Multi-Person 3D Motion Prediction with Multi-Range Transformers

Anonymous Author(s)
Affiliation
Address
email

Abstract

We propose a novel framework for multi-person 3D motion trajectory prediction. Our key observation is that a human's action and behaviors may highly depend on the other persons around. Thus, instead of predicting each human pose trajectory in isolation, we introduce a *Multi-Range Transformer* model which contains of a local-range encoder for individual motion and a global-range encoder for social interactions. The Transformer decoder then performs prediction for each person by taking a corresponding pose as a query which attends to both local and global-range encoder features. Our model not only outperforms state-of-the-art methods on long-term 3D motion prediction, but also generates diverse social interactions. More interestingly, our model can even predict 15-person motion simultaneously by automatically dividing the persons into different interaction groups. Videos are available at: https://anonymousaut.github.io/Anonymous-Result/

1 Introduction

Given a few time steps of human motion, we are able to forecast how the person will continue to move and imagine the complex dynamics of their motion in the future. The ability to perform such predictions allows us to react and plan our own behaviors. Similarly, a predictive model for human motion is an essential component for many real world computer vision applications such as surveillance systems, and collision avoidance for robotics and autonomous vehicles. The research on 3D human motion prediction has caught a lot of attention in recent years [44, 43], where deep models are designed to take a few steps of 3D motion trajectory as inputs and predict a long-term future 3D motion trajectory as the outputs.

While encouraging results have been shown in previous work, most of the research focus on single human 3D motion prediction. Our key observation is that, how a human acts and behaves ma highly depend on the people around. Especially during interactions with multiple agents, an agent will need to predict the other agents' intentions, and then response accordingly [53]. Thus instead of predicting each human motion in isolation, we propose to build a model to predict multi-person 3D motion and interactions. Such a model will need have the following properties: (i) understand each agent's own motion in previous time steps to obtain smooth and natural future motion; (ii) within a crowd of agents, understand which agents are interacting with each other and learn to predict based on the social interactions; (iii) the time scale for prediction needs to be long-term.

In this paper, we introduce a Multi-Range Transformer for multi-person 3D motion trajectory prediction. The Transformer [62] has shown to be very effective in modeling long-term relations in language modeling [16] and recently in visual recognition [17]. Inspired by these encouraging results, we propose to explore Transformer models for predicting long-term human motion (3 seconds into the future). Our Multi-Range Transformer contains a local-range Transformer encoder for each individual person trajectory, a global-range Transformer encoder for modeling social interactions, and a Transformer decoder for predicting each person's future motion trajectory in 3D.

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

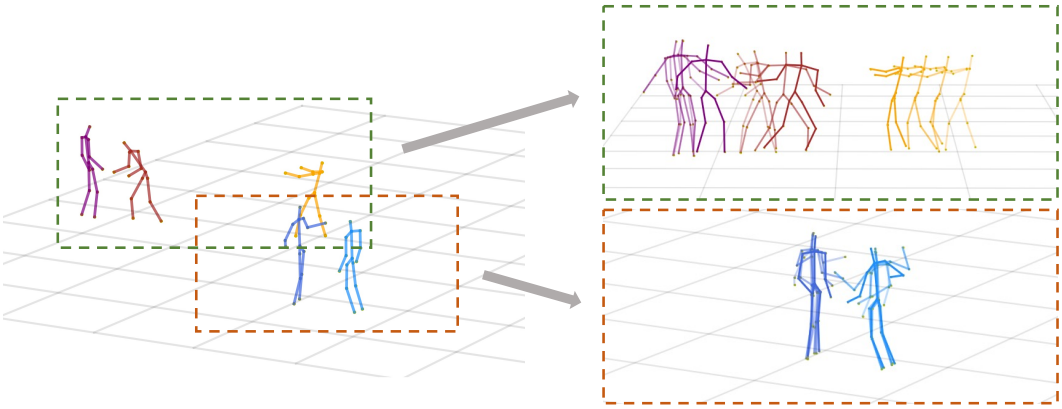

Figure 1: Our motion prediction results. **Left**: The last time step of the input sequence. **Right**: The predicted diverse and continuous motion with multi-person social interactions. We use different color to indicate different persons and the darker color for the further in future for predictions.

Specifically, given the human pose joints (with 3D locations in the world coordinate) in 1-second time steps as inputs, the local-range Transformer encoder processes each person's trajectory separately and focuses on the local motion for smooth and natural prediction. The global-range Transformer encoder performs self-attention on 3D pose joints across different persons and different time steps, and it automatically learns which persons that one person should be attending to model their social interactions. Our Transformer decoder will then take a single human 3D pose in *one time step* as the query input and encoder features as the key and value inputs to compute attention for prediction. We perform prediction for different persons by using different query pose inputs. By using only one time step person pose as the query for the decoder instead of a sequence of motion steps, we create a bottleneck to force the Transformer to exploit the relations between different time steps and persons in the encoders, instead of just repeating the existing motion alone [43].

We perform our experiments on multiple datasets including CMU-Mocap [1], MuPoTS-3D [48], 3DPW [63] for multi-person motion prediction in 3D (with $2 \sim 3$ persons). Our method achieves a significant improvement over state-of-the-art approaches for long-term predictions and the gain enlarges as we increase the future prediction time steps from 1 second to 3 seconds. Qualitatively, we visualize that our method can predict interesting behaviors and interactions between different persons while previous approaches will repeat the same poses as it goes to further steps in the future. More interestingly, we extend the task to perform prediction with $9 \sim 15$ persons by mixing the CMU-Mocap [1] and the Panoptic [24] datasets. We visualize part of the prediction results with multi-person interactions in Fig. 1. We show that Multi-Range Transformer can not only perform predictions with a crowd of persons, but also learn to automatically group the persons into different social interaction clusters using the attention mechanism.

2 Related work

3D Motion Prediction. Predicting future human pose in 3D has been widely studied with Recurrent Neural Networks (RNNs) [15, 19, 31, 41, 46, 51, 20, 22, 26, 71, 21]. For example, Fragkiadaki *et al.* [19] propose a Encoder-Recurrent-Decoder (ERD) model which incorporates nonlinear encoder and decoder networks before and after recurrent layers. Besides using RNNs, temporal convolution networks have also show promising results on modeling long-term motion [37, 28, 10, 10, 44, 43, 5]. For example, Li *et al.* [37] use a convolutional long-term encoder to encode the given history motion into hidden variable and then use a decoder to predict the future sequence. While these approaches show encouraging results, most of these studies fix the pose center and ignore the global body trajectory. Instead of solving two problems separately, recent works start looking into jointly predict human pose and the trajectory in the world coordinate [69, 72, 70, 64, 11]. For example, Cao *et al.* [11] introduce to predict human motion under the constraint of 3D scene context. Our work also predicts the human motion with both 3D poses and the trajectory movements at the same time. Going beyond single human prediction, we predict multi-human motion and interaction.

