Learning Progressive Joint Propagation for Human Motion Prediction

Yujun Cai[1], Lin Huang[3], Yiwei Wang[5], Tat-Jen Cham[1], Jianfei Cai[1,2], Junsong Yuan[3], Jun Liu[6], Xu Yang[1], Yiheng Zhu[4], Xiaohui Shen[4], Ding Liu[4], Jing Liu[4], and Nadia Magnenat Thalmann[1]

[1] Nanyang Technological University, Singapore.
{yujun001,s170018}@e.ntu.edu.sg,{astjcham,nadiathalmann}@ntu.edu.sg
[2] Monash University, Australia jianfei.cai@monash.edu
[3] State University of New York at Buffalo University, USA
{lhuang27,jsyuan}@buffalo.edu
[4] ByteDance Research
{shenxiaohui,yiheng.zhu,liuding,jing.liu}@ bytedance.com
[5] National University of Singapore wangyw_seu@foxmail.com
[6] SUTD, Singapore jun_liu@sutd.edu.sg

Abstract. Despite the great progress in human motion prediction, it remains a challenging task due to the complicated structural dynamics of human behaviors. In this paper, we address this problem in three aspects. First, to capture the long-range spatial correlations and temporal dependencies, we apply a transformer-based architecture with the global attention mechanism. Specifically, we feed the network with the sequential joints encoded with the temporal information for spatial and temporal explorations. Second, to further exploit the inherent kinematic chains for better 3D structures, we apply a progressive-decoding strategy, which performs in a central-to-peripheral extension according to the structural connectivity. Last, in order to incorporate a general motion space for high-quality prediction, we build a memory-based dictionary, which aims to preserve the global motion patterns in training data to guide the predictions. We evaluate the proposed method on two challenging benchmark datasets (Human3.6M and CMU-Mocap). Experimental results show our superior performance compared with the state-of-the-art approaches.

Keywords: 3D motion prediction, transformer network, progressive decoding, dictionary module

1 Introduction

Human motion prediction aims to forecast a sequence of future dynamics based on an observed series of human poses. It has extensive applications in robotics, computer graphics, healthcare and public safety [20, 24, 26, 41, 40], such as human robot interaction [25], autonomous driving [35] and human tracking [18].

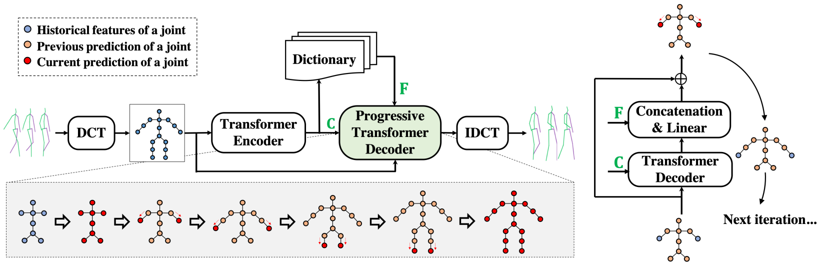

Historical features of a joint
Previous prediction of a joint
Current prediction of a joint
Dictionary
F
DCT
Transformer Encoder
C
Progressive Transformer Decoder
IDCT
F
Concatenation & Linear
C
Transformer Decoder
Next iteration...

Fig. 1: Left: **Overview of our proposed network architecture for 3D motion prediction.** Given a sequence of 3D human poses, we apply Discrete Cosine Transform (DCT) to encode the temporal information of each joint into frequency coefficients. The DCT coefficients of sequential joints are then fed into the transformer-based architecture for progressive predictions. Additionally, we use memory-based dictionary to incorporate the global motion knowledge into the model. We apply inverse DCT (IDCT) to convert the predicted DCT coefficients back to the temporal domain. Right: **Progressive decoding strategy.** We predict the DCT coefficients of the target joints progressively, which performs in a central-to-peripheral manner in accordance with the kinematic chains (bottom left), with the encoded context feature **C** and the dictionary information **F**.

Due to the inherent temporal nature of this task, many existing methods [33, 14, 43] resort to recurrent neural networks (RNN) and their variants for temporal modeling. However, simply relying on the temporal coherence is not enough, since the bio-mechanical dynamics of human behavior are extremely complicated, which not only correspond to temporal smoothness, but also highly relate to spatial joint dependencies. To address this issue, previous work attempted to embed the spatial configurations into the modeling space, so as to enhance the validity of the 3D structures. For instance, Li *et al.* [28] relied on a convolutional filter to capture the dependencies across the spatial and temporal domains. The range of such dependencies, however, is strongly limited by the size of the convolutional kernel. Mao *et al.* [32] applied Discrete Cosine Transform (DCT) to encode temporal information and designed a Graph Neural Network (GNN) to model spatial correlations. Although achieving good results, it forgoes the prevailing sequential decoding architecture and thus cannot explicitly leverage context features that may lead to further improvement.

Based on these observations, we aim to efficiently capture long-range spatial-temporal dependencies while also incorporating the advantage of sequential modeling. In particular, motivated by substantial performance gains achieved by transformer-based networks [42, 12, 37, 11] in Natural Language Processing (NLP), we propose to apply the transformer architecture to simultaneously model the spatial and temporal dependencies of human motion. A key benefit of the transformer is that it can capture the global dependencies among the input and output sequences with the help of the attention mechanism. Note that instead of directly feeding sequential poses into the network, following [32] we encode the temporal trajectories of each joint into the frequency domain, before transferring these embedded temporal features to the network. In this way, the model essentially works in the trajectory domain while simultaneously drawing global attention among different joints, as well as between the input historical trajectories and the output predictions.

Moreover, we would like to point out that simply using the transformer for motion prediction does not fully exploit the kinematic chains of body skeletons, yet these are important since they underlie the motions in human behavior. For instance, absolute displacement of a wrist is often mediated by initial movement of the shoulder, followed by the elbow. Inspired by spatial explorations in 3D human pose estimation [27, 8], we propose to exploit the structural configurations by predicting the joint trajectories progressively in a central-to-peripheral manner. More precisely, as depicted in Figure 1 (bottom left), we first estimate the future dynamics of the central body as seed points, and then sequentially propagate the joint predictions based on the kinematic connections.

