SINC: Spatial Composition of 3D Human Motions for Simultaneous Action Generation

Nikos Athanasiou*[1] Mathis Petrovich*[1,2] Michael J. Black[1] Gül Varol[2]
[1]Max Planck Institute for Intelligent Systems, Tübingen, Germany
[2]LIGM, École des Ponts, Univ Gustave Eiffel, CNRS, France
sinc.is.tue.mpg.de

Abstract

Our goal is to synthesize 3D human motions given textual inputs describing simultaneous actions, for example 'waving hand' while 'walking' at the same time. We refer to generating such simultaneous movements as performing spatial compositions. In contrast to temporal compositions that seek to transition from one action to another, spatial compositing requires understanding which body parts are involved in which action, to be able to move them simultaneously. Motivated by the observation that the correspondence between actions and body parts is encoded in powerful language models, we extract this knowledge by prompting GPT-3 with text such as "what are the body parts involved in the action <action name>?", while also providing the parts list and few-shot examples. Given this action-part mapping, we combine body parts from two motions together and establish the first automated method to spatially compose two actions. However, training data with compositional actions is always limited by the combinatorics. Hence, we further create synthetic data with this approach, and use it to train a new state-of-the-art text-to-motion generation model, called SINC ("SImultaneous actioN Compositions for 3D human motions"). In our experiments, we find that training with such GPT-guided synthetic data improves spatial composition generation over baselines. Our code is publicly available at sinc.is.tue.mpg.de.

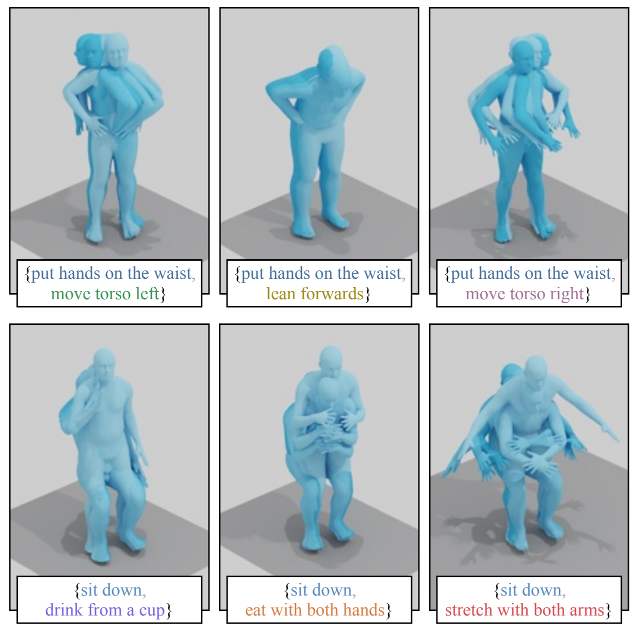

Figure 1. **Goal:** We demonstrate the task of *spatial compositions* in human motion synthesis. We generate 3D motions for a pair of actions, defined by a pair of textual descriptions. Here, we provide six sample input-output illustrations from our model. For example, we input the set of actions{'put hands on the waist', 'move torso left'} and generate one motion that simultaneously performs both.

1. Introduction

Text-conditioned 3D human motion generation has recently attracted increasing interest in the research community [4, 15, 44], where the task is to input natural language descriptions of actions and to output motion sequences that semantically correspond to the text. Such controlled motion synthesis has a variety of applications in fields that rely on motion capture data, such as special effects, games, and virtual reality. While there have been promising results in this direction, *fine-grained* descriptions remain out of reach. Consider the scenario in which a movie production needs a particular motion of someone jumping down from a building. One may generate an initial motion with one description, and then gradually refine it until the desired motion is obtained, e.g., {'jumping down', 'with arms behind the back', 'while bending the knees'}. State-of-the-art methods [9, 44] often fail to produce reasonable motions when conditioned on fine-grained text describing multiple actions. In this work, we take a step towards this goal by focusing on the *spatial composition* of motions. In other words, we aim to generate one motion depicting multiple simultaneous actions; see Figure 1. This paves the way for

*Equal contribution

further research on fine-grained human motion generation.

Previous work [2, 13, 33, 44] initially explored the text-conditioned motion synthesis problem on the small-scale KIT Motion-Language dataset [46]. Recently, work [4, 15] has shifted to the large-scale motion capture collection AMASS [37], and its language labels from BABEL [47] or HumanML3D [15]. In particular, similar to this work, TEACH [4] focuses on fine-grained descriptions by addressing temporal compositionality, that is, generating a sequence of actions, one *after* the other. We argue that composition in time is simpler for a model to learn since the main challenge is to smoothly transition between actions. This does not necessarily require action-specific knowledge, and a simple interpolation method such as Slerp [51] may provide a decent solution. On the other hand, there is no such trivial solution for compositions in *space*, since one needs to know action-specific body parts to combine two motions. If one knows that 'waving' involves the hand and 'walking' involves the legs, then compositing the two actions can be performed by cutting and pasting the hand motion into the walking motion. This is often done manually in the animation industry.

To automate this process, we observe that pretrained language models such as GPT-3 [7] encode knowledge about which body parts are involved in different actions. This allows us to first establish a spatial composition baseline (analogous to the Slerp baseline for temporal compositions); i.e., independently generating actions then combining with heuristics. Not surprisingly, we find that this is suboptimal. Instead, we use the synthesized compositions of actions as additional training data for a text-to-motion network. This enriched dataset enables our model, called SINC ("SImultaneous actioN Compositions for 3D human motions"), to outperform the baseline. Our GPT-based approach is similar in spirit to work that incorporates external linguistic knowledge into visual tasks [6, 60, 64].

While BABEL [47] and HumanML3D [15] have relatively large vocabularies of actions, they contain a limited number of *simultaneous* actions. A single temporal segment is rarely annotated with multiple texts. For example, BABEL contains only roughly 2.5K segments with simultaneous actions, while it has ∼25K segments with only one action. This highlights the difficulty of obtaining compositional data at scale. Moreover, for any reasonably large set of actions, it is impractical to collect data for all possible pairwise, or greater, combinations of actions such that there exists no unseen combination at test time [62, 64]. With existing datasets, it is easy to learn spurious correlations. For example, if waving is only ever observed by someone standing, a model will learn that waving involves moving the arm with straight legs. Thus generating waving and sitting would be highly unlikely. In our work, we address this challenge by artificially creating compositional data for training using GPT-3. By introducing more variety, our generative model is better able to understand what is essential to an action like 'waving'.

Our method, SINC, extends the generative text-to-motion model TEMOS [44] such that it becomes robust to input text describing more than one action, thanks to our synthetic training. We intentionally build on an existing model to focus the analysis on our proposed synthetic data. Given a mix of real single actions, real pairs of actions, and synthetic pairs of actions, we train a probabilistic text-conditioned motion generation model. We introduce several baselines to measure sensitivity to the model design, as well as to check whether our learned motion decoder outperforms a simpler compositing technique (i.e., simply using our GPT-guided data creation approach, along with a single-action generation model). We observe limited realism when compositing different body parts together, and need to incorporate several heuristics, for example when merging motions whose body parts overlap. While such synthetic data is imperfect, it helps the model disentangle the body parts that are relevant for an action and avoid learning spurious correlations. Moreover, since our motion decoder has also access to real motions, it learns to generate realistic motions, eliminating the realism problem of the synthetic composition baseline.

Our contributions are the following: (i) We establish a new benchmark on the problem of spatial compositions for 3D human motions, compare a number of baseline models on this new problem, and introduce a new evaluation metric that is based on a motion encoder that has been trained with text supervision. (ii) To address the data scarcity problem, we propose a GPT-guided synthetic data generation scheme by combining action-relevant body parts from two motions. (iii) We provide an extensive set of experiments on the BABEL dataset, including ablations that demonstrate the advantages of our synthetic training, as well as an analysis quantifying the ability of GPT-3 to assign part labels to actions. Our code is available for research purposes.

