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

MoDi: Unconditional Motion Synthesis from Diverse Data

Sigal Raab [1] Inbal Leibovitch [1] Peizhuo Li [2]
Kfir Aberman [3] Olga Sorkine-Hornung [2] Daniel Cohen-Or [1]

[1] Tel-Aviv University [2] ETH Zurich [3] Google Research
sigal.raab@gmail.com

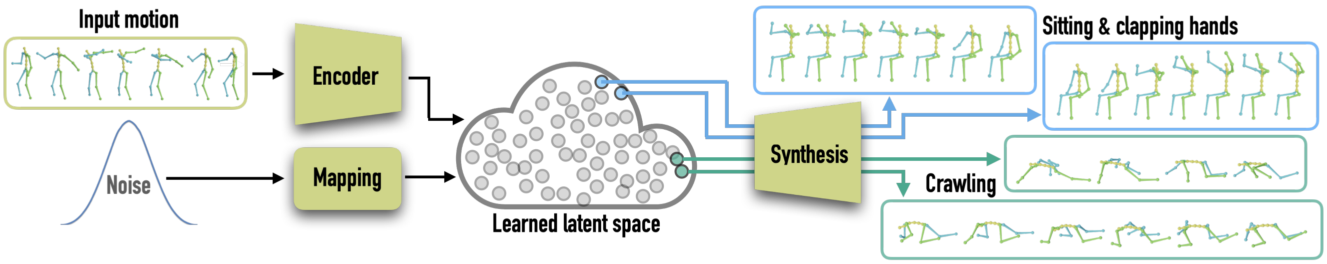

Figure 1. Our generative model is learned in an unsupervised setting from a diverse, unstructured and unlabeled motion dataset and yields a highly semantic, clustered, latent space that facilitates synthesis operations. An encoder and a mapping network enable the employment of real and generated motions, respectively.

Abstract

The emergence of neural networks has revolutionized the field of motion synthesis. Yet, learning to unconditionally synthesize motions from a given distribution remains challenging, especially when the motions are highly diverse. In this work, we present MoDi – a generative model trained in an unsupervised setting from an extremely diverse, unstructured and unlabeled dataset. During inference, MoDi can synthesize high-quality, diverse motions. Despite the lack of any structure in the dataset, our model yields a well-behaved and highly structured latent space, which can be semantically clustered, constituting a strong motion prior that facilitates various applications including semantic editing and crowd animation. In addition, we present an encoder that inverts real motions into MoDi's natural motion manifold, issuing solutions to various ill-posed challenges such as completion from prefix and spatial editing. Our qualitative and quantitative experiments achieve state-of-the-art results that outperform recent SOTA techniques. Code and trained models are available at https://sigal-raab.github.io/MoDi.

1. Introduction

The field of motion synthesis includes a wide range of long-standing tasks whose goal is to generate a sequence of temporally coherent poses that satisfy given cues and/or spatio-temporal constraints and importantly, look natural. In particular, learning to synthesize human motion from a given data distribution is a challenging task, especially when the dataset is highly diverse, unstructured, and unlabeled. In recent years, deep neural networks have become a popular tool for motion generation, and their excellent performance is imputed to their ability to learn motion priors from large-scale datasets. However, learning a motion prior from a diverse dataset remains a challenge.

Previous works focus on synthesizing specific types of motion of limited diversity [28], conditioned by a set of frames [62] or by a label indicating a text or an action [51].

In this work, we present MoDi, an unconditional generative model that synthesizes diverse motions. MoDi is unsupervised and is trained on diverse, unstructured, and unlabeled datasets, yielding a well-behaved, highly semantic latent space, facilitating a variety of synthesis operations.

Our design is inspired by the powerful architecture of StyleGAN [34], which has become a foundation for synthesis in the imaging domain, as it learns a well-structured latent space that allows incredible semantic editing capabilities [10]. However, there is a significant gap between the imaging and motion domains; Images possess a regularized 2D spatial structure with a relatively large number of degrees of freedom (DoF), while motion data is irregular, consisting of a skeletal graph with a temporal axis that has a smaller number of DoF. To mitigate this gap, we have conducted a thorough study of potential operators (2D vs. 3D convolutions, with and without skeleton-aware [2]), architectural variations (order and number of blocks, resolution, layers), and even propose a new building block (a convolutional scaler) that enables us to achieve state-of-the-art results in unconditional motion synthesis.