In addition, the typical approach for most encoder-decoder frameworks, when decoding the motion predictions, is to mainly focus on the single source video that is being processed. This may not be the optimal, since partial motions of many actions follow certain types of general patterns (*e.g.* walking feet, waving hands and bending knees), which may appear in multiple videos with similar but not identical context. Thus, we further propose to incorporate a general motion space into the predictions. Specially, inspired by the memory scheme that is widely utilized in Question Answering (QA) [39, 46], we design a memory-based dictionary to store the common actions across different videos in training data. From the dictionary, we can query the historical motions $\mathbf{C}$ and construct the future dynamics $\mathbf{F}$ to guide the predictions, as shown in Figure 1 (left).

In summary, our contributions of this work are threefold:

- We propose to leverage the transformer-based architecture to simultaneously exploit the spatial correlations and the temporal smoothness of human motion, by treating the sequential joints with the encoded temporal features as the input of the network.
- To further exploit the structural connectivity of human skeletons, we deploy a progressive decoding strategy to predict the future joint dynamics in a central-to-peripheral manner in accordance with the kinematic chains of a human body.
- To incorporate the general motion space for high quality results, we build a memory-based dictionary to guide the predictions, which preserves the correspondences between the historical motion features and the representative future dynamics.

We conducted comprehensive experiments on two widely-used benchmarks for human motion prediction: the Human3.6M dataset and the CMU-Mocap dataset, and our proposed method improves state-of-the-art performance in both datasets.

2 Related Work

Human motion prediction. Human motion predictions have been extensively studied in the past few years. Early approaches tackled this problem with Hidden Markov Model [4], linear dynamics system [36], and Gaussian Process latent variable models [44], *etc.*, which commonly suffer from the computational resources and can be easily stuck in non-periodical actions. Recently, due to the success of the sequence-to-sequence inference, RNN-based architectures have been widely used in state-of-the-art approaches [17, 14, 5, 2, 45]. For instance, Fragkiadaki *et al.* [14] proposed a Encoder- Recurrent-Decoder (ERD) framework, which maps pose data into a latent space and propagates it across the temporal domain through LSTM cells. To facilitate more realistic human motions, Gui *et al.* [19] introduced an adversarial training and Wang *et al.* [43] employed imitation learning into the sequential modeling. While pushing the boundaries of the motion predictions, many of these RNN-based models directly use a fully-connected layer to learn the representation of human pose, which to some extent overlook the inherent spatial configurations of human body.

Structural-aware Prediction. Several recent works [32, 22, 27, 1, 29, 31, 30] tried to embed the spatial articulations of human body to enhance the validity of the 3D structures. For example, Jain *et al.* [22] proposed to encode the spatial and temporal structure of the pose via a manually designed spatio-temporal graph. Although taking structural configurations into account, these graphs, however, have limited flexibility for discovering long-range interactions between different joints. To address this issue, Mao *et al.* [32] leveraged GNN-based architectures, where all joints are linked together for full explorations. While achieving good results, this method does not explicitly utilize the kinematic chains of body structure. In contrast, to leverage the long-range connections while also exploiting the structural connectivity of body skeletons, we apply a transformer-based architecture to capture the long-range spatial and temporal dependencies of human motion. Additionally, we propose to progressively propagate the joint predictions in a central-to-peripheral manner to further exploit the spatial configurations.

Transformer Network. The transformer has become the state-of-the-art approach in Natural Language Processing (NLP), with extensive architectures such as Bert [12], GPT [37], XLNet [11]. Recently, it is also investigated in Computer Vision, such as Image GPT[10] and Object Detection[9]. Compared with the traditional recurrent neural network (RNN) that explicitly models the compatibility of adjacent tokens, the transformer takes an entirely different global attention mechanism, which allows to capture the long-term dependencies between the input and the output sequences. Inspired from this, we propose to lever-

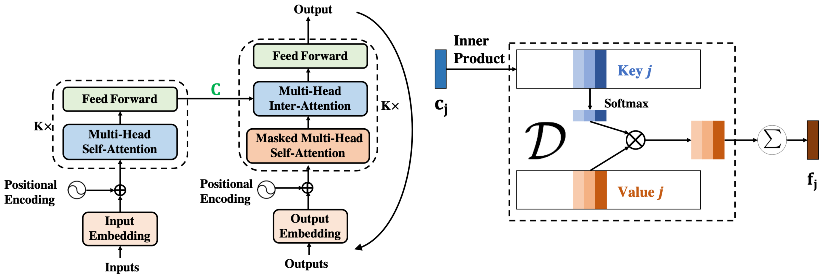


Fig. 2: Left: **The architecture of the conventional transformer [42]**, where the far left side is the transformer-encoder that encodes the input into context features **C**, while the relatively right side is the decoder that recursively generates the output sequence with the encoded context features. Right: **Query and reconstruction procedure in memory-based dictionary for joint j.** The input is the observed context features encoded from the historical trajectories, and the output is the constructed features for predicting future dynamics.

age transformer-based architecture to capture the spatio-temporal correlations of human motion. Particularly, instead of directly taking the sequential poses as the input, we follow [32] to apply DCT to encode the trajectory of each joint. The sequential joints with encoded temporal patterns are then fed into the network for global explorations.

3 Methodology

3.1 Overview

Figure 1 gives an overview of our proposed network architecture. Given a series of human motion poses $\mathbf{X}_{1:T} = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_T]$, where $\mathbf{x}_t$ denotes the pose at frame t, our target is to generate a corresponding prediction $\mathbf{X}_{T+1:T+T_f}$ for the future T_f frames. To achieve this goal, different from most existing work that employ RNN-based architectures to model the temporal information of human motions, we leverage the transformer network to capture the long-range spatial and temporal correlations of human motions with the help of the attention mechanism. Specifically, we apply Discrete Cosine Transform (DCT) to encode the temporal information of each joint into frequency space and feed the network with the sequential joints with encoded temporal patterns. Additionally, motivated by the inherent structural connectivity of body skeletons, we explicitly stagger the decoding into predefined progressive steps. This is performed in a central-to-peripheral manner according to the kinematic chains, with a total update at the final stage (see Figure 1 bottom-left). To create a generalized and full-spectrum human motion space across the different videos in training data, we further introduce a dictionary as an auxiliary decoder to enhance the prediction quality. The whole model is trained in an end-to-end manner with backpropagation. Next, we describe the individual components in detail.