Social interaction with multiple persons. Multi-person trajectory prediction has been a long standing problem in decades [25, 47, 66, 52, 74, 7, 6, 23, 18, 49, 36, 42, 55, 8, 34, 60, 68, 61].

For example, Alahi *et al.* [7] present a LSTM [27] model which jointly reasons across multiple individuals in a scene. Gupta *et al.* [23] propose to predict socially plausible futures with Generative Adversarial Networks. However, most of these approaches focus on the global movement of the humans. To model more fine-grained human-human interactions, recent research have proposed to predict multi-person poses and trajectories at the same time [57, 56, 3, 2]. For example, Shu *et al.* [57] propose a MCMC based algorithm automatically discovers semantically meaningful interactive social affordance from RGB-D videos. Adeli *et al.* [2] introduce to combine context constraints in multi-person motion prediction. Inspired by these works, we propose a novel Multi-Range Transformer which scales up the long-term prediction with even more than 10 persons.

Transformers. Transformer is first introduced by Vaswani *et al.* [62] and has been widely applied in language processing [16, 54, 9] and computer vision [65, 29, 50, 14, 35, 17, 12, 13, 59, 67, 58, 45, 39]. For example, Dosovitskiy *et al.* [17] introduce that using Transformer alone can lead to competitive performance in image classification. Beyond recognition tasks, Transformers have also been applied in modeling the human motion [11, 40, 38, 5, 43]. Aksan *et al.* [5] use space-time self-attention mechanism with skeleton joints to synthesize long-term motion. Mao *et al.* [43] propose to extract motion attention to capture the similarity between the current motion and the historical motion. Beyond modeling single human motion, we introduce multi-range encoders to combine local motion trajectory and global interactions, and a decoder to predict multi-person motion.

3 Method

We propose a Multi-Range Transformer architecture, which allows each person to query other persons' and their own history of motion for generating socially and physically plausible future motion in 3D.

Given a scene with N persons and their corresponding history motion, our goal is to predict their future 3D motion. Specifically, given $X^n_{1:k} = [x^n_1, ..., x^n_k]$ representing the history motion of person n where $n = 1, ..., N$, and k is the time step. We aim to predict the future motion $X^n_{k+1:T}$ where T represents the end of the sequence. We use a vector $x^n_k \in \mathbb{R}^{3J}$ containing the Cartesian coordinates of the J skeleton joints to represent the pose of the person n at time step k. In contrast to most previous motion prediction works which center the pose (joint positions) at the origin, we instead use the absolute joint positions in the world coordinate. In our method, x^n_k contains both the trajectory and the pose information. For simplicity, we omit subscript n when n only represents an arbitrary person, e.g., taking $x^n_{1:k}$ as $x_{1:k}$.

3.1 Network Architecture

The proposed architecture is composed of a motion predictor $\mathcal{P}$ and a motion discriminator $\mathcal{D}$. In the predictor $\mathcal{P}$, two Transformer-based encoders encode the individual (local) and global motion separately and one Transformer-based decoder decodes a smooth and natural motion sequence. The motion discriminator $\mathcal{D}$ is a Transformer-based classifier to determine whether the generated motion is natural. The network architecture is shown in Fig. 2. We apply Discrete Cosine Transform (DCT) [4, 44] to encode the inputs for the encoders and the Inverse Discrete Cosine Transform (IDCT) for the decoder outputs. We introduce the architectures for each component as following.

3.1.1 Local-range Transformer Encoder.

When predicting one person's motion, we first use our Local-range Transformer encoder to process this person's history motion. We use offset $\Delta x_i = x_{i+1} - x_i$ between two time steps to represent the motion. We apply DCT and a linear layer to $\Delta x_{1:k}$ and then add the sinusoidal positional embedding [62] to get the local motion embedding for k time steps $l_{1:k} = [l_1, ..., l_k]$. We concatenate them as a set of tokens $E_{loc} = [l_1, ..., l_k]^T$ and feed them to the Transformer encoder. We have L stack alternating layers in the local-range Transformer encoder and we introduce the technique we use in each layer.

Firstly, a Multi-Head Attention is used for extracting sequential information,

$$\begin{aligned} \text{MultiHead}(Q, K, V) &= [\text{head}_1; ...; \text{head}_h]W^O \\ \text{where head}_i &= \text{softmax}(\frac{Q^i(K^i)^T}{\sqrt{d_K}})V^i \end{aligned} \tag{1}$$

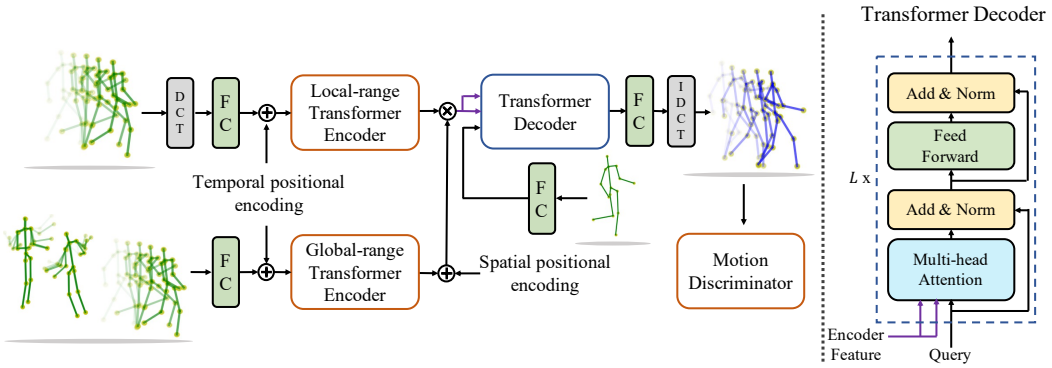

Figure 2: **Network architecture.** Individual input motion is sent to the Local-range Transformer Encoder and all the person's motions are sent to the Global-range Transformer Encoder. $\otimes$ represents concatenate and $\oplus$ represents add. The encoded motion features are used as the key and value together with the query person skeleton for the Transformer Decoder. The output is the future motion prediction results. On the right, we show the architecture of the Transformer decoder. The encoder architecture is similar with the decoder except that the query, key and value are from the same input.