2. Related Work

Human motion generation. While motion prediction [5, 10, 34, 38, 41, 49, 65, 71], synthesis [18, 31] and in-betweening [19, 27, 54, 73] represent the most common motion-generation tasks, conditional synthesis through other modalities (e.g., text) has recently received increasing interest. Example conditions include music [32, 40], speech [1, 17], scenes [20, 53, 59, 69], action [16, 43] or text [2, 4, 13, 15, 33, 44]. In the following, we focus on work involving text-conditioned motion synthesis, which is most closely related to our work.

3D human motion and natural language. Unlike methods that use categorical action labels to control the motion synthesis [16, 36, 43], text-conditioned methods [2, 4, 13, 15, 33, 44] seek to input free-form language descriptions that go beyond a closed set of classes. The KIT-ML dataset [46] comprises textual annotations for motion capture data, representing the first benchmark for this task. More recently, the larger scale AMASS [37] motion capture collection is labeled with language descriptions by BABEL [47] and HumanML3D [15]. A common solution to text-conditioned synthesis is to design a cross-modal joint space between motions and language [2, 13, 44]. TM2T [14] introduces a framework to jointly

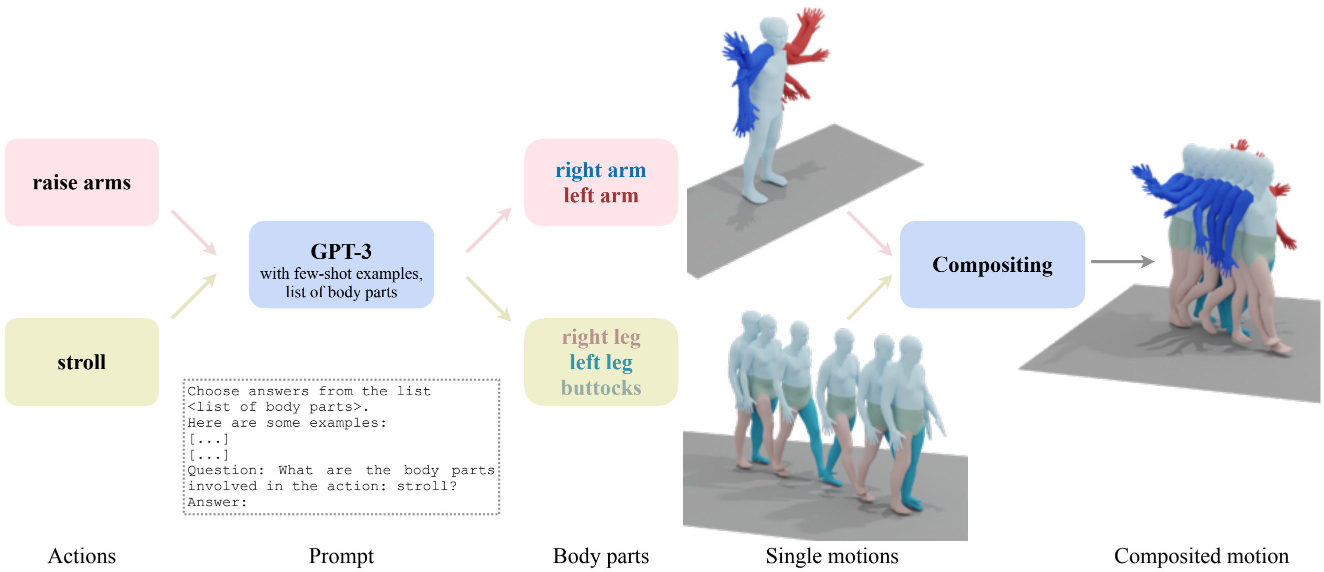

Figure 2. **GPT-guided synthetic training data creation:** We illustrate our procedure to generate Synth-Pairs. Here, we combine two motion sequences from the training set with the corresponding labels 'stroll' and 'raise arms'. We first prompt GPT-3 with the instructions, few-shot examples containing question-answer pairs, and giving the action of interest in the last question without the answer. We minimally post-process the output of GPT-3 to assign this action to a set of body parts. The relevant body parts from each motion are then stitched together to form a new synthetically composited motion.

perform text-to-motion and motion-to-text, integrating a back-translation loss. In contrast to the deterministic methods of [2, 13], TEMOS [44] employs a VAE-based probabilistic approach (building on ACTOR [43]) that can generate multiple motions per textual input, and establishes the state of the art on the KIT benchmark [46] with a non-autoregressive architecture. Following the success of diffusion models [22, 52], very recently, MDM [56], FLAME [28], MotionDiffuse [67], and MoFusion [12] demonstrate diffusion-based motion synthesis. Recent work [9] shows the potential of latent diffusion to address the slow inference limitation. On the other hand, T2M-GPT [66] obtains competitive performance compared with diffusion using VQ-VAEs. Our approach is complementary and applicable to existing models for text-to-motion synthesis. In this work, we adopt TEMOS [44] and retrain it on the data from [47] together with our proposed synthetic compositions.

In contrast to previous work, our focus is on the composition of *simultaneous* actions. Prior work on compositional actions focuses on *temporal* compositions; i.e., inputting a *sequence* of textual descriptions). Early influential work [3] employs dynamic-programming approaches to compose existing motions from a motion database with action labels. Recently, Wang et al. [59] generate a sequence of actions in 3D scenes by synthesizing pose anchors that are then placed in the scene and refined by infilling. TEACH [4] extends TEMOS [44] by incorporating an action-level recursive design that generates the next action conditioned on the past motion. ActionGPT [25] improves this model by retraining it with text augmentations using language models. Concurrently, MultiAct [30] similarly aims to produce continuous transitions between generated actions. In contrast to previous work [4, 25, 30], we focus on *spatial* compositionality, inputting text that describes *simultaneous actions*. In this direction, MotionCLIP [55] and MDM [56] test the compositional capabilities of their methods, but only show preliminary analyses. The concurrent work of MotionDiffuse [67] injects manually labeled body-part information and performs noise interpolation to obtain spatial compositionality.

External linguistic knowledge. Large language models have been exploited for many visual tasks such as instruction-conditioned image editing [6], visual relationship detection [64], and human-object reconstruction [60], among others. Similar to us, Wang et al. [60] incorporate GPT by asking what body part is in contact with a given object, which in turn is used for image-based 3D human-object reconstruction. On the other hand, we exploit GPT to extract knowledge about body parts that are involved in an action. To the best of our knowledge, we are the first to systematically model such body part associations from textual descriptions.

Training with synthetic data. Using synthetic data to train machine learning models is a standard approach for solving many visual recognition tasks, such as 3D body pose estimation [8, 42], 2D body part segmentation [58], 3D hand pose estimation [21], video action recognition [57], 2D body pose estimation [48] pedestrian detection [45], and optical flow estimation [24]. In a similar spirit to us, the recent work of HUMANISE [61] creates a synthetic dataset of human-scene interactions by combining 4 actions from BABEL [47] with 3D scenes, and pairing them with language descriptions. In this work, we generate synthetic training data by combining existing 3D motion assets and language labels to overcome the data scarcity problem for compositional learning, helping our method to avoid learning spurious correlations.

Compositionality. Compositionality has been explored in other areas of computer vision, such as visual relation detection [62], learning object attributes [39], human-object interaction [26], video prediction [63], and video action recognition [11]. For example, Shuffle-then-assemble [62] explicitly forces their visual relation detection model to become object-agnostic to achieve generalization to unseen object pairs. Similarly, COINS [70] aims to generate compositions of human-scene static interactions, where poses that match a text description are generated in a 3D scene. Here, we focus on action compositionality in space, i.e., simultaneity in time.