Our results show that MoDi learns a structured latent space that can be clustered into regions of semantically similar motions without any supervision. This latent space facilitates applications on diverse motions, including semantic editing, semantic interpolation between motions, and crowd animation.

In addition, we present an encoder architecture that leverages the knowledge we acquired on the generative model architecture, to invert unseen motions into MoDi's latent space, facilitating the usage of our generative prior on real-world motions. Inversion by an encoder enables us to project motions into the latent space within a feed-forward pass, instead of optimizing the latent code which requires several minutes for a single input [45]. Importantly, an encoder can better project a given motion to the well-behaved part of the latent space and can better assist in solving ill-posed problems (*e.g.*, motion prediction from prefix) as it learns to target its projections to the healthy regions of the latent space instead of overfitting to the input motion.

We evaluate our model qualitatively and quantitatively on the Mixamo [5] and HumanAct12 [21] datasets and show that it outperforms SOTA methods in similar settings. The strength of our generative prior is demonstrated through the various applications we show in this work, such as motion fusion, denoising, and spatial editing.

2. Related Work

The emergence of neural networks has transformed the field of motion synthesis, and many novel neural models have been developed in recent years [29,30]. Most of these models focus on specific human motion related tasks, conditioned on some limiting factors, such as motion prefix [7,9,22,25,59,61], in-betweening [15,23,24,35], motion retargeting or style transfer [2–4,27,52], music [8,37,40,49], action [13,21,43,51], or text [6,11,20,44,50,51,60].

A large number of models focus on action conditioned generation. These works are closer in spirit to ours, hence in the following we elaborate about them. These models can be roughly divided to autoregressive [16,17,21,22,32,41,43,62], diffusion-based [51] and GAN-based [14,54,56,58].

Petrovich *et al.* [43] learn an action-aware latent representation by training a VAE. They sample from the learned latent space and query a series of positional encodings to synthesize motion sequences conditioned on an action. They employ a transformer for encoding and decoding a sequence of parametric SMPL human body models. Maheshwari *et al.* [41] generate single or multi-person pose-based action sequences with locomotion. They present generations conditioned by 120 action categories. They use a Conditional Gaussian Mixture Variational Autoencoder to enable intra and inter-category diversity. Wang *et al.* [53] employ a sequence of recurrent autoencoders. They replace the KL divergence loss with a discriminator to ensure the bottleneck distribution.

Some GAN-based models are combined with factors that limit their generalization, such as Gaussian processes [56] or auto encoders [54,58]. Degardin *et al.* [14] fuse the architectures of GANs and GCNs to synthesize the kinetics of the human body. Like us, they borrow a mapping network from StyleGAN [34]. However, their model does not utilize important aspects of StyleGAN such as multi-level style injection. As we demonstrate, these aspects significantly ameliorate the quality of the synthesized motions. Unlike the above conditional models, we present an unconditional method.

Only a few works enable pure unconditioned synthesis. Holden *et al.* [29] present a pioneering work in deep motion synthesis. Their latent space is not sufficiently disentangled, so they train a separated feed-forward network for each editing task, while MoDi either performs editing in the latent space with no need to train an additional network (Sec. 4.1), or uses a single encoder for a variety of applications (Sec. 4.2). Several conditional works [14, 56] use an unconditional baseline but focus on the conditional task. Another model that supports an unconstrained setting is MDM [51]. Although they use state-of-the-art diffusion models, we demonstrate that MoDi outperforms their unconditional synthesis setting (Sec. 5.1). See our sup. mat. for an elaborated discussion about unconditional works.

To process motion with deep learning, some works convert it into a pseudo image, where joints and time-frames correspond to image height and width, and joint features (*e.g.* coordinates) are equivalent to RGB channels [25, 29, 41, 43]. Although intuitive, this approach does not account for the fact that joints may not be adjacent like image pixels. TSSI [57] partially solves this problem by replicating some joints to ensure skeletal continuity in convolution. However, it does not reflect all neighborhood degrees. The emergence of graph-based convolutional networks has been adopted by the motion research community [56], since the human skeleton can be naturally represented by a graph, where the joints and bones are represented with vertices and edges, respectively. A full motion is then considered as a spatio-temporal graph [14, 58]. Since a single kernel shared by all joints cannot capture the fine nuances of each joint, more advanced techniques [2,38,48,55,56] exploit the advantage of using finite size skeletons with predefined topology. Each skeletal joint is unique in the way it relates to its neighbors. In our work, we adopt this approach and dedicate a unique kernel for each joint.

