Synthesis of Compositional Animations from Textual Descriptions

Anindita Ghosh *[1,3], Noshaba Cheema[1,2,3], Cennet Oguz[1,3], Christian Theobalt[2,3], and Philipp Slusallek[1,3]

[1]German Research Center for Artificial Intelligence (DFKI)
[2]Max-Planck Institute for Informatics
[3]Saarland Informatics Campus

Abstract

"How can we animate 3D-characters from a movie script or move robots by simply telling them what we would like them to do?" "How unstructured and complex can we make a sentence and still generate plausible movements from it?" These are questions that need to be answered in the long-run, as the field is still in its infancy. Inspired by these problems, we present a new technique for generating compositional actions, which handles complex input sentences. Our output is a 3D pose sequence depicting the actions in the input sentence. We propose a hierarchical two-stream sequential model to explore a finer joint-level mapping between natural language sentences and 3D pose sequences corresponding to the given motion. We learn two manifold representations of the motion, one each for the upper body and the lower body movements. Our model can generate plausible pose sequences for short sentences describing single actions as well as long complex sentences describing multiple sequential and compositional actions. We evaluate our proposed model on the publicly available KIT Motion-Language Dataset containing 3D pose data with human-annotated sentences. Experimental results show that our model advances the state-of-the-art on text-based motion synthesis in objective evaluations by a margin of 50%. Qualitative evaluations based on a user study indicate that our synthesized motions are perceived to be the closest to the ground-truth motion captures for both short and compositional sentences.

1. Introduction

Manually creating realistic animation of humans performing complex motions is always a challenge. Motion synthesis based on textual descriptions substantially simplifies this task and has a wide range of applications, including language-based task planning for robotics and virtual assistants [3], designing instructional videos, creating public safety demonstrations [40], and visualizing movie scripts [27]. However, mapping natural language text descriptions to 3D pose sequences for human motions is non-trivial. The input texts may describe single actions with sequential information, *e.g.*, "a person walks four steps forward", or may not correspond to the discrete time steps of the pose sequences to be generated, such as for compositional actions, *e.g.*, "a person is spinning around while walking". This necessitates a machine-level understanding of the syntax and the semantics of the text descriptions to generate the desired motions [4]. While translating a sentence to a pose sequence, we need to identify the different parts of speech in the given sentence and how they impact the output motion. A verb in the sentence describes the type of action, whereas an adverb may provide information on the direction, place, frequency, and other circumstances of the denoted action. These need to be mapped into the generated pose sequence in the correct order, laying out additional challenges for motion modeling systems.

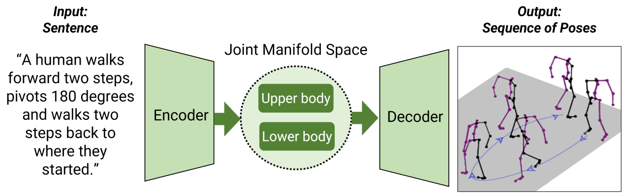

Input:
Sentence
"A human walks forward two steps, pivots 180 degrees and walks two steps back to where they started."
Encoder
Joint Manifold Space
Upper body
Lower body
Decoder
Output:
Sequence of Poses

Figure 1: Overview of our proposed method to generate motion from complex natural language sentences.[1]

Existing text-to-motion mapping methods either generate motions from sentences describing one action only [53] or produce sub-par motions from descriptions of composi-

*Corresponding Author: anindita.ghosh@dfki.de.

[1] Code and additional resources available at
https://github.com/anindita127/Complextext2animation

tional actions [4]. They fail to translate the long-range dependencies and correlations in complex sentences and do not generalize well to motions outside of locomotion [4].

We propose a method to handle complex sentences, meaning sentences that describe a person performing multiple actions either sequentially or simultaneously. For example, the input sentence “a person is stretching his arms, taking them down, walking forwards for four steps and raising them again” describes multiple sequential actions such as raising the arms, taking down the arms, and walking, as well as the direction and number of steps for the action. To the best of our knowledge, our method is the first to synthesize plausible motions from such varieties of complex textual descriptions, which is an essential next step to improve the practical applicability of text-based motion synthesis systems. To achieve this goal, we propose a hierarchical, two-stream, sequential network that synthesizes 3D pose sequences of human motions by parsing the long-range dependencies of complex sentences, while preserving the essential details of the described motions. Our output is a sequence of 3D poses corresponding to the motions described in the sentence (Fig. 1). Our main contributions in this paper are as follows:

Hierarchical joint embedding space. In contrast to JL2P [4], we separate our intermediate pose embeddings into two embeddings, one each for the upper body and the lower body. We further separate these embeddings hierarchically to limb embeddings. Our model learns the semantic variations in a sentence ascribing speed, direction, frequency of motion, and maps them to temporal pose sequences by decoding the combined embeddings. This results in the synthesis of pose sequences that correlate strongly with the descriptions given in the input sentences.

Sequential two-stream network. We introduce a sequential two-stream network with an autoencoder architecture, with different layers focusing on different parts of the body, and combine them hierarchically to two representations for the pose in the manifold space, one for the upper body and the other for the lower body. This reduces the smoothing of upper body movements (such as wrist movements for playing violin) in the generated poses and makes the synthesized motion more robust.

Contextualized BERT embeddings. In contrast to previous approaches [4, 53], which do not use any contextualized language model, we use the state-of-the-art BERT model [16] with handpicked word feature embeddings to improve text understanding. The BERT model is pretrained on a large corpus of unlabelled text including the entire Wikipedia and the Book Corpus [73].

Additional loss terms and pose discriminator. We add a set of loss terms to the network training to better condition the learning of the velocity and the motion manifold [36]. We also add a pose discriminator with an adversarial loss to further improve the plausibility of the synthesized motions.

Experimental results show that our method outperforms the baseline methods of JL2P [4] and Lin et al. [44] significantly on both the quantitative metrics we discuss in Sec. 4.3 and on qualitative evaluations.

2. Related Work

This section briefly summarizes prior works in the related areas of data-driven human motion modeling and text-based motion synthesis.

