Human Motion Prediction Using Graph Scattering Network

Baris Tura
Robotics, Cognition, Intelligence
Technical University of Munich
baris.tura@tum.de

Vitalii Rusinov
Informatics
Technical University of Munich
ge56cug@mytum.de

Abstract

The task of human motion prediction is concerned with forecasting a sequence of motion in the future given the past motion sequences. The difficulty of addressing this issue lies in the fact that it is difficult to embed both spatial and temporal interdependencies of the joints and carry over the information from the history while still leaving room for unseen motion modalities. Some models may be prone to overfitting and oversmoothing. We introduce a novel graph neural network architecture that includes a graph scattering network with spectral attention applied to a spatiotemporal graph that encodes human motion. We propose to use wavelet filters that are computed based on a non-negative, symmetric, and learnable adjacency matrix. Both the learnable graph wavelets and the spectral attention help significantly reduce the mean per joint position error. Our model outperforms the state-of-the-art model for deterministic human motion prediction using certain configurations of the graph adjacency matrix in long-term predictions on the Human3.6M dataset.

1. Introduction

The focus of the project is the prediction of human body motion given the past motion sequence. The task is highly beneficial for robot motion planning, virtual reality, computer animation, and sports analytics. Extensive effort has been put into human motion prediction. Conventional methods, such as Hidden Markov Models and Linear Dynamic Systems, were good at capturing simple motion patterns, but they may be less effective at the modeling of the extremely complex biomechanics of human motion compared to modern deep learning methods. Recurrent Neural Networks have shown impressive results with generating short sequences, however many of the methods tend to lose context, converge to a mean pose, or diverge for longer sequences. The deterministic motion prediction methods may produce accurate predictions, but these methods tend to duplicate seen sequences and lack diversity. The stochastic motion prediction methods incorporate motion variations in the prediction but are less capable of capturing the repetitive nature of human motion.

In this work, we focus on deterministic human motion prediction. Our contributions can be summarized as follows. (i) We propose a novel neural network architecture that for the first time uses a spectral spatiotemporal GCN for the task of human motion prediction; (ii) We use graph scattering transform with non-negative, symmetric, and learnable adjacency matrices and show that it significantly improves the results; (iii) Our model allows us to improve on state of the art using certain configurations of the human motion graph adjacency matrix.

2. Related Work

The task of human motion prediction can be subdivided into two main categories, stochastic motion prediction and deterministic motion prediction.

2.1. Deterministic Human Motion Prediction

Given an input sequence of human poses, the aim is to deterministically predict the future poses, which are expected to be close to ground truth and hence capture the patterns in the history well. Martinez et al. propose an RNN with residual connections linking the input and the output, and design a sampling-based loss to compensate for prediction errors during training. Mao et al. [6] propose a transformer based model, where they encode the information sequence-wise rather than frame-wise. Hereby, they map the motion sequences into frequency space based on the heuristic that human motion is repetitive on all levels. They then make use of attention mechanisms to encode the history information, which is aggregated decoded by means of several GCN layers.

2.2. Stochastic Human Motion Prediction

Contrary to deterministic case, stochastic predictions produce a range of plausible motion sequences and hence have more room for diverse and unseen motion modalities.

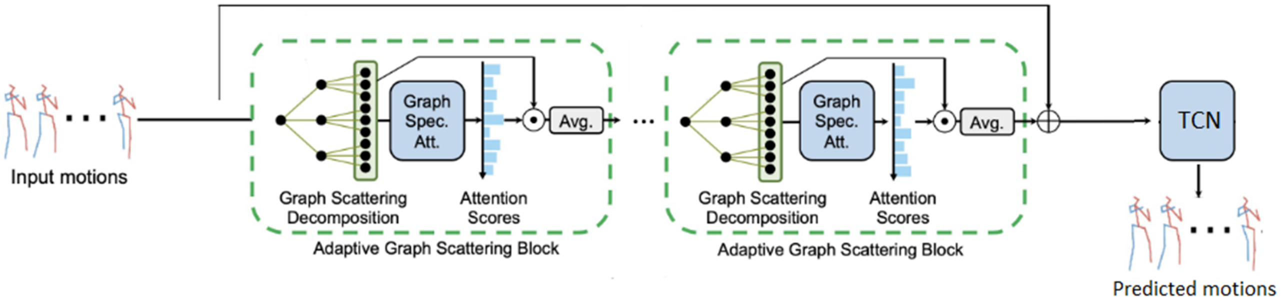
Input motions
Graph Spec. Att.
Avg.
Graph Scattering Decomposition
Attention Scores
Adaptive Graph Scattering Block
Graph Spec. Att.
Avg.
Graph Scattering Decomposition
Attention Scores
Adaptive Graph Scattering Block
TCN
Predicted motions

Figure 1. Method overview.