3.2 Revisiting Transformer

The transformer architecture is a core pillar underpinning of many state-of-the-art methods in Natural Language Processing (NLP) since [12], showing superior performance compared to conventional RNN-based structures. This is mainly because RNNs have difficulties in modelling long-term dependencies, while the transformer overcomes this limitation by leveraging the global attention mechanism to draw the dependencies between the entire input and output sequences, without regard to their distances. In particular, as shown in Figure 2 (left), the transformer employs an attention-based encoder-decoder framework, where the encoder applies self-attention to extract the useful context from the input sequence, and the decoder consecutively produces the prediction based on the global dependencies between the context features and the previous output sequences. To make use of the sequential order, the transformer additionally inserts a "positional encoding" module to the embeddings at the bottom of the encoder and decoder stacks, assigning each dimension of each token with a unique encoded value.

3.3 Transformer for Pose Prediction

Motivated by the substantial performance gain induced by the transform architecture in NLP, we propose to solve the pose prediction problem with the help of a transformer-based network. A straightforward way is to take the human pose at each time step as corresponding to a "word" in the machine translation task, and then predict the pose at the next time step as akin to predicting the next word. However, doing so blindly ignores the spatial dependencies between joints, which have proven to be highly effective in state-of-the-art methods for pose estimation [8, 27, 29, 13, 15, 16] and pose prediction [28, 32].

To tackle this issue and leverage both the spatial and temporal dependencies of human poses, following [32] we encode the temporal information of each joint into frequency space. Specifically, we first replicate the last pose $\mathbf{x}_T$ for T_f times to generate a temporal sequence of length $T + T_f$, and then compute the DCT coefficients of each joint. In this way, the task becomes that of generating an output sequence $\hat{\mathbf{X}}_{1:T+T_f}$ from an input sequence $\mathbf{X}_{1:T+T_f}$, with our true objective to predict $\mathbf{X}_{T+1:T+T_f}$.

We then feed the obtained DCT coefficients into the transformer-based network, so as to capture the spatial dependencies with the help of the attention mechanism. A key benefit of this design is that the network essentially works in the trajectory space while simultaneously modeling the spatial dependencies among the input and the output joint sequences. Moreover, thanks to the positional encoding, the joint index can be explicitly injected into motion features, allowing the network to not only learn the trajectory of each joint, but also incorporates the joint identities into this process. To encourage smoothness between the input and output trajectories, we also apply the residual scheme at the end of the decoding (see Figure 1 right).

3.4 Progressive Joint Propagation

The conventional transformer decoder works auto-regressively during inference, that is, conditioning each output word on previously generated outputs. In terms of pose prediction, we observe that human motion is naturally propagated sequentially based on the kinematic chains of body skeletons. For instance, a person may initiate movement of the left shoulder, which then drives the movement of the left elbow and eventually that of the left wrist.

Motivated by this, we propose to progressively express the 3D pose predictions in a similar manner. In particular, as shown in Figure 1 bottom-left, we treat the central eight joints as the seed joints, and estimate their future motions first, based on their historical motions. Next we sequentially propagate the joint predictions from center to periphery, according to the structural connectivity of the body skeleton. Figure 1 (right) depicts the details of the progressive decoding process: we iteratively predict the residual DCT coefficients of the joints, given the encoded context feature $\mathbf{C} = [\mathbf{c}_1, \mathbf{c}_2, ..., \mathbf{c}_J]$ and the auxiliary information $\mathbf{F} = [\mathbf{f}_1, \mathbf{f}_2, ..., \mathbf{f}_J]$ from the dictionary, where J is the number of joints. Mathematically, for the s^{th} progressive decoding, we formulate the computational process as:

$$\mathbf{X}_{in}^{(s)} = [\hat{\mathbf{X}}_p; \mathbf{X}_h^{(s)}], \tag{1}$$

$$\hat{\mathbf{X}}_{out}^{(s)} = \mathbf{X}_h^{(s)} + \text{Decoder}(\mathbf{X}_{in}^{(s)}, \mathbf{C}, \mathbf{F}^{(s)}), \tag{2}$$

$$\hat{\mathbf{X}}_p = [\hat{\mathbf{X}}_p; \hat{\mathbf{X}}_{out}^{(s)}], \tag{3}$$

where $\mathbf{X}_{in}^{(s)}$ denotes the input of the progressive decoder at stage s, which is a combination of the previously predicted joint sequence $\hat{\mathbf{X}}_p$ and the historical motion of the target joints $\mathbf{X}_h^{(s)}$. $\hat{\mathbf{X}}_{out}^{(s)}$ is the output of the s^{th} decoder, which summarizes the historical motions $\mathbf{X}_h^{(s)}$ and the generated residual DCT coefficients. $\mathbf{F}^{(s)}$ refers to the auxiliary information used for stage s, containing the guided future dynamics of the target joints at stage s. Note that each time we generate the estimation of certain joints $\hat{\mathbf{X}}_{out}$, we merge them into the previously predicted joint sequence $\hat{\mathbf{X}}_p$ for the next iteration. In this way, we ensure that the estimation is propagated along the structural connectivity and the entire body prediction is constructed in the order of the kinematic chains via the progressive joint propagation. To further refine the full-body prediction, we add a total updating stage at the end of the progressive decoder.

3.5 Dictionary

One potential limitation of the encoder-decoder framework (*e.g.* the transformer, the RNN-based modeling) is that the decoder mainly focuses on one input sequence that is currently being processed while decoding. However, partial motions of many actions follow certain common patterns (*e.g.* walking feet, bending knees), which may appear in multiple videos with similar but not identical context features. To incorporate this generalized motion prior knowledge for better prediction quality, we design a memory-based dictionary to guide the motion prediction, inspired by the memory scheme [39, 46, 47] leveraged to preserve a knowledge base for comprehensive understanding. The dictionary is built to store the full spectrum of correspondences between the observed motion features and the representative future dynamics of each joint across different videos in training data. Note that although the correspondences are mainly constructed for each joint, the global motion of the full body is also taken into account, due to the self-attention mechanism of the transformer encoder.