2.1. Human Motion modeling

Data-driven motion synthesis is widely used to generate realistic human motion for digital human models [33, 31, 17]. Different strategies have been implemented over the years using temporal convolutional neural networks [14, 41, 10], graph convolutional networks [5, 50] and recurrent neural networks [47, 26, 68, 38]. Pose forecasting attempts to generate short [20, 51] and long-term motions [23, 43, 62] by predicting future sequence of poses given their history. Prior works have encoded the observed information of poses to latent variables and perform predictions based on the latent variables [36, 35]. Holden et al. [34] used a feed-forward network to map high-level parameters to character movement. Xu et al. [70] proposed a hierarchical style transfer-based motion generation, where they explored a self-supervised learning method to decompose a long-range generation task hierarchically. Aristidou et al. [6] decomposed the whole motion sequences into short-term movements defining motion words and clustered them in a high-dimensional feature space. Generative adversarial networks [24] have also gained considerable attention in the field of unsupervised learning-based motion prediction [8, 39]. Li et al. [42] used a convolutional discriminator to model human motion sequences to predict realistic poses. Gui et al. [25] proposed the adversarial geometry aware encoder-decoder (AGED) framework, where two global recurrent discriminators distinguish the predicted pose from the ground-truth. Cui et al. [15] proposed a generative model for pose modeling based on graph networks and adversarial learning. Other works include pixel-level motion predictions with human pose as an intermediate variable [66, 67], and forecasting locomotion trajectories [29, 28, 46]. Researchers have also explored audio-, speech-, and image-conditioned pose forecasting[7]. For instance, Ferreira et al. [19] explored generating skeleton pose sequences for dance movements from audio, Chao et al. [9, 69] predicted pose sequences from static images. Ahuja [2] linked pose prediction with speech and audio. Takeuchi et al.[61] tackled speech conditioned forecasting for only the upper body, modeling the non-verbal behaviors such as head nods, pose switches, and hand waving for a character without providing knowledge on the character’s next movements. Chiu et al. [11] rely solely on the history of poses to predict what motion will follow.

2.2. Text-based Motion Synthesis

A subset of prior works have opted to train deep learning models to translate linguistic instructions to actions for virtual agents [30, 32, 48, 72]. Takano et al. [60, 57] developed a mapping between human motion and word labels using Hidden Markov Models. Takano et al. also used statistical methods [58, 59] using bigram models for natural languages to generate motions. Yamada et al. [71] used separate autoencoders for text and animations with a shared latent space to generate animations from text. Ahn et al. [1] generated actions from natural language descriptions for video data. However, their method only applies to upper-body joints (neck, shoulders, elbows, and wrist joints) with a static root. More recent methods use RNN based sequential networks to map text inputs to motion. Plappert et al. [53] proposed a bidirectional RNN network to map a text input to a series of Gaussian distributions representing the joint angles of the skeleton. However, their input sequence is encoded into a single one-hot vector that cannot scale as the number of sequences increases. Lin et al. [44] used an autoencoder architecture trained on mocap data without language descriptions first, and then used an RNN to map descriptions into these motion representations. JL2P [4] learned a joint embedding space for both pose and language using a curriculum learning approach. Training a model jointly with both pose and sentence inputs improves the generative power of the model. However, these methods are limited to synthesizing motion from simple sentences. Our model, by contrast, handles long sentences describing multiple actions.

3. Proposed Method

We train our model end-to-end with a hierarchical two-stream pose autoencoder, a sentence encoder, and pose discriminator (Fig. 2). Our model learns a joint-embedding between the natural language and the poses of the upper body and the lower body. Our input motion $P = [P_0, \ldots, P_{T-1}]$ is a sequence of T poses, where $P_t \in \mathbb{R}^{J \times 3}$ is the pose at the t^{th} time step, consisting of the (x, y, z) coordinates of the J joints in the pose. Our hierarchical two-stream pose encoder pe encodes the ground-truth pose sequence P into two manifold vectors,

$$pe\,(P) = (Z^p_{ub}, Z^p_{lb})\,, \tag{1}$$

where $Z^p_{ub}, Z^p_{lb} \in \mathbb{R}^h$ represent the features for the upper body and the lower body, respectively, and h denotes the dimension of the latent space.

Our input sentence $S = [S_1, S_2, \ldots, S_W]$ is a sequence of W words converted to word embeddings $\widetilde{S}_w$ using the pre-trained BERT model [16]. $\widetilde{S}_w \in \mathbb{R}^K$ represents the word embedding vector of the w^{th} word in the sentence and K is the dimension of the word embedding vector. Our two-stream sentence encoder se encodes the word embeddings and maps them to two latent vectors in the latent space as,

$$se\,(S) = (Z^s_{ub}, Z^s_{lb})\,, \tag{2}$$

where $Z^s_{ub}, Z^s_{lb} \in \mathbb{R}^h$ represent the sentence embeddings for the upper body and the lower body, respectively. Using an appropriate loss (see Sec. 3.2), we ensure that (Z^p_{ub}, Z^p_{lb}) and (Z^s_{ub}, Z^s_{lb}) lie close in the joint embedding space and carry similar information.

Our hierarchical two-stream pose decoder de learns to generate poses from these two manifold vectors. As an initial input, the pose decoder uses the initial pose P_0 to generate the pose $\hat{P}_0$, and generate each subsequent pose $\hat{P}_{t+1}$ recursively using pose $\hat{P}_t$. $\hat{P} \in \mathbb{R}^{T \times J \times 3}$ denotes a generated pose sequence. The output of our decoder module is a sequence of T poses $\hat{P}^p \in \mathbb{R}^{T \times J \times 3}$ generated from the pose embeddings, and $\hat{P}^s \in \mathbb{R}^{T \times J \times 3}$ generated from the language embeddings as

$$\hat{P}^p = de\,(Z^p_{ub}, Z^p_{lb}) \tag{3}$$

$$\hat{P}^s = de\,(Z^s_{ub}, Z^s_{lb})\,. \tag{4}$$

We use a pose prediction loss term to ensure that $\hat{P}^p$ and $\hat{P}^s$ are close to each other (Sec. 3.2). $\hat{P} = \hat{P}^s$ is our final output pose sequence for a given sentence.

3.1. Network Architecture

We describe the three main modules in our network, the two-stream hierarchical pose encoder, the two-stream sentence encoder and the two-stream hierarchical pose decoder.

3.1.1 Two-Stream Hierarchical Pose Encoder

We structure the pose encoder such that it learns features based on five major parts of the body, and combine those features hierarchically. Following Du et al. [18], we decompose the human skeleton into five major parts as the left arm, right arm, trunk, left leg, and right leg. Our hierarchical pose encoder, as shown in Fig. 2, encodes these five parts using five linear layers with output dimension h_1. We combine the trunk representation with that of the left arm, right arm, left leg, and right leg and pass them through another set of linear layers to obtain combined representations of (left arm, trunk), (right arm, trunk), (left leg, trunk), and (right leg, trunk) each of dimension h_2. Two separate GRUs [12] encode the combined representation for the arms with the trunk and the legs with the trunk respectively, thus creating manifold representations for the upper body, $Z^p_{ub} \in \mathbb{R}^h$, and for the lower body, $Z^p_{lb} \in \mathbb{R}^h$. The two GRUs then output the two manifold representations of dimension h.