3. Spatial Composition of Motions from Textual Descriptions

Given a set of action descriptions in the form of text, such as {"walk in a circle", "wave with the right hand"}, and a desired motion duration F, the goal is to probabilistically generate realistic 3D human motions such that all the given actions are performed simultaneously in each generated sequence. We refer to this problem as spatial composition. Note that as a proof of concept, we perform our experiments mainly with pairs of actions, but the framework is applicable beyond pairs.

In the following, we first introduce our framework to generate synthetic training data by extracting correspondence between actions and body parts from large language models (Section 3.1). Then, we describe our model training with synthetically augmented data (Section 3.2), and finally present implementation details (Section 3.3).

3.1. GPT-guided synthetic training data creation

As explained in Section 1, we leverage a large language model, GPT-3 [7], to automatically assign a given action description to a set of body parts from a predefined list. Given such correspondence, we then synthetically combine existing motions together to create compositional training data. This process is illustrated in Figure 2.

Body part label extraction from GPT-3. We process the entire set of motion descriptions in the dataset to associate each action description to a set of body parts. We use the Text-Completion tool from OpenAI's API of GPT-3 [7] to extract the body part correspondence for a given language description. Specifically, for each individual action description in the dataset, we construct a prompt consisting of three parts. (i) We specify the instruction in the form of "choose answers from the list <list of body parts>', where the list is ['left arm', 'right arm', 'left leg', 'right leg', 'torso', 'neck', 'buttocks', 'waist']". (ii) We provide few-shot examples as question-answer pairs, where the question is 'What are the body parts involved in the action: <action>?', and the answer is the list of manually labeled body parts. (iii) The last part has the same form as the question, but we do not give the answer.

With this approach, GPT-3 outputs require minimal processing, i.e., the responses are words that correspond almost always to the provided list in (i). We post-process GPT-3's responses by removing punctuation, lowercasing, and mapping to a list of SMPL [35] body parts that we define separately, and use in the subsequent steps of our approach to generate synthetic data. We take a subset of SMPL body parts: ['left arm', 'right arm', 'left leg', 'right leg', 'torso', 'global orientation']. We coarsely define these six different body parts, but dealing with more fine-grained body parts is certainly possible.

From the first list, 'neck' is mapped to 'torso', and ['waist', buttocks'] are mapped to 'global orientation'. This is because, when prompting for free-form outputs without providing a list (i) or few-shot examples (ii), we qualitatively observe that GPT-3 refers to changes in global orientation of the body using words such as 'waist' or 'buttocks'. Hence, we replace 'global orientation' with these two words instead. GPT-3 also outputs the word 'neck' in some cases even when it is not included in the list, which motivated us to add it to our list.

To evaluate our choices for the prompt, in Table 1 we measure the contribution of providing (i) the list, and (ii) few-shot examples in the prompt. For this, we manually label 100 action descriptions from BABEL. For each action, we annotate each body part as Yes/No/Sometimes to mark whether that body part is involved with that action. Note that we use 'Sometimes' for ambiguous cases, where it is acceptable to include, but not necessarily mandatory. For example 'hands' may or may not be involved in 'walking'. We then check the accuracy of GPT-3 body part labeling, by counting Yes/No as 1/0, ignoring optional body parts to not bias our evaluation.

A prompt asking for a free-form answer (i.e., "List the body parts involved in this action: <action>") complicates the required post-processing as one needs to handle over-detailed answers such as 'deltoids', 'triceps', or different ways of referring to the same body part. We manually built a lookup table to map from GPT-3 outputs to SMPL body parts but obtained suboptimal results. As can be seen from Table 1, providing the list (rows a vs b) significantly boosts the labeling accuracy, especially for picking the correct left/right arm/leg, which is further improved by providing few-shot examples (row c). We provide examples from GPT-3's responses for various prompts in Section B of the Appendix.

Could we extract body part labels without GPT-3? To test the effectiveness of our GPT-based body part labeling, we also implement an alternative body-part labeling approach based on part velocity magnitude. The assumption is that we have action-motion pairs, and if a body part movement is above a threshold, that part should be involved with the associated action. Specifically, we compute average positional velocities

Body part labeling	Global	Torso	Left arm	Right arm	Left leg	Right leg	Mean
Part velocity magnitude	0.72	0.68	0.60	0.55	0.58	0.67	0.65
GPT-based (a) free-form	0.72	0.70	0.85	0.86	0.80	0.83	0.79
GPT-based (b) choose from list	0.79	0.68	0.89	0.90	0.88	0.89	0.84
GPT-based (c) choose from list + few-shot examples	**0.84**	**0.72**	**0.89**	**0.89**	**0.89**	**0.90**	**0.85**

Table 1. **GPT body part labeling performance:** We report the part-labeling accuracy of GPT-3, as well as a simpler baseline based on part velocity magnitudes. For GPT-3, we experiment with various types of prompts on 100 manually annotated actions. (a) Asking which body parts are involved with an action, and post-processing free-form language outputs to associate to part labels. (b) Asking to choose from a given list of body parts, and (c) additionally also providing few-shot examples. See Section 3.1 for more details on these prompts.

across frames for each body part, standardize (subtracting the mean, dividing by the standard deviation over frames), and determine a threshold (by visual inspection) to decide if a body part is involved in a given motion. This heuristic baseline has the disadvantage that it may suffer from spurious correlations (e.g., if we only see waving while walking, we will think that leg motion is critical to waving). From the first row of Table 1, we observe that the accuracy of this approach is significantly lower than the GPT-based approaches.

Body part composition to create new motions. Given a set of labeled motions to combine, and the extracted GPT-3 body parts involved, we first determine if the actions are compatible; i.e., whether a valid motion can be composited, based on the descriptions. For example, the actions ['walking', 'kicking with the right leg'] may not be performed at the same time as they both include the body part 'right leg'. For the synthetic training data, we only create compositions for valid pairs that are compatible in terms of their body part involvement, and use *real* motions from the database. Next, we detail the data creation procedure.

Given two motions A and B, along with the corresponding selected body parts extracted by GPT-3, we compose these motions into a new one by performing the following steps: (1) We trim the longer motion to match the length of the shorter one; (2) We order the motions A and B such that motion B always has fewer body parts than motion A; (3) If motion B involves at least one leg or the global orientation, we also select both legs, the global orientation, and translation from motion B (otherwise, we obtain these 4 values from motion A); (4) The remaining unselected body parts (if any) are taken from motion A; (5) The composited motion is obtained by combining selected body parts from motion A and B, along with the translation according to step 3. We perform step 3 to retain plausibility as much as possible, as the leg motions are highly correlated with changes in global translation and orientation. This procedure ensures realism and accuracy of the compositions to some extent; but does not provide a guarantee.

Note that we also employ this approach as a baseline in our experiments, where we combine the motions under these assumptions using two *generated* motions from a single-action trained model. In this case, body part incompatibilities may

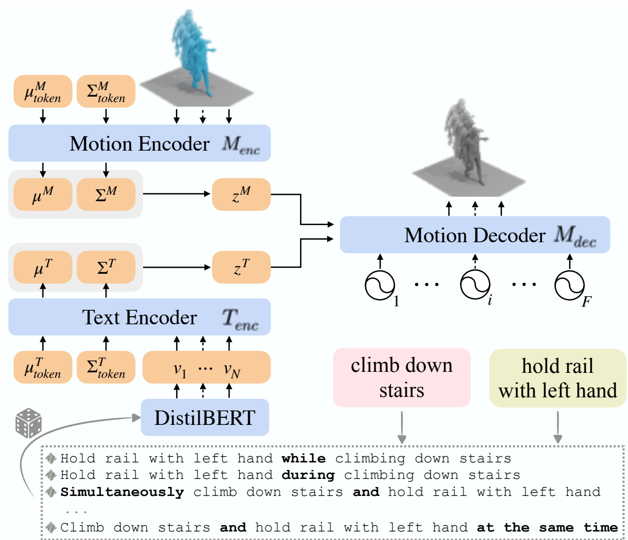

Figure 3. **Model architecture:** We extend TEMOS [44] such that it is trained with compositional actions. We build multiple descriptions given two action labels, by adding words such as 'while', 'during', etc. We then randomly sample one version during training as input to the text encoder.

occur ('walking' and 'kicking' both involve the leg), and body parts from motion B override the conflicting parts from motion A (see Section C of the Appendix for further details).