3. Method

At the crux of our approach lays a deep generative model trained in an unsupervised manner on an extremely diverse, unstructured and unlabeled motion dataset. Our network receives a noise vector drawn from an i.i.d Gaussian distribution and outputs a natural, temporally coherent human motion sequence. Once the generator is trained, the learned prior can be leveraged for various applications, and can be applied to either synthetic or real motions using an encoder model that receives a motion tensor and inverts it into the latent space of the generative model.

In recent years, generative works in the image domain have attained unprecedented synthesis quality [12, 26, 36], and our framework is inspired by one of the prominent methods – StyleGAN [33, 34]. However, StyleGAN as is *cannot* be used for motion synthesis since there is a significant domain gap between images and motions that makes the adaptation non-trivial. First, images possess a regularized spatial structure with an inductive bias of pixel neighborhood which is strongly exploited, while motions are irregular, consisting of joints whose features are adjacent in a tensor but are unnecessarily adjacent in the skeletal topology. Second, images have a relatively larger number of DoF compared to the DoF of motion which is limited by the number of joints.

To bridge the gap, our architectural design employs structure-aware neural filters that enable coping with irregular motion representation. Unlike prior works, we employ 3D convolutions instead of 1D or 2D ones, facilitating convolutional operators with a dedicated kernel for each skeletal joint. The benefit of 3D filters is detailed in the sup. mat. To address the low DoF and avoid over-fitting, we use a shallower hierarchy than in the imaging domain. Moreover, we suggest a novel skeleton-aware convolutional-pooling filter to boost the performance of our networks.

Next, we discuss our structure-aware modules and network architecture and present an inversion technique that projects a given motion into the learned latent space. In Sec. 4.1 we show that our latent space is semantically clustered and demonstrate semantic editing applications, and in Sec. 4.2 we demonstrate that ill-posed problems can be solved with our encoder. Finally, we show a quantitative and qualitative evaluation of our framework and compare it to state-of-the-art alternatives (Section 5). We refer the reader to the supplementary video to see the results of our work.

3.1. Motion Representation

We describe a motion using temporally coherent 3D joint rotations, $\mathbf{R} \in \mathbb{R}^{T \times J \times K}$, where T, J, and K are the numbers of frames, joints and rotation features, respectively. Unit quaternions (4D) attain the best empirical results for rotation representation. The root joint position is represented by a sequence of global displacements, $\mathbf{P} \in \mathbb{R}^{T \times 3}$, from which we extract their velocities, $\mathbf{V} \in \mathbb{R}^{T \times 3}$. In addition, our network learns to refrain from foot sliding artifacts using binary foot contact labels, $\mathbf{F} \in \{0,1\}^{T \times 2}$, that are concatenated to the joints axis. We zero-pad the feature dimension of the root location and the foot contact labels to the size of the rotation feature, K, and add an extra dimension, so all entities ($\mathbf{R}$, $\mathbf{V}$ and $\mathbf{F}$) possess the same number of features. Altogether we have $\mathbf{R} \in \mathbb{R}^{T \times J \times K}$ (unchanged), $\hat{\mathbf{V}} \in \mathbb{R}^{T \times 1 \times K}$, and $\hat{\mathbf{F}} \in \mathbb{R}^{T \times 2 \times K}$. Once all features share the same size, we concatenate them to obtain the complete space by

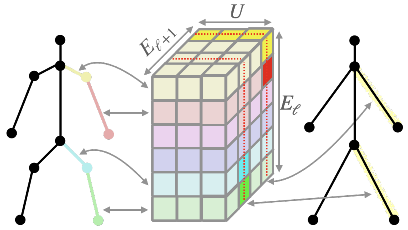

Figure 2. 3D convolutional scaler: Each horizontal slice affects one entity in the fine character (left), and each vertical slice (xz plane) affects one entity in the coarse character (right). Each entity in the coarse character “sees” only weights related to relevant entities of the fine character, emphasized with saturated colors in the filter.

$$\mathcal{M}_{full} \equiv \mathbb{R}^{T \times E \times K}, \tag{1}$$

where $E = J + 3$ is the number of entities ($\mathbf{R}$, $\mathbf{V}$ and $\mathbf{F}$).