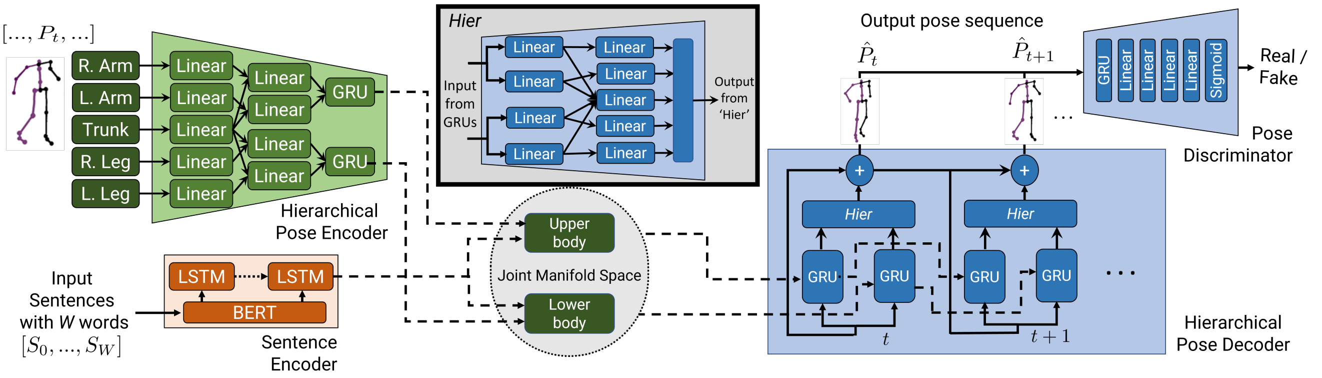


Figure 2: Structure of our hierarchical two-stream model with our pose discriminator (top right). The model learns a joint embedding for both pose and language. The embedding has separate representations for the upper body and lower body movements. We show the hierarchy of the pose decoder as *Hier* in the inset box (top center).

3.1.2 Two-Stream Sentence Encoder

To represent the text input, we use the pre-trained large-case model of BERT [16] as a contextualized language model. It comprises of 24 layers, each representing different linguistic notions of syntax or semantics [13]. To find the layers focused on local context, *e.g.*, adverbs of a verb [63], we use the attention visualization tool [65] with randomly selected samples of the KIT Motion Language dataset [52]. Thus, we select the layers 12 (corresponding to subjects), 13 (adverbs), 14 (verbs) and 15 (prepositional objects) and concatenate the hidden states of these layers to represent the corresponding words. Formally, $\widetilde{S}_w \in \mathbb{R}^K$ represents the word embedding vector of the w^{th} word in the sentence S, and K is the dimension of the word embedding vector used. Our Sentence encoder se uses Long-Short Term Memory units (LSTMs) [54] to capture the long-range dependencies of complex sentences. We input the word embeddings to a two-layer LSTM, which generates

$$Z^s = \text{LSTM}\left(\widetilde{S}_w\right) = [Z^s_{ub}, Z^s_{lb}], \tag{5}$$

where $Z^s \in \mathbb{R}^{2h}$ is the latent embedding of the whole sentence, with $\widetilde{S}_w = \text{BERT}\left(S_w\right)$. We choose the first half of this embedding as $Z^s_{ub} \in \mathbb{R}^h$ to represent the upper body and the second half as $Z^s_{lb} \in \mathbb{R}^h$ to represent the lower body.

3.1.3 Two-Stream Hierarchical Pose Decoder

We can conceptually unfold our pose decoder as a series of T hierarchical decoder units, each constructing the output pose $\hat{P}_t, \forall t = 0, \ldots, T$ time steps in a recurrent fashion by taking in the generated pose at the corresponding previous time step. We add a residual connection between the inputs and the outputs of the individual decoder units. Each decoder unit consists of two GRUs, and a *Hier* unit (inset box in Fig. 2) consisting of a series of linear layers in a hierarchical structure mirroring that of the pose encoder. Conditioned by the latent space vector representing the previous frames, the *Hier* unit outputs the reconstructed pose $\hat{P}_{t+1}$ at the $(t+1)^{\text{th}}$ frame given the previous pose $\hat{P}_t$.

3.2. Optimizing the Training Procedure

We train our model end-to-end with a hierarchical two-stream pose autoencoder along with a sentence encoder as shown in Fig. 2. Our model learns a joint embedding space between the natural language and the poses of the upper body and the lower body. Our decoder is trained with the tuples (Z^p_{ub}, Z^p_{lb}) obtained from pe to generate the pose sequence $\hat{P}^p$, and (Z^s_{ub}, Z^s_{lb}), obtained from se to generate the pose sequence $\hat{P} = \hat{P}^s$.

Loss functions. We use the smooth ℓ_1 loss to train our model. The smooth ℓ_1 loss is less sensitive to outliers than the smoother ℓ_2 loss, and more stable than the ℓ_1 loss as it is differentiable near $x = 0$ for all $x \in \mathbb{R}$ [4]. We use the following five losses to train our model:

- **Pose Prediction loss.** It minimizes the difference between the input ground-truth motion P and the predicted motions $\hat{P} = \hat{P}^s$ and $\hat{P}^p$. We measure it as,

$$L_R = \mathcal{L}\left(\hat{P}^s, P\right) + \mathcal{L}\left(\hat{P}^p, P\right), \tag{6}$$

  where $\mathcal{L}$ denotes the Smooth ℓ_1 Loss between the two terms.

- **Manifold reconstruction loss.** This encourages a reciprocal mapping between the generated motions and the manifold representations to improve the manifold space [36]. We reconstruct the manifold representations from the generated poses as $\hat{Z}^p_{ub} = pe\left(\hat{P}\right)$ and $\hat{Z}^p_{lb} = pe\left(\hat{P}\right)$, and compare them with the manifold representations obtained from input pose sequence. We compute the loss as,

$$L_M = \mathcal{L}\left(\hat{Z}^p_{ub}, Z^p_{ub}\right) + \mathcal{L}\left(\hat{Z}^p_{lb}, Z^p_{lb}\right). \tag{7}$$

- **Velocity reconstruction loss.** We minimize the difference between the velocity of the reconstructed motion $\left(\hat{P}_{vel}\right)$ and the velocity of the input motion (P_{vel}). We compute the velocity of the t^{th} frame of a pose P as $P_{vel}\left(t\right) = P_{(t+1)} - P_{(t)}$. We compute L_V as ,
$$L_V = \mathcal{L}\left(\hat{P}_{vel}, P_{vel}\right). \tag{8}$$
- **Embedding similarity loss.** This loss ensures that the manifold representations, Z^s_{ub} and Z^s_{lb}, generated by the sentence encoder, are close to the manifold representations Z^p_{ub} and Z^p_{lb} generated by the pose encoder. We measure it as,
$$L_E = \mathcal{L}\left(Z^p_{ub}, Z^s_{ub}\right) + \mathcal{L}\left(Z^p_{lb}, Z^s_{lb}\right). \tag{9}$$
- **Adversarial loss.** We further employ a binary cross-entropy discriminator D to distinguish between the real and generated poses. We compute the corresponding discriminator and generator losses as,
$$L_D = \mathcal{L}_2\left(D\left(\hat{P}\right), 0\right) + \mathcal{L}_2\left(D\left(P\right), 1\right) \tag{10}$$
$$L_G = \mathcal{L}_2\left(D\left(\hat{P}\right), 1\right), \tag{11}$$
where $\mathcal{L}_2$ denotes the Binary Cross Entropy loss, and the generator is the decoder of our autoencoder.