3.2. Learning to generate spatial compositions

We employ the recent architecture TEMOS [44], which encodes the text into a distribution via a Transformer encoder (text encoder $\mathcal{T}_{enc}$), and produces motions by using a Transformer decoder (motion decoder $\mathcal{M}_{dec}$). Similar to Language2Pose [2], TEMOS contains a motion encoder ($\mathcal{M}_{enc}$) and encourages a cross-modal joint space between text and motion embeddings. A simplified overview of the architecture can be seen in Figure 3. At test time, the motion encoder is not used.

The motion encoder takes as input a body motion sequence $B \in \mathbb{R}^{l \times d_f}$ where d_f is the feature dimension and l the maximum motion length and outputs a single latent vector z^M and a distribution $\mathcal{N}(\mu^M, \Sigma^M)$. Similarly, the text encoder outputs z^T, which is sampled from the distribution $\mathcal{N}(\mu^T, \Sigma^T)$. These distribution parameters are obtained by appending two extra learnable tokens in the transformer encoder, and taking their corresponding outputs [43]. The latent vectors are sampled using the re-parametrization trick [29]. The motion decoder then takes as input (a) the duration encoded by positional encodings $F \in \mathbb{R}^{l \times d}$, where l is the maximum motion length and d the latent dimension, (b) along with either the motion z^M or text z^T latent vector.

The model is supervised with the standard normal distribution losses, $\mathcal{L}_{\mathcal{KL}}^{\mathcal{T}} = \mathcal{KL}(\mathcal{N}(\mu^T, \Sigma^T), \mathcal{N}(0, I))$ and $\mathcal{L}_{\mathcal{KL}}^{\mathcal{M}} = \mathcal{KL}(\mathcal{N}(\mu^M, \Sigma^M), \mathcal{N}(0, I))$ for the text and motion distributions, respectively. Moreover, $\mathcal{L}_{\mathcal{Z}} = \tilde{\mathcal{L}}_1(z^T, z^M)$ is used to force the text latent vectors to be close to the motion latent vector, where $\tilde{\mathcal{L}}_1$ is the smooth L1 loss. Finally, the distributions of different texts and the motion are supervised via $\mathcal{L}_{\mathcal{KL}}^{\mathcal{M}\|\mathcal{T}} = \mathcal{KL}(\mathcal{N}(\mu^T, \Sigma^T), \mathcal{N}(\mu^M, \Sigma^M))$ and its symmetric version $\mathcal{L}_{\mathcal{KL}}^{\mathcal{T}\|\mathcal{M}}$. The reconstruction losses for the generated motions, $\hat{B}^M$ and $\hat{B}^T$, from both the motion and the text branches, $\mathcal{L}_{\mathcal{R}} = \tilde{\mathcal{L}}_1(B, \hat{B}^T) + \tilde{\mathcal{L}}_1(B, \hat{B}^M)$, are added to the total loss:

$$\mathcal{L} = \mathcal{L}_{\mathcal{KL}}^{\mathcal{T}} + \mathcal{L}_{\mathcal{KL}}^{\mathcal{M}} + \mathcal{L}_{\mathcal{KL}}^{\mathcal{M}\|\mathcal{T}} + \mathcal{L}_{\mathcal{KL}}^{\mathcal{T}\|\mathcal{M}} + \mathcal{L}_{\mathcal{R}} + \mathcal{L}_{\mathcal{Z}}. \qquad (1)$$

While our experiments use TEMOS [44], our synthetic data strategy is applicable to any text-to-motion generation model. We provide further evidence on the benefits of synthetic training on a diffusion-based approach (similar to MLD [9]) in Section A of the Appendix.

Input text format and augmentations. Here, we describe how we provide the input to the text encoder. In case of a single motion that is described by one action label, we simply input the original label as in [44]. In case of two or more descriptions, which is the focus of this work, we combine multiple descriptions into a single text. Specifically, we use several keywords to describe simultaneous actions (e.g., 'while', 'at the same time', 'simultaneously', 'during', etc.), and randomly place them in the text description to form an input that imitates a free-form input. Moreover, we shuffle the order of the labels, and add inflections to verbs such as gerunds when grammatically applicable; e.g., when using 'while'. Figure 3 shows some examples. Such and input formation allows users to enter free-form language descriptions at test time, which is a natural interface for humans. During training, we pick a random text augmentation, and at test time, we evaluate all the models using the conjunction word 'while'. In Section E.1 of the Appendix, we provide results with more conjunction words both seen and unseen during training.

3.3. Implementation details

We define a 3D human motion as a sequence of human poses using the SMPL body model [35]. As in TEMOS [23, 44], we represent the motion using the 6D rotations [72] for body joints and the 2D-projection of the x,y trajectory along with the z translation. This results in $d_f = 135$ for each body pose in each motion sequence. All the motions are canonicalized to face the same forward direction and are standardized.

The input text is encoded with DistilBERT [50] (whose parameters are frozen), followed by a learnable linear projection. The latent dimension is fixed to $d = 256$. We use 6 layers and heads in the transformers with a linear projection of size 1024. We set the batch size to 64 and the learning rate to $3 \cdot 10^{-4}$ for all our experiments.

Our model is applicable to arbitrary numbers of actions for a given motion. Therefore, we jointly train on single actions, and multiple actions. Single actions are from real data. Multiple actions can be (i) from synthetic pairs that are randomly generated 'on the fly' or (ii) from real data where most such motions have two labels, but we also include those with more than two; see the supplementary video on our project page for more details. For each sequence in a mini-batch, if it is a real single action, with probability p, we combine it randomly with another compatible action.

4. Experiments

We present data and evaluation metrics (Section 4.1), followed by the baselines we introduce (Section 4.2). We report quantitative experimental results with ablations (Sections 4.3 and 4.4). We conclude with a qualitative analysis (Section 4.5) and a discussion of limitations (Section 4.6).

4.1. Data and evaluation metrics

We use the **BABEL** dataset [47], to exploit its unique potential to study simultaneous actions. Some BABEL motions come with multiple language descriptions where annotations can overlap in time. We extract all such simultaneous action pairs for both training (2851 motions), and validation sets (1232 motions). We only consider the sequences that have a length between 600 (20 sec.) and 15 (0.5 sec.) frames. From the validation set, we exclude redundant pairs with the label 'stand', because this commonly occurs in the data while not representing challenging cases. We also remove pairs that are *seen* in the training set, and end up with 667 sequences that contain two simultaneous actions. The results on the full validation set are provided in Section E.4 of the Appendix. Besides the simultaneous pairs, we include the single-action data from BABEL in training. Specifically, there are 24066 and 8711 single-action motions for training and validation sets, respectively. In our experiments, we denote the simultaneous actions from BABEL with **Real-Pairs**, the single-motion segments from BABEL with **Real-Singles**, and our synthetic data created by using body-part labels from GPT with **Synth-Pairs**. We perform evaluation only on the real spatial pairs of the BABEL validation set to assess the quality of simultaneous action generation. We use the validation set as test set and train all of our models for 500 epochs.