W^O is a projection parameter matrix, d_K is the dimension of the key and h is the number of the heads we use. We use self-attention and get the query Q_{loc}, key K_{loc}, and value V_{loc} from E_{loc} for each head$_i$,

$$Q^i_{loc} = E_{loc}W^{(Q,i)}_{loc},\ K^i_{loc} = E_{loc}W^{(K,i)}_{loc},\ V^i_{loc} = E_{loc}W^{(V,i)}_{loc} \tag{2}$$

where $W^{(Q,i)}_{loc}$, $W^{(K,i)}_{loc}$, $W^{(V,i)}_{loc}$ are projection parameter matrices in W^O. loc represents the local-range. We then employ a residual connection and the layer normalization techniques to our architecture. We further apply a feed forward layer, again followed by a residual connection and a layer normalization following [62]. The whole process forms one layer of local-range Transformer encoder. We stack L such Transformer encoders to update update the local motion embedding and obtain the local motion feature $e_{1:k} = [e_1, ..., e_k]$ as the output, with e_i represents the feature for time step i.

3.1.2 Global-range Transformer Encoder.

In the global-range Transformer encoder, we aim to encode all the N people's motion in the scene. In our method, this only needs to be calculated one time and then can contact with any person's local motion feature to predict the correspond person's future motion. We first apply a linear layer to each person's motion $x^n_{1:k}$ and plus the sinusoidal positional embedding to get the global motion embedding $g^{1:N}_{1:k} = [g^1_1, ..., g^1_k, ..., g^N_1, ..., g^N_k]$ for N persons in k time steps. We use L layers of Transformers to encode the global motion embedding. We apply the Multi-head Attention mechanism similar to Eq. 1 to the global embedding which calculated as,

$$Q^i_{glob} = E_{glob}W^{(Q,i)}_{glob},\ K^i_{glob} = E_{glob}W^{(K,i)}_{glob},\ V^i_{glob} = E_{glob}W^{(V,i)}_{glob} \tag{3}$$

where $E_{glob} = [g^1_1, ..., g^1_k, ..., g^N_1, ..., g^N_k]^T$ and $W^{(Q,i)}_{glob}$, $W^{(K,i)}_{glob}$ and $W^{(V,i)}_{glob}$ are projection parameter matrices. $glob$ represents the global-range. Then we feed them to the normalization and feed-forward layers same as local-range Transformer encoder. After applying such L Transformer encoders to the global embedding, we can get the output $o^{1:N}_{1:k}$. We add our spatial positional encoding to the output and get the global motion feature $f^{1:N}_{1:k}$. We concatenate the local $e_{1:k}$ and global $f^{1:N}_{1:k}$ features together as $H = [e_1, ..., e_k, f^1_1, ..., f^1_k, ..., f^N_1, ..., f^N_k]^T$ and feed them to the decoder.

3.1.3 Positional encoding

Temporal positional encoding. We apply the sinusoidal positional encoding [62] to the inputs in both local and global-range encoder. This technique is routine in many Transformer-based methods for injecting information about the relative or absolute temporal position to the models.

Spatial positional encoding. We propose a spatial positional encoding (SPE) technique on the outputs of the global-range Transformer encoder. Before forwarding to the Transformer decoder, we

want to provide the spatial distance between the query token x_k and the tokens of every time step of each person $x_{1:k}^{1:N}$. Intuitively, the location information helps clustering different persons in different social interactions groups, especially in a scene with a crowd of persons. We calculate SPE as,

$$\text{SPE}(x_t^n, x_k) = \exp(-\frac{1}{3J}||x_t^n - x_k||_2^2) \tag{4}$$

where $n = 1, ..., N$, $t = 1, ..., k$, and $x_t^n, x_k \in \mathbb{R}^{3J}$. SPE$(\cdot)$ will explicitly calculate the spatial distance between two persons. All the SPE (x_t^n, x_k) will then add to o_t^n (the output of the global-range encoder) respectively in order to get the global feature $f_{1:k}^{1:N}$.

3.1.4 Transformer decoder

We send the local-global motion feature H together with a static human pose x_k at time step k into the decoder. Different from the Transformer encoders, we use this single pose to as the query and use the feature from the encoders to get keys and values. We apply L layers of Transformer decoder. Specifically, we apply a linear layer to x_k and then get q in order to get the query Q_{dec}, we get the key K_{dec} and value V_{dec} both from the local-global motion feature H as,

$$Q_{dec}^i = q^T W_{dec}^{(Q,i)},\ K_{dec}^i = HW_{dec}^{(K,i)},\ V_{dec}^i = HW_{dec}^{(V,i)} \tag{5}$$

where $W_{dec}^{(Q,i)}$,$W_{dec}^{(K,i)}$ and $W_{dec}^{(V,i)}$ are projection parameter matrices. At the end of the decoder, we apply two fully connected layers followed by Inverse Discrete Cosine Transform (IDCT) [4, 44] and output an offset motion sequence $[\Delta\hat{x}_k, ..., \Delta\hat{x}_{T-1}]$ which can easily lead to the future 3D motion trajectory $\hat{x}_{k+1:T}$. The architecture outputs sequence directly prevents generating freezing motion [40]. Note we also add residual connections and layer normalization between layers.

3.1.5 Motion discriminator

The design of such encoder-decoder architecture helps to predict the future motion. To ensure a natural and continuous long-term motion, we use a discriminator $\mathcal{D}$ to adversially train the Predictor $\mathcal{P}$. The output motion $\hat{x}_{k+1:T}$ is given as input to the Transformer encoder with the same architecture of the local-range encoder and we further use another two fully connected layers to predict values $\in \{1, 0\}$ representing that $\hat{x}_{k+1:T}$ are real or fake poses. We use the ground-truth future poses to provide as the positive examples. We train the predictor $\mathcal{P}$ and discriminator $\mathcal{D}$ jointly.

3.2 Training and Inference

We train our predictor $\mathcal{P}$ with both the reconstruction loss and the adversarial loss,

$$L_{\mathcal{P}} = \lambda_{rec} L_{rec} + \lambda_{adv} L_{adv} \tag{6}$$

where λ_{adv} and λ_{rec} are constant coefficients to balance the training loss. We calculate the L_{rec} and L_{adv} as follows,

$$\begin{aligned} L_{rec} &= \frac{1}{T-k}\Sigma_{t=k}^{T-1}||\Delta\hat{x}_t - \Delta x_t||_2^2 \\ L_{adv} &= \frac{1}{T-k}||\mathcal{D}(\hat{x}_{k+1:T}) - \mathbf{1}||^2 \end{aligned} \tag{7}$$

We train our discriminator $\mathcal{D}$ following [33] with loss $L_{\mathcal{D}}$,

$$L_{\mathcal{D}} = \frac{1}{T-k}||\mathcal{D}(\hat{x}_{k+1:T})||^2 + \frac{1}{T-k}||\mathcal{D}(y_{k+1:T}) - \mathbf{1}||^2 \tag{8}$$

where $\hat{x}_{k+1:T}$ is from the predicted motion and $y_{k+1:T}$ is from the real motion. We train the discriminator that classifies the real ones as $\mathbf{1}$, where $\mathbf{1} \in \mathbb{R}^{T-k}$ represents all the poses are natural.