We propose to learn this dictionary via query and construction processes. This dictionary $\mathcal{D}$ is defined as:

$$\mathcal{D} = \left\{ \left(\mathbf{D}_j^{key}, \mathbf{D}_j^{value} \right) \mid j = 1, 2, \ldots, J \right\}, \tag{4}$$

where $\mathbf{D}_j^{key}, \mathbf{D}_j^{value} \in \mathbb{R}^{N \times M}$ are the key and the value matrices respectively for joint j, N is the number of memory cells / clusters for each joint, and M is the dimension of a feature stored in a memory cell. As shown in Figure 2 (right), the key matrix is used to score an observed motion query of each memory cell so as to better combine the value elements. Mathematically, given an encoded context motion feature $\mathbf{c}_j \in \mathbb{R}^M$ of joint j, the query process can be written as:

$$\mathbf{q}_j = \text{softmax}\left(\mathbf{D}_j^{key} \mathbf{c}_j \right), \tag{5}$$

where $\mathbf{q}_j$ is the query result in the memory network for joint j. Then we define the construction process as

$$\mathbf{f}_j = \left(\mathbf{D}_j^{value} \right)^T \mathbf{q}_j, \tag{6}$$

where $\mathbf{f}_j$ is the feature vector constructed by the memory network for joint j, representing the future dynamics summarized from the learned generalized motion space. In our implementation, we set N as 100 and M as 512.

3.6 Training Strategy

Since the construction of the dictionary relies on the context features from the transformer encoder, we first train the whole network without the dictionary module. Then we learn the dictionary and finetune the whole model subsequently. For the first stage, we employ the following loss function, which aims to minimize the differences between the predicted sequential poses converted from the DCT coefficients and the corresponding ground truth:

$$L = \sum_{s=1}^{S} \lambda_s \sum_{j \in \mathcal{J}_s} \sum_{t=1}^{T+T_f} \mathcal{P}(\hat{x}_{j,t}, x_{j,t}^{GT}) \tag{7}$$

Here $\hat{x}_{j,t}$ refers to the prediction of the j^{th} joint in frame t and $x^{GT}_{j,t}$ is the corresponding ground truth. S is the number of the progressive decoding stages, $\mathcal{J}_s$ is the set of joints to be predicted at stage s, λ_s is the weight for each stage, T and T_f represent the length of observed frames and predicted frames respectively. $\mathcal{P}$ is a distance function that uses L_1 loss for joint angle representation and L_2 loss for 3D joint coordinates, both of which are typical representations in motion prediction literature. Following [32], we sum the errors over both future and observed time steps, to provide more signals for the learning process.

Having trained the transformer encoder and decoder, we next learn the dictionary module. Specifically, given the observed motion features $\mathbf{C}$, we seek to query for similar historical motion patterns and produce the auxiliary information $\mathbf{F}$ containing future dynamics for each joint. The auxiliary information is typically concatenated with the features generated from the decoder and sent into the final linear layer of the progressive decoder (see Figure 1 right), so as to produce the prediction of each joint. Formally, we train the dictionary by penalizing the difference between the produced joint predictions and the corresponding ground truth:

$$L_d = \sum_{j=1}^{J} \sum_{t=1}^{T+T_f} \mathcal{P}(\hat{x}_{j,t}, x^{GT}_{j,t}) \tag{8}$$

where J is the number of joints.

Finally, we finetune the whole model in an end-to-end manner, with the same loss function (Equation (7)) as proposed in the first training stage.

4 Experiments

4.1 Implementation details

In our experiments, we chose the conventional transformer proposed in [42], with 8 headers and 512 hidden units for each module. Both the encoder and the progressive decoder contain a stack of $K = 4$ identical layers. To accelerate the convergence, we applied the scheduled sampling scheme [3] during training, which randomly replaces part of the previous joint predictions with the ground truth in the input to the progressive decoder. The whole model was implemented within the PyTorch framework. For the first training stage described in Section 3.6, we set $\lambda_s = 1$, and trained for 40 epochs with the Adam optimizer [23]. The learning rate started from 5e-4, with a shrink factor of 0.96 applied every two epochs. For the second stage, we learned the dictionary for 20 epochs with the learning rate of 5e-4. Finally, the whole network was finetuned in an end-to-end manner, using a relatively small learning rate of 5e-5. All experiments were conducted on a single NVIDIA Titan V GPU, with a batch size of 128 for both training and evaluation.

Table 1: Short-term prediction results in Mean Angle Error (MAE) on Human3.6M for the main actions due to limited space. The best result is marked in bold. The full table can be found in supplementary.

	Walking				Eating				Smoking				Directions				Greeting				Average			
milliseconds	80	160	320	400	80	160	320	400	80	160	320	400	80	160	320	400	80	160	320	400	80	160	320	400
zero-velocity [33]	0.39	0.68	0.99	1.15	0.27	0.48	0.73	0.86	0.26	0.48	0.97	0.95	0.39	0.59	0.79	0.89	0.54	0.89	1.30	1.49	0.40	0.78	1.07	1.21
Residual sup. [33]	0.28	0.49	0.72	0.81	0.23	0.39	0.62	0.76	0.33	0.61	1.05	1.15	0.26	0.47	0.72	0.84	0.75	1.17	1.74	1.83	0.36	0.67	1.02	1.15
convSeq2Seq [28]	0.33	0.54	0.68	0.73	0.22	0.36	0.58	0.71	0.26	0.49	0.96	0.92	0.39	0.60	0.80	0.91	0.51	0.82	1.21	1.38	0.38	0.68	1.01	1.13
AGED w/o adv [19]	0.28	0.42	0.66	0.73	0.22	0.35	0.61	0.74	0.30	0.55	0.98	0.99	0.26	0.46	0.71	0.81	0.61	0.95	1.44	1.61	0.32	0.62	0.96	1.07
AGED w/ adv [19]	0.22	0.36	0.55	0.67	0.17	**0.28**	0.51	0.64	0.27	0.43	**0.82**	0.84	0.23	0.39	0.63	0.69	0.56	0.81	1.30	1.46	0.31	0.54	0.85	0.97
Imitation [43]	0.21	0.34	0.53	0.59	0.17	0.30	0.52	0.65	0.23	0.44	0.87	0.85	0.27	0.46	0.81	0.89	0.43	0.75	1.17	1.33	0.31	0.57	0.90	1.02
GNN [32]	0.18	0.31	**0.49**	0.56	0.16	0.29	0.50	0.62	0.22	0.41	0.86	0.80	0.26	0.45	0.71	0.79	0.36	0.60	0.95	1.13	0.27	0.51	0.83	0.95
ours	**0.17**	**0.30**	0.51	**0.55**	**0.16**	0.29	**0.50**	**0.61**	**0.21**	**0.40**	0.85	**0.78**	**0.22**	**0.39**	**0.62**	**0.69**	**0.34**	**0.58**	**0.94**	**1.12**	**0.25**	**0.49**	**0.83**	**0.94**

Table 2: Short-term prediction results in Mean Per Joint Position Error (MPJPE) on Human3.6M for the main actions due to limited space. The best result is marked in bold. A $3D$ suffix to a method indicates that the method was directly trained on 3D joint positions. Otherwise, the results were obtained by converting the joint angle to 3D positions. The best result is marked in bold and the full table can be found in supplementary.

