This ICCV workshop paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Skeleton Graph Scattering Networks for 3D Skeleton-based Human Motion Prediction

Maosen Li[1,2], Siheng Chen[✉1], Zihui Liu[1], Zijing Zhang[3], Lingxi Xie[2], Qi Tian[2], and Ya Zhang[✉1]

[1]Cooperative Medianet Innovation Center, Shanghai Jiao Tong University
[2] Huawei Cloud & AI
[3] State Key Laboratory of Modern Optical Instrumentation, Zhejiang University

{maosen_li, sihengc, zihui_liu} @sjtu.edu.cn, 11730048@zju.edu.cn, 198808xc@gmail.com,
tianqi1@huawei.com, ya_zhang@sjtu.edu.cn

Abstract

To achieve 3D skeleton-based human motion prediction, many graph-convolution-based methods are proposed for promising results; however, due to only preserving low-pass information over graphs, those graph convolution methods suffer from over-smoothing, causing the predicted poses staying the same in the long term. To resolve the over-smoothing issue, we propose a novel skeleton graph scattering network (SGSN), which leverages graph scattering to extract comprehensive motion information from multiple graph spectrum bands. The core of the proposed SGSN is the adaptive graph scattering block (AGSB), including two key modules: i) graph scattering decomposition, which decomposes information into various graph spectrum bands and updates the trainable features in each band, as well as ii) graph spectrum attention, which aggregates those features in various graph spectrum bands via trainable attention weights. Extensive experiments reveal that SGSN outperforms state-of-the-art methods by 8.5%, 9.0% *and* 3.9% *of 3D mean per joint position error (MPJPE) in average on Human3.6M, CMU Mocap and 3DPW datasets, respectively. We also test the mean angle error (MAE) on Human3.6M, which is lower by* 3.3% *than previous methods. Moreover, SGSN outperforms even more in the long-term prediction because of the alleviation of the over-smoothing.*

1. Introduction

In recent years, increasing attention has been attracted by processing the signals on the ubiquitous graphs in various scenarios, such as social networks [49], human behaviors [45, 55] and molecule structures [25, 57]. Moreover, graphs can also depict the spatial dependencies in some dynamic systems for kinetic modeling. The dynamic systems usually contain multiple agents performing interactive movements or complex evolution [26]. For example, skeleton-based human bodies carry body-joints during actions [33]; pedestrians in intersections show social effects [20]. In this work, we focus on the 3D skeleton-based human motion prediction, which aims to forecast the future poses conditioned on the historical ones and plays critical roles in various applications, such as human-computer interaction [16] and autonomous driving [6].

Human motion prediction has been studied for a long period. Traditional attempts employ state models [51, 50, 30, 44] to depict shallow distribution. In the deep learning era, algorithms capture flexible patterns. Considering the sequential formats, some methods [11, 54, 39, 15] build recurrent networks to predict poses step-by-step along time, while they would accumulate prediction errors due to the sequential generation mechanisms. Some feed-forward models [31] directly output the whole motion sequence to alleviate the errors caused the noisy self-regression, while they neglect the spatial dependencies on the body. Recently, graph-based models [38, 34, 7, 37] explicitly model the inherent body relations. Although many have improved the prediction quality, the deep and multi-step low-pass information propagation averages the joint feature to enhance their similarity and loses the dynamics variance among different joints, causing the over-smoothing issue. For example, the generated poses tend to show mode collapse and converge to a mean pose without clear movements, especially in the long-term future.

To address the mentioned problem, in this work, we propose *skeleton graph scattering networks* (SGSNs), which combine mathematically designed graph scatterings and trainable feature embeddings to capture motion features in various graph spectrum bands and alleviate the over-smoothing. In our SGSN, we build a deep feed-forward

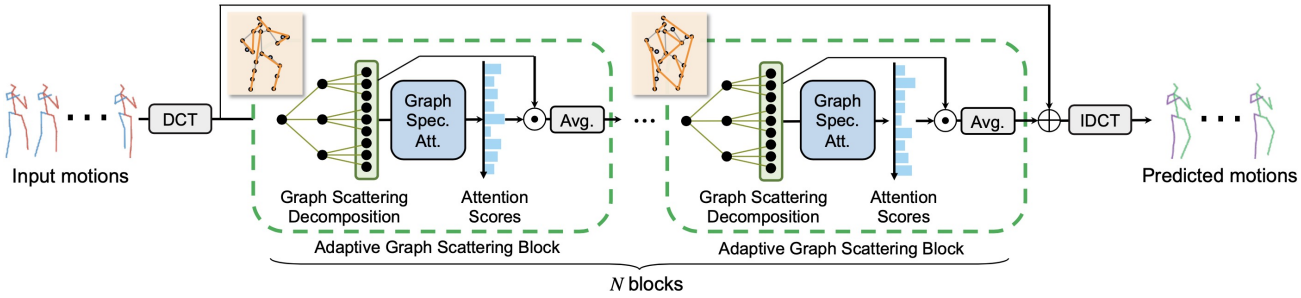

Figure 1. Architecture overview of the SGSN model. Taking the historical motions as inputs, the SGSN first applies DCT to convert the body-joint positions along time to the frequency domain. Then, N cascaded adaptive graph scattering blocks (AGSB) are built for deep feature extraction. Finally, we use inverse DCT to recover the temporal information and build a skip-connection for stable prediction. Note that different AGSBs employ dynamic and trainable pose graphs

network, which learns informative motion representation in spectrum to achieve effective motion prediction. The core module of SGSN is an *adaptive graph scattering block* (AGSB), which is formed by two key operations. The first is the graph scattering decomposition, which builds a tree structure with multi-layer graph scatterings to provide spectrum information; on the tree node, mathematically designed graph filter is computed from the body-graph structure and applied on the learnable joint representation. In this way, we could learn the flexible features that adapt to different actions. The second operation is the graph spectrum attention mechanism, which reflects the importance of sperctal features by calculating the attention scores. We further aggregate the spectrum weighted by the attention for information fusion.

As for the entire system (see Figure 1), our SGSN constructs hierarchical AGSBs in cascade to learn the underlying dynamics; note that, each AGSB carries dynamic and trainable body graph to adaptively depict the implicit interaction and dependencies during motions. Taking the 3D motions as inputs, SGSN first converts the temporal features by discrete cosine transform (DCT) to obtain more compact representation, which removes the complexity of temporal modeling [38] and helps the model focus on spatial graphs. The DCT-formed features are then fed to the AGSB pipeline. Finally, an inverse DCT recovers the output features to the temporal domain. We also build a skip-connection between inputs and outputs to force to predict the residual DCT coefficients, leading to stable prediction.

Extensive experiments are conducted for both short-term and long-term human motion prediction on various datasets, i.e., Human3.6M [23], CMU Mocap [1] and 3DPW [53]. We demonstrate that, the proposed SA-GCN could significantly outperform state-of-the-art methods in terms of different metrics inluding mean per joint position error (MPJPE) and mean angle error (MAE), respectively. The prediction visualization also reveals the rationality of our method. The main contributions of our work are summarized here:

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

- We propose the skeleton graph scattering networks (SGSN) to extract motion features at various graph spectrum bands and resolve the over-smoothing issue in the 3D skeleton-based human motion prediction.
- In our SGSN, we propose the adaptive graph scattering block (AGSB), which contains two operations: graph scattering decomposition and graph spectrum attention to learn rich spectral representation and aggregate comprehensive features for effective dynamics learning.
- We conduct experiments to quantatitively and qualitively verify that our SGSN outperforms state-of-the-art works by 8.5%, 9.0% and 3.9% of MPJPE for motion prediction on Human3.6M, CMU Mocap and 3DPW datasets, respectively. We also obtain lower MAE by 3.3% than previous methods on Human3.6M.