Since we are using Transformer-based encoder, there is no limit to the length of the input sequence. During **training**, we can provide $x_{1:k}$ to predict $\hat{x}_{k+1:2k}$ (x_k as the query), or we can provide $x_{1:2k}$ to predict $\hat{x}_{2k+1:3k}$ (x_{2k} as the query) and so on. We sample different lengths of sequence in this way input during training. During **inference**, we predict the future motion in an auto-regressive way. Given $x_{1:k}$, our model predicts $\hat{x}_{k+1:2k}$. Then, given the input and our prediction results, we use them as inputs for our model again to predict $\hat{x}_{2k+1:3k}$, and so on. The advantage of such design is that when predicting longer motions, we still maintain the early motions as inputs to the model, instead of using a fixed length to predict each of the future time step [44, 43], which may cause the loss of early interactive information. At the same time, a fixed-length output makes the result more stable and prevent producing freezing motion [40].

4 Experiments

4.1 Datasets

We perform our experiments on multiple datasets. CMU-Mocap [1] contains high quality motion sequences of $1 \sim 2$ persons in each scene. Panoptic [24], MuPoTS-3D [48] and 3DPW [63] datasets are collected using cameras with pose estimation and optimization. Consequently, the estimated joint positions in the latter datasets contain more noise and are less smooth in comparison to CMU-Mocap. For multi-person data, there are about $3 \sim 7$ persons in each scene in Panoptic, $2 \sim 3$ persons in MuPoTS-3D each scene and 2 persons in 3DPW. It is worth noting that in the Panoptic dataset, most of the scenes are people standing and chatting, with small movement. The skeleton joint positions in 3DPW are obtained from a moving camera so that there is more unnatural foot skating.

Dataset Settings. We design two different settings to evaluate our method. The first setting consists of a small number of people ($2 \sim 3$). We use CMU-Mocap as our training data. CMU-Mocap contains a large number of scenes with a single person moving and a small number of scenes with two persons interacting and moving. We sample from these two parts and mix them together as our training data. We make all the CMU-Mocap data consists of 3 persons in each scene. We sample test set from CMU-Mocap in a similar way. We evaluate the *generalization* ability of our model by testing on MuPoTS-3D and the 3DPW dataset with the model trained on the entire CMU-Mocap dataset.

The second setting consists of scenes with more people. For the training data, We sample motions from CMU-Mocap and Panoptic and then mix them. We integrate CMU-Mocap and Panoptic scenes to build a single scene with more people. For the test data, we sample one version from both CMU-Mocap and Panoptic, namely Mix1. And we sample one version from CMU-Mocap, MuPoTS-3D and 3DPW, namely Mix2. There are $9 \sim 15$ persons in each scene in Mix1 and 11 persons in Mix2. The positive motion data for discriminator is sampled from CMU-Mocap's single person motion. We visualize one example for each dataset we used in Fig. 3. More details about the dataset setup will be discussed in the supplementary material.

4.2 Implementation details

In our experiments, we give 1 second history motion ($k = 15$ time steps) as input and recursively predict the future 3 seconds (45 time steps) as Sec. 3.2 described. We use $L = 3$ alternating layers with 8 heads in each Transformer. We use Adam [32] as the optimizer for our networks. During training, we set 3×10^{-4} as the learning rate for predictor $\mathcal{P}$ and 5×10^{-4} as the learning rate for discriminator $\mathcal{D}$. We set $\lambda_{rec} = 1$ and $\lambda_{adv} = 5 \times 10^{-4}$. For experiments with $2 \sim 3$ persons, we set a batch size of 32 and for scene with more people, we set a batch size of 8.

4.3 Metrics and methods for comparisons

Metrics. We use Mean Per Joint Position Error (MPJPE) [30] without aligning as the metric to compare the multi-person motion prediction results in 1, 2 and 3 seconds. Without aligning, MPJPE will reflect the error caused by both trajectory and pose. Further, we compare the distribution of the movement between the start and end of the outputs on different datasets. Specifically, we measure this movement by calculating the mean L2 distance between the start and end skeleton joint positions. This surrogate metric is designed to measure whether a method could predict a plausible movement, instead of standing and repeating. We also perform a user study using Amazon Mechanical Turk (AMT) to compare the realness of the predicted motion. We let the user to score each motion prediction from 1 to 5, with a higher score implying the video looks more natural inspired by [64, 73].

Methods for Comparisons. We argue that for multi-person motion prediction, joint positions are more relevant compared to angle-based representations, since most of the data suitable for this task is obtained by cameras and pose estimation, containing only the position representations. Another consideration for comparison method selection is many related studies only model the pose fixing at the origin, while we need to select methods that could predict absolute motion. We select two competitive state-of-the-art person motion prediction methods: LTD [44] is a graph-based method and HRI [43] is an attention-based method. Both of them allow absolute coordinate inputs, which fits our task and settings well. Most relevant to our work is SocialPool [2], a method which uses GRU [15] to model the motion sequences and proposes to use a social pool to model the interaction. We remove the image input of it as well as the image feature, keeping the sequence to sequence and social pool structures. We use the same data to train these methods.

method	CMU-Mocap (3 persons)		
	1 s	2s	3s
LTD [44]	1.37	2.19	3.26
HRI [43]	1.49	2.60	3.07
SocialPool [2]	1.15	2.71	3.90
Ours w/o Global	0.99	1.71	2.50
Ours w/o $\mathcal{D}$	1.13	1.84	2.57
Ours w/o SPE	1.05	1.68	2.37
Ours	**0.96**	**1.57**	**2.18**

Table 1: MPJPE on CMU-Mocap. We compare the MPJPE with the previous SOTA methods and ablative baselines of predicting 1, 2 and 3 seconds motion. Best results are shown in boldface.