We train our model end-to-end with the pose autoencoder, the sentence encoder and the discriminator modules on a weighted sum of these loss terms as,

$$\begin{aligned}&\min_{pe,se,de}\left(L_R + \lambda_M L_M + \lambda_V L_v + \lambda_E L_E + \lambda_G L_G\right)\\&\min_{D}\left(\lambda_G L_D\right),\end{aligned} \tag{12}$$

where $\lambda_M = 0.001$, $\lambda_V = 0.1$, $\lambda_E = 0.1$ and $\lambda_G = 0.001$ are weight parameters, obtained experimentally.

4. Experiments

This section describes the dataset we use for our experiments and reports the quantitative and qualitative performances of our method. We also highlight the benefits of the different components of our method via ablation studies.

4.1. Dataset

We evaluate our model on the publicly available KIT Motion-Language Dataset [52], which consists of 3,911 recordings of human whole-body motion in the Master Motor Map representation [64, 45], and natural language descriptions corresponding to each motion. It has a total of 6,278 annotations in the English language, with each motion recordings having one or multiple annotations describing the task. The sentences range from describing simple actions such as walking forwards or waving the hand to describing motions with complicated movements such as waltzing. Moreover, there are longer, more descriptive sentences describing a sequence of multiple actions, *e.g.*, "*A human walks forwards two steps, pivots 180 degrees and walks two steps back to where they started.*" We randomly split the dataset in the ratio of 0.6, 0.2, and 0.2 for our training, validation, and test sets. For a fair comparison with the baselines, we follow the steps of Lin et al. [44] and JL2P [4] to sub-sample the motion sequences from 100 Hz to 12.5 Hz, and pre-process the motion data. Following the approach of Holden et al. [34], we use the character's joint positions with respect to the local coordinate frame and the character's trajectory of movement in the global coordinate frame. We have the (x, y, z) coordinates of $J = 21$ joints, and a separate dimension for representing the global trajectory for the root joint.

4.2. Implementation Details

We train our model for 350 epochs using the Adam Optimizer [37], which takes approximately 15 hours on an NVIDIA Tesla V100 GPU. The dimensions of our hidden layers in the hierarchical autoencoder are $h_1 = 32$, $h_2 = 128$ and $h = 512$. We used a batch size of 32 and a learning rate of 0.001 with exponential decay. For training the sentence encoder, we converted given sentences to word embeddings of dimension $K = 4{,}096$ using the pre-trained BERT-large-case model (Sec. 3.1.2). We encode these embeddings to a dimension of 1,024 through the sentence encoder, and split them to obtain two manifold representations of dimension $h = 512$ each.

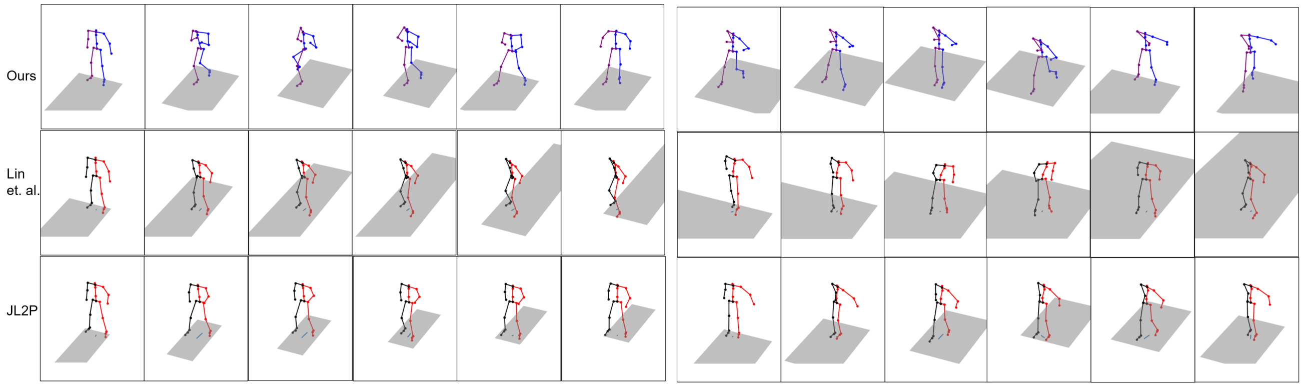
4.3. Quantitative Evaluation Metrics

To quantitatively evaluate the correctness of our motion, we use the Average Position Error (APE). APE measures the average positional difference for a joint j between the generated and the ground-truth pose sequences as,

$$\text{APE}\left[j\right] = \frac{1}{NT}\sum_{n\in N}\sum_{t\in T}\left\|P_t\left[j\right] - \hat{P}_t\left[j\right]\right\|_2, \tag{13}$$

where T is the total time steps and N is the total number of data in our test dataset and $[j]$ indicates the index.

Given our setting of natural language descriptions and corresponding free-form movements, it is naturally difficult to find a quantitative measure that does justice to both modalities. For example, in a walking setting, sentences that do not mention any direction correspond to a wider variety of plausible motions, while specifying a direction narrows the possibilities. To account for such discrepancies, we separate the APEs between the local joint positions and the global root trajectory. The former corresponds to the error of the overall poses, while the latter corresponds to the overall direction and trajectory of the motion.



A human is performing a low kick with the left foot, getting the arms up in a defense position. (Perplexity =[31.5968])

A human performs the steps of a waltz dance while it is holding its hands like it is leading a partner with its hands. (Perplexity = [29.6535])

Figure 3: Comparison of consecutive motion frames from our method (top row) with Lin et al. [44] (middle row) and JL2P [4] (bottom row) for the given sentences. Our method generates clear kicking and dancing motions in contrast to JL2P and Lin et al., that shows no prominent movements. The perplexity values of the sentences are according to Plappert et al. [52].