2. Related Works

2.1. Human Motion Prediction

3D skeleton-based human motion prediction is a critical task that has been widely explored. Many traditional methods develop algorithms based on state models [51, 50, 30, 44]. Recently, some recurrent-network-based models consider the sequential motion states. ERD [11] bridges encoder and decoder with a recurrent learner. Structural-RNN [24] builds recurrent networks to propagate information between body-parts. Pose-VAE [54] constructs an LSTM-based VAE. Res-sup [39], AGED [15] and DMGNN [34] model the pose displacements in an RNN model. Besides, some feed-forward networks abandon recurrent computation, even directly predict the whole sequences. CSM [31] builds both encoder and decoder with spatio-temporal convolutions. Furthermore, considering an articulated pose structures, some methods exploit the correlations between body-components. TrajGCN [38] and LDR [7] build adaptive body relations in spatial domain and apply graph convolutions to learn patterns. HisRep [37] builds a self-attention mechanism along time to emphasize periodic motion patterns. LPJP [4] designs progressive information propagation strategies in a transformer-based networks. Compared

to previous models, our methods aim to exploit rich band-pass spectrums based on pose graphs to alleviate the over-smoothing caused by previous low-pass graph convolutions for more precise prediction.

2.2. Graph Representation Learning

Graphs represent numerous data with non-grid structures by depicting the vertices relations [56, 46], which could be leveraged in various scenarios such as social networks [49], human behaviors [45, 55, 60, 56, 33, 47, 19, 59, 5, 29, 10, 36], and dynamic system analysis [26, 58, 21, 28, 32, 20]. To capture the graph patterns, some works study the graph neural networks (GNNs) mainly from two perspectives: a spectral perspective and a vertex perspective. From the spectral perspective, graphs are converted into its spectrum based on the eigen-decomposition of graph Laplacian or spectrum approximation [61, 3, 9, 27]. From a vertex perspective, feature aggregation resembles the classic convolution [18] or modify the propagation strategies, including node sampling, learning edge attentions or building recurrent networks [17, 42, 52, 35, 8].

Recently, to alleviate the over-smoothing caused by classic GNNs, graph scattering transform (GST) and related deep models are developed [22, 14, 40, 43]. GSTs generalize the grid-like scattering transforms [2, 48, 1] to the graph domains, showing theoretical justification in terms of spectrum properties and stability. [62] employs energy-preserving graph wavelets. [12] develops diffusion wavelets. [13] proves the stability for a large graph wavelet family. [22] designs a pruning algorithms to sample the important scattering channels. [40, 41] propose hybrid scattering GCNs with trainable feature GST. [43] expands GSTs on the spatio-temporal domain. In this work, we combine the nontrainable GST based on mathematically designed graph filters with trainable graph feature embedding network to extract highly flexible pattern in spectrum.

3. Skeleton Graph Scattering Network

3.1. Problem Formulation

Skeleton-based motion prediction generates the future pose sequences given the observed ones. Mathematically, let $\mathbf{X}^{(t)} \in \mathbb{R}^{M\times 3}$ be a pose matrix carrying the 3D coordinates of M body joints at time t, $\mathbb{X} = [\mathbf{X}^{(1)}, \dots, \mathbf{X}^{(T)}] \in \mathbb{R}^{T\times M\times 3}$ be a three-mode tensor that concatenates moving poses within T timestamps, where $\mathbb{X}^{[t,m,c]}$ is the cth coordinate of the mth joint at timestamp t. Based on these notations, let $\mathbb{X}^- = [\mathbf{X}^{(-T+1)}, \dots, \mathbf{X}^{(0)}] \in \mathbb{R}^{T\times M\times 3}$ represent T historical poses, $\mathbb{X}^+ = [\mathbf{X}^{(1)}, \dots, \mathbf{X}^{(\Delta T)}] \in \mathbb{R}^{\Delta T\times M\times 3}$ represent ΔT future poses. In motion prediction, we aim to propose a predictor $\mathcal{F}_{\text{pred}}(\cdot)$ to predict the future motions $\widehat{\mathbb{X}}^+ = \mathcal{F}_{\text{pred}}(\mathbb{X}^-)$ to approximate the ground-truth $\mathbb{X}^+$.

In this work, we further consider the spatial dependencies in a human pose, which could be depicted as a graph $G(\mathcal{V}, \mathbf{X}, \mathbf{A})$. $\mathcal{V} = \{v_1, \dots, v_M\}$ denotes the vertex set containing M nodes, whose features are recorded by $\mathbf{X} \in \mathbb{R}^{M\times 3}$. $\mathbf{A} \in \{0, 1\}^{M\times M}$ is the adjacency matrix, where $\mathbf{A}_{i,j} = 1$ if there is connection between v_i and v_j; otherwise $\mathbf{A}_{i,j} = 0$. We attempt to exploit the underlying features from the graph signals and motion dynamics.

3.2. Framework Architecture

Here we propose our *Skeleton Graph Scattering Network* (SGSN) for motion prediction. Figure 1 sketches the framework architecture of the proposed SGSN. Given the input motion $\mathbb{X}^-$, we first apply the discrete cosine transform (DCT) on the time axis to encode each joint's dynamics into the frequency domain; that is, $\mathbf{X}^- = \text{DCT}(\mathbb{X}^-) \in \mathbb{R}^{M\times C}$, where C denotes the number of DCT coefficients as well as the dimension of the transformed features. This processing enables a compact representation [38], and we eliminate the complexity to embed the temporal dynamics for an easy and stable training process.

Then, at the trunk of SGSN, we stack N adaptive graph scattering blocks (AGSBs) as the core components to learn the hierarchical motion features. In each AGSB, we build adaptively tuned graph structures to model the implicit constraints and interactions among joints at different feature levels; for example, in shallow layers, bone-connected joints might have strong constraints, while in deep layers, distant joints carry stronger interactions. Furthermore, each AGSB employs mathematically designed graph filters as well as trainable networks to extract rich and flexible spectral information and integrates the spectrum according to the importance of each channel; see details in Section 3.3. At the output layer, we apply inverse DCT to recover the temporal information for prediction. Moreover, we build skip connections between the input and output to capture the residual feature displacements for stable prediction.

Some methods [38, 37, 7] connect multiple graph convolution layers in a deep pipeline; however, the graph convolution pushes the edge-connected joint feature to be increasingly similar while weakening the joint difference, thus we call the graph convolution a low-pass graph filtering. Due to the iterative graph convolution, rich high-frequency information is removed and the joints lose their variations, causing nearly static 'mean poses'. Compared to these methods, our SGSN leverages graph scattering techniques to explicitly preserve information in a wide-range graph spectrum band, effectively alleviating the over-smoothing during joint feature propagation for precise and reasonable prediction of the highly dynamic motions.

3.3. Adaptive Graph Scattering Block

According to the SGSN framework, the core components are a series of *Adaptive Graph Scattering Blocks* (AGSBs), which extract features in large spectrum bands based on pose graphs. In our AGSB, there are two key operations:

1) graph scattering decomposition and **2) graph spectrum attention**, which learn band-pass features and aggregate them. The AGSBs have parameterized graph structures, feature extractors, and spectrum attentions to reflect the flexible pose representations.

Graph Scattering Decomposition. To capture the informative spectrum from the input motions, we propose our graph scattering decomposition, which combines the mathematically designed filter banks and data-driven feature embeddings together to learn the graph spectrum.

Concretely, the graph scattering decomposition forms a tree-structure network, having L layers that have exponentially growing numbers of tree nodes. These tree nodes carry different band-pass graph learner to extract corresponding features in parallel channels. Here we consider the first layer as an example, and we could expand the design to any layers. Let the DCT-formed pose feature be $\mathbf{X} \in \mathbb{R}^{M\times C}$, and a graph adjacency matrix $\mathbf{A} \in \mathbb{R}^{M\times M}$ is built to connect related body-joints. We set $\mathbf{A}$ to be adaptively tuned to estimate the implicit spatial relations on the entire body. Given the pose graph, we build the filter bank $\{h_k(\mathbf{A})|k = 0, 1, \ldots, K\}$ and further obtain a series of features $\{\mathbf{H}_{(k)} \in \mathbb{R}^{M\times C'}\}_{k=0}^{K}$ with any

$$\mathbf{H}_{(k)} = \sigma(h_k(\mathbf{A})\mathbf{X}\mathbf{W}_{(k)}),$$

where $\mathbf{W}_{(k)}$ denotes the trainable weights corresponding to the kth filter, and the nonlinear activation $\sigma(\cdot)$ disperses the graph frequency representation on the spectrum [22]. In this work, we set $\sigma(\cdot)$ to be Tanh function, which effectively constrain the feature values for stable prediction.