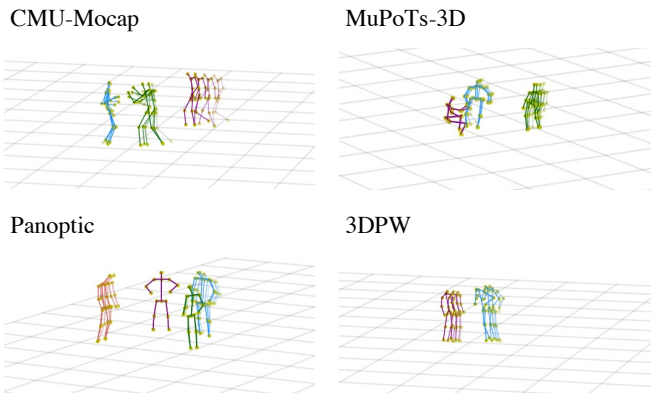

Figure 3: Examples of the ground truth motion from different datasets. Darker pose color represents the further in future.

method	MuPoTS-3D (2 $\sim$ 3 persons)			3DPW (2 persons)		
	1s	2s	3s	1s	2s	3s
LTD [44]	1.19	1.81	2.34	4.67	7.10	8.71
HRI [43]	0.94	1.68	2.29	4.07	6.32	8.01
SocialPool [2]	0.92	1.67	2.51	4.17	7.17	9.27
Ours w/o Global	0.92	1.67	2.50	4.17	6.85	8.91
Ours w/o $\mathcal{D}$	0.92	1.62	2.26	4.17	6.41	8.09
Ours w/o SPE	0.92	**1.51**	2.23	3.92	6.18	**7.79**
Ours	**0.89**	1.59	**2.22**	**3.87**	**6.12**	7.83

Table 2: MPJPE on MPoTS-3D and 3DPW of 1 second, 2 seconds and 3 seconds motion. We train the model on CMU-Mocap. Best results are shown in boldface.

method	Mix1 (9 $\sim$ 15 persons)			Mix2 (11 persons)		
	1s	2s	3s	1s	2s	3s
LTD [44]	2.10	3.19	4.15	1.72	2.58	3.45
HRI [43]	1.80	3.14	4.21	1.60	2.71	3.67
SocialPool [2]	1.85	3.39	4.84	1.72	3.06	4.26
Ours w/o Global	1.77	3.10	4.19	1.42	2.29	3.06
Ours w/o $\mathcal{D}$	1.75	3.00	4.00	1.34	2.19	2.95
Ours w/o SPE	1.75	3.09	4.13	1.31	2.15	2.92
Ours	**1.73**	**2.99**	**3.97**	**1.29**	**2.09**	**2.82**

Table 3: MPJPE on Mix1 and Mix2 of 1 second, 2 seconds and 3 seconds motion. We train the model with the mix of CMU-Mocap and Panoptic. Best results are shown in boldface.

4.4 Evaluation results

4.4.1 Quantitative results

We report the MPJPE in 0.1 meters of 1 second, 2 seconds and 3 seconds predicted motion on CMU-Mocap (Tab. 1), MuPoTS-3D and 3DPW (Tab. 2), Mix1 and Mix2 (Tab. 3) respectively. Tab. 1 and Tab. 2 correspond to settings with a small number of people and we only use training data from CMU-Mocap. Further we conduct experiments for scenes with more people in Tab. 3. In both cases with a small number and a large number of people, our method achieves state-of-the-art performance for different prediction time lengths. We achieve up to 20% improvement when compared to the previous single-person-based methods [44, 43] and achieve up to 30% improvement compared to the multi-person-based method [2]. In SocialPool [2], the same global feature is added to all the persons which interferes with the model's prediction for each individual, especially when there are a large number of people. Because of this, SocialPool's performance is even not as good as the previous single-person-based method. However, in our design the model can use the features corresponding to one person to query the global motion feature which automatically allows it to use the motion information belonging to other persons. Therefore, our method can achieve good results on scenes with any number of people.

We show the distribution of the moving distance between the start and the end of the predictions in Fig. 4. It shows that other methods intend to predict a motion with less movement, with much higher value in y-axis close to 0 in x-axis the distributions. We observe the results of other methods sometimes just stay in the same spatial location while the ground truth is moving with a large distance. On the contrary, the distribution of our result is very similar to the ground truth on CMU-Mocap and even on completely unseen MuPoTS-3D and 3DPW datasets. The reason is that our Transformer encoders can take any time length motion as input for modeling the long-term motion. And using the decoder to output a Δx motion is also beneficial to model the position movement. For user study, we report the average and the standard error of the score in Tab. 4. A higher average score means users think that the results are more "natural looking" and a lower standard error indicates that the scores are more stable. Our results get better reviews consistently cross all datasets.

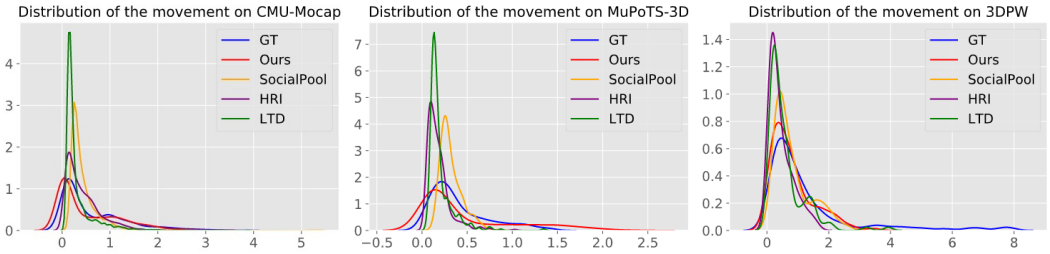

Figure 4: Distribution of the movement between the start and end of the predictions on different datasets. X-axis shows the moving distance. Ours is the most similar to the ground truth while the others intend to predict smaller movement. The model is only trained on CMU-Mocap.

method	CMU-Mocap	MuPoTS-3D	3DPW	Mix1	Mix2
LTD [44]	3.61±0.83	3.66±**0.93**	3.65±0.76	3.71±0.93	3.75±0.90
HRI [43]	3.36±0.96	3.59±1.25	3.76± **0.72**	3.67±0.89	3.71±0.90
SocialPool [2]	3.49±0.87	3.66±1.19	3.66±0.86	3.62±0.92	3.49±1.02
Ours	**3.62±0.78**	**3.68**±0.98	**3.78**±0.82	**3.74±0.83**	**3.77±0.82**
GT	3.78±0.76	3.85±0.96	3.77±0.81	3.77±0.87	3.88±0.79

Table 4: **User study.** We perform a user study using the AMT. We provide the average of the human evaluated score w.r.t. the average ± the standard deviations. Best results are shown in boldface.

4.4.2 Qualitative results

We provide qualitative comparisons in Fig. 5. More results can be found in the supplementary material. Our predictions are more natural and smooth while being close to the real record. It can be seen that RNN-based method (SocialPool [2]) will quickly produce freezing motion, which is consistent with the claims in [5, 40]. Another finding is that when predicting the absolute skeleton joint positions, decoding based on an input seed sequence (HRI [43]) or adding the input sequential residual (LTD [44]) to the output, will make the predicted motion have hysteresis and repeat the history. For example, in a forward motion, the prediction may jump back into temporally unreasonable position and then continue to move forward. We argue the positions of the past sequences have a negative impact on the model's prediction using the previously proposed design. This is also consistent with the conclusion we get in Fig. 4. However, our method, using a static pose as query and predicting a Δx sequence, could solve this problem effectively.