However, the average position of each joint simply corresponds to a mean compared to the dataset. To understand the full statistics of the overall distribution compared to the dataset, we also compute the Average Variance Error (AVE), which measures the difference of variances of individual joints of the generated poses compared to the ground-truth poses. We calculate the variance of an individual joint j for a pose P with T time steps as,

$$\sigma[j] = \frac{1}{T-1}\sum_{t\in T}\left(P_t[j] - \widetilde{P}[j]\right)^2, \tag{14}$$

where $\widetilde{P}[j]$ is the mean pose over T time steps for the joint j. Calculating the variance for all joints of the ground-truth poses and the generated poses, we use their root mean square error as the AVE metric as follows:

$$\text{AVE}[j] = \frac{1}{N}\sum_{n\in N}\|\sigma[j] - \hat{\sigma}[j]\|_2, \tag{15}$$

where σ refers to the ground-truth pose variance and $\hat{\sigma}$ refers to generated pose variance.

However, even this measure does not account for any information regarding the sentences or sentence encodings themselves. Therefore, we propose a Content Encoding Error (CEE), which corresponds to the embedding similarity loss L_E in Eq. 9 by measuring the effectiveness of the embedding space. We calculate CEE as the difference between manifold representations $Z^p = [Z^p_{ub}, Z^p_{lb}]$, obtained by encoding the input poses P through the pose encoder pe, and the manifold representations $Z^s = [Z^s_{ub}, Z^s_{lb}]$, obtained by encoding the corresponding input sentences using the sentence encoder se. We write it as,

$$\text{CEE}(S,P) = \frac{1}{MN}\sum_{n\in N}\sum_{m\in M}\|Z^s - Z^p\|_2, \tag{16}$$

where M is the number of features in the manifold representation, and N is the total number of data. The idea is to measure how well the joint embedding space correlates the latent embeddings of poses with the latent embeddings of the corresponding sentences.

To also account for style factors in the motion and the sentences, we further propose a Style Encoding Error (SEE). SEE compares a summary statistics of the sentence embeddings Z^s and the pose embeddings Z^p to account for general style information. We compute the Gram matrix [22, 21] G on the corresponding embeddings:

$$G_s = Z^s \cdot {Z^s}^\top \tag{17}$$

$$G_p = Z^p \cdot {Z^p}^\top. \tag{18}$$

We compute SEE as

$$\text{SEE}(S,P) = \frac{1}{MN}\sum_{n\in N}\sum_{m\in M}\|G_s - G_p\|_2, \tag{19}$$

where M is the number of features in the manifold representation and N is the total number of data.

4.4. Ablation Studies

We compare the performance of our model with the following four ablated versions:

- **Ablation 1: Two-stream hierarchical model without jointly training the embedding space (w/o JT).** Instead of end-to-end training of the model, we train the hierarchical pose encoder and decoder first, using the loss terms L_R, L_M, L_V, L_G and L_D (Sec. 3.2). We then train the model with the sentence encoder and the pose decoder with the losses L_R and L_E. This indicates that the model is not learning a joint embedding

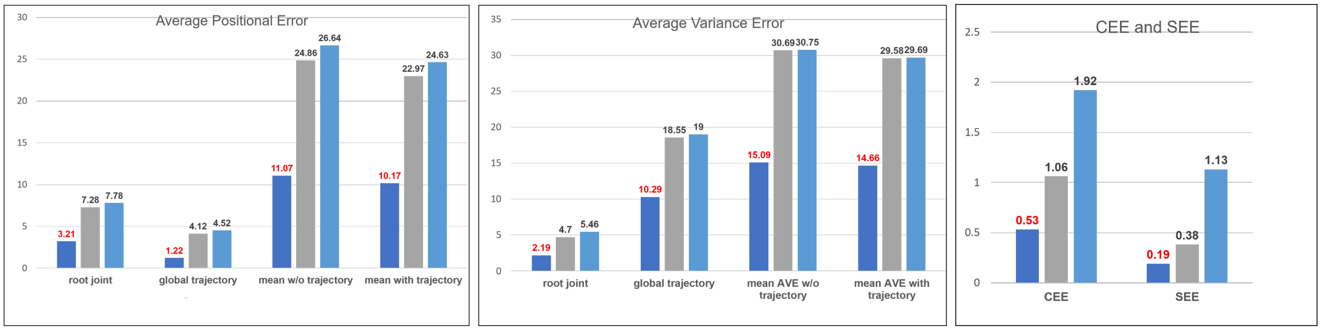


Figure 4: Plots showing the APE (left), AVE (middle), and CEE and SEE (right) in mm for our model compared to those of JL2P [4] and Lin et al. [44]. Dark blue denotes our method, grey denotes JL2P and light blue denotes Lin et al. method. Lower values are better. We see our method improves on the baselines by over 50% on all benchmarks.

space, but learns an embedding for poses first and then fine-tunes it to map the sentences.

- **Ablation 2: Hierarchical model without the two-stream representation (w/o 2-St).** We use a single manifold representation for the whole body instead of separating the upper and lower body and train the model jointly on language and pose inputs.
- **Ablation 3: Training the hierarchical two-stream model without the extra losses (w/o Lo).** We train our model with only the pose prediction loss L_R, discarding all other loss terms described in Sec. 3.2.
- **Ablation 4: Using a pre-trained language model instead of selected layers of BERT (w/o BERT).** Instead of selecting layers of BERT as described in Sec. 3.1.2, we use a pre-trained Word2Vec model [49], to convert the input sentence into word embeddings, as done in JL2P [4]. This ablation shows how BERT as a contextualized language model helps focus on the local context within a sentence.

4.5. User Study

To evaluate our ablation studies, we conduct a user study to observe the subjective judgment of the quality of our generated motions compared to the quality of motions generated from the ablations described in Sec. 4.4. We asked 23 participants to rank 14 motion videos from the five methods and from the ground-truth motion-captures, based on whether the motion corresponds to the input text, and by the quality and naturalness of the motions. The five methods include our method and the four ablations of our model, ‘w/o JT’, ‘w/o 2-St’, ‘w/o Lo’, and ‘w/o BERT’. We quantified the user study with two preference scores, the first one describing if the participants found the motions to correspond to the input sentence (“yes/no”), and the second one rating the overall quality of the motion in terms of naturalness (from 1 =“most natural” to 6 =“least natural”, which we then scaled to $[0, 1]$ and inverted). We observe that our method has a preference score of $\sim$ 40% in both cases, second only to the ground-truth motion, as seen in Fig. 5. [2]

5. Results and Discussion

We compare our method with the baseline Joint Language to Pose (JL2P) method [4], and the method of Lin et al. [44]. We use the pre-trained models for both these methods, made available in JL2P [4], to calculate the quantitative results. We compute all the results on our test set.