Based on the same graph structure $\mathbf{A}$, we employs a series of mathematically designed graph filters to obtain $\{h_k(\mathbf{A})\}_{k=0}^{K}$, including one low-pass graph convolution filter for $k = 0$ and K band-pass graph wavelet filters for $k = 1, \ldots, K$, to explicitly focus on various bands. Mathematically, the graph convolution filter is directly defined as $h_0(\mathbf{A}) = \mathbf{A}$, which models the one-order joint relations and similarities, and derives the graph convolution,

$$\mathbf{H}_{(0)} = \sigma\left(h_0(\mathbf{A})\mathbf{X}\mathbf{W}_{(0)}\right) = \sigma\left(\mathbf{A}\mathbf{X}\mathbf{W}_{(0)}\right), \quad (1)$$

where $\mathbf{W}_{(0)}$ denotes the trainable weights and $\sigma(\cdot)$ is the Tanh. Employing a graph-convolution-based information propagation, the graph convolution filter averages related nodes, capturing the low-frequency responses while impairing much high-frequency features. To capture comprehensive spectrums, the K graph wavelet filters are defined as

$$\begin{aligned} h_k(\mathbf{A}) &= \mathbf{\Psi}_k = \mathbf{I} - \mathbf{P}, && k = 1; \\ h_k(\mathbf{A}) &= \mathbf{\Psi}_k = \mathbf{P}^{2^{k-2}} - \mathbf{P}^{2^{k-1}}, && k = 2, \ldots, K, \end{aligned} \quad (2)$$

where $\mathbf{P} = 1/2(\mathbf{I} + \mathbf{A}/\|\mathbf{A}\|_F^2)$ is the normalized propagation matrix, which handles the amplitude of the values in the freely trainable adjacency matrix $\mathbf{A}$. In this way, given the graph wavelet filter $\mathbf{\Psi}_k$, the corresponding feature extraction is formulated as

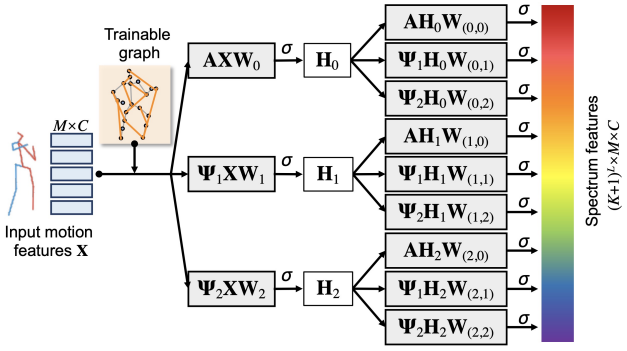

Figure 2. An example sketch of graph scattering decomposition, where we consider multi-layer graph scattering with designed filters and trainable networks to learn the informative spectrum.

$$\mathbf{H}_{(k)} = \sigma\left(h_k(\mathbf{A})\mathbf{X}\mathbf{W}_{(k)}\right) = \sigma\left(\mathbf{\Psi}_k\mathbf{X}\mathbf{W}_{(k)}\right), \quad (3)$$

Combining Eq. (1) and Eq. (3) together, we obtain the bank of spectrum features, i.e., $\{\mathbf{H}_{(0)}, \mathbf{H}_{(1)}, \ldots, \mathbf{H}_{(K)}\}$, which form the output of the first layer of the graph scattering decomposition. Note that, besides the mathematically designed filters, we introduce trainable graph topologies and network weights to embed the inputs. The benefits include that the graph information could be flexibly converted, which improves dynamics learning effectively.

At the next layer of graph scattering, we leverage the same filters $\{h_k(\mathbf{A})|k = 0, 1, \ldots, K\}$ and repeat graph scattering on each $\mathbf{H}_k$. Therefore, in the scattering tree, we could index a tree node at the ℓth layer by the path $p^{(\ell)} = \{k^{(1)}, \ldots, k^{(\ell)}\}$ to denote the sequence with ℓ filterings on the inputs. The scattering at the channel indexed by $(p_{(\ell)}, k)$ at layer $\ell + 1$ is

$$\mathbf{H}_{(p_{(\ell)},k)} = \sigma\left(h_k(\mathbf{A})\mathbf{H}_{(p_{(\ell)})}\mathbf{W}_{(p_{(\ell)},k)}\right) \quad (4)$$

Thus, in this scattering tree, each non-leaf feature has $K+1$ new branches, and the ℓth layer has $(K+1)^\ell$ responses. For visual understanding, the architecture of the graph scattering decomposition is sketched in Figure 2, where we show two layers and three filters as an example.

Different from previous deep graph convolution models [38, 34, 7, 37], we employ designed graph scatterings to explicitly enrich the large-band graph spectrum. Previous Traj-GCN [38] is a special case of our methods which just learn motion features by $\mathbf{H}_{(0)} = \sigma(\mathbf{A}\mathbf{X}\mathbf{W}_{(0)})$, thus only low-pass information is preserved to potentially cause the over-smoothing in this 'fine-grained' motion prediction task. Different from previous Sc-GCN [40] which builds trainable filtering in cascade, our scattering decomposition methods preserve more comprehensive information for reliable pattern learning.

Graph Spectrum Attention. Since the graph scattering decomposition provides spectral features at different bands, we need to leverage these information to abstract the

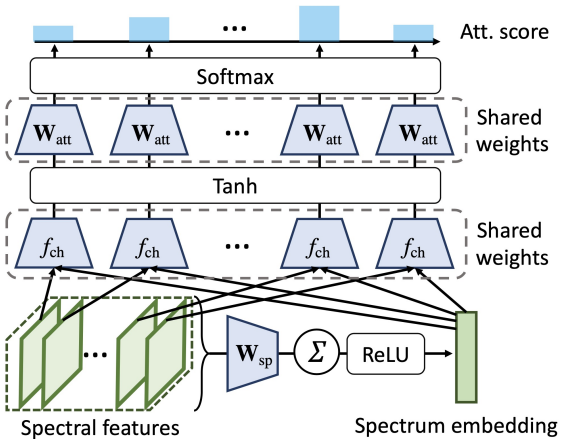

Figure 3. The architecture of the graph spectrum attention mechanism, where we compare the relations between individual channels and the whole spectrum to reflect the feature importance.

core dynamics. A straightforward way is averaging all the scattering channels, while this method cannot recognize the most crucial frequency bands to improve prediction. To address this problem, we propose a graph spectrum attention, which calculates the attention scores of each scattering feature, and integrate the features weighted by the scores to generate the final representation.

In our graph spectrum attention, we aim to compare the similarities between each scattering feature and the full bands to reflect the importance of one channel over the entire spectrum. Let the scattering features obtained by the L-layer graph scattering decomposition be $\{\mathbf{H}_{(k)}\}_{k=0}^{(K+1)^L}$ for any $\mathbf{H}_{(k)} \in \mathbb{R}^{M\times C}$. We first embed all the features and aggregate them as a spectrum embedding $\mathbf{H}_{\text{sp}} \in \mathbb{R}^{M\times C}$, which is formulated as

$$\mathbf{H}_{\text{sp}} = \text{ReLU}\left(\frac{1}{(K+1)^L}\sum\nolimits_{k=0}^{(K+1)^L}\mathbf{H}_{(k)}\mathbf{W}_{\text{sp}}\right), \quad (5)$$

where $\mathbf{W}_{\text{sp}}$ denotes trainable parameters and $\text{ReLU}(\cdot)$ is the ReLU activation function. $\mathbf{H}_{\text{sp}}$ carries the comprehensive information over the full bands learned by the graph scattering decomposition. Then, given the spectrum embedding and individual scattering features, the attention score vector $\boldsymbol{\alpha} \in [0,1]^{(K+1)^L}$ is calculated, whose jth element is obtained through a softmax function,

$$\boldsymbol{\alpha}_j = \frac{\exp\left(\mathbf{w}_{\text{att}}^\top \tanh\left(f_{\text{ch}}\left([\mathbf{H}_{\text{sp}}, \mathbf{H}_{(j)}]\right)\right)\right)}{\sum_{k=0}^{(K+1)^L}\exp\left(\mathbf{w}_{\text{att}}^\top \tanh\left(f_{\text{ch}}\left([\mathbf{H}_{\text{sp}}, \mathbf{H}_{(k)}]\right)\right)\right)}$$

where $\mathbf{w}_{\text{att}} \in \mathbb{R}^C$ denotes a trainable vector for feature projection; $f_{\text{ch}}(\cdot)$ is an MLP; $\tanh(\cdot)$ is the Tanh function; and $[\cdot,\cdot]$ denotes concatenation on the feature dimension. Considering the intermediate state learned by $f_{\text{ch}}(\cdot)$, we employ underlying motion representation that are beneficial in guiding the key spectral features. The architecture of the graph spectrum attention is illustrated in Figure 3, where we show the computation process of the importance of each spectral feature over the spectrum.