	Walking				Eating				Smoking				Directions				Greeting				Average			
milliseconds	80	160	320	400	80	160	320	400	80	160	320	400	80	160	320	400	80	160	320	400	80	160	320	400
Residual sup. [33]	21.7	38.1	58.9	68.8	15.1	28.6	54.8	67.4	20.8	39.0	66.1	76.1	27.9	44.8	63.5	78.2	29.3	56.0	110.2	125.6	27.9	51.6	88.9	103.4
Residual sup. 3D [33]	23.8	40.4	62.9	70.9	17.6	34.7	71.9	87.7	19.7	36.6	61.8	73.9	36.5	56.4	81.5	97.3	37.9	74.1	1390	158.8	30.8	57.0	99.8	115.5
convSeq2Seq [28]	21.8	37.5	55.9	63.0	13.3	24.5	48.6	60.0	15.4	25.5	39.3	44.5	26.7	43.3	59.0	72.4	30.4	58.6	110.0	122.8	24.9	44.9	75.9	88.1
convSeq2Seq 3D [28]	17.1	31.2	53.8	61.5	13.7	25.9	52.5	63.3	11.1	21.0	33.4	38.3	22.0	37.2	59.6	73.4	24.5	46.2	90.0	103.1	19.6	37.8	68.1	80.2
GNN [32]	11.1	19.0	32.0	39.1	9.2	19.5	40.3	48.9	9.2	16.6	26.1	29.0	11.2	23.2	52.7	64.1	14.2	**27.7**	67.1	82.9	13.5	27.0	54.2	65.0
GNN 3D [32]	8.9	15.7	29.2	**33.4**	8.8	18.9	39.4	47.2	7.8	14.9	25.3	28.7	12.6	24.4	48.2	58.4	14.5	30.5	74.2	89.0	12.1	25.0	51.0	61.3
Ours	9.6	18.0	33.1	39.1	9.1	19.5	40.2	48.8	7.2	14.2	24.7	29.7	**9.3**	**22.0**	51.6	63.2	15.4	30.7	71.8	82.8	11.9	26.1	53.2	64.5
Ours 3D	**7.9**	**14.5**	**29.1**	34.5	**8.4**	**18.1**	**37.4**	**45.3**	**6.8**	**13.2**	**24.1**	**27.5**	11.1	22.7	**48.0**	**58.4**	**13.2**	28.0	**64.5**	**77.9**	**10.7**	**23.8**	**50.0**	**60.2**

Table 3: Short and long-term prediction of 3D joint positions in MPJPE on CMU-Mocap dataset.

	Basketball					Basketball Signal					Directing Traffic					Jumping					Running				
milliseconds	80	160	320	400	1000	80	160	320	400	1000	80	160	320	400	1000	80	160	320	400	1000	80	160	320	400	1000
Residual sup. 3D [33]	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
GNN 3D [32]	14.0	25.4	49.6	61.4	106.1	3.5	6.1	**11.7**	**15.2**	**53.9**	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
Ours 3D	**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**

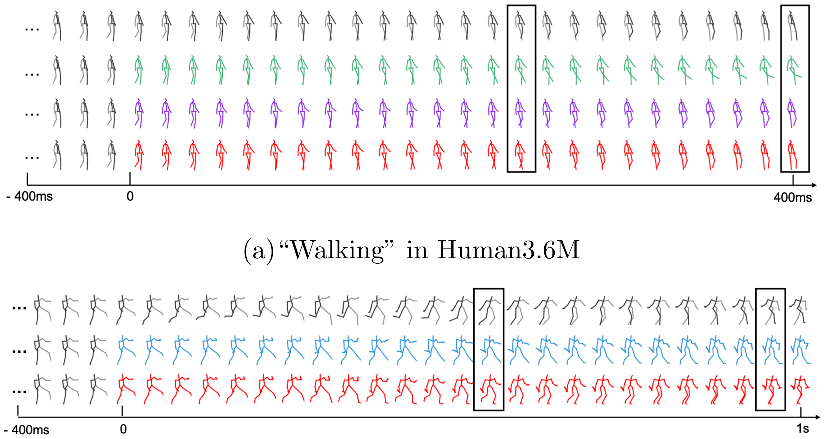
	Soccer					Walking					Washwindow					Average				
milliseconds	80	160	320	400	1000	80	160	320	400	1000	80	160	320	400	1000	80	160	320	400	1000
Residual sup. 3D [33]	20.3	39.5	71.3	84	129.6	8.2	13.7	21.9	24.5	52.2	8.4	15.8	29.3	**35.4**	**61.1**	16.8	30.5	54.2	63.6	99.0
GNN 3D [32]	11.3	21.5	44.2	55.8	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
Ours 3D	**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**

4.2 Datasets and Evaluation Metrics

We evaluated our method on two publicly available datasets: the Human3.6M dataset [21] and the CMU-Mocap dataset[7] for 3D human motion prediction.

Human3.6M: The Human3.6M dataset [21] is a large-scale and commonly used dataset for human motion prediction, which consists of 7 subjects performing 15 actions, such as "Walking", "Sitting" and "Smoking". Following the standard setup in [33, 28, 19, 43, 32], the global rotations, translations and constant joints were excluded from our experiments. We down-sampled each sequence to 25 frames per second and applied the evaluation on subject 5 (S5), as proposed in [32, 28, 33].

[7] Available at http://mocap.cs.cmu.edu/



(a) "Walking" in Human3.6M

(b) "Soccer" in CMU-Mocap

Fig. 3: (a) **Qualitative comparison of long-term prediction on Human 3.6M dataset.** From top to bottom, we show the ground truth, the results of Residual sup.[33], GCN [32] and our method. The results show that our approach generates more realistic and accurate results.(b) **Qualitative analysis for the impact of the dictionary.** From top to bottom, we show the ground truth, the results without and with the dictionary module. We see that adding the dictionary facilitates more descriptive future dynamics.