4.4.3 Ablation study

We perform ablation study on different modules of our network. We provide the results on different datasets in Tab. 1,Tab. 2 and Tab. 3 respectively. We remove the global-range encoder, discriminator and the spatial positional encoding respectively. After removing each module, the overall performance of the model has declined. We prove the effectiveness of the global-range encoder and an adversial training would also improve the prediction results. It is worth noting that our spatial positional encoding performs better in a scene with more people, which is in line with our expectations. We also report the MPJPE with different decoder query input length on CMU-Mocap in Tab. 5. We successfully prove that using only a single pose as the query for the decoder instead of a sequence of motion could create a *bottleneck* to force the Transformer to learn to predict the future instead of repeating the existing motion. We further explore the suitable number of the Transformer layers. We conduct experiments using 1, 3, 6 and 9 layers respectively on CMU-Mocap. The results in Tab. 6 show that a 3-layer-Transformer is suitable for both shorter and longer prediction.

	1f	2f	4f	half	all
1 second	**0.96**	**0.96**	**0.96**	1,03	1.03
2 seconds	**1.57**	1.58	**1.57**	1.71	1.73
3 seconds	**2.18**	2.22	2.26	2.44	2.50

Table 5: MPJPE with different decoder query input length on CMU-Mocap. "f" represents frame. "half" represents half of the sequence and "all" represents the whole sequence.

	1-layer	3-layer	6-layer	9-layer
1 second	1.02	**0.96**	1.07	2.69
2 seconds	1.71	**1.57**	1.79	4.71
3 seconds	2.47	**2.18**	2.58	6.83

Table 6: MPJPE with different number of Transformer layers on CMU-Mocap.

Figure 5: Qualitative comparison with other methods. Left two columns are input and right three columns are outputs. Our result is the closest to the real record and the others fail to predict a walking motion and predict a less accurate interaction motion. The input data is from CMU-Mocap.

4.5 Analysis of attention

In this section, we discuss the attention results of each person in the decoder. In our model, the person in the last frame will query his/her past motion as well as the history motion of everyone. The latter is what we are interested in when considering interaction. We give a visualization of the attention score in the first layer between the query and global motion feature in Fig. 6. On the left we show the history motion of three persons in the scene. On the right, we show different person's query and how the attention score looks like. Since we use a single frame x_k to query, the query q is only a vector, so is the attention score. We concatenate the attention score vectors from each head together and draw them as a matrix. Thus each row represents a head for a person and each column represents different persons at different time steps $[f_1^1, ..., f_k^1, ..., f_1^3, ..., f_k^3]$. Fig. 6 shows that different query person could automatically weight different people at different time. Furthermore, we found that people in different interaction groups in a scene will have more similar attention score distributions. This shows that our method has the ability to automatically divide the persons into different interaction groups in the decoder. See more results in supplementary material.

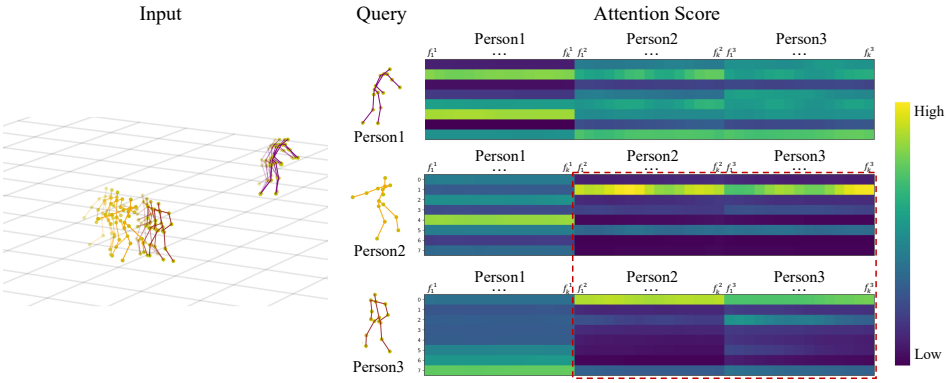

Figure 6: Attention scores in the decoder. The dotted red box shows that the two interacting people have a more similar attention score distribution. Darker pose color represents the further in future.

5 Conclusion

In this paper, we propose a novel framework to predict multi-person trajectory motion. We design a Multi-Range Transformer architecture to encode both individual motion and social interaction and output a natural long-term motion with a correspond pose as the query in the decoder. Compared with previous methods, our framework can predict more accurate and natural multi-person 3D motion.

Broader Impact. The original intention of our research is to protect people's safety in surveillance systems, and collision avoidance for robotics and autonomous vehicles. However, we remain concerned about the invasion of people's privacy through the human motion and behavior.

References

[1] Cmu graphics lab motion capture database. http://mocap.cs.cmu.edu/. 2, 6

[2] Vida Adeli, Ehsan Adeli, Ian Reid, Juan Carlos Niebles, and Hamid Rezatofighi. Socially and contextually aware human motion and pose forecasting. *IEEE Robotics and Automation Letters*, 5(4):6033–6040, 2020. 3, 6, 7, 8

[3] Vida Adeli, Mahsa Ehsanpour, Ian Reid, Juan Carlos Niebles, Silvio Savarese, Ehsan Adeli, and Hamid Rezatofighi. Tripod: Human trajectory and pose dynamics forecasting in the wild. *arXiv preprint arXiv:2104.04029*, 2021. 3

[4] Nasir Ahmed, T_ Natarajan, and Kamisetty R Rao. Discrete cosine transform. *IEEE transactions on Computers*, 100(1):90–93, 1974. 3, 5

[5] Emre Aksan, Peng Cao, Manuel Kaufmann, and Otmar Hilliges. Attention, please: A spatio-temporal transformer for 3d human motion prediction. *arXiv preprint arXiv:2004.08692*, 2020. 2, 3, 8

[6] Alexandre Alahi, Kratarth Goel, Vignesh Ramanathan, Alexandre Robicquet, Li Fei-Fei, and Silvio Savarese. Social lstm: Human trajectory prediction in crowded spaces. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 961–971, 2016. 2

[7] Alexandre Alahi, Vignesh Ramanathan, and Li Fei-Fei. Socially-aware large-scale crowd forecasting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2203–2210, 2014. 2, 3

[8] Javad Amirian, Jean-Bernard Hayet, and Julien Pettré. Social ways: Learning multi-modal distributions of pedestrian trajectories with gans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019. 2