Zhang et al. employ a CVAE [9]. They map the input space into frequency space, and pass it through a GRU based CVAE structure, where they enforce the latent space to have two distinct parts: low and high frequency. The sampling is then accomplished by using a GRU+MLP based network. Yuan et al. propose diversifying latent flows (DLow) to exploit the latent space of an RNN-based VAE, which generates highly diverse but accurate future motions.

3. Our approach

Figure 1 gives an overview of the method. Its main components are the GNN encoder and the TCN decoder.

3.1. GNN encoder

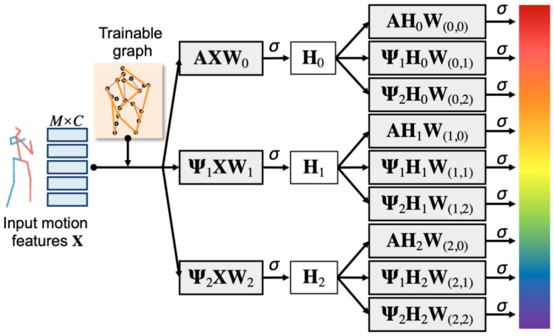

Figure 2. Architecture of graph scattering decomposition.

The GNN encoder is similar to the one used in [4]. However, we introduce several novelties into its structure. The encoder consists of four Adaptive Graph Scattering Blocks. Each AGSB consists of two operations: graph scattering decomposition and graph spectrum attention.

The graph scattering decomposition operation is illustrated in figure 2. The motion features X serve as the input to the operation. X has size (V*T) x C. V is the number of skeleton joints. T is the number of time steps in the motion history input sequence (we use T=10). In the first layer, X stores the xyz-coordinates of the V skeleton joints at each of the T time steps. X is first multiplied by the filter matrix on the left and then by the weight matrix on the right. Our filter bank consists of 3 filters. A is the adjacency matrix (the elements of A can be negative). It is used as the first filter in the filter bank. In contrast to [4], we first transform the adjacency matrix A into a symmetric matrix that only has non-negative coefficients (we further denote it as matrix A*) before computing the wavelet filters Ψ_1 and Ψ_2 from the matrix A*. A_{symm} is obtained by filling its elements below the main diagonal with the corresponding elements of A^T.

$$A^* = ReLu(A_{symm})$$

$$\Psi_1 = I - P$$
$$\Psi_2 = P - P^2$$

$$P = \frac{1}{2}(I + \frac{A^*}{\|A^*\|_F^2})$$

$$H_k = \sigma(\Psi_k X W_k)$$

The graph adjacency matrix A is of size (V*T) x (V*T). It is fully adaptive (learnable). Matrices W have size C x C. We define $\Psi_0 = A$. Each of the 9 outputs of the graph scattering layer is of size (V*T) x C.

The graph spectrum attention operation is illustrated in figure 3.

It inputs the 9 outputs of the graph scattering operation and learns 9 attention scores. For details on the graph spectrum attention architecture, please refer to [4]. However, we use PReLU activations in the spectral attention block instead of TanH activations for the purpose of training stability. Each of the 9 attention scores is multiplied by its corresponding output H_k. The sum of the multiplication results serve as the output of the AGSB.