5.1. Objective Evaluation

Fig. 4 shows the improvement of our method compared to JL2P and Lin et al. for all the metrics described in Sec. 4.3. Our method shows an improvement of 55.4% in the mean APE calculated for all local joints compared to JL2P and by 58.4% compared to Lin et al. When including the global trajectory, our method shows an improvement of 55.7% in mean APE compared to JL2P and 58.7% compared to Lin et al. (Fig. 4 left). [3] We also observe that high error in the root joint leads to either foot sliding in the motion or averages out the whole motion. Improvement in the error values for the root joint indicates high-quality motions without such artifacts. Further, our method shows closer resemblances to the variance of the ground-truth motion compared to the baseline models (Fig. 4 center). Our method has an improvement of 50.4% and 50.6% in the AVE over the mean of all joints with the global trajectory compared to JL2P and Lin et al. respectively. We provide detailed APE and AVE values of individual joints in the supplementary material.

We also show improvements of 50% in the CEE and SEE metrics compared to JL2P. Compared to Lin et al., we show

[2] We decided to exclude JL2P [4] and Lin et al. [44] from the user study, based on overwhelming feedback from participants that our method beats the baselines in the most obvious ways.

[3] We note that our reported numbers for the baseline methods in the APE metric are different from the original paper. However, we were unable to replicate the numbers in the original paper using the code and the pre-trained model provided by the authors.

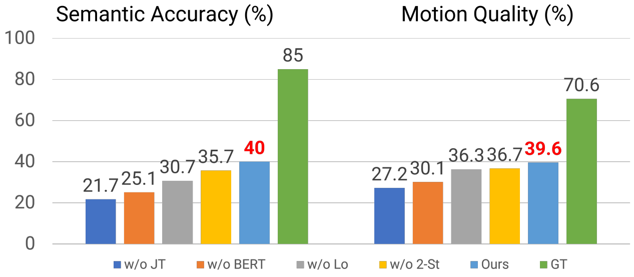


Figure 5: Semantic accuracy in percent, denoting how well the motions correspond visually to the input sentences (left) and motion quality in percent, showing the overall quality of the motions in terms of naturalness (right). Higher values are better. Our method (red) is second-best only to the ground-truth.

improvements of 72.3% and 83.1% in CEE and SEE respectively (Fig. 4 right). These results show that the joint embedding space learned by our method can correlate the poses and corresponding sentences better than the baselines.

5.2. Qualitative Results

To qualitatively compare our model with JL2P [4] and Lin et al. [44], we examine the generated motions from all three methods. Fig. 3 shows two motions with comparatively high sentence perplexities [52]. Our method (top row left) accurately generates the kicking action with the correct foot and right arm positions as described in the sentence, while the baseline models fail to generate a kick at all (middle and bottom rows left). Fig. 3 (right) further shows that the Waltz dance is more prominent in our model, compared to both baselines where arm movements appear to be missing completely, and the skeleton tends to slide than actually step. Fig. 6 shows screenshots with motions generated from complex sentence semantics. Our method (left) accurately synthesizes a trajectory that matches the semantics of the sentence. Although JL2P [4] generates a circular trajectory (bottom right), the walking direction does not match the semantics of the sentence. Lin et al. [44] fail to generate a circular trajectory at all. Further, neither method can synthesize the correct turning motions (middle and right).

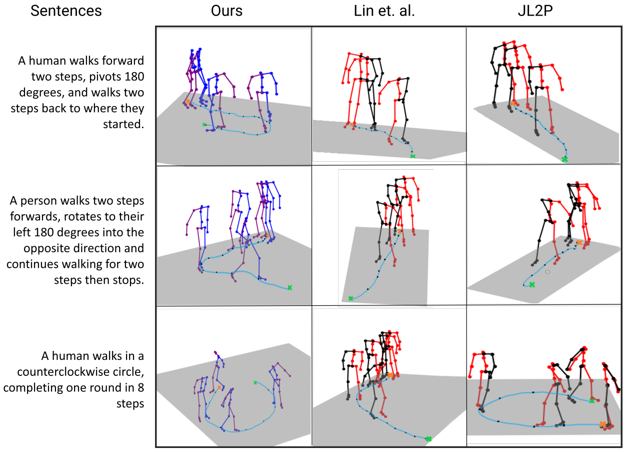

Sentences
Ours
Lin et. al.
JL2P
A human walks forward two steps, pivots 180 degrees, and walks two steps back to where they started.
A person walks two steps forwards, rotates to their left 180 degrees into the opposite direction and continues walking for two steps then stops.
A human walks in a counterclockwise circle, completing one round in 8 steps

Figure 6: Comparison of generated motions of our method (left) with Lin et al. [44] (middle) and JL2P [4] (right) for long sentences indicating the direction and the number of steps. Orange and green crosses respectively denote the start- the end-points of the motion. Blue curve on the plane denotes the trajectory and the black dots represent the foot steps. Our method is able to follow the semantics of the sentences, while the baselines fail.

6. Limitations, Future Work and Conclusion

We presented a novel framework that advances the state-of-the-art on text-based motion synthesis on qualitative evaluations and several objective benchmarks. While our model accurately synthesizes compositional actions encountered during training, it cannot always synthesize novel motions successfully. We intend to extend our model to a zero- or few-shot paradigm [56] such that it generates compositional actions from input sentences without being trained on those specific motions. We also plan to experiment with narration-based transcripts that describe long sequences of step-by-step actions involving multiple people, *e.g.*, narration-based paragraphs depicting step-by-step movements for performing complex actions such as dance and professional training. To this end, a different embedding that explicitly models the sequential nature of the task may be more suitable, but that may also reduce the ability of the model to synthesize actions not described in a sequential manner. We also plan to introduce physical constraints [55] to improve on the general motion quality, such as foot sliding, limb constraints, and biomechanical plausibility.

Being able to model a variety of motions and handle such complex sentence structures is an essential next step in generating realistic animations for mixtures of actions in the long-term and improving the practical applicability of text-based motion synthesis systems. To the best of our knowledge, this is the first work to achieve this quality of motion synthesis on a benchmark dataset and is an integral step towards script-based animations.

Acknowledgements

We would like to thank all the participants in our user study, as well as the XAINES partners for their valuable feedback. This research was funded by the BMBF grants XAINES (01|W20005) and IMPRESS (01|S20076), as well as by the EU Horizon 2020 grant Carousel+ (101017779) and an IMPRS-CS Fellowship. Computational resources were provided by the BMWi under the grants 01MK20004D and 01MD19001B.