Given the learned attention scores, we then make use of them to weight spectral channels for information aggregation. Let the feature bank at the Lth graph scattering decomposition layer be $\{\mathbf{H}_{(k)}\}_{k=0}^{(K+1)^L}$, The aggregated spectrum features $\mathbf{H} \in \mathbb{R}^{M\times C}$ is formulated as

$$\mathbf{H} = \sum\nolimits_{k=0}^{(K+1)^L}\boldsymbol{\alpha}_k\mathbf{H}_{(k)}, \quad (6)$$

where a larger $\boldsymbol{\alpha}_k$ reflects a higher influence level of the corresponding $\mathbf{H}_{(k)}$ in the derived comprehensive feature, which is further fed to the next AGSB as the input feature.

Compared to previous GSAN [41], which proposes self-attension to measure the relations between the graph node features before and after graph filtering, besides the difference from a series of detailed operations, our method emphasizes the influence level of scattering features over the spectrum, thus we measure the relations between individual scattering features with the spectrum embedding to reflect the attention statuses.

3.4. Loss Function

To train the proposed SGSN, we here define the loss function. Suppose that we take N samples in a mini-batch as inputs, and let the nth ground-truth and predicted motion sample be $\mathbb{X}_n^+$ and $\widehat{\mathbb{X}}_n^+$. The loss function $\mathcal{L}$ is defined as the average ℓ_2 distance between the targets and predictions:

$$\mathcal{L} = \frac{1}{N}\sum_{n=1}^{N}\|\mathbb{X}_n^+ - \widehat{\mathbb{X}}_n^+\|^2 \quad (7)$$

Given the loss function, all the parameters in our SGSN are trained end-to-end, including the body graph structures, weights in the graph scattering decomposition and the graph spectrum attention module.

4. Experiments

4.1. Datasets and Model Configuration

Dataset 1: Human 3.6M (H3.6M) H3.6M dataset [23] has 7 subjects performing 15 classes of actions. There are 32 joints in each subject. Along the time axis, all sequences are downsampled by two. Following previous paradigms [39], the models are trained on 6 subjects and tested on the specific clips of the 5th subject.

Dataset 2: CMU motion capture (CMU Mocap) CMU Mocap consists of 5 general classes of actions, where each subject has 38 joints and we preserve 26 joints with non-zero exponential maps. Following [31], we use 8 actions: 'basketball', 'basketball signal', 'directing traffic', 'jumping', 'running', 'soccer', 'walking' and 'washing window'.

Dataset 3: 3D Pose in the Wild (3DPW) The 3D Pose in the Wild dataset (3DPW) [53] is a large-scale dataset that contains more than 51k frames with 3D poses for challenging indoor and outdoor activities. We adopt the training,

Table 1. Comparison of the MPJPEs of various models for short-term motion prediction on H3.6M dataset. We also introduce an SGSN variant called SGSN (no Att.), which replace the graph spectrum attention by averaging the spectrum information.

Motion	Walking				Eating				Smoking				Discussion				Directions				Greeting			
millisecond	80	160	320	400	80	160	320	400	80	160	320	400	80	160	320	400	80	160	320	400	80	160	320	400
Res-sup [39]	23.8	40.4	62.9	70.9	17.6	34.7	71.9	87.7	19.7	36.6	61.8	73.9	31.7	61.3	96.0	103.5	36.5	56.4	81.5	97.3	37.9	74.1	139.0	158.8
CSM [31]	17.1	31.2	53.8	61.5	13.7	25.9	52.5	63.3	11.1	21.0	33.4	38.3	18.9	39.3	67.7	75.7	22.0	37.2	59.6	73.4	24.5	46.2	90.0	103.1
Traj-GCN [38]	8.9	15.7	29.2	33.4	8.8	18.9	39.4	47.2	7.8	14.9	25.3	28.7	9.8	22.1	39.6	44.1	12.6	24.4	48.2	58.4	14.5	30.5	74.2	89.0
DMGNN [34]	9.3	15.1	28.6	35.2	8.5	15.4	37.2	46.8	8.5	14.4	27.1	30.4	10.2	20.8	39.7	46.3	12.9	26.2	48.8	58.0	14.3	29.6	74.5	87.8
HisRep [37]	8.4	15.6	27.4	32.1	8.1	17.6	37.0	44.3	7.1	14.7	26.0	28.7	9.2	21.3	38.3	43.2	11.9	23.0	45.9	57.7	13.1	28.3	72.2	88.2
LPJP [4]	**7.9**	**14.5**	29.1	34.5	8.4	18.1	37.4	45.3	**6.8**	**13.2**	**24.1**	**27.5**	**8.3**	21.7	43.9	48.0	11.1	22.7	48.0	58.4	13.2	28.0	64.5	86.9
SGSN (no Att.)	8.6	15.6	28.0	32.9	8.2	17.4	36.6	44.0	7.5	14.1	24.4	28.4	9.4	21.6	37.3	43.4	10.9	23.7	53.2	65.5	13.0	26.9	65.9	82.2
SGSN	8.3	15.0	**26.7**	**31.5**	**7.9**	**17.4**	**35.8**	**43.7**	7.0	13.8	23.6	28.2	**8.0**	**19.1**	**34.7**	**40.4**	**10.6**	**21.6**	**45.4**	**56.3**	**12.6**	**26.5**	**63.8**	**79.6**
Motion	Phoning				Posing				Purchases				Sitting				Sitting Down				Taking Photo			
millisecond	80	160	320	400	80	160	320	400	80	160	320	400	80	160	320	400	80	160	320	400	80	160	320	400
Res-sup [39]	25.6	44.4	74.0	84.2	27.9	54.7	131.3	160.8	40.8	71.8	104.2	109.8	34.5	69.9	126.3	141.6	28.6	55.3	101.6	118.9	23.6	47.4	94.0	112.7
CSM [31]	17.2	29.7	53.4	61.3	16.1	35.6	86.2	105.6	29.4	54.9	82.2	93.0	19.8	42.4	77.0	88.4	17.1	34.9	66.3	77.7	14.0	27.2	53.8	66.2
Traj-GCN [38]	11.5	20.2	37.9	43.2	9.4	23.9	66.2	82.9	19.6	38.5	64.4	72.2	10.7	24.6	50.6	62.0	11.4	27.6	56.4	67.6	6.8	15.2	38.2	49.6
DMGNN [34]	11.2	18.6	37.1	45.8	9.0	23.6	67.3	84.2	19.8	37.7	62.8	74.3	10.5	24.3	49.8	61.9	12.8	28.4	55.2	69.1	8.2	15.6	38.9	53.7
HisRep [37]	11.1	19.1	38.0	44.7	8.8	24.4	68.2	83.0	19.0	38.9	63.5	72.6	10.2	24.6	52.0	64.0	10.2	**26.2**	55.6	68.7	6.5	15.8	42.8	53.6
LPJP [4]	**10.8**	19.6	37.6	46.8	8.3	22.8	65.6	81.8	18.5	38.1	61.8	69.6	**9.5**	23.9	49.8	61.8	11.2	29.9	59.8	68.4	6.3	14.5	38.8	49.4
SGSN (no Att.)	11.3	18.7	36.0	41.8	8.6	23.2	67.2	83.7	18.8	38.4	65.9	72.1	10.2	24.0	50.1	60.6	10.3	26.0	51.9	**59.2**	6.4	14.8	39.3	51.4
SGSN	10.9	**18.1**	**36.2**	**41.4**	**8.2**	**22.7**	**64.8**	**80.9**	**18.4**	**36.9**	**60.0**	**68.5**	9.8	**23.0**	**46.2**	**56.4**	**10.1**	**24.7**	**51.0**	60.2	**6.0**	**13.9**	**36.3**	**47.8**
Motion	Waiting				Walking Dog				Walking Together				Average											
millisecond	80	160	320	400	80	160	320	400	80	160	320	400	80	160	320	400								
Res-sup [39]	29.5	60.5	119.9	140.6	60.5	101.9	160.8	188.3	23.5	45.0	71.3	82.8	30.8	57.0	99.8	115.5								
CSM [31]	17.9	36.5	72.9	90.7	40.6	74.7	116.6	138.7	15.0	29.9	54.3	65.8	19.6	37.8	68.1	80.2								
Traj-GCN [38]	9.5	22.0	57.5	73.9	32.2	58.0	102.2	122.7	8.9	18.4	35.3	44.3	12.1	25.0	51.0	61.3								
DMGNN [34]	9.0	21.4	56.7	72.8	30.4	57.2	105.6	120.8	8.6	19.0	35.7	45.2	12.2	24.5	51.0	62.1								
HisRep [37]	9.2	22.6	58.7	73.9	27.1	**49.4**	**98.8**	**118.3**	8.6	18.4	**33.6**	**39.8**	11.2	24.0	50.5	60.9								
LPJP [4]	8.4	21.5	53.9	69.8	**22.9**	50.4	100.8	119.8	8.7	18.3	34.2	44.1	10.7	23.8	50.0	60.2								
SGSN (no Att.)	8.7	21.3	53.4	68.3	27.1	54.7	93.8	112.7	8.3	17.7	35.2	45.6	11.2	23.8	49.3	59.5								
SGSN	**8.1**	**20.1**	**52.8**	**67.8**	25.7	53.0	93.0	111.4	**8.1**	**17.6**	34.5	43.8	**10.6**	**22.9**	**47.0**	**56.9**								