We report evaluation metrics adopted by [4, 13, 44]: Average Positional Error (APE), and Average Variational Error (AVE). However, we observe that these metrics do not always correlate well with the visual quality of motions, nor their semantic correspondence. We introduce, and additionally report, a new

Model	Tr. Data		TEMOS ↑	Average Positional Error ↓				Average Variance Error ↓			
	Real-P	Real-S	score	root joint	global traj.	mean local	mean global	root joint	global traj.	mean local	mean global
Single-action	✗	✓	0.601	0.592	0.551	0.286	0.712	0.076	0.075	0.013	0.083
Single-action GPT-compositing	✗	✓	0.618	0.546	0.507	0.282	0.666	0.076	0.075	0.013	0.082
SINC-STE	✓	✗	0.614	0.636	0.615	0.275	0.743	0.082	0.081	0.014	0.090
SINC	✓	✗	0.631	0.703	0.682	0.269	0.815	0.107	0.106	0.013	0.114
SINC	✓	✓	**0.640**	0.601	0.573	0.268	0.724	0.093	0.092	0.012	0.100

Table 2. **Baseline comparison:** We train only with Real-Pairs of the BABEL dataset and report performance when compositing naively or with GPT-3 annotations. Furthermore, we ablate the model design for handling multiple textual inputs when extending TEMOS [44]. We observe better performance at handling action pairs with a single text encoder (SINC) that takes as input the two text labels as a single free-form description with various augmentations, as described in Section 3.2, compared to separate text encodings of the labels (SINC-STE). Moreover, we report the performance of SINC when adding Real-Singles, as well.

Synthetic data	Training Data			TEMOS ↑	Average Positional Error ↓				Average Variance Error ↓			
	Real-P	Real-S %	Synth-P %	score	root joint	global traj.	mean local	mean global	root joint	global traj.	mean local	mean global
N/A	✓	0	0	0.631	0.703	0.682	0.269	0.815	0.107	0.106	0.013	0.114
	✓	100	0	0.640	0.601	0.573	0.268	0.724	0.093	0.092	0.012	0.100
Random composition	✗	0	100	0.539	0.489	0.434	0.291	0.595	0.075	0.074	0.012	0.082
	✗	50	50	0.540	0.587	0.535	0.288	0.687	0.077	0.076	0.012	0.083
	✓	0	100	0.619	0.485	0.438	0.272	0.602	0.074	0.073	0.011	0.081
	✓	50	50	0.617	0.454	0.394	0.272	0.560	0.069	0.068	0.011	0.075
GPT composition	✗	0	100	0.618	0.478	0.451	0.265	0.610	0.063	0.062	0.012	0.070
	✗	50	50	0.541	0.646	0.598	0.290	0.747	0.078	0.077	0.012	0.085
	✓	0	100	0.642	0.553	0.527	0.266	0.671	0.061	0.060	0.011	0.068
	✓	50	50	**0.644**	0.481	0.452	0.261	0.605	0.064	0.062	0.011	0.070

Table 3. **Contribution of the synthetic data:** We report performance when including two types of synthetic data created by body part combination, either determined by GPT or randomly. We further experiment (i) with different percentages of sampling ratios between the Real-Singles and Synth-Pairs, and (ii) with the inclusion of Real-Pairs.

TEMOS score, which compares the cosine similarity between the generated motion and the ground truth after encoding them into the motion encoder of TEMOS [44], which is trained on BABEL Real-Singles (we do not observe significant changes when altering this model with TEMOS trained on different data; see Section E.2 of the Appendix). This is similar in spirit to BERTScore [68], which evaluates text generation quality by comparing to the ground truth in the text embedding space. More details can be found in Section D of the Appendix. While this metric is also imperfect (e.g., it still assumes a single ground truth action), we observe that it better correlates with realism and motion semantics as it has been trained to encode motions controlled by text descriptions. An alternative performance measure is adopted by [15] that reports motion-to-text retrieval metrics, randomly selecting for each motion 31 negative text descriptions along with the ground truth. Finally, we include diversity metrics in Section E.3 of the Appendix.

4.2. Single-action baselines

In the following, we introduce and describe two baselines using a model trained with one description per motion: (i) A naive single-action baseline that relies on a text-to-motion synthesis model trained on single actions, tested on pairs of actions. (ii) Our proposed GPT-compositing applied on independent motion generations from a single-action model.

Single-action model. Our first baseline tests the ability of single-action models to synthesize compositions by only modifying the input text. We train with Real-Singles from BABEL. At test time, we concatenate the text descriptions using 'while' as a keyword and evaluate the generated motions.

Single-action GPT-compositing. Another single-action baseline generates two independent motions given two texts, which are then combined using our proposed GPT-guided composition, stitching body parts from two motions (as described in our synthetic data creation; see Section 3.1). Note that unlike the synthetic data, which combines real motions, this baseline combines generated motions. The disadvantage of this model is that it requires GPT at test time, and is based on heuristics that may be error-prone, such as trimming the motions to the same duration, and resolving common body part labels (see the supplementary video on our project page for details). In the presence of a model that is trained only on individual actions (Real-Singles), we observe that the GPT-based compositing of two independent generations improves the performance over the single-action baseline (as shown in Table 2 top). Based on qualitative observation (see Section 4.5), the single-action baseline often generates one out of the two actions. The GPT-compositing baseline better captures both actions; however, lacks realism due to composing actions with heuristics. SINC, which trains on compositional data, alleviates both issues.

4.3. The effect of the input text format

To confirm whether our free-form input format sacrifices performance compared to a more controlled alternative of keeping the two action texts separate, we experiment with a variant of our SINC model by changing the text encoding. Instead of a single text combining two actions, we concatenate them together with a learnable separation token in between after independently encoding the actions with DistilBERT. We refer to this separate text encoding variant as SINC-STE. In Table 2, we compare SINC with SINC-STE when trained only with Real-Pairs, and observe a better TEMOS score with the free-form text augmentations, at the cost of worse positional errors. We observe that metrics based on joint positions may score high even in the absence of the second action, especially if it involves a fine-grained motion (see supplementary video). Besides quantitative performance, SINC has the advantage of allowing more flexible inputs.

4.4. Training with different sets of data

Contribution of Real-Singles and Real-Pairs. In Table 2, we report the performance of SINC when adding both Real-Pairs and Real-Singles to training. We see that training with the large number of single actions of BABEL, in addition to the small amount of action pairs, improves performance, and highlights the limited scale of the available pairs.

Contribution of GPT-guided Synth-Pairs. We experiment with different training sources in Table 3, mainly to assess the effect of adding synthetic training data. The percentages (0, 50, or 100) reflect the probability p that a real-single action is composited synthetically with another action (see Section 3.3). When using all training data (i.e., Real-P, Real-S 50%, Synth-P 50%), we obtain the best TEMOS score, and more importantly observe better qualitative results (see Figure 5). In particular, the model trained with GPT-guided synthetic data demonstrates superior generalization capability to unseen combinations. In the supplementary video, we provide results with input combinations that are unseen both in the real training and validation sets.

Synthetic data without GPT guidance. We further test whether our GPT-guidance to generate synthetic data is better than just randomly mixing body parts (Random composition). In Table 3, GPT compositions outperform Random compositions, especially when training only on synthetic data (0.539 vs 0.618 TEMOS score).

4.5. Qualitative analysis

In Figure 5 (a), we present simultaneous action generations using SINC for the validation set of BABEL. We show one random generation from our model for each description pair (left), along with the ground truth (right). Note that we display one sample due to space constraints, but the model can synthesize multiple diverse motions per input. We observe that, while being sometimes different from the ground-truth motion, our generations follow the semantics of *both* actions, achieving spatial compositionality. Moreover, we qualitatively compare different models trained with and without synthetic data in Figure 5 (b), for the pair {'stretch', 'sit down'} and {'bend torso right', 'put hands on hips'}. This action pair combination is unseen in Real-Pairs, but is seen in the Synthetic-Pairs data. In both cases, the Single-action model and the model that has not been trained on Synthetic-Pairs (first two columns) fail to generate the motion in contrast to SINC which is trained on spatial compositions.