References

[1] H. Ahn, T. Ha, Y. Choi, H. Yoo, and S. Oh. Text2action: Generative adversarial synthesis from language to action. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5915–5920, 2018. 3

[2] Chaitanya Ahuja. Coalescing narrative and dialogue for grounded pose forecasting. In *2019 International Conference on Multimodal Interaction*, pages 477–481, 2019. 2

[3] Chaitanya Ahuja, Dong Won Lee, Yukiko I Nakano, and Louis-Philippe Morency. Style transfer for co-speech gesture animation: A multi-speaker conditional-mixture approach. In *European Conference on Computer Vision*, pages 248–265. Springer, 2020. 1

[4] C. Ahuja and L. Morency. Language2pose: Natural language grounded pose forecasting. In *2019 International Conference on 3D Vision (3DV)*, pages 719–728, 2019. 1, 2, 3, 4, 5, 6, 7, 8

[5] Emre Aksan, Manuel Kaufmann, and Otmar Hilliges. Structured prediction helps 3d human motion modelling. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7144–7153, 2019. 2

[6] Andreas Aristidou, Daniel Cohen-Or, Jessica K Hodgins, Yiorgos Chrysanthou, and Ariel Shamir. Deep motifs and motion signatures. *ACM Transactions on Graphics (TOG)*, 37(6):1–13, 2018. 2

[7] Tadas Baltrušaitis, Chaitanya Ahuja, and Louis-Philippe Morency. Multimodal machine learning: A survey and taxonomy. *IEEE transactions on pattern analysis and machine intelligence*, 41(2):423–443, 2018. 2

[8] Emad Barsoum, John Kender, and Zicheng Liu. Hp-gan: Probabilistic 3d human motion prediction via gan. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 1418–1427, 2018. 2

[9] Yu-Wei Chao, Jimei Yang, Brian Price, Scott Cohen, and Jia Deng. Forecasting human dynamics from static images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 548–556, 2017. 2

[10] Noshaba Cheema, Somayeh Hosseini, Janis Sprenger, Erik Herrmann, Han Du, Klaus Fischer, and Philipp Slusallek. Dilated temporal fully-convolutional network for semantic segmentation of motion capture data. *arXiv preprint arXiv:1806.09174*, 2018. 2

[11] Hsu-kuang Chiu, Ehsan Adeli, Borui Wang, De-An Huang, and Juan Carlos Niebles. Action-agnostic human pose forecasting. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1423–1432. IEEE, 2019. 3

[12] 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. 3

[13] Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D Manning. What does bert look at? an analysis of bert's attention. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 276–286, 2019. 4

[14] Qiongjie Cui, Huaijiang Sun, Yue Kong, Xiaoqian Zhang, and Yanmeng Li. Efficient human motion prediction using temporal convolutional generative adversarial network. *Information Sciences*, 545:427–447, 2021. 2

[15] Qiongjie Cui, Huaijiang Sun, and Fei Yang. Learning dynamic relationships for 3d human motion prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6519–6527, 2020. 2

[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. 2, 3, 4

[17] Han Du, Erik Herrmann, Janis Sprenger, Noshaba Cheema, Somayeh Hosseini, Klaus Fischer, and Philipp Slusallek. Stylistic locomotion modeling with conditional variational autoencoder. In *Eurographics (Short Papers)*, pages 9–12, 2019. 2

[18] Yong Du, Wei Wang, and Liang Wang. Hierarchical recurrent neural network for skeleton based action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1110–1118, 2015. 3

[19] Joao P Ferreira, Thiago M Coutinho, Thiago L Gomes, José F Neto, Rafael Azevedo, Renato Martins, and Erickson R Nascimento. Learning to dance: A graph convolutional adversarial network to generate realistic dance motions from audio. *Computers & Graphics*, 94:11–21, 2020. 2

[20] 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

[21] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. Texture synthesis using convolutional neural networks. *arXiv preprint arXiv:1505.07376*, 2015. 6

[22] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2414–2423, 2016. 6

[23] 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

[24] Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *arXiv preprint arXiv:1406.2661*, 2014. 2

[25] 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

[26] Ikhsanul Habibie, Daniel Holden, Jonathan Schwarz, Joe Yearsley, and Taku Komura. A recurrent variational autoencoder for human motion synthesis. In *28th British Machine Vision Conference*, 2017. 2

[27] Eva Hanser, Paul Mc Kevitt, Tom Lunney, and Joan Condell. Scenemaker: Intelligent multimodal visualisation of natural language scripts. In *Irish Conference on Artificial Intelligence and Cognitive Science*, pages 144–153. Springer, 2009. 1

[28] Irtiza Hasan, Francesco Setti, Theodore Tsesmelis, Vasileios Belagiannis, Sikandar Amin, Alessio Del Bue, Marco Cristani, and Fabio Galasso. Forecasting people trajectories and head poses by jointly reasoning on tracklets and vislets. *IEEE transactions on pattern analysis and machine intelligence*, 2019. 2

[29] Irtiza Hasan, Francesco Setti, Theodore Tsesmelis, Alessio Del Bue, Marco Cristani, and Fabio Galasso. " seeing is believing": Pedestrian trajectory forecasting using visual frustum of attention. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1178–1185. IEEE, 2018. 2

[30] Jun Hatori, Yuta Kikuchi, Sosuke Kobayashi, Kuniyuki Takahashi, Yuta Tsuboi, Yuya Unno, Wilson Ko, and Jethro Tan. Interactively picking real-world objects with unconstrained spoken language instructions. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3774–3781. IEEE, 2018. 3

[31] Gustav Eje Henter, Simon Alexanderson, and Jonas Beskow. Moglow: Probabilistic and controllable motion synthesis using normalising flows. *ACM Transactions on Graphics (TOG)*, 39(6):1–14, 2020. 2

[32] Karl Moritz Hermann, Felix Hill, Simon Green, Fumin Wang, Ryan Faulkner, Hubert Soyer, David Szepesvari, Wojciech Marian Czarnecki, Max Jaderberg, Denis Teplyashin, et al. Grounded language learning in a simulated 3d world. *arXiv preprint arXiv:1706.06551*, 2017. 3

[33] Daniel Holden, Taku Komura, and Jun Saito. Phase-functioned neural networks for character control. *ACM Transactions on Graphics (TOG)*, 36(4):1–13, 2017. 2

[34] Daniel Holden, Jun Saito, and Taku Komura. A deep learning framework for character motion synthesis and editing. *ACM Transactions on Graphics (TOG)*, 35(4):1–11, 2016. 2, 5

[35] 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

[36] Deok-Kyeong Jang and Sung-Hee Lee. Constructing human motion manifold with sequential networks. In *Computer Graphics Forum*, volume 39, pages 314–324. Wiley Online Library, 2020. 2, 4

[37] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5

[38] Jogendra Nath Kundu, Himanshu Buckchash, Priyanka Mandikal, Anirudh Jamkhandi, Venkatesh Babu RADHAKRISHNAN, et al. Cross-conditioned recurrent networks for long-term synthesis of inter-person human motion interactions. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2724–2733, 2020. 2

[39] Jogendra Nath Kundu, Maharshi Gor, and R Venkatesh Babu. Bihmp-gan: Bidirectional 3d human motion prediction gan. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 8553–8560, 2019. 2