Table 2. Comparison of the MAEs of various models for short-term motion prediction on H3.6M dataset. We also introduce an SGSN variant called SGSN (no Att.), which replace the graph spectrum attention by averaging the spectrum information.

Motion	Walking				Eating				Smoking				Discussion				Directions				Average			
millisecond	80	160	320	400	80	160	320	400	80	160	320	400	80	160	320	400	80	160	320	400	80	160	320	400
Res-sup [39]	0.28	0.49	0.72	0.81	0.23	0.39	0.62	0.76	0.33	0.61	1.05	1.15	0.31	0.68	1.01	1.09	0.26	0.47	0.72	0.84	0.36	0.67	1.02	1.15
CSM [31]	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	0.39	0.60	0.80	0.91	0.38	0.68	1.01	1.13
Traj-GCN [38]	0.18	0.31	0.49	0.56	**0.16**	0.29	0.50	0.62	0.22	0.41	0.86	0.80	0.20	0.51	**0.77**	0.85	0.26	0.45	0.71	0.79	0.27	0.52	0.83	0.95
DMGNN [34]	0.18	0.31	0.49	0.58	0.17	0.30	0.49	0.59	0.21	0.40	**0.81**	**0.78**	0.26	0.65	0.92	0.99	0.25	0.44	0.65	0.71	0.27	0.52	0.82	0.94
HisRep [37]	0.18	**0.30**	**0.46**	**0.51**	**0.16**	0.29	0.49	0.60	0.22	0.42	0.86	0.80	0.20	0.52	0.78	0.87	0.25	0.43	**0.60**	0.69	0.27	0.52	0.82	0.94
LPJP [4]	**0.17**	**0.30**	0.51	0.55	**0.16**	0.29	0.50	0.61	0.21	0.40	0.85	0.78	**0.19**	0.54	0.89	0.94	0.22	0.39	0.62	0.69	**0.25**	0.49	0.83	0.94
SGSN (no Att.)	**0.17**	0.31	0.50	0.56	**0.16**	0.29	0.49	0.60	0.22	0.41	0.83	0.80	0.21	0.52	0.80	0.86	0.24	0.41	0.63	0.73	0.26	0.51	0.82	0.93
SGSN	**0.17**	**0.30**	0.47	0.53	**0.16**	**0.27**	**0.47**	**0.58**	**0.20**	**0.39**	0.82	**0.78**	0.20	**0.50**	**0.77**	**0.84**	**0.21**	**0.39**	**0.60**	**0.68**	**0.25**	**0.48**	**0.79**	**0.91**

test and validation separation suggested by the official setting. The frame rate of the 3D poses is 30Hz.

Model settings. We implement our SGSN with PyTorch 1.4 on one NVIDIA Tesla V100 GPU. We set 10 AGSBs to form the entire model. In each AGSBs, we consider 2 layers of graph scattering decomposition, and each tree node applies 1 graph convolution and $K = 2$ graph wavelets. The hidden dimension in each AGSB is 256. We use Adam optimizer to train our model with batch size 16. The learning rate is 0.0005 with a 0.96 decay for every two epochs, and the gradients are clipped to a maximum $\ell 2$-norm of 1.

4.2. Baselines and Evaluation Metrics

Baselines. We compare our model to several recent effective methods, including the RNN-based Res-sup [39], CNN-based CSM [31], and graph-based Traj-GCN [38], DMGNN [34], HisRep [37] and LPJP [4].

Evaluation Metrics. We consider two metrics. First, we use the Mean Per Joint Position Error (MPJPE) in 3D Euclidean space, where the input motions are formed in 3D space. Second, we consider a more conventional metric called Mean Angle Error (MAE) in angle space, where the input motions are in form of exponential maps. We mainly focus MPJPE, which covers larger ranges of error values for clearer comparison.

4.3. Comparison to State-of-the-Art Methods

To validate the proposed SGSN, we show the quantitative performance for both short-term and long-term motion prediction on H3.6M, CMU Mocap and 3DPW. We also illustrate the predicted samples for qualitative evaluation.

Short-term prediction. Short-term motion prediction aims to predict the poses within 500 milliseconds. On H3.6M, we compare SGSN to state-of-the-art methods for predicting poses in 400 milliseconds. Table 1 shows MPJPEs of various methods on 15 actions and the average MPJPE over these actions. Besides the previous models, we also present a degraded SGSN variant; that is, we replace our graph spectrum attention by just averaging the spectrum information (SGSN (no Att.)). We see that i) the SGSN with graph spectrum attention obtains more precise prediction than the variant with spectrum averaging, showing the effectiveness of the graph spectrum attention for channel aggregation; ii) compared to previous methods, SGSN has much lower MPJPEs by 5.3% in average.

Moreover, we compare our SGSN to baselines on H3.6M for short-term prediction in terms of MAE metric. We present MAEs of the 5 representative actions and the average MPJPEs over all the 15 actions in Table 2. Compared to previous methods, SGSN achieves lower MAEs by 3.3% in average.

Table 3. Prediction MPJPEs of different methods for long-term prediction on the 15 actions of H3.6M dataset. We also present the average prediction results across all the actions.

Motion	Walking		Eating		Smoking		Discussion		Directions		Greeting		Phoning		Posing	
milliseconds	560	1k	560	1k	560	1k	560	1k	560	1k	560	1k	560	1k	560	1k
Res-sup. [39]	73.8	86.7	101.3	119.7	85.0	118.5	117.7	144.6	105.3	132.5	129.9	161.2	103.2	132.0	145.1	191.3
CSM [31]	59.2	71.3	66.5	85.4	42.0	67.9	84.1	116.9	92.8	122.1	110.7	139.4	93.0	120.6	124.2	**179.8**
Traj-GCN [38]	42.3	51.3	56.5	68.6	32.3	60.5	70.5	103.5	87.8	113.9	96.2	90.8	65.2	115.8	111.0	210.1
DMGNN [34]	41.9	49.6	57.0	68.4	36.4	65.7	71.1	99.8	88.4	116.2	100.5	97.1	66.7	118.3	113.6	215.8
HisRep [37]	41.5	49.0	59.6	73.0	37.9	68.8	**66.6**	**96.5**	86.0	105.8	99.5	93.4	63.5	110.2	107.3	213.1
SGSN	**36.6**	**43.6**	**56.0**	**68.2**	**31.6**	**58.8**	69.1	**96.5**	**79.0**	**101.0**	**93.4**	**86.4**	**63.0**	**100.6**	**104.7**	204.5
Motion	Purchases		Sitting		Sitting Down		Taking Photo		Waiting		Walking Dog		Walking Toge		Average	
milliseconds	560	1k	560	1k	560	1k	560	1k	560	1k	560	1k	560	1k	560	1k
Res-sup. [39]	124.5	159.8	118.9	160.6	139.5	181.7	115.9	162.4	108.6	142.2	131.8	163.3	84.5	100.7	105.7	143.8
CSM [31]	110.5	146.3	104.2	129.3	119.7	162.0	96.9	135.2	**89.1**	**121.4**	**117.0**	**153.5**	74.1	96.5	85.6	123.2
Traj-GCN [38]	92.4	127.9	84.4	115.7	90.2	140.6	77.9	90.3	101.2	168.0	140.8	174.3	60.2	82.5	79.9	114.3
DMGNN [34]	94.8	126.5	86.1	114.4	92.8	147.5	75.6	93.4	103.3	171.9	144.7	172.6	57.9	82.8	82.1	116.0
HisRep [37]	95.4	126.1	83.6	110.2	88.3	143.8	76.7	91.9	100.8	166.5	138.4	162.1	56.5	79.8	74.1	112.7
SGSN	**87.9**	**118.9**	**80.3**	**109.8**	**84.5**	**126.8**	**69.6**	**86.3**	97.7	162.4	127.5	158.6	**56.3**	**79.8**	**67.3**	**100.8**