[9] Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020. 3

[10] Judith Butepage, Michael J Black, Danica Kragic, and Hedvig Kjellstrom. Deep representation learning for human motion prediction and classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6158–6166, 2017. 2

[11] Zhe Cao, Hang Gao, Karttikeya Mangalam, Qi-Zhi Cai, Minh Vo, and Jitendra Malik. Long-term human motion prediction with scene context. In *European Conference on Computer Vision*, pages 387–404. Springer, 2020. 2, 3

[12] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European Conference on Computer Vision*, pages 213–229. Springer, 2020. 3

[13] Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, and Ilya Sutskever. Generative pretraining from pixels. In *International Conference on Machine Learning*, pages 1691–1703. PMLR, 2020. 3

[14] Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. Generating long sequences with sparse transformers. *arXiv preprint arXiv:1904.10509*, 2019. 3

[15] Kyunghyun Cho, Bart Van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*, 2014. 2, 6

[16] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 1, 3

[17] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 1, 3

[18] Tharindu Fernando, Simon Denman, Sridha Sridharan, and Clinton Fookes. Soft+ hardwired attention: An lstm framework for human trajectory prediction and abnormal event detection. *Neural networks*, 108:466–478, 2018. 2

[19] Katerina Fragkiadaki, Sergey Levine, Panna Felsen, and Jitendra Malik. Recurrent network models for human dynamics. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4346–4354, 2015. 2

[20] Partha Ghosh, Jie Song, Emre Aksan, and Otmar Hilliges. Learning human motion models for long-term predictions. In *2017 International Conference on 3D Vision (3DV)*, pages 458–466. IEEE, 2017. 2

[21] Anand Gopalakrishnan, Ankur Mali, Dan Kifer, Lee Giles, and Alexander G Ororbia. A neural temporal model for human motion prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12116–12125, 2019. 2

[22] Liang-Yan Gui, Yu-Xiong Wang, Xiaodan Liang, and José MF Moura. Adversarial geometry-aware human motion prediction. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 786–803, 2018. 2

[23] Agrim Gupta, Justin Johnson, Li Fei-Fei, Silvio Savarese, and Alexandre Alahi. Social gan: Socially acceptable trajectories with generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2255–2264, 2018. 2, 3

[24] Lei Tan Lin Gui Bart Nabbe Iain Matthews Takeo Kanade Shohei Nobuhara Hanbyul Joo, Hao Liu and Yaser Sheikh. Panoptic studio: A massively multiview system for social motion capture. In *ICCV*, 2015. 2, 6

[25] Dirk Helbing and Peter Molnar. Social force model for pedestrian dynamics. *Physical review E*, 51(5):4282, 1995. 2

[26] Alejandro Hernandez, Jurgen Gall, and Francesc Moreno-Noguer. Human motion prediction via spatio-temporal inpainting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7134–7143, 2019. 2

[27] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997. 3

[28] Daniel Holden, Jun Saito, Taku Komura, and Thomas Joyce. Learning motion manifolds with convolutional autoencoders. In *SIGGRAPH Asia 2015 Technical Briefs*, pages 1–4. 2015. 2

[29] Han Hu, Jiayuan Gu, Zheng Zhang, Jifeng Dai, and Yichen Wei. Relation networks for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3588–3597, 2018. 3

[30] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3. 6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE transactions on pattern analysis and machine intelligence*, 36(7):1325–1339, 2013. 6

[31] Ashesh Jain, Amir R Zamir, Silvio Savarese, and Ashutosh Saxena. Structural-rnn: Deep learning on spatio-temporal graphs. In *Proceedings of the ieee conference on computer vision and pattern recognition*, pages 5308–5317, 2016. 2

[32] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[33] Muhammed Kocabas, Nikos Athanasiou, and Michael J. Black. Vibe: Video inference for human body pose and shape estimation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 5

[34] 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. 2

[35] Juho Lee, Yoonho Lee, Jungtaek Kim, Adam Kosiorek, Seungjin Choi, and Yee Whye Teh. Set transformer: A framework for attention-based permutation-invariant neural networks. In *International Conference on Machine Learning*, pages 3744–3753. PMLR, 2019. 3

[36] Namhoon Lee, Wongun Choi, Paul Vernaza, Christopher B Choy, Philip HS Torr, and Manmohan Chandraker. Desire: Distant future prediction in dynamic scenes with interacting agents. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 336–345, 2017. 2

[37] Chen Li, Zhen Zhang, Wee Sun Lee, and Gim Hee Lee. Convolutional sequence to sequence model for human dynamics. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5226–5234, 2018. 2

[38] Jiaman Li, Yihang Yin, Hang Chu, Yi Zhou, Tingwu Wang, Sanja Fidler, and Hao Li. Learning to generate diverse dance motions with transformer. *arXiv preprint arXiv:2008.08171*, 2020. 3

[39] Ke Li, Shijie Wang, Xiang Zhang, Yifan Xu, Weijian Xu, and Zhuowen Tu. Pose recognition with cascade transformers. *arXiv preprint arXiv:2104.06976*, 2021. 3

[40] Ruilong Li, Shan Yang, David A. Ross, and Angjoo Kanazawa. Learn to dance with aist++: Music conditioned 3d dance generation, 2021. 3, 5, 8

[41] Zimo Li, Yi Zhou, Shuangjiu Xiao, Chong He, Zeng Huang, and Hao Li. Auto-conditioned recurrent networks for extended complex human motion synthesis. *arXiv preprint arXiv:1707.05363*, 2017. 2

[42] Karttikeya Mangalam, Harshayu Girase, Shreyas Agarwal, Kuan-Hui Lee, Ehsan Adeli, Jitendra Malik, and Adrien Gaidon. It is not the journey but the destination: Endpoint conditioned trajectory prediction. In *European Conference on Computer Vision*, pages 759–776. Springer, 2020. 2

[43] Wei Mao, Miaomiao Liu, and Mathieu Salzmann. History repeats itself: Human motion prediction via motion attention. In *European Conference on Computer Vision*, pages 474–489. Springer, 2020. 1, 2, 3, 5, 6, 7, 8

[44] Wei Mao, Miaomiao Liu, Mathieu Salzmann, and Hongdong Li. Learning trajectory dependencies for human motion prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9489–9497, 2019. 1, 2, 3, 5, 6, 7, 8

[45] Weian Mao, Yongtao Ge, Chunhua Shen, Zhi Tian, Xinlong Wang, and Zhibin Wang. Tfpose: Direct human pose estimation with transformers. *arXiv preprint arXiv:2103.15320*, 2021. 3

[46] Julieta Martinez, Michael J Black, and Javier Romero. On human motion prediction using recurrent neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2891–2900, 2017. 2