[40] Jasper LaFortune and Kristen L Macuga. Learning movements from a virtual instructor: Effects of spatial orientation, immersion, and expertise. *Journal of Experimental Psychology: Applied*, 24(4):521, 2018. 1

[41] Colin Lea, Michael D Flynn, Rene Vidal, Austin Reiter, and Gregory D Hager. Temporal convolutional networks for action segmentation and detection. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 156–165, 2017. 2

[42] 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

[43] 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

[44] Angela S Lin, Lemeng Wu, Rodolfo Corona, Kevin Tai, Qixing Huang, and Raymond J Mooney. Generating animated videos of human activities from natural language descriptions. *Learning*, 2018:2, 2018. 2, 3, 5, 6, 7, 8

[45] Christian Mandery, Ömer Terlemez, Martin Do, Nikolaus Vahrenkamp, and Tamim Asfour. Unifying representations and large-scale whole-body motion databases for studying human motion. *IEEE Transactions on Robotics*, 32(4):796–809, 2016. 5

[46] Karttikeya Mangalam, Ehsan Adeli, Kuan-Hui Lee, Adrien Gaidon, and Juan Carlos Niebles. Disentangling human dynamics for pedestrian locomotion forecasting with noisy supervision. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2784–2793, 2020. 2

[47] 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

[48] Hongyuan Mei, Mohit Bansal, and Matthew Walter. Listen, attend, and walk: Neural mapping of navigational instructions to action sequences. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 30, 2016. 3

[49] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. *arXiv preprint arXiv:1310.4546*, 2013. 7

[50] 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

[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] Matthias Plappert, Christian Mandery, and Tamim Asfour. The KIT motion-language dataset. *Big Data*, 4(4):236–252, dec 2016. 4, 5, 6, 8

[53] Matthias Plappert, Christian Mandery, and Tamim Asfour. Learning a bidirectional mapping between human whole-body motion and natural language using deep recurrent neu-

ral networks. *Robotics and Autonomous Systems*, 109:13 – 26, 2018. 1, 2, 3

[54] Jürgen Schmidhuber and Sepp Hochreiter. Long short-term memory. *Neural Comput*, 9(8):1735–1780, 1997. 4

[55] Soshi Shimada, Vladislav Golyanik, Weipeng Xu, and Christian Theobalt. Physcap: Physically plausible monocular 3d motion capture in real time. *ACM Transactions on Graphics (TOG)*, 39(6):1–16, 2020. 8

[56] Richard Socher, Milind Ganjoo, Hamsa Sridhar, Osbert Bastani, Christopher D Manning, and Andrew Y Ng. Zero-shot learning through cross-modal transfer. *arXiv preprint arXiv:1301.3666*, 2013. 8

[57] Wataru Takano, Dana Kulic, and Yoshihiko Nakamura. Interactive topology formation of linguistic space and motion space. In *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1416–1422. IEEE, 2007. 3

[58] Wataru Takano and Yoshihiko Nakamura. Bigram-based natural language model and statistical motion symbol model for scalable language of humanoid robots. In *2012 IEEE International Conference on Robotics and Automation*, pages 1232–1237. IEEE, 2012. 3

[59] Wataru Takano and Yoshihiko Nakamura. Statistical mutual conversion between whole body motion primitives and linguistic sentences for human motions. *The International Journal of Robotics Research*, 34(10):1314–1328, 2015. 3

[60] Wataru Takano and Yoshihiko Nakamura. Symbolically structured database for human whole body motions based on association between motion symbols and motion words. *Robotics and Autonomous Systems*, 66:75–85, 2015. 3

[61] Kenta Takeuchi, Dai Hasegawa, Shinichi Shirakawa, Naoshi Kaneko, Hiroshi Sakuta, and Kazuhiko Sumi. Speech-to-gesture generation: A challenge in deep learning approach with bi-directional lstm. In *Proceedings of the 5th International Conference on Human Agent Interaction*, pages 365–369, 2017. 2

[62] Yongyi Tang, Lin Ma, Wei Liu, and Weishi Zheng. Long-term human motion prediction by modeling motion context and enhancing motion dynamic. *arXiv preprint arXiv:1805.02513*, 2018. 2

[63] Ian Tenney, Patrick Xia, Berlin Chen, Alex Wang, Adam Poliak, R Thomas McCoy, Najoung Kim, Benjamin Van Durme, Samuel R Bowman, Dipanjan Das, et al. What do you learn from context? probing for sentence structure in contextualized word representations. *arXiv preprint arXiv:1905.06316*, 2019. 4

[64] Ömer Terlemez, Stefan Ulbrich, Christian Mandery, Martin Do, Nikolaus Vahrenkamp, and Tamim Asfour. Master motor map (mmm)—framework and toolkit for capturing, representing, and reproducing human motion on humanoid robots. In *2014 IEEE-RAS International Conference on Humanoid Robots*, pages 894–901. IEEE, 2014. 5

[65] Jesse Vig. A multiscale visualization of attention in the transformer model. *arXiv preprint arXiv:1906.05714*, 2019. 4

[66] Ruben Villegas, Jimei Yang, Yuliang Zou, Sungryull Sohn, Xunyu Lin, and Honglak Lee. Learning to generate long-term future via hierarchical prediction. In *international conference on machine learning*, pages 3560–3569. PMLR, 2017. 2

[67] Jacob Walker, Kenneth Marino, Abhinav Gupta, and Martial Hebert. The pose knows: Video forecasting by generating pose futures. In *Proceedings of the IEEE international conference on computer vision*, pages 3332–3341, 2017. 2

[68] Zhiyong Wang, Jinxiang Chai, and Shihong Xia. Combining recurrent neural networks and adversarial training for human motion synthesis and control. *IEEE transactions on visualization and computer graphics*, 27(1):14–28, 2019. 2

[69] Erwin Wu and Hideki Koike. Real-time human motion forecasting using a rgb camera. In *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pages 1575–1577. IEEE, 2019. 2

[70] Jingwei Xu, Huazhe Xu, Bingbing Ni, Xiaokang Yang, Xiaolong Wang, and Trevor Darrell. Hierarchical style-based networks for motion synthesis. *arXiv preprint arXiv:2008.10162*, 2020. 2

[71] Tatsuro Yamada, Hiroyuki Matsunaga, and Tetsuya Ogata. Paired recurrent autoencoders for bidirectional translation between robot actions and linguistic descriptions. *IEEE Robotics and Automation Letters*, 3(4):3441–3448, 2018. 3

[72] Tatsuro Yamada, Shingo Murata, Hiroaki Arie, and Tetsuya Ogata. Dynamical integration of language and behavior in a recurrent neural network for human–robot interaction. *Frontiers in neurorobotics*, 10:5, 2016. 3

[73] Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2015. 2