Table 4. Prediction MPJPEs of different methods on the 8 actions of CMU Mocap for both short-term and long-term motion prediction. We also present the average prediction results across all the actions.

Motion	Basketball					Basketball Signal					Directing Traffic					Jumping					Running				
milliseconds	80	160	320	400	1000	80	160	320	400	1000	80	160	320	560	1000	80	160	320	560	1000	80	160	320	560	1000
Res-sup. [39]	18.4	33.8	59.5	70.5	106.7	12.7	23.8	40.3	46.7	77.5	15.2	29.6	55.1	66.1	127.1	36.0	68.7	125.0	145.5	195.5	15.6	19.4	31.2	36.2	43.3
CSM [31]	16.7	30.5	53.8	64.3	91.5	8.4	16.2	30.8	37.8	76.5	10.6	20.3	38.7	48.4	**115.5**	22.4	44.0	87.5	106.3	162.6	**14.3**	**16.3**	**18.0**	**20.2**	**27.5**
Traj-GCN [38]	14.0	25.4	49.6	61.4	106.1	3.5	6.1	11.7	15.2	53.3	7.4	15.1	31.7	42.2	152.4	16.9	34.4	76.3	96.8	164.6	25.5	36.7	39.3	39.9	58.2
DMGNN [34]	13.6	24.9	49.4	62.0	105.7	3.3	5.9	13.1	15.6	55.5	7.6	14.5	30.9	41.6	148.3	16.6	34.0	74.6	95.8	162.4	25.1	38.3	39.5	39.9	59.7
LPJP [4]	11.6	21.7	44.4	57.3	90.9	2.6	4.9	12.7	18.7	75.8	6.2	12.7	29.1	39.6	149.1	**12.9**	**27.6**	73.5	92.2	176.6	23.5	34.2	35.2	36.1	43.1
SGSN	**11.1**	**20.2**	**41.3**	**52.9**	**89.1**	**2.2**	**4.1**	**9.7**	**14.7**	**51.5**	**5.8**	**11.0**	**24.7**	**32.1**	137.6	13.8	29.9	**71.5**	**90.8**	**160.2**	19.8	24.7	26.6	30.2	44.2
Motion	Soccer					Walking					Washing Window					Average									
milliseconds	80	160	320	560	1000	80	160	320	560	1000	80	160	320	560	1000	80	160	320	560	1000					
Res-sup. [39]	20.3	39.5	71.3	84.0	129.6	8.2	13.7	21.9	24.5	32.2	8.4	15.8	29.3	**35.4**	**61.1**	16.8	30.5	54.2	63.6	96.6					
CSM [31]	12.1	21.8	41.9	52.9	94.6	7.6	12.5	23.0	27.5	49.8	8.2	15.9	32.1	39.9	58.9	12.5	22.2	40.7	49.7	84.6					
Traj-GCN [38]	11.3	21.5	44.2	55.8	117.5	7.7	11.8	19.4	23.1	40.2	5.9	11.9	30.3	40.0	79.3	11.5	20.4	37.8	46.8	96.5					
DMGNN [34]	11.9	21.4	44.5	56.1	115.8	8.3	12.4	21.9	23.6	41.0	5.8	11.5	29.7	39.3	76.8	11.5	20.3	38.0	46.7	95.5					
LPJP [4]	**9.2**	18.4	39.2	**49.5**	**93.9**	6.7	10.7	21.7	27.5	37.4	5.4	11.3	29.2	39.6	79.1	9.8	17.6	35.7	45.1	93.2					
SGSN	**9.2**	**18.1**	**38.8**	**49.5**	103.3	**5.9**	**9.6**	**18.6**	**22.8**	**31.2**	**4.9**	**10.1**	**28.1**	37.3	71.1	**9.1**	**16.0**	**32.4**	**41.3**	**86.0**					

Table 5. The average prediction MPJPEs across the test set of 3DPW at various prediction time steps.

	Average MAE						
milliseconds	100	200	400	600	800	900	1000
Res-sup. [39]	102.4	113.9	173.1	191.9	201.1	205.8	210.7
CSM [31]	57.3	71.6	124.9	155.4	174.7	183.1	187.5
Traj-GCN [38]	16.4	35.6	67.8	**90.6**	106.9	113.4	117.8
DMGNN [34]	17.1	37.3	70.1	94.5	109.7	117.8	123.6
HisRep [37]	15.7	35.9	66.5	91.2	105.4	111.6	114.9
SGSN	**14.9**	**32.3**	**62.2**	91.5	**103.3**	**107.0**	**110.0**

Long-term prediction. Long-term motion prediction aims to predict the poses over 500 milliseconds, which is challenging due to the action variation. Table 3 presents the MPJPEs of models for prediction at the 560 ms and 1000 ms on the H3.6M. SGSN achieves more effective prediction on most actions and has lower MPJPEs by 11.1% in average.

We also test our SGSN for short-term and long-term prediction on CMU Mocap. Table 4 shows the MPJPEs within in the future 1000 ms. We see that, SGSN significantly outperforms the state-of-the-art methods on most actions at various prediction step, and the average prediction MPJPE is much lower by 9.0% than previous methods.

Furthermore, we test our SGSN on 3DPW dataset for both short-term ($\leq$ 500 ms) and long-term motion prediction ($>$ 500 ms). We present the average MPJPEs across all the test samples at different prediction steps in Table 5. We see that, compared to the state-of-the-art methods, the proposed SGSN outperforms previous methods, where the prediction MPJPE is lower by 3.9% in average.

4.4. Ablation Studies

Here we study the effects of various configurations of our SGSN, including different numbers of AGSBs, graph scattering layers and spectral channels at each scattering node.

Effects of AGSBs and graph scattering decomposition layers. We analyze the SGSN with various numbers (9-12) of AGSBs, where each AGSB use 1-3 layers of graph scattering decompositions. We test all the architectures on H3.6M for both short-term and long-term prediction, where the average MPJPEs are presented in Table 6. We see that the most effective motion prediction is achieved with 10 AGSBs and 2 scattering layers. The model performance keeps stable when we use fewer AGSBs or scattering layers. However, a larger number of AGSBs and scattering layers would cause over-fitting to damage the prediction.

Different spectral channels. Then, we investigate the effects of the numbers of spectral channels generated by each graph scattering node. We vary the numbers of channels from 1 to 5, and test these model variants on H3.6M; see the average MPJPEs in Table 7. We see that 3 channels lead to the lowest prediction errors. The model with only 1 channel indicates just using low-pass graph convolution, which causes large prediction erros due to over-smoothing. More than 3 channels introduce heavy parameters in the graph scattering decomposition and cause over-fitting.

4.5. Visualization

Prediction Visualization. We compare the synthesized

Table 6. Performance analysis of SGSNs with different numbers of graph scattering blocks, each of which uses different numbers of graph scattering layers.