[47] Ramin Mehran, Alexis Oyama, and Mubarak Shah. Abnormal crowd behavior detection using social force model. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 935–942. IEEE, 2009. 2

[48] Dushyant Mehta, Oleksandr Sotnychenko, Franziska Mueller, Weipeng Xu, Srinath Sridhar, Gerard Pons-Moll, and Christian Theobalt. Single-shot multi-person 3d pose estimation from monocular rgb. In *3D Vision (3DV), 2018 Sixth International Conference on*. IEEE, sep 2018. 2, 6

[49] Abduallah Mohamed, Kun Qian, Mohamed Elhoseiny, and Christian Claudel. Social-stgcnn: A social spatio-temporal graph convolutional neural network for human trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14424–14432, 2020. 2

[50] Niki Parmar, Ashish Vaswani, Jakob Uszkoreit, Lukasz Kaiser, Noam Shazeer, Alexander Ku, and Dustin Tran. Image transformer. In *International Conference on Machine Learning*, pages 4055–4064. PMLR, 2018. 3

[51] Dario Pavllo, David Grangier, and Michael Auli. Quaternet: A quaternion-based recurrent model for human motion. *arXiv preprint arXiv:1805.06485*, 2018. 2

[52] Stefano Pellegrini, Andreas Ess, Konrad Schindler, and Luc Van Gool. You'll never walk alone: Modeling social behavior for multi-target tracking. In *2009 IEEE 12th International Conference on Computer Vision*, pages 261–268. IEEE, 2009. 2

[53] David Premack and Guy Woodruff. Does the chimpanzee have a theory of mind? *Behavioral and brain sciences*, 1(4):515–526, 1978. 1

[54] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019. 3

[55] Amir Sadeghian, Vineet Kosaraju, Ali Sadeghian, Noriaki Hirose, Hamid Rezatofighi, and Silvio Savarese. Sophie: An attentive gan for predicting paths compliant to social and physical constraints. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1349–1358, 2019. 2

[56] Tianmin Shu, Xiaofeng Gao, Michael S Ryoo, and Song-Chun Zhu. Learning social affordance grammar from videos: Transferring human interactions to human-robot interactions. In *2017 IEEE international conference on robotics and automation (ICRA)*, pages 1669–1676. IEEE, 2017. 3

[57] Tianmin Shu, Michael S Ryoo, and Song-Chun Zhu. Learning social affordance for human-robot interaction. *arXiv preprint arXiv:1604.03692*, 2016. 3

[58] Lucas Stoffl, Maxime Vidal, and Alexander Mathis. End-to-end trainable multi-instance pose estimation with transformers. *arXiv preprint arXiv:2103.12115*, 2021. 3

[59] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. Videobert: A joint model for video and language representation learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7464–7473, 2019. 3

[60] Jianhua Sun, Qinhong Jiang, and Cewu Lu. Recursive social behavior graph for trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 660–669, 2020. 2

[61] Jianhua Sun, Yuxuan Li, Hao-Shu Fang, and Cewu Lu. Three steps to multimodal trajectory prediction: Modality clustering, classification and synthesis. *arXiv preprint arXiv:2103.07854*, 2021. 2

[62] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *arXiv preprint arXiv:1706.03762*, 2017. 1, 3, 4

[63] Timo von Marcard, Roberto Henschel, Michael Black, Bodo Rosenhahn, and Gerard Pons-Moll. Recovering accurate 3d human pose in the wild using imus and a moving camera. In *European Conference on Computer Vision (ECCV)*, sep 2018. 2, 6

[64] Jiashun Wang, Huazhe Xu, Jingwei Xu, Sifei Liu, and Xiaolong Wang. Synthesizing long-term 3d human motion and interaction in 3d scenes. *arXiv preprint arXiv:2012.05522*, 2020. 2, 6

[65] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7794–7803, 2018. 3

[66] Kota Yamaguchi, Alexander C Berg, Luis E Ortiz, and Tamara L Berg. Who are you with and where are you going? In *CVPR 2011*, pages 1345–1352. IEEE, 2011. 2

[67] Sen Yang, Zhibin Quan, Mu Nie, and Wankou Yang. Transpose: Towards explainable human pose estimation by transformer. *arXiv preprint arXiv:2012.14214*, 2020. 3

[68] Cunjun Yu, Xiao Ma, Jiawei Ren, Haiyu Zhao, and Shuai Yi. Spatio-temporal graph transformer networks for pedestrian trajectory prediction. In *European Conference on Computer Vision*, pages 507–523. Springer, 2020. 2

[69] Ye Yuan and Kris Kitani. Diverse trajectory forecasting with determinantal point processes. *arXiv preprint arXiv:1907.04967*, 2019. 2

[70] Ye Yuan, Shih-En Wei, Tomas Simon, Kris Kitani, and Jason Saragih. Simpoe: Simulated character control for 3d human pose estimation. *arXiv preprint arXiv:2104.00683*, 2021. 2

[71] Jason Y Zhang, Panna Felsen, Angjoo Kanazawa, and Jitendra Malik. Predicting 3d human dynamics from video. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7114–7123, 2019. 2

[72] Yan Zhang, Michael J Black, and Siyu Tang. We are more than our joints: Predicting how 3d bodies move. *arXiv preprint arXiv:2012.00619*, 2020. 2

[73] Yan Zhang, Mohamed Hassan, Heiko Neumann, Michael J. Black, and Siyu Tang. Generating 3d people in scenes without people. In *Computer Vision and Pattern Recognition (CVPR)*, June 2020. 6

[74] Bolei Zhou, Xiaogang Wang, and Xiaoou Tang. Understanding collective crowd behaviors: Learning a mixture model of dynamic pedestrian-agents. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2871–2878. IEEE, 2012. 2

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] See Section 4 and supplementary material.

(c) Did you discuss any potential negative societal impacts of your work? [Yes] See Section 5 and supplementary material.

(d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]

2. If you are including theoretical results...

(a) Did you state the full set of assumptions of all theoretical results? [N/A]

(b) Did you include complete proofs of all theoretical results? [N/A]

3. If you ran experiments...

(a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] See supplementary material.

(b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] see Section 3.2, 4.1 and 4.2.

(c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [N/A]

(d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See supplementary material.

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...

(a) If your work uses existing assets, did you cite the creators? [Yes]

(b) Did you mention the license of the assets? [N/A]

(c) Did you include any new assets either in the supplemental material or as a URL? [Yes] See supplementary material.

(d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]

(e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]

5. If you used crowdsourcing or conducted research with human subjects...

(a) Did you include the full text of instructions given to participants and screenshots, if applicable? [Yes] See supplementary material.

(b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]