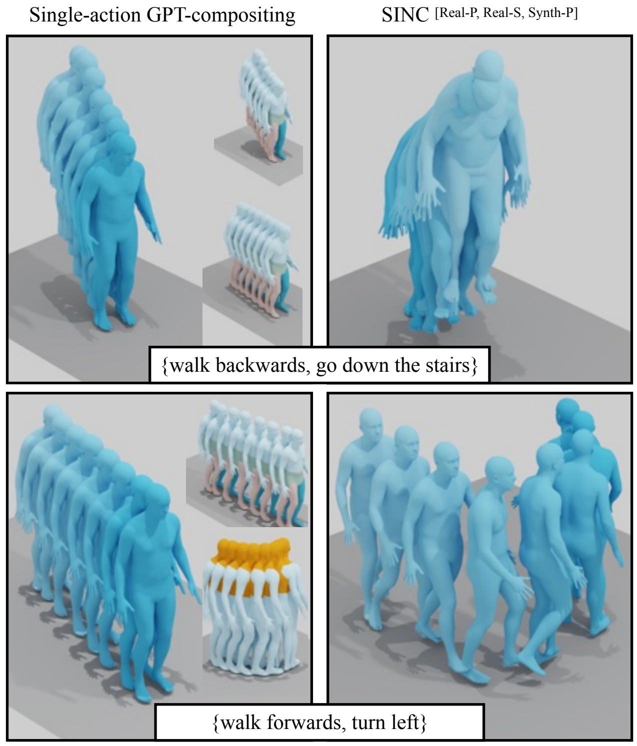

Figure 4. **Single-action GPT-compositing vs SINC:** We show two examples that highlight the advantage of our model compared to GPT compositions. Top: The detected body parts overlap causing the stitching to generate a forwards movement. Bottom: The global orientation is taken from the 'walk forwards' failing to generate a left turn.

Finally, in Figure 4 we show failure cases of GPT-composition. Our baseline fails to generate a motion that corresponds to the instruction when the body parts are overlapping (top row). Another failure case happens when global orientation is important for the semantics of an action ('turn left') and is assigned to the walking action since it involves both feet (bottom row).

4.6. Limitations

Our framework relies on synthetic data creation by combining arbitrary motions together. Even if the body parts are compatible, in real life, not all actions appear simultaneously together. Future work should also explore the *semantic* compatibility between actions by extracting this knowledge from language models to construct semantically meaningful compositions. However, language models are also prone to mistakes. In particular, GPT-3 body part labels may be insufficient or ambiguous (e.g.,

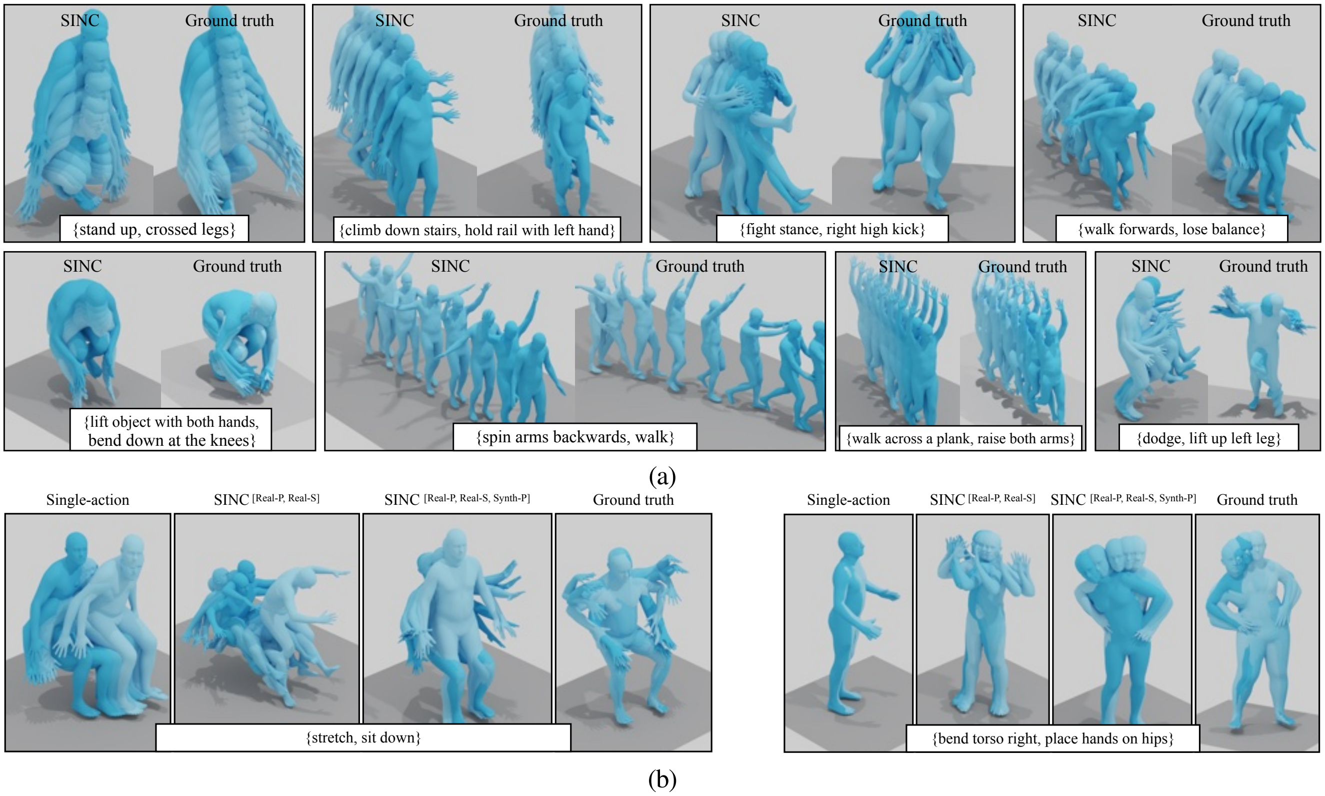

Figure 5. **Qualitative analysis:** (a) We present qualitative results for our final model, SINC, for various description pairs from the validation set. Our generations correctly correspond to the input semantics even when they are different from the ground truth, highlighting the challenge of coordinate-based (positional) performance measures. We display the ground truth (GT) for reference to define what the given actions mean. (b) We compare different models on two simultaneous action pairs. Both the Single-action model and the model not trained on synthetic data fail to generate those two compositions. Our model trained with the synthetic data successfully generates the composition in both cases. We include more comparisons in the supplementary video on our project page.

'walking' may or may not involve hands). Additionally, going beyond our 6 course parts to obtain fine-grained body part label association is important. In particular, this could involve the fingers and even facial expressions. Another limitation of our work (and the whole field) concerns the evaluation metrics. Despite introducing a new TEMOS score, perceptually meaningful performance measures are still missing. Finally, our model is conceptually not limited to pairs, but since it is rare to simultaneously perform more than two actions, we only focus on pairs in this work.

5. Conclusions

In this work, we established a new method to create spatial compositions of 3D human motions. Given a set of textual descriptions, our SINC model is able to generate motions that simultaneously perform multiple actions presented as textual input. We make use of the GPT-3 language model to obtain a mapping between actions and body parts to automatically create synthetic combinations of compatible actions. We use these synthetic motions to enrich the training of our model and find that it helps it generalize to new, complex, motions. We introduce multiple baselines and experiment with different data sources for this new problem. Our findings will open up possibilities for further research in fine-grained motion synthesis. While here we focus on spatial composition, future work should explore jointly modeling spatial and temporal action composition.