Let $\mathcal{M}_{nat}$ denote the space of natural motions that are plausible for humans to enact. Each motion $m \in \mathcal{M}_{nat}$ is represented by a tuple, $[\mathbf{R}_m, \hat{\mathbf{V}}_m, \hat{\mathbf{F}}_m]$. Note that the subspace of all human motions, $\mathcal{M}_{nat} \subset \mathcal{M}_{full}$, is extremely sparse, as most of the values in $\mathcal{M}_{full}$ correspond to motions that are unnatural or impossible for humans.

Our network has a hierarchical structure that evolves the motion representation from coarse to fine. At each level ℓ, the number of frames, joints, entities, and features is denoted by T_ℓ, J_ℓ, E_ℓ, and K_ℓ, respectively. The number of frames T_ℓ increases between two consecutive levels by a factor of 2, and the number of joints increases by a topologically specified factor in order to obtain a meaningful refinement of the skeleton [2, 14]. More representation considerations are detailed in the sup. mat.

3.2. Structure-aware Neural Modules

We consider the human skeleton as a directed graph, where the joints stand for vertices and the bones stand for directed edges. We associate each skeletal joint with the edge that is directed towards it, hence they share the same features. The root joint, to which no edge is directed, is associated with an abstract edge that starts at the origin.

Some works [14, 58] use Graph Convolutional Networks (GCNs) for neural computation. Like GCNs, they employ the same kernels to all graph vertices. Unlike general graphs, the topology of the skeleton is known in advance, and has a finite size. These facts can be exploited to get better sensitivity to each joint's unique role in the skeleton. We follow the works that exploit the knowledge of skeletal topology [2,56] and dedicate separate kernels for each joint.

However, these works use naïve pooling to up/down sample the skeletal (spatial) domain, which are essentially mere copying and averaging. Alternatively, we present a spatio-temporal convolutional operator, that scales the skeleton topology, as well as the temporal dimension. We use the same filter architecture for convolution during down-sampling and transposed convolution for up-sampling. We achieve the desired functionality by adding a dimension to the kernels for the outgoing joints, similar to the way a dimension is added for the outgoing channels. The dimensions of each filter are then $K_{\ell+1} \times K_\ell \times E_{\ell+1} \times E_\ell \times U$, where U is the filter width. Fig. 2 visualizes our novel convolutional scaler filter and the sup. mat. elaborates on it.

In addition, we use one existing skeleton-aware module, namely in-place convolution [2], and add a third dimension to it too. The motivation for the extra dimension is the convenience of applying modulation, explained in the sup. mat. The sup. mat. also describes skeleton-aware modules in existing works (convolutional and pooling).

3.3. Generative Network Architecture

Our network receives a noise vector drawn from an i.i.d Gaussian distribution, $\mathcal{Z}$, and outputs a natural, temporally coherent, human motion sequence, as depicted in Fig. 3. Our generator G consists of two main parts: a mapping network that maps noise into a well-behaved, structured, latent space, and a fully convolutional neural network that maps a learned constant and the latent code into the final motion.

Mapping network Let $\mathcal{Z} = \mathcal{N}(\vec{0}, \mathbf{I})$ be a multivariate normal distribution. Given a latent code $z \in \mathcal{Z}$, a non-linear mapping network produces a latent value, $w \in \mathcal{W}$, where $\mathcal{W}$ is known to be disentangled and well behaved, as studied for images [42,46] and for motions [14].

Motion synthesis network We use a hierarchical framework that learns to convert a learned constant into a motion representation via a series of skeleton-aware convolutional layers (Sec. 3.2), where the traditional skeletal pooling layer is replaced by our convolutional scaler. The layers in the motion synthesis network are modulated by style codes that are injected in each level and modify second-order statistics of the channels, in a spatially invariant manner [31]. The style codes are learned from the outputs of the mapping network, using affine transformation.

