MSR-GCN: Multi-Scale Residual Graph Convolution Networks for Human Motion Prediction
— Supplementary Material —

Lingwei Dang[1], Yongwei Nie[1*], Chengjiang Long[2], Qing Zhang[3] and Guiqing Li[1]

[1]School of Computer Science and Engineering, South China University of Technology, China
[2]JD Finance America Corporation, USA
[3]School of Computer Science and Engineering, Sun Yat-sen University, China

Abstract

In this supplementary material, we provide the loss function used to train the proposed MSR-GCN model. We also describe the detailed structure of MSR-GCN in a table.

1. Loss Function

We use ℓ_2 loss to optimize MSR-GCN. Let the j^{th} joint position in the t^{th} frame at s^{th} scale be $\hat{p}^s_{j,t}$, and the corresponding ground-truth be $p^s_{j,t}$, then the loss function for N training pose sequences each having J^s joints and T frames is written as

$$\mathcal{L}^s = \frac{1}{N \times J^s \times T} \sum_{n=1}^{N} \sum_{j=1}^{J^s} \sum_{t=1}^{T} \left\| \hat{p}^s_{j,t} - p^s_{j,t} \right\|_2. \quad (1)$$

The above loss is calculated at all S scales and added up to optimize the proposed model, that is,

$$\mathcal{P}^* = \arg\min_{\mathcal{P}} \sum_{s=1}^{S} \lambda^s \mathcal{L}^s, \quad (2)$$

where $\mathcal{P}$ indicates network parameters, and λ denotes hyper parameters and we set them as 1 at all scales.

2. Model Structure

The detailed MSR-GCN model structure is shown in Table 1. As mentioned in the main paper, our proposed approach is composed of three kinds of GCNs, called "Start GCNs", "Descending (D0-D3)/Ascending (A0-A3) GCNs", and "End/Decoding GCNs (E0-E3)".

*Corresponding author: nieyongwei@scut.edu.cn

Module	Layers	Output Size	Operations
Start GCN	GCL	66 × 64	GCL, A(66 × 66), W(35 × 64)
	GCLs	66 × 64	2-layer GCLs, A(66 × 66), W(64 × 64), res-connection
D0	GCLs	66 × 64	6-layer GCLs, A(66 × 66), W(64 × 64), res-connection
Downsampling 0	Linear1	36 × 64	linear transformation, W(66 × 36)
	Linear2	36 × 128	linear transformation,W(64 × 128)
D1	GCLs	36 × 128	6-layer GCLs, A(66 × 66), W(64 × 64), res-connection
Downsampling 1	Linear1	21 × 128	linear transformation, W(36 × 21)
	Linear2	21 × 256	linear transformation,W(128 × 256)
D2	GCLs	21 × 256	6-layer GCLs, A(21 × 21), W(256 × 256), res-connection
Downsampling 2	Linear1	12 × 256	linear transformation, W(21 × 12)
	Linear2	12 × 512	linear transformation,W(256 × 512)
D3	GCLs	12 × 512	6-layer GCLs, A(12 × 12), W(512 × 512), res-connection
A3	GCLs	12 × 512	6-layer GCLs, A(12 × 12), W(512 × 5124), res-connection
Upsampling 2	Linear1	21 × 21	linear transformation, W(12 × 21)
	Linear2	21 × 256	linear transformation,W(512 × 256)
A2	GCLs	21 × 256	6-layer GCLs, A(21 × 21), W(256 × 256), res-connection
Upsampling 1	Linear1	21 × 36	linear transformation, W(21 × 36)
	Linear2	256 × 128	linear transformation,W(256 × 128)
A1	GCLs	36 × 128	6-layer GCLs, A(36 × 36), W(128 × 128), res-connection
Upsampling 0	Linear1	66 × 128	linear transformation, W(36 × 66)
	Linear2	66 × 64	linear transformation,W(128 × 64)
A0	GCLs	66 × 64	6-layer GCLs, A(66 × 66), W(64 × 64), res-connection
E0	GCLs	66 × 64	3-layer GCLs, A(66 × 66), W(64 × 64), res-connection
	GCL	66 × 35	GCL, A(66 × 66), W(64 × 35)
E1	GCLs	36 × 128	3-layer GCLs, A(36 × 36), W(128 × 128), res-connection
	GCL	36 × 35	GCL, A(36 × 36), W(128 × 35)
E2	GCLs	21 × 256	3-layer GCLs, A(21 × 21), W(256 × 256), res-connection
	GCL	21 × 35	GCL, A(21 × 21), W(256 × 35)
E3	GCLs	12 × 512	3-layer GCLs, A(12 × 12, W(512 × 512), res-connection
	GCL	12 × 35	GCL, A(12 × 12), W(512 × 35)

Table 1. Detailed architecture of MSR-GCN.

The most basic buildingblock is Graph Convolution Layer (GCL), which is consist of a graph convolution layer, a batch normalization layer, a tanh activation layer, and a dropout layer (with rate 0.1). A graph convolution layer has an adjacency matrix A and parameters W.

Each GCN is composed of several GCLs. The size of A and W of these GCLs are shown in the table. We use linear layers for downsampling and upsampling. The size of the parameters in these linear layers are also shown in the table. In the third column of the table, we give the output size of the corresponding layer. We hope with these information the proposed MSR-GCN can be re-implemented by others.