Acknowledgments. This work was granted access to the HPC resources of IDRIS under the allocation 2022-AD011012129R2 made by GENCI. GV acknowledges the ANR project CorVis ANR-21-CE23-0003-01. The authors would like to thank Peter Kulits for proofreading and Benjamin Pellkofer for IT support. **MJB Disclosure:** https://files.is.tue.mpg.de/black/CoI_ICCV_2023.txt

References

[1] Hyemin Ahn, Timothy Ha, Yunho Choi, Hwiyeon Yoo, and Songhwai Oh. Text2Action: Generative adversarial synthesis from language to action. In *International Conference on Robotics and Automation (ICRA)*, 2018. 2

[2] Chaitanya Ahuja and Louis-Philippe Morency. Language2Pose: Natural language grounded pose forecasting. In *International Conference on 3D Vision (3DV)*, 2019. 2, 3, 5

[3] Okan Arikan, David A. Forsyth, and James F. O'Brien. Motion synthesis from annotations. *Transactions on Graphics (TOG)*, 2003. 3
[4] Nikos Athanasiou, Mathis Petrovich, Michael J. Black, and Gül Varol. TEACH: Temporal action compositions for 3D humans. In *International Conference on 3D Vision (3DV)*, 2022. 1, 2, 3, 6
[5] Emad Barsoum, John Kender, and Zicheng Liu. HP-GAN: probabilistic 3D human motion prediction via GAN. In *Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2017. 2
[6] Tim Brooks, Aleksander Holynski, and Alexei A Efros. InstructPix2Pix: Learning to follow image editing instructions. In *Computer Vision and Pattern Recognition (CVPR)*, 2023. 2, 3
[7] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2020. 2, 4
[8] W. Chen, H. Wang, Y. Li, H. Su, Z. Wang, C. Tu, D. Lischinski, D. Cohen-Or, and B. Chen. Synthesizing training images for boosting human 3D pose estimation. In *International Conference on 3D Vision (3DV)*, 2016. 3
[9] Xin Chen, Biao Jiang, Wen Liu, Zilong Huang, Bin Fu, Tao Chen, Jingyi Yu, and Gang Yu. Executing your commands via motion diffusion in latent space. In *Computer Vision and Pattern Recognition (CVPR)*, 2023. 1, 3, 6
[10] Enric Corona, Albert Pumarola, G. Alenyà, and F. Moreno-Noguer. Context-aware human motion prediction. In *Computer Vision and Pattern Recognition (CVPR)*, 2020. 2
[11] Rodrigo Santa Cruz, Anoop Cherian, Basura Fernando, Dylan Campbell, and Stephen Gould. Inferring temporal compositions of actions using probabilistic automata. In *CVPR Workshop on Compositionality in Computer Vision*, 2020. 4
[12] Rishabh Dabral, Muhammad Hamza Mughal, Vladislav Golyanik, and Christian Theobalt. MoFusion: A framework for denoising-diffusion-based motion synthesis. In *Computer Vision and Pattern Recognition (CVPR)*, 2023. 3
[13] Anindita Ghosh, Noshaba Cheema, Cennet Oguz, Christian Theobalt, and Philipp Slusallek. Synthesis of compositional animations from textual descriptions. In *International Conference on Computer Vision (ICCV)*, 2021. 2, 3, 6
[14] Chuan Guo, Xinxin Xuo, Sen Wang, and Li Cheng. TM2T: Stochastic and tokenized modeling for the reciprocal generation of 3D human motions and texts. In *European Conference on Computer Vision (ECCV)*, 2022. 2
[15] Chuan Guo, Shihao Zou, Xinxin Zuo, Sen Wang, Wei Ji, Xingyu Li, and Li Cheng. Generating diverse and natural 3D human motions from text. In *Computer Vision and Pattern Recognition (CVPR)*, 2022. 1, 2, 7
[16] Chuan Guo, Xinxin Zuo, Sen Wang, Shihao Zou, Qingyao Sun, Annan Deng, Minglun Gong, and Li Cheng. Action2Motion: Conditioned generation of 3d human motions. In *ACM International Conference on Multimedia (MM)*, 2020. 2
[17] Ikhsanul Habibie, Mohamed A. Elgharib, Kripasindhu Sarkar, Ahsan Abdullah, Simbarashe Linval Nyatsanga, Michael Neff, and Christian Theobalt. A motion matching-based framework for controllable gesture synthesis from speech. In *International Conference on Computer Graphics and Interactive Techniques (SIGGRAPH)*, 2022. 2
[18] I. Habibie, Daniel Holden, Jonathan Schwarz, J. Yearsley, and T. Komura. A recurrent variational autoencoder for human motion synthesis. In *British Machine Vision Conference (BMVC)*, 2017. 2
[19] Félix G. Harvey, Mike Yurick, Derek Nowrouzezahrai, and Cristopher Joseph Pal. Robust motion in-betweening. *Transactions on Graphics (TOG)*, 2020. 2
[20] Mohamed Hassan, Duygu Ceylan, Ruben Villegas, Jun Saito, Jimei Yang, Yi Zhou, and Michael Black. Stochastic Scene-Aware Motion Prediction. In *International Conference on Computer Vision (ICCV)*, 2021. 2
[21] Yana Hasson, Gül Varol, Dimitrios Tzionas, Igor Kalevatykh, Michael J. Black, Ivan Laptev, and Cordelia Schmid. Learning joint reconstruction of hands and manipulated objects. In *Computer Vision and Pattern Recognition (CVPR)*, 2019. 3
[22] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2020. 3
[23] Daniel Holden, Jun Saito, and Taku Komura. A deep learning framework for character motion synthesis and editing. *Transactions on Graphics (TOG)*, 2016. 6
[24] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox. FlowNet 2.0: Evolution of optical flow estimation with deep networks. In *Computer Vision and Pattern Recognition (CVPR)*, 2017. 3
[25] Sai Shashank Kalakonda, Shubh Maheshwari, and Ravi Kiran Sarvadevabhatla. Action-GPT: Leveraging large-scale language models for improved and generalized zero shot action generation. *arXiv:2211.15603*, 2022. 3
[26] Keizo Kato, Yin Li, and Abhinav Gupta. Compositional learning for human object interaction. In *European Conference on Computer Vision (ECCV)*, 2018. 4
[27] Jihoon Kim, Taehyun Byun, Seungyoung Shin, Jungdam Won, and Sungjoon Choi. Conditional motion in-betweening. *Pattern Recognition*, 2022. 2
[28] Jihoon Kim, Jiseob Kim, and Sungjoon Choi. FLAME: Free-form language-based motion synthesis & editing. In *AAAI Conference on Artificial Intelligence*, 2023. 3
[29] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *International Conference on Learning Representations (ICLR)*, 2014. 6
[30] Taeryung Lee, Gyeongsik Moon, and Kyoung Mu Lee. MultiAct: Long-term 3D human motion generation from multiple action labels. In *AAAI Conference on Artificial Intelligence*, 2023. 3
[31] Peizhuo Li, Kfir Aberman, Zihan Zhang, Rana Hanocka, and Olga Sorkine-Hornung. GANimator: Neural motion synthesis from a single sequence. *Transactions on Graphics (TOG)*, 2022. 2
[32] Ruilong Li, Shan Yang, D. A. Ross, and Angjoo Kanazawa. AI choreographer: Music conditioned 3D dance generation with AIST++. In *International Conference on Computer Vision (ICCV)*, 2021. 2
[33] 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. In *Visually Grounded Interaction and Language (ViGIL) NeurIPS Workshop*,