We employ a discriminator [18], D, that holds the reverse architecture of the synthesis network. It receives generated or real motion and processes it in skeleton-aware neural blocks that downscale gradually. A recap of StyleGAN, and details on training setups and hyperparameters, are given in the sup. mat.

We train our network with all the StyleGAN losses [34]: adversarial loss, path length regularization, and $R1$ regularization. For completeness, these losses are detailed in the sup. mat. We add two skeleton-related losses.

Accurate foot contact is a major factor in motion quality. There is already special care for foot contact in the adversarial loss, as $\mathcal{M}_{nat}$ contains foot contact labels. However, we notice that encouraging contact between the feet and the ground improves the naturalness of the motions, and discourages the phenomenon of "floating" feet. Hence, we add an encouragement regulation

$$\mathcal{L}^G_{tch} = \mathop{\mathbb{E}}_{z \sim \mathcal{Z}} \left[-\log s(G(z)_F) \right], \qquad (2)$$

where $(\cdot)_F$ is the contact-label component of the motion, and $s(\cdot)$ is the sigmoid function.

In addition, we use contact consistency loss [39, 47], which requires that a high velocity should not be possible while a foot is touching the ground:

$$\mathcal{L}^G_{fcon} = \mathop{\mathbb{E}}_{z \sim \mathcal{Z}} \left[\left\| FK\left(G(z)\right)_f \right\|_2^2 \cdot s\left(G(z)_F\right) \right], \qquad (3)$$

where FK is a forward kinematic operator yielding joint locations, and $(\cdot)_f$ is feet velocity extracted from them.

Although our foot contact losses notably mitigate sliding artifacts, we further clean foot contact with a fully automatic procedure using standard IK optimization [39].

3.4. Encoder Architecture

Our encoder projects input motions onto the learned latent space such that it can be reconstructed by our synthesis network (with fixed weights), depicted in Fig. 5. The input motion can originate from video footage (using 3D motion reconstruction [19]), a dataset, or a motion-capture system. Thus, our encoder enables using the advantages of our well-behaved learned latent space on real motions rather than on generated ones. Moreover, the encoder aims its projections to the healthy part of the latent space, resulting in natural-looking results to ill-posed tasks (Sec. 4.2).

Our encoder, I, holds the reverse architecture of the synthesis network, similar to the discriminator D. It processes the input motion in skeleton-aware neural blocks that downscale gradually, as shown in Fig. 4. Inversion quality is improved when using $\mathcal{W}+$ [1] rather than $\mathcal{W}$. See the sup. mat. for a recap regarding $\mathcal{W}+$.

In order to train the encoder we split $\mathcal{M}_{nat}$ into sets of train $\mathcal{M}_{trn}$ and test $\mathcal{M}_{tst}$, with a ratio of 80:20, respectively, and train the encoder on the training set only. The encoder is trained with several losses.

Reconstruction loss The main goal of the encoder is to predict a latent variable $I(m) \in \mathcal{W}+$ such that $G(I(m))$ is as close as possible to the input motion m:

$$\mathcal{L}^I_{rec} = \mathop{\mathbb{E}}_{m \sim \mathcal{M}_{trn}} \left[\| m - G(I(m)) \|_2^2 \right]. \qquad (4)$$

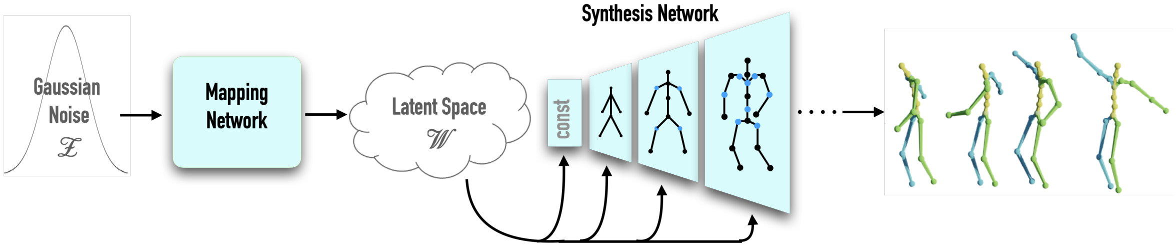

Figure 3. Our motion generator combines structure-aware neural modules with a mapping network and style codes injected to multiple levels of the generator. A detailed description of the architecture (e.g., layers, hyperparameters) is given in the sup. mat.

