 'Washing window' in CMU Mocap dataset. The baselines are the CSM [11] and Traj-GCN [13].

For the action of 'Basketball', the main challenge of motion prediction is the running legs and swaying arms. We illustrate the generated samples of three models in Figure 5. We see that the errors of the predictions from CSM and Traj-GCN rise after the 320th ms (blue box); two baselines give unreasonable postures at the 1000th ms in long-term (yellow box); that is, CSM has wrong tilt orientation of the body and the left leg (purple) of the pose predicted by Traj-GCN has inaccurate position; DMGNN could predict motions with smaller errors in both short-term and long-term.

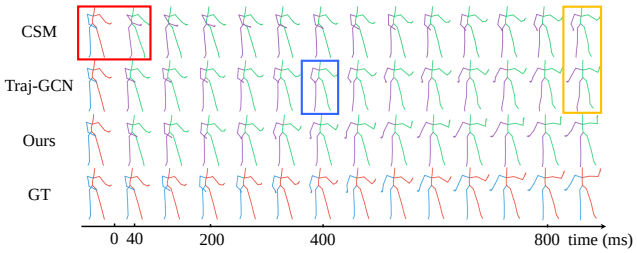

Figure 5. Predicted samples of the action of 'Washing window' in CMU Mocap dataset from three models in a long term.

For the action of 'Washing window', we also predict the future poses in 1000 ms and illustrate them in Figure 5. We see that the prediction of CSM has large discontinuity between the last observed pose the first predicted one (red box); Traj-GCN tends to have large errors after the 400th ms, since the pose does not raise the left arm (blue box); two baselines give poses at the 1000th ms with large errors (yellow box); but DMGNN could predict motions with smaller errors in both short-term and long-term.

References

[1] Hsukuang Chiu, Ehsan Adeli, Borui Wang, DeAn Huang, and Juan Niebles. Action-agnostic human pose forecasting. *CoRR*, abs/1810.09676, 2018.

[2] Katerina Fragkiadaki, Sergey Levine, Panna Felsen, and Jitendra Malik. Recurrent network models for human dynamics. In *The IEEE International Conference on Computer Vision (ICCV)*, pages 4346–4354, December 2015.

[3] Partha Ghosh, Jie Song, Emre Aksan, and Otmar Hilliges. Learning human motion models for long-term predictions. *CoRR*, abs/1704.02827, 2017.

[4] Anand Gopalakrishnan, Ankur Mali, Dan Kifer, Lee Giles, and Alexander Ororbia. A neural temporal model for human motion prediction. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12116–12125, June 2019.

[5] Liangyan Gui, Yuxiong Wang, Xiaodan Liang, and Jose Moura. Adversarial geometry-aware human motion prediction. In *The European Conference on Computer Vision (ECCV)*, pages 786–803, September 2018.

[6] Xiao Guo and Jongmoo Choi. Human motion prediction via learning local structure representations and temporal dependencies. In *AAAI Conference on Artificial Intelligence*, February 2019.

[7] Du Huynh. Metrics for 3d rotations: Comparison and analysis. *Journal of Mathematical Imaging and Vision*, 35(2):155–164, October 2009.

[8] 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 (TPAMI)*, 36(7):1325–1339, July 2014.

[9] Ashesh Jain, Amir Zamir, Silvio Savarese, and Ashutosh Saxena. Structural-rnn: Deep learning on spatio-temporal

graphs. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5308–5317, June 2016.

[10] JogendraNath Kundu, Maharshi Gor, and RVenkatesh Babu. Bihmp-gan: Bidirectional 3d human motion prediction gan. In *AAAI Conference on Artificial Intelligence*, February 2019.

[11] Chen Li, Zhen Zhang, Wee Sun Lee, and Gim Hee Lee. Convolutional sequence to sequence model for human dynamics. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5226–5234, June 2018.

[12] Zhenguang Liu, Shuang Wu, Shuyuan Jin, Qi Liu, Shijian Lu, Roger Zimmermann, and Li Cheng. Towards natural and accurate future motion prediction of humans and animals. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10004–10012, June 2019.

[13] Wei Mao, Miaomiao Liu, Mathieu Salzmann, and Hongdong Li. Learning trajectory dependencies for human motion prediction. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019.

[14] Julieta Martinez, Michael Black, and Javier Romero. On human motion prediction using recurrent neural networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4674–4683, July 2017.

[15] Dario Pavllo, David Grangier, and Michael Auli. Quaternet: A quaternion-based recurrent model for human motion. In *British Machine Vision Converence (BMVC)*, pages 1–14, September 2018.

[16] Borui Wang, Ehsan Adeli, Hsukuang Chiu, Dean Huang, and JuanCarlos Niebles. Imitation learning for human pose prediction. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019.