2018. 2
[34] Yuejiang Liu, Riccardo Cadei, Jonas Schweizer, Sherwin Bahmani, and Alexandre Alahi. Towards robust and adaptive motion forecasting: A causal representation perspective. In *Computer Vision and Pattern Recognition (CVPR)*, 2022. 2
[35] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. In *International Conference on Computer Graphics and Interactive Techniques in Asia (SIGGRAPH ASIA)*, 2015. 4, 6
[36] Thomas Lucas, Fabien Baradel, Philippe Weinzaepfel, and Grégory Rogez. PoseGPT: Quantizing human motion for large scale generative modeling. In *European Conference on Computer Vision (ECCV)*, 2022. 2
[37] Naureen Mahmood, Nima Ghorbani, Nikolaus F. Troje, Gerard Pons-Moll, and Michael J. Black. AMASS: Archive of motion capture as surface shapes. In *International Conference on Computer Vision (ICCV)*, 2019. 2
[38] Julieta Martinez, Michael J. Black, and Javier Romero. On human motion prediction using recurrent neural networks. In *Computer Vision and Pattern Recognition (CVPR)*, 2017. 2
[39] Ishan Misra, Abhinav Gupta, and Martial Hebert. From red wine to red tomato: Composition with context. In *Computer Vision and Pattern Recognition (CVPR)*, 2017. 4
[40] Davide Moltisanti, Jinyi Wu, Bo Dai, and Chen Change Loy. BRACE: The breakdancing competition dataset for dance motion synthesis. In *European Conference on Computer Vision (ECCV)*, 2022. 2
[41] Dirk Ormoneit, Michael J. Black, T. Hastie, and H. Kjellström. Representing cyclic human motion using functional analysis. In *Image and Vision Computing (IVC)*, 2005. 2
[42] Priyanka Patel, Chun-Hao P. Huang, Joachim Tesch, David T. Hoffmann, Shashank Tripathi, and Michael J. Black. AGORA: Avatars in geography optimized for regression analysis. In *Computer Vision and Pattern Recognition (CVPR)*, 2021. 3
[43] Mathis Petrovich, Michael J. Black, and Gül Varol. Action-conditioned 3D human motion synthesis with transformer VAE. In *International Conference on Computer Vision (ICCV)*, 2021. 2, 3, 6
[44] Mathis Petrovich, Michael J. Black, and Gül Varol. TEMOS: Generating diverse human motions from textual descriptions. In *European Conference on Computer Vision (ECCV)*, 2022. 1, 2, 3, 5, 6, 7
[45] Leonid Pishchulin, Arjun Jain, Christian Wojek, Mykhaylo Andriluka, Thorsten Thormählen, and Bernt Schiele. Learning people detection models from few training samples. In *Computer Vision and Pattern Recognition (CVPR)*, 2011. 3
[46] Matthias Plappert, Christian Mandery, and Tamim Asfour. The KIT motion-language dataset. *Big Data*, 2016. 2, 3
[47] Abhinanda R. Punnakkal, Arjun Chandrasekaran, Nikos Athanasiou, Alejandra Quiros-Ramirez, and Michael J. Black Black. BABEL: Bodies, action and behavior with English labels. In *Computer Vision and Pattern Recognition (CVPR)*, 2021. 2, 3, 6
[48] Grégory Rogez and Cordelia Schmid. MoCap-guided data augmentation for 3D pose estimation in the wild. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2016. 3
[49] Tim Salzmann, Marco Pavone, and Markus Ryll. Motron: Multimodal probabilistic human motion forecasting. In *Computer Vision and Pattern Recognition (CVPR)*, 2022. 2
[50] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. In *Energy Efficient Machine Learning and Cognitive Computing NeurIPS Workshop*, 2019. 6
[51] Ken Shoemake. Animating rotation with quaternion curves. In *International Conference on Computer Graphics and Interactive Techniques (SIGGRAPH)*, 1985. 2
[52] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning (ICML)*, 2015. 3
[53] Sebastian Starke, He Zhang, Taku Komura, and Jun Saito. Neural state machine for character-scene interactions. *Transactions on Graphics (TOG)*, 2019. 2
[54] Xiangjun Tang, He Wang, Bo Hu, Xu Gong, Ruifan Yi, Qilong Kou, and Xiaogang Jin. Real-time controllable motion transition for characters. *Transactions on Graphics (TOG)*, 2022. 2
[55] Guy Tevet, Brian Gordon, Amir Hertz, Amit H. Bermano, and Daniel Cohen-Or. MotionCLIP: Exposing human motion generation to CLIP space. In *European Conference on Computer Vision (ECCV)*, 2022. 3
[56] Guy Tevet, Sigal Raab, Brian Gordon, Yonatan Shafir, Daniel Cohen-Or, and Amit H. Bermano. Human motion diffusion model. In *International Conference on Learning Representations (ICLR)*, 2023. 3
[57] Gül Varol, Ivan Laptev, Cordelia Schmid, and Andrew Zisserman. Synthetic humans for action recognition from unseen viewpoints. *International Journal of Computer Vision (IJCV)*, 2021. 3
[58] Gül Varol, Javier Romero, Xavier Martin, Naureen Mahmood, Michael J. Black, Ivan Laptev, and Cordelia Schmid. Learning from synthetic humans. In *Computer Vision and Pattern Recognition (CVPR)*, 2017. 3
[59] Jingbo Wang, Yu Rong, Jingyuan Liu, Sijie Yan, Dahua Lin, and Bo Dai. Towards diverse and natural scene-aware 3D human motion synthesis. In *Computer Vision and Pattern Recognition (CVPR)*, 2022. 2, 3
[60] Xi Wang, Gen Li, Yen-Ling Kuo, Muhammed Kocabas, Emre Aksan, and Otmar Hilliges. Reconstructing action-conditioned human-object interactions using commonsense knowledge priors. In *International Conference on 3D Vision (3DV)*, 2022. 2, 3
[61] Zan Wang, Yixin Chen, Tengyu Liu, Yixin Zhu, Wei Liang, and Siyuan Huang. HUMANISE: Language-conditioned human motion generation in 3D scenes. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2022. 3
[62] Xu Yang, Hanwang Zhang, and Jianfei Cai. Shuffle-Then-Assemble: Learning object-agnostic visual relationship features. In *European Conference on Computer Vision (ECCV)*, 2018. 2, 4
[63] Yufei Ye, Maneesh Singh, Abhinav Gupta, and Shubham Tulsiani. Compositional video prediction. In *International Conference on Computer Vision (ICCV)*, 2019. 4
[64] Ruichi Yu, Ang Li, Vlad I. Morariu, and Larry S. Davis. Visual relationship detection with internal and external linguistic knowledge distillation. In *International Conference on Computer Vision (ICCV)*, 2017. 2, 3
[65] Ye Yuan and Kris M. Kitani. DLow: Diversifying latent flows for diverse human motion prediction. In *European Conference on Computer Vision (ECCV)*, 2020. 2
[66] Jianrong Zhang, Yangsong Zhang, Xiaodong Cun, Shaoli Huang, Yong Zhang, Hongwei Zhao, Hongtao Lu, and Xi Shen. T2M-GPT: Generating human motion from textual descriptions

with discrete representations. In *Computer Vision and Pattern Recognition (CVPR)*, 2023. 3

[67] Mingyuan Zhang, Zhongang Cai, Liang Pan, Fangzhou Hong, Xinying Guo, Lei Yang, and Ziwei Liu. MotionDiffuse: Text-driven human motion generation with diffusion model. *arXiv:2208.15001*, 2022. 3

[68] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. BERTScore: Evaluating text generation with BERT. In *International Conference on Learning Representations (ICLR)*, 2020. 7

[69] Yan Zhang and Siyu Tang. The wanderings of Odysseus in 3D scenes. In *Computer Vision and Pattern Recognition (CVPR)*, 2022. 2

[70] Kaifeng Zhao, Shaofei Wang, Yan Zhang, Thabo Beeler, and Siyu Tang. Compositional human-scene interaction synthesis with semantic control. In *European Conference on Computer Vision (ECCV)*, 2022. 4

[71] Chongyang Zhong, Lei Hu, Zihao Zhang, Yongjing Ye, and Shi hong Xia. Spatio-temporal gating-adjacency gcn for human motion prediction. In *Computer Vision and Pattern Recognition (CVPR)*, 2022. 2

[72] Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. In *Computer Vision and Pattern Recognition (CVPR)*, 2019. 6

[73] Yi Zhou, Jingwan Lu, Connelly Barnes, Jimei Yang, Sitao Xiang, and Hao Li. Generative tweening: Long-term inbetweening of 3D human motions. *arXiv:2005.08891*, 2020. 2