	Scatter-layers			Average MPJPE					
Blocks	1	2	3	80	160	320	400	560	1000
9	✓			**10.5**	23.2	47.2	57.0	67.6	103.2
		✓		10.8	23.1	**47.0**	57.2	67.6	101.4
			✓	10.7	23.8	47.7	58.2	68.9	105.7
10	✓			10.6	**22.8**	47.2	57.3	68.2	103.6
		✓		10.6	22.9	**47.0**	**56.9**	**67.3**	**100.8**
			✓	11.4	24.3	47.7	57.8	69.6	105.5
11	✓			11.1	23.1	47.6	57.5	69.0	107.7
		✓		10.8	22.9	47.3	57.1	68.4	107.2
			✓	11.2	23.8	48.3	59.2	70.9	110.1
12	✓			11.3	23.2	47.4	57.3	68.8	108.3
		✓		11.4	23.3	47.6	57.6	69.3	110.0
			✓	12.0	24.5	48.7	59.7	72.1	113.5

Table 7. Comparison of SGSNs with different numbers of filtering channels on each non-leaf scattering feature.

	Average MAE					
channel numbers	80	160	320	400	560	1000
1	12.1	25.0	51.0	61.3	79.9	114.3
2	10.8	**22.9**	47.2	57.4	68.5	102.1
3	**10.6**	**22.9**	**47.0**	**56.9**	**67.3**	**100.8**
4	11.1	23.4	47.6	57.9	67.9	105.3
5	11.4	23.7	48.0	58.8	70.4	109.3

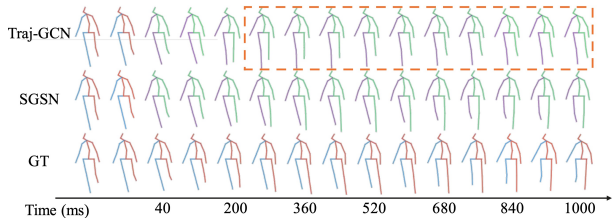

Figure 4. Prediction samples of different methods on action 'Walking' of H3.6M for long-term prediction.

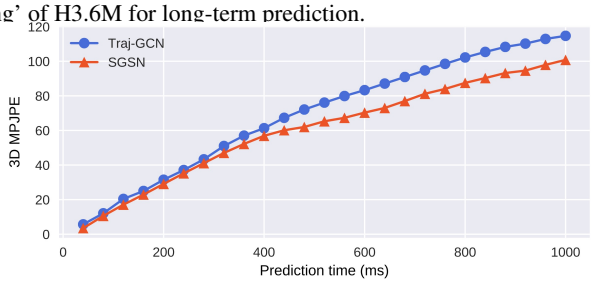

Figure 5. MPJPE of SGSN and Traj-GCN as a function of time on H3.6M. SGSN outperforms more in the long-term prediction.

samples of SGSN to those of Traj-GCN on H3.6M. Fig. 4 illustrates the future poses of 'Walking' in 1000 ms with the frame interval of 80 ms. Compared to the baseline, SGSN completes the action more accurately. The predictions of Traj-GCN start to suffer from large errors at the 280th ms (orange box); also, Traj-GCN converges to static 'mean poses' in long terms.

We further compare the SGSN to Traj-GCN by visualizing the average MPJPE at various timestamps; see Figure 5. At the short terms, the two models have similar

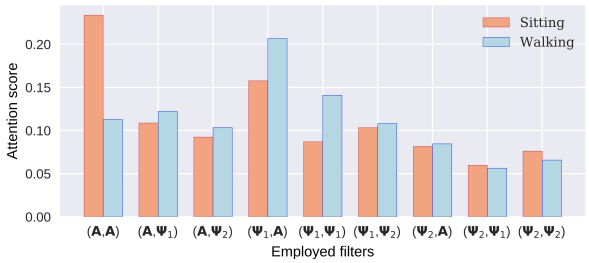

Figure 6. Learned spectrum attention scores on different actions.

MPJPEs since it is easy to model the short-term motions that have relatively consistent dynamics with the observations; at longer terms, SGSN outperforms Traj-GCN with large margins. Due to the complex dynamics far from the limited observations in long terms, it is harder to learn the highly nonlinear movements. Figure 5 reflects that SGSN preserves much clearer dynamics especially in long terms.

Attention Visualization. We investigate the learned importance of the spectrum for different actions; that is, the attention scores in AGSBs on different actions. For SGSN on H3.6M, we illustrate the attention scores calculated by the last AGSB on actions 'Sitting' and 'Walking'; see Figure 6, where the x-axis denotes the filter combinations in branches of the graph scattering decomposition. We see that, different actions lead to different attention distributions. For 'Sitting', the poses show slow movements, thus the low-pass features dominate the spectrum to stabilize pattern learning; as for 'Walking', poses keep large movements, thus some high-frequency information is preserved.

5. Conclusion

We propose a skeleton graph scattering network (SGSN), which leverages graph scattering to extract motion information from graph spectrum bands to achieve 3D skeleton-based human motion prediction. The core of our SGSN is the adaptive graph scattering block (AGSB), including i) graph scattering decomposition, which decomposes information into various graph spectrum bands and update the trainable features, and ii) graph spectrum attention, which aggregates those features via trainable attention weights. Extensive experiments reveal the superiority of our SGSN for both short-term and long-term motion prediction on Human3.6M, CMU Mocap and 3DPW datasets, respectively.

Acknowledgement

This work is supported by the National Key Research and Development Program of China (No. 2019YFB1804304), SHEITC (No. 2018-RGZN-02046), 111 plan (No. BP0719010), Shanghai "Science and Technology Innovation Plan" Key Research Program of Artificial Intelligence (No. 21511100900), STCSM (No. 18DZ2270700), State Key Laboratory of UHD Video and Audio Production and Presentation, and Huawei Cloud.

References

[1] Joakim Andén and Stéphane Mallat. Deep scattering spectrum. *IEEE Transactions on Signal Processing*, 62(16):4114–4128, 2014.

[2] Joan Bruna and Stephane Mallat. Invariant scattering convolution networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1872–1886, 2013.

[3] Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. Spectral networks and locally connected networks on graphs. In *International Conference on Learning Representations (ICLR)*, Apr. 2014.

[4] Yujun Cai, Lin Huang, Yiwei Wang, Tat-Jen Cham, Jianfei Cai, Junsong Yuan, Jun Liu, Xu Yang, Yiheng Zhu, Xiaohui Shen, Ding Liu, Jing Liu, and Nadia Magnenat Thalmann. Learning progressive joint propagation for human motion prediction. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *The European Conference on Computer Vision (ECCV)*, pages 226–242, 2020.

[5] G. Chen, X. Song, H. Zeng, and S. Jiang. Scene recognition with prototype-agnostic scene layout. *IEEE Transactions on Image Processing*, 29:5877–5888, 2020.

[6] Siheng Chen, Baoan Liu, Chen Feng, Carlos Vallespi-Gonzalez, and Carl Wellington. 3d point cloud processing and learning for autonomous driving. *IEEE Transactions on Signal Processing*, 2020.

[7] Qiongjie Cui, Huaijiang Sun, and Fei Yang. Learning dynamic relationships for 3d human motion prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[8] Hanjun Dai, Bo Dai, and Le Song. Discriminative embeddings of latent variable models for structured data. In *ICML*, June 2016.

[9] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. In *Advances in Neural Information Processing Systems (NeurIPS)*, Dec. 2016.

[10] Lifeng Fan, Wenguan Wang, Siyuan Huang, Xinyu Tang, and Song-Chun Zhu. Understanding human gaze communication by spatio-temporal graph reasoning. In *ICCV*, Oct. 2019.

[11] Katerina Fragkiadaki, Sergey Levine, Panna Felsen, and Jitendra Malik. Recurrent network models for human dynamics. In *ICCV*, pages 4346–4354, December 2015.

[12] Fernando Gama, Alejandro Ribeiro, and Joan Bruna. Diffusion scattering transforms on graphs. In *International Conference on Learning Representations (ICLR)*, May 2019.

[13] Fernando Gama, Alejandro Ribeiro, and Joan Bruna. Stability of graph scattering transforms. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, December 2019.

[14] Feng Gao, Guy Wolf, and Matthew Hirn. Geometric scattering for graph data analysis. In *ICML*, pages 2122–2131, June 2019.