Each AGSB produces an output of size (V*T) x C. Subsequently, 1x1 convolution is applied to rescale the hidden

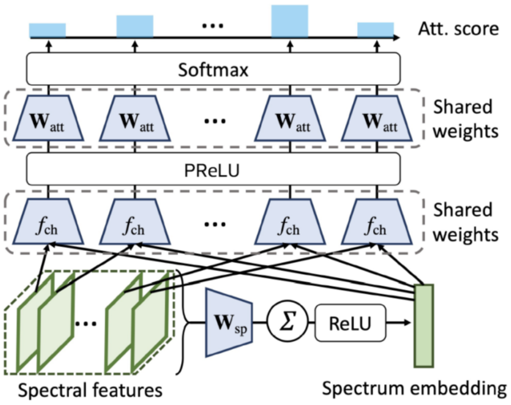

Figure 3. Architecture of graph spectrum attention.

dimension of the graph nodes, followed by batch normalization and dropout. The 4 AGSBs have the following hidden dimensions: 3, 64, 32, 64, 3.

3.2. TCN decoder

The TCN decoder is made of 4 layers, each consisting of a 3x3 convolution, batch normalization, and ReLU activation. For implementation details, please refer to [8].

4. Key experiments

For all experiments, we used the Human3.6M dataset [2]. We used Mean Per Joint Position Error as the error metric.

$$L_{MPJPE} = \frac{1}{VT}\sum_{k=1}^{T}\sum_{v=1}^{V}\|x_{vk} - \hat{x_{vk}}\|_2$$

4.1. Improvement of History Repeats Itself

We reimplemented the model described in [6] from scratch and conducted several experiments to improve the method results. In Experiment 1, we replaced the attention layer with a 2-layer convolutional network with filters in the temporal domain, followed by a discrete cosine transform. In Experiment 2, we replaced the attention layer with 1x1 convolution that learned constant attention coefficients for the previous subsequences. In Experiment 3, we replaced the GCN with a two-layer MLP. In Experiments 1 and 2, the validation error decreased compared to the baseline method. In Experiment 3, the validation error increased considerably. We concluded that the GCN made a larger contribution to the validation error reduction than the attention layer.

4.2. Basic improvements to STS-GCN

We explored various modifications to the model described in [8], including adding adjacency matrix prior, using bone length loss, creating banks of GCN layers applied to the same input in parallel, experimenting with the temporal graph convolution. These modifications did not lead to visible improvements in terms of the test error, therefore we decided to explore the spectral extensions of the method.

4.3. Exploring the spectral extensions of the method STS-GCN

Having implemented the model described in [4] using the framework of History Repeats Itself [6], we applied it to the spatiotemporal graph used in STS-GCN [8]. We used the adjacency matrix of size (V*T) x (V*T). The ablation study showed the importance of all the model components. We later found that using the non-negative and symmetric adjacency matrix A* for wavelet filter computation resulted in significant improvements.

5. Results

5.1. Quantitative results

	ST (400 ms)	LT (1000 ms)
ConvSeq2Seq [3]	72.7	124.2
LTD-10-10 [7]	58.9	114.0
HistRepItself [6]	58.3	112.1
MPUTC [5]	47.9	96.4
STSGCN [8]	38.3	75.6
Ours (separable adj. matrix)	41.4	76.9
Ours (full adj. matrix)	43.0	79.5

Table 1. Average MPJPE. ST denotes short-term predictions, LT stands for long-term predictions. Here STSGCN method uses different spatial adjacency matrices for different time steps.

Experiments show that our model outperforms strong baselines.

	LT (1000 ms)
HisRepItself [6]	112.1
Baseline with full adjacency matrix [8]	84.8
Ours – full model	79.5
Ours – without spectral attention	84.3
Ours – signed A, signed A*	80.6
Ours – non-negative A, non-negative A*	82.1
Ours – fixed (non-trainable) A*	84.1

Table 2. Ablation study, model variants. Baseline with full adjacency matrix denotes the STSGCN method [8] variant with the full adjacency matrix A of size (V*T) x (V*T).

The ablation study proves the importance of the key model components. The model variant with spectral attention, signed adjacency matrices A (for graph convolution), and non-negative symmetric adjacency matrices A* (for wavelet filter computation) performed the best.