CMU-Mocap: To show the generalization ability of our proposed method, we also evaluated our performance on the CMU mocap dataset (CMU-Mocap). Following [32, 28], we selected eight actions for evaluation, including "basketball", "baseball", "soccer", etc. The data processing is the same as for Human3.6M.

Evaluation Metric: The evaluation was performed under two metrics. Following [33, 28, 19, 43, 32], we first report the Euclidean distance between the predicted and the ground-truth joint angles in Euler angle representation, which can be referred to as Mean Angle Error (MAE). In [32], an alternative metric of Mean Per Joint Position Error (MPJPE) in millimeters is adopted, which is also widely used in 3D pose estimation field [13, 34, 27, 8, 7, 6, 15, 38, 48]. Compared with MAE, MPJPE has been noted to be more effective in measuring the predicted human poses due to the inherent ambiguity in angle space, where two different sets of angles can yield the same 3D pose. To show this, we measured the MPJPE in two ways: directly using 3D coordinates to train the network (via DCT/IDCT), and converting Euler angles into 3D joint locations.

Table 4: Long-term prediction of 3D joint positions in MPJPE on Human 3.6M dataset. Our method using 3D coordinates yields the best performance.

	Walking		Eating		Smoking		Discussion		Average	
milliseconds	560	1000	560	1000	560	1000	560	1000	560	1000
Residual sup. [33]	79.4	91.6	82.6	110.8	89.5	122.6	121.9	154.3	93.3	119.8
Residual sup. 3D [33]	73.8	86.7	101.3	119.7	85.0	118.5	120.7	147.6	95.2	118.1
convSeq2Seq [28]	69.2	81.5	71.8	91.4	50.3	85.2	101.0	143.0	73.1	100.3
convSeq2Seq 3D [28]	59.2	71.3	66.5	85.4	42.0	67.9	84.1	116.9	62.9	85.4
GNN [32]	55.0	60.8	68.1	79.5	42.2	70.6	93.8	119.7	64.8	82.6
GNN 3D [32]	42.3	51.3	**56.5**	68.6	32.3	60.5	**70.5**	103.5	50.4	71.0
Ours	51.8	58.7	59.3	76.5	40.3	76.8	82.6	107.7	58.5	79.9
Ours 3D	**36.8**	**41.2**	58.4	**67.9**	**29.2**	**58.3**	74.0	**103.1**	**49.6**	**67.6**

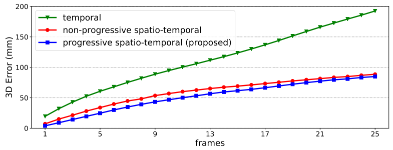


Fig. 4: Average 3D position error of each predicted frame for all actions on Human 3.6M dataset. The error accumulates much faster with the temporal model and our proposed progressive spatio-temporal method achieves the best results.

4.3 Comparison with the State-of-the-art Methods

For fair comparison, we report both short-term (10 frames in 400 milliseconds) and long-term predictions (25 frames in 1 second) for the two datasets, given the input of consecutive 10-frame human poses.

Results on Human3.6M: For short term predictions, we evaluated our results under both MAE (Table 1) and MPJPE (Table 2) protocols, in comparison to state-of-the-art baselines [33, 28, 19, 43, 32]. As previously mentioned, for MPJPE we can either directly use 3D coordinates or convert angles to 3D joint locations. As can be seen, our proposed method consistently outperformed all the state-of-the-art methods on most actions for both MAE and MPJPE protocols. The improvement is more obvious when measuring with the MPJPE metric, for which the best performance was achieved when directly using 3D joint locations during training. Moreover, we would like to point out that a high error in angle space (*e.g.* Phoning Action under MAE protocol) does not necessarily generate worse results in 3D (Phoning under MPJPE protocol). This can be explained by the inherent ambiguity of the angle representation, since two different sets of angles can generate the same 3D human pose. Based on this observation, for the following experiments, we mainly report our results under the MPJPE metric, using the 3D coordinates for training.

Besides the short term predictions, we also compared our results with the state-of-the-art methods [32, 33, 28] in long-term scenarios. For fair comparison, we report our results for 4 main classes used in the previous work, including the "Walking", "Eating", "Smoking" and"Discussion" actions under MPJPE evaluation. As shown in Table 4, similar to the short-term results, our results surpassed all other state-of-the-art methods, reducing the average errors to 49.6mm in 560ms and 67.6mm in 1000ms predictions when directly training and evaluating with the 3D joint locations.

For qualitative analysis, we provided visual comparisons with the state-of-the-art approaches [32, 33] for long-term (Figure 3 (a)) scenario, which fur-

Table 5: Influence of the spatial temporal explorations and the progressive-decoding strategy on 4 actions of Human3.6M. For fair comparison, we exclude the dictionary module for all models.

	Walking				Eating				Smoking				Discussion				Average			
milliseconds	80	160	320	400	80	160	320	400	80	160	320	400	80	160	320	400	80	160	320	400
temporal	28.9	47.2	69.7	77.8	17.2	31.6	58.3	69.0	18.8	35.7	65.4	76.1	27.9	53.1	82.0	87.9	23.2	41.9	68.9	77.7
non-progressive spatial-temporal	10.5	17.1	31.9	35.7	10.1	21.2	40.7	47.5	8.6	15.9	26.5	30.4	10.6	24.1	47.5	51.3	9.9	19.5	36.6	41.2
progressive spatial-temporal (proposed)	**8.3**	**15.1**	**30.3**	**35.2**	**8.8**	**19.3**	**39.0**	**46.1**	**7.1**	**14.0**	**24.9**	**28.1**	**8.9**	**22.1**	**44.3**	**49.1**	**8.3**	**17.6**	**34.6**	**39.6**

Table 6: Impact of the propagating directions on Human3.6 M dataset under the MPJPE protocol. The outward direction performs in the proposed central-to-peripheral extension while the inward direction contrastly propagate the predictions from the outside to inside body.

	Walking				Eating				Smoking				Discussion				Average			
milliseconds	80	160	320	400	80	160	320	400	80	160	320	400	80	160	320	400	80	160	320	400
Inward Propagation	8.9	15.8	30.5	34.9	8.9	19.5	38.9	46.7	7.6	15.0	25.0	29.2	9.4	23.0	45.3	49.9	8.7	18.3	34.9	40.2
Outward Propagation(proposed)	**7.9**	**14.5**	**29.1**	**33.5**	**8.4**	**18.1**	**37.4**	**45.3**	**6.8**	**13.2**	**24.1**	**27.5**	**8.3**	**21.7**	**43.9**	**48.0**	**7.8**	**16.8**	**33.6**	**38.5**

Table 7: The impact of the dictionary module on Human3.6M dataset under the MJMPE metric.