[15] 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, Sept. 2018.

[16] Liangyan Gui, Kevin Zhang, Yuxiong Wang, Xiaodan Liang, Jose Moura, and Manuela Veloso. Teaching robots to predict human motion. In *IEEE International Conference on Intelligent Robots and Systems*, Oct. 2018.

[17] Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems (NeurIPS)*, Dec. 2017.

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[19] Guyue Hu, Bo Cui, and Shan Yu. Skeleton-based action recognition with synchronous local and non-local spatio-temporal learning and frequency attention. In *ICME*, July 2019.

[20] Yue Hu, Siheng Chen, Ya Zhang, and Xiao Gu. Collaborative motion prediction via neural motion message passing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[21] Yingfan Huang, Huikun Bi, Zhaoxin Li, Tianlu Mao, and Zhaoqi Wang. Stgat: Modeling spatial-temporal interactions for human trajectory prediction. In *ICCV*, pages 6272–6281, 2019.

[22] Vassilis N. Ioannidis, Siheng Chen, and Georgios B. Giannakis. Pruned graph scattering transforms. In *International Conference on Learning Representations (ICLR)*, Apr. 2020.

[23] C. Ionescu, D. Papava, V. Olaru, and C. 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, 2014.

[24] Ashesh Jain, Amir Zamir, Silvio Savarese, and Ashutosh Saxena. Structural-rnn: Deep learning on spatio-temporal graphs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5308–5317, June 2016.

[25] Wengong Jin, Regina Barzilay, and Tommi Jaakkola. Junction tree variational autoencoder for molecular graph generation. In *ICML*, pages 2323–2332, 2018.

[26] Thomas Kipf, Ethan Fetaya, Kuan-Chieh Wang, Max Welling, and Richard Zemel. Neural relational inference for interacting systems. In *ICML*, pages 2688–2697, 2018.

[27] Thomas Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR)*, Apr. 2017.

[28] Vineet Kosaraju, Amir Sadeghian, Roberto Martín-Martín, Ian Reid, S Hamid Rezatofighi, and Silvio Savarese. Social-bigat: Multimodal trajectory forecasting using bicycle-gan and graph attention networks. *arXiv preprint arXiv:1907.03395*, 2019.

[29] S. Lee, J. Lim, and I. H. Suh. Progressive feature matching: Incremental graph construction and optimization. *IEEE Transactions on Image Processing*, 29:6992–7005, 2020.

[30] A. Lehrmann, P. Gehler, and S. Nowozin. Efficient nonlinear markov models for human motion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1314–1321, June 2014.

[31] Chen Li, Zhen Zhang, Wee Sun Lee, and Gim Hee Lee. Convolutional sequence to sequence model for human dynamics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[32] Jiachen Li, Fan Yang, Masayoshi Tomizuka, and Chiho Choi. Evolvegraph: Multi-agent trajectory prediction with dynamic relational reasoning. *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[33] Maosen Li, Siheng Chen, Xu Chen, Ya Zhang, Yanfeng Wang, and Qi Tian. Actional-structural graph convolutional networks for skeleton-based action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[34] Maosen Li, Siheng Chen, Yangheng Zhao, Ya Zhang, Yanfeng Wang, and Qi Tian. Dynamic multiscale graph neural networks for 3d skeleton based human motion prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[35] Yujia Li, Daniel Tarlow, Marc Brockschmidt, and Richard Zemel. Gated graph sequence neural networks. In *International Conference on Learning Representations (ICLR)*, May 2016.

[36] Xiankai Lu, Wenguan Wang, Martin Danelljan, Tianfei Zhou, Jianbing Shen, and Luc Van Gool. Video object segmentation with episodic graph memory networks. In *The European Conference on Computer Vision (ECCV)*, pages 661–679, 2020.

[37] Wei Mao, Miaomiao Liu, and Mathieu Salzmann. History repeats itself: Human motion prediction via motion attention. In *The European Conference on Computer Vision (ECCV)*, Aug. 2020.

[38] Wei Mao, Miaomiao Liu, Mathieu Salzmann, and Hongdong Li. Learning trajectory dependencies for human motion prediction. In *ICCV*, Oct. 2019.

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

[40] Yimeng Min, Frederik Wenkel, and Guy Wolf. Scattering gcn: Overcoming oversmoothness in graph convolutional networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 14498–14508, Dec. 2020.

[41] Yimeng Min, Frederik Wenkel, and Guy Wolf. Geometric scattering attention networks. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8518–8522, 2021.

[42] Mathias Niepert, Mohamed Ahmed, and Konstantin Kutzkovl. Learning convolutional neural networks for graphs. In *ICML*, June 2016.

[43] Chao Pan, Siheng Chen, and Antonio Ortega. Spatio-temporal graph scattering transform. In *International Conference on Learning Representations (ICLR)*, May 2021.

[44] Vladimir Pavlovic, James M Rehg, and John MacCormick. Learning switching linear models of human motion. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2001.

[45] Siyuan Qi, Wenguan Wang, Baoxiong Jia, Jianbing Shen, and Song-Chun Zhu. Learning human-object interactions by graph parsing neural networks. In *The European Conference on Computer Vision (ECCV)*, pages 401–417, 2018.

[46] M. Rizkallah, X. Su, T. Maugey, and C. Guillemot. Geometry-aware graph transforms for light field compact representation. *IEEE Transactions on Image Processing*, 29:602–616, 2020.

[47] Lei Shi, Yifan Zhang, Jian Cheng, and Hanqing Lu. Skeleton-based action recognition with directed graph neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[48] Laurent Sifre and Stephane Mallat. Rotation, scaling and deformation invariant scattering for texture discrimination. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1233–1240, June 2013.

[49] Shazia Tabassum, Fabiola SF Pereira, Sofia Fernandes, and João Gama. Social network analysis: An overview. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 8(5), 2018.

[50] Graham Taylor and Geoffrey Hinton. Factored conditional restricted Boltzmann machines for modeling motion style. In *ICML*, June 2009.

[51] Graham Taylor, Geoffrey Hinton, and Sam Roweis. Modeling human motion using binary latent variables. In *Advances in Neural Information Processing Systems (NeurIPS)*, December 2007.

[52] Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *International Conference on Learning Representations (ICLR)*, Apr. 2018.

[53] Timo von Marcard, Roberto Henschel, Michael J. Black, Bodo Rosenhahn, and Gerard Pons-Moll. Recovering accurate 3d human pose in the wild using imus and a moving camera. In *The European Conference on Computer Vision (ECCV)*, September 2018.

[54] Jacob Walker, Kenneth Marino, Abhinav Gupta, and Martial Hebert. The pose knows: Video forecasting by generating pose futures. In *ICCV*, pages 3332–3341, Oct. 2017.

[55] Wenguan Wang, Hailong Zhu, Jifeng Dai, Yanwei Pang, Jianbing Shen, and Ling Shao. Hierarchical human parsing with typed part-relation reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[56] Sijie Yan, Yuanjun Xiong, and Dahua Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *AAAI*, Feb. 2018.

[57] Jiaxuan You, Bowen Liu, Rex Ying, Vijay Pande, and Jure Leskovec. Graph convolutional policy network for goal-directed molecular graph generation. *arXiv preprint arXiv:1806.02473*, 2018.

[58] Cunjun Yu, Xiao Ma, Jiawei Ren, Haiyu Zhao, and Shuai Yi. Spatio-temporal graph transformer networks for pedestrian trajectory prediction. In *The European Conference on Computer Vision (ECCV)*, pages 507–523. Springer, 2020.

[59] J. Zhang, F. Shen, X. Xu, and H. T. Shen. Temporal reasoning graph for activity recognition. *IEEE Transactions on Image Processing*, 29:5491–5506, 2020.

[60] X. Zhang, C. Xu, X. Tian, and D. Tao. Graph edge convolutional neural networks for skeleton-based action recognition. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–14, 2019.

[61] C. Zheng, L. Pan, and P. Wu. Multimodal deep network embedding with integrated structure and attribute information. *IEEE Transactions on Neural Networks and Learning Systems*, 31(5):1437–1449, 2020.

[62] Dongmian Zou and Gilad Lerman. Graph convolutional neural networks via scattering. *Applied and Computational Harmonic Analysis*, 49(3):1046–1074, 2020.