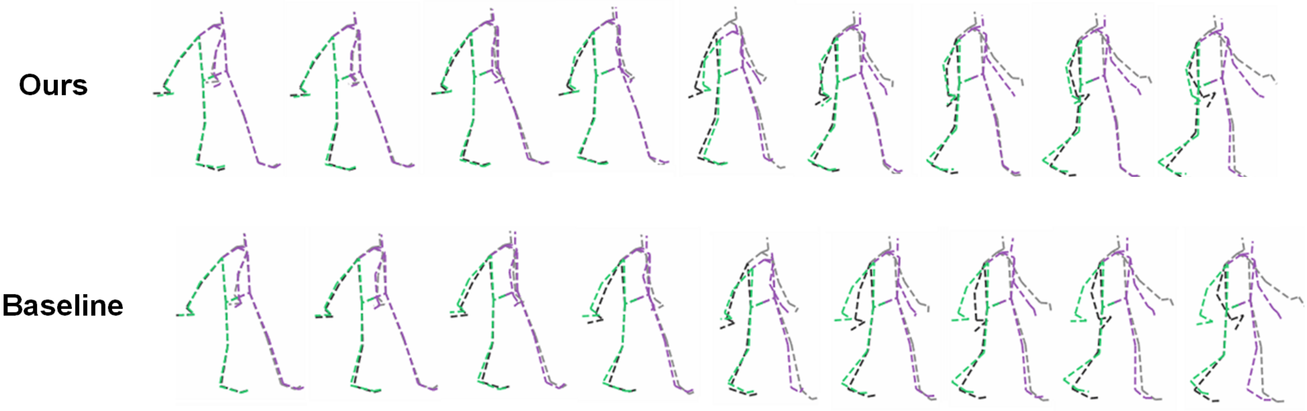

Figure 4. Qualitative results.

	LT (1000 ms)
Ours – GCNII model	82.5
Ours – full model with bone length loss	99.2
Ours – full model with separable adjacency matrix	76.9
Baseline with separable adjacency matrix [8]	78.6

Table 3. Additional experiments.

The results of our additional experiments show that GCNII [1] model seems to be less effective at fighting oversmoothing than the graph scattering transform. Our model also outperformes the baseline model when separable adjacency matrix is used (the signal is first propagated through time and then through space). However, in these experiments, we used shared spatial adjacency matrices for all time steps and different temporal adjacency matrices for different vertices.

5.2. Qualitative results

Figure 4 shows the comparison between our method and the baseline method.

6. Conclusion / Future work

We propose a novel neural network architecture that for the first time uses a spectral spatiotemporal GCN for the task of human motion prediction. Both wavelets and spectral attention help significantly reduce the test error. We discovered that wavelets work better with unsigned and symmetric learnable adjacency matrices, this change significantly improved the results. In the future, we plan to improve our method for the case of the separable adjacency matrix with different spatial adjacency matrices for different time steps.

References

[1] Ming Chen, Zhewei Wei, Zengfeng Huang, Bolin Ding, and Yaliang Li. Simple and deep graph convolutional networks. In *International Conference on Machine Learning*, pages 1725–1735. PMLR, 2020. 4

[2] 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. 3

[3] Chen Li, Zhen Zhang, Wee Sun Lee, and Gim Hee Lee. Convolutional sequence to sequence model for human dynamics. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5226–5234, 2018. 3

[4] Maosen Li, Siheng Chen, Zihui Liu, Zijing Zhang, Lingxi Xie, Qi Tian, and Ya Zhang. Skeleton graph scattering networks for 3d skeleton-based human motion prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 854–864, 2021. 2, 3

[5] Zhenguang Liu, Pengxiang Su, Shuang Wu, Xuanjing Shen, Haipeng Chen, Yanbin Hao, and Meng Wang. Motion prediction using trajectory cues. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13299–13308, 2021. 3

[6] Wei Mao, Miaomiao Liu, and Mathieu Salzmann. History repeats itself: Human motion prediction via motion attention. *Computer Vision – ECCV 2020*, page 474–489, 2020. 1, 3

[7] Wei Mao, Miaomiao Liu, Mathieu Salzmann, and Hongdong Li. Learning trajectory dependencies for human motion prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9489–9497, 2019. 3

[8] Theodoros Sofianos, Alessio Sampieri, Luca Franco, and Fabio Galasso. Space-time-separable graph convolutional network for pose forecasting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11209–11218, 2021. 3, 4

[9] Yan Zhang, Michael J. Black, and Siyu Tang. We are more than our joints: Predicting how 3d bodies move. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2