	Walking				Eating				Smoking				Discussion				Average			
milliseconds	80	160	320	400	80	160	320	400	80	160	320	400	80	160	320	400	80	160	320	400
w/o dictionary	8.3	15.1	30.3	35.2	8.8	19.3	39.0	46.1	7.1	14.0	24.9	28.1	8.9	22.1	44.3	49.1	8.3	17.6	34.6	39.6
w/ dictionary(proposed)	**7.9**	**14.5**	**29.1**	**33.5**	**8.4**	**18.1**	**37.4**	**45.3**	**6.8**	**13.2**	**24.1**	**27.5**	**8.3**	**21.7**	**43.9**	**48.0**	**7.8**	**16.8**	**33.6**	**38.5**

ther underscored how our method generates more realistic results, matching the ground truth better.

Results on CMU-Mocap: Table 3 compared the performance of our approach with the previously reported results [32, 33] on the CMU-Mocap dataset. For fair comparison, all methods were directly trained with 3D joint coordinates and evaluated under the MPJPE protocol. It can be seen that compared with the state-of-the-art methods, our approach achieved the best results on average and over most of the action classes.

4.4 Ablation Study

Advantages of spatio-temporal correlation & joint propagation. We first quantify the importance of leveraging both spatial and temporal correlations and assess the effectiveness of the progressive-decoding strategy. For fair comparisons, we excluded the dictionary from the model and ablated our proposed method (progressive spatio-temporal) with the following baselines: **1) temporal:** We used the straightforward way of applying the transformer network, which treats a pose at each time step as a "word" in machine translation task and sequentially generates the pose prediction of each frame; **2) non-progressive spatio-temporal:** We used DCT to encode the temporal information of each joint and fed the sequential joints into the transformer network, so as to capture the spatial and temporal dependencies. However, the decoder generates the whole body predictions at one step, without progressively producing the results.

As shown in Table 5, compared with the temporal baseline, exploiting both the spatial and temporal correlations (non-progressive spatio-temporal) considerably improved the performance by a large margin, reducing the average MPJPE error from 77.7 mm to 41.2 mm in 400ms prediction. This result can be further enhanced by applying our proposed progressive-decoding strategy, dropping the MPJPE error to 39.6mm in 400ms prediction. Moreover, as illustrated in Figure 4, the 3D errors accumulated much faster with the temporal model than the spatio-temporal approaches, and the proposed progressive joint propagation consistently outperformed the non-progressive counterpart across all time steps.

Impact of the propagating direction. Despite the overall effectiveness of progressive joint propagation, we wanted to investigate how the propagation direction impacts the results. To address this, we employed the progressive-decoding in two directions: the outward (proposed) direction that propagates from the body center to the periphery, and the inward (opposite) direction that propagates from outside to the center. As shown in Table 6, the outward propagation yielded superior performance, indicating the benefit of guiding joint extension with the more stable motion cues from the center body.

Impact of using the dictionary. We examined the impact of the dictionary quantitatively and qualitatively. As presented in Table 7, adding the dictionary consistently reduced the 3D errors among the four main action classes on the Human 3.6M dataset, which quantitatively shows the effectiveness of dictionary module. To gain more insight into what the dictionary has learned and how the dictionary enhances the prediction quality, in Figure 3 (b), we qualitatively compared our method with or without the dictionary. As can be seen, when adding the dictionary for general motion guidance, we produce more plausible and descriptive future dynamics, such as "smooth running" after kicking a ball when playing soccer.

5 Acknowledgement

This research / project is supported by the National Research Foundation, Singapore under its International Research Centres in Singapore Funding Initiative. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of National Research Foundation, Singapore This research is partially supported by the Monash FIT Start-up Grant, start-up funds from University at Buffalo and SUTD project PIE-SGP-Al-2020-02.

References

1. Aksan, E., Kaufmann, M., Hilliges, O.: Structured prediction helps 3d human motion modelling. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 7144–7153 (2019)
2. Barsoum, E., Kender, J., Liu, Z.: Hp-gan: Probabilistic 3d human motion prediction via gan. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. pp. 1418–1427 (2018)
3. Bengio, S., Vinyals, O., Jaitly, N., Shazeer, N.: Scheduled sampling for sequence prediction with recurrent neural networks. In: Advances in Neural Information Processing Systems. pp. 1171–1179 (2015)
4. Brand, M., Hertzmann, A.: Style machines. In: Proceedings of the 27th annual conference on Computer graphics and interactive techniques. pp. 183–192 (2000)
5. Butepage, J., Black, M.J., Kragic, D., Kjellstrom, H.: Deep representation learning for human motion prediction and classification. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 6158–6166 (2017)
6. Cai, Y., Ge, L., Cai, J., Magnenat-Thalmann, N., Yuan, J.: 3d hand pose estimation using synthetic data and weakly labeled rgb images. IEEE Transactions on Pattern Analysis and Machine Intelligence (2020)
7. Cai, Y., Ge, L., Cai, J., Yuan, J.: Weakly-supervised 3d hand pose estimation from monocular rgb images. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 666–682 (2018)
8. Cai, Y., Ge, L., Liu, J., Cai, J., Cham, T.J., Yuan, J., Thalmann, N.M.: Exploiting spatial-temporal relationships for 3d pose estimation via graph convolutional networks. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 2272–2281 (2019)
9. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. arXiv preprint arXiv:2005.12872 (2020)
10. Chen, M., Radford, A., Child, R., Wu, J., Jun, H., Dhariwal, P., Luan, D., Sutskever, I.: Generative pretraining from pixels (2020)
11. Dai, Z., Yang, Z., Yang, Y., Carbonell, J., Le, Q.V., Salakhutdinov, R.: Transformer-xl: Attentive language models beyond a fixed-length context. arXiv preprint arXiv:1901.02860 (2019)
12. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
13. Fang, H.S., Xu, Y., Wang, W., Liu, X., Zhu, S.C.: Learning pose grammar to encode human body configuration for 3d pose estimation. In: Thirty-Second AAAI Conference on Artificial Intelligence (2018)
14. Fragkiadaki, K., Levine, S., Felsen, P., Malik, J.: Recurrent network models for human dynamics. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 4346–4354 (2015)
15. Ge, L., Cai, Y., Weng, J., Yuan, J.: Hand pointnet: 3d hand pose estimation using point sets. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 8417–8426 (2018)
16. Ge, L., Ren, Z., Li, Y., Xue, Z., Wang, Y., Cai, J., Yuan, J.: 3d hand shape and pose estimation from a single rgb image. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 10833–10842 (2019)
17. Ghosh, P., Song, J., Aksan, E., Hilliges, O.: Learning human motion models for long-term predictions. In: 2017 International Conference on 3D Vision (3DV). pp. 458–466. IEEE (2017)

18. Gong, H., Sim, J., Likhachev, M., Shi, J.: Multi-hypothesis motion planning for visual object tracking. In: 2011 International Conference on Computer Vision. pp. 619–626. IEEE (2011)
19. Gui, L.Y., Wang, Y.X., Liang, X., Moura, J.M.: Adversarial geometry-aware human motion prediction. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 786–803 (2018)
20. Gupta, A., Martinez, J., Little, J.J., Woodham, R.J.: 3d pose from motion for cross-view action recognition via non-linear circulant temporal encoding. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2601–2608 (2014)
21. Ionescu, C., Papava, D., Olaru, V., Sminchisescu, C.: 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)
22. Jain, A., Zamir, A.R., Savarese, S., Saxena, A.: Structural-rnn: Deep learning on spatio-temporal graphs. In: Proceedings of the ieee conference on computer vision and pattern recognition. pp. 5308–5317 (2016)
23. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
24. Koppula, H., Saxena, A.: Learning spatio-temporal structure from rgb-d videos for human activity detection and anticipation. In: International conference on machine learning. pp. 792–800 (2013)
25. Koppula, H.S., Saxena, A.: Anticipating human activities for reactive robotic response. In: IROS. p. 2071. Tokyo (2013)
26. Kovar, L., Gleicher, M., Pighin, F.: Motion graphs. In: ACM SIGGRAPH 2008 classes, pp. 1–10 (2008)
27. Lee, K., Lee, I., Lee, S.: Propagating lstm: 3d pose estimation based on joint interdependency. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 119–135 (2018)
28. Li, C., Zhang, Z., Sun Lee, W., Hee Lee, G.: Convolutional sequence to sequence model for human dynamics. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5226–5234 (2018)
29. Liu, J., Ding, H., Shahroudy, A., Duan, L.Y., Jiang, X., Wang, G., Chichung, A.K.: Feature boosting network for 3d pose estimation. IEEE transactions on pattern analysis and machine intelligence (2019)
30. Liu, J., Shahroudy, A., Wang, G., Duan, L.Y., Kot, A.C.: Skeleton-based online action prediction using scale selection network. IEEE transactions on pattern analysis and machine intelligence **42**(6), 1453–1467 (2019)
31. Liu, J., Shahroudy, A., Xu, D., Kot, A.C., Wang, G.: Skeleton-based action recognition using spatio-temporal lstm network with trust gates. IEEE transactions on pattern analysis and machine intelligence **40**(12), 3007–3021 (2017)
32. Mao, W., Liu, M., Salzmann, M., Li, H.: Learning trajectory dependencies for human motion prediction. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 9489–9497 (2019)
33. Martinez, J., Black, M.J., Romero, J.: On human motion prediction using recurrent neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2891–2900 (2017)
34. Martinez, J., Hossain, R., Romero, J., Little, J.J.: A simple yet effective baseline for 3d human pose estimation. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 2640–2649 (2017)

35. Paden, B., Čáp, M., Yong, S.Z., Yershov, D., Frazzoli, E.: A survey of motion planning and control techniques for self-driving urban vehicles. IEEE Transactions on intelligent vehicles **1**(1), 33–55 (2016)
36. Pavlovic, V., Rehg, J.M., MacCormick, J.: Learning switching linear models of human motion. In: Advances in neural information processing systems. pp. 981–987 (2001)
37. Radford, A., Narasimhan, K., Salimans, T., Sutskever, I.: Improving language understanding by generative pre-training. URL https://s3-us-west-2. amazonaws. com/openai-assets/researchcovers/languageunsupervised/language understanding paper. pdf (2018)
38. Siyuan Yang, Jun Liu, S.L.M.H.E.A.C.K.: Collaborative learning of gesture recognition and 3d hand pose estimation with multi-order feature analysis. In: Proceedings of the European Conference on Computer Vision (ECCV) (2020)
39. Sukhbaatar, S., Weston, J., Fergus, R., et al.: End-to-end memory networks. In: Advances in neural information processing systems. pp. 2440–2448 (2015)
40. Tianjiao Li, Jun Liu, W.Z.L.D.: hard-net: Hardness-aware discrimination network for 3d early activity prediction. In: Proceedings of the European Conference on Computer Vision (ECCV) (2020)
41. Troje, N.F.: Decomposing biological motion: A framework for analysis and synthesis of human gait patterns. Journal of vision **2**(5), 2–2 (2002)
42. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: Advances in neural information processing systems. pp. 5998–6008 (2017)
43. Wang, B., Adeli, E., Chiu, H.k., Huang, D.A., Niebles, J.C.: Imitation learning for human pose prediction. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 7124–7133 (2019)
44. Wang, J.M., Fleet, D.J., Hertzmann, A.: Gaussian process dynamical models for human motion. IEEE transactions on pattern analysis and machine intelligence **30**(2), 283–298 (2007)
45. Wang, Z., Yu, P., Zhao, Y., Zhang, R., Zhou, Y., Yuan, J., Chen, C.: Learning diverse stochastic human-action generators by learning smooth latent transitions. arXiv preprint arXiv:1912.10150 (2019)
46. Xiong, C., Merity, S., Socher, R.: Dynamic memory networks for visual and textual question answering. In: International conference on machine learning. pp. 2397–2406 (2016)
47. Yang, X., Tang, K., Zhang, H., Cai, J.: Auto-encoding scene graphs for image captioning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 10685–10694 (2019)
48. Zhipeng Fan, Jun Liu, Y.W.: Adaptive computationally efficient network for monocular 3d hand pose estimation. In: Proceedings of the European Conference on Computer Vision (ECCV) (2020)