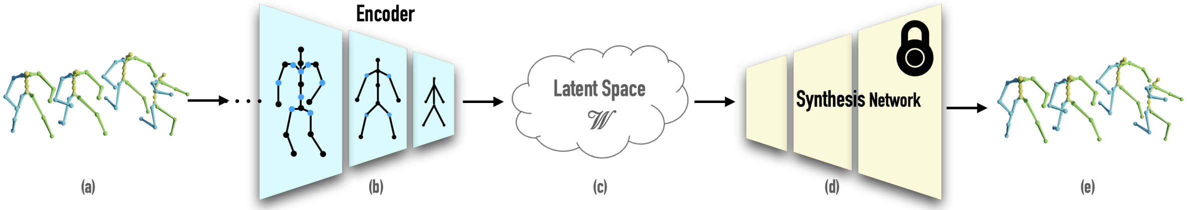

Figure 4. Our encoder receives an input motion (a), learns a hierarchy of skeleton aware layers (b) and outputs the latent value of that motion (c). Once the projected latent value is fed into the synthesis network (d) (with fixed weights), the input motion is reconstructed (e).

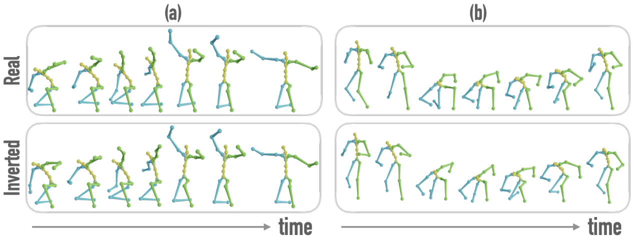

Figure 5. Inversion of two motions, (a) and (b). The original and the reconstructed motions are depicted at the top and bottom rows, respectively. The reconstruction is done by first projecting the real motions into $\mathcal{W}+$ using the encoder, and then running the obtained latent values in the generator.

Foot contact loss Unlike the unsupervised foot contact loss of the generator, the following loss [47] is supervised:

$$\mathcal{L}^{I}_{fcon} = \mathop{\mathbb{E}}_{m \sim \mathcal{M}_{trn}} \left[BCE(m_F, s(G(I(m))_F)) \right], \quad (5)$$

where BCE is a binary cross entropy function, $(\cdot)_F$ is the contact-label of the motion, and $s(\cdot)$ is the sigmoid function.

Root loss We notice that the positions and rotations of the root converge slower than the rotations of the other joints, hence, we add a dedicated loss term:

$$\mathcal{L}^{I}_{root} = \mathop{\mathbb{E}}_{m \sim \mathcal{M}_{trn}} \left[\| m_{root} - G(I(m))_{root} \|_2^2 \right], \quad (6)$$

where $(\cdot)_{root}$ denotes root velocity and rotation in a motion. This loss prioritizes the root components in $\mathcal{L}rec$.

Position loss In addition to the reconstruction loss that mainly supervises rotation angles, we regularize our encoder by supervision on the joint position themselves:

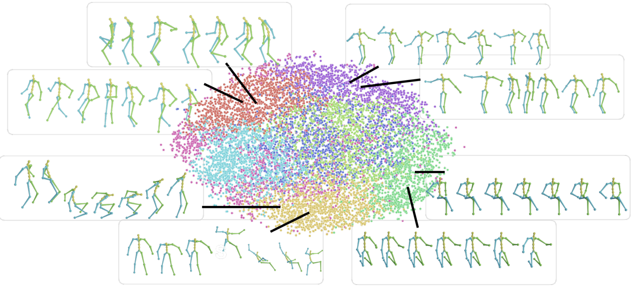

Figure 6. The latent space $\mathcal{W}$, split into 8 clusters using K-means, and visualized using T-SNE. Each point relates to one $\mathcal{W}$ space instance, generated from random noise $z \in \mathcal{Z}$. The visualized motions are the result of running these latent variables through our generator G. We observe that the clusters indeed represent semantic grouping of the data.

$$\mathcal{L}^{I}_{pos} = \mathop{\mathbb{E}}_{m \sim \mathcal{M}_{trn}} \left[\| FK(m) - FK(G(I(m))) \|_2^2 \right]. \quad (7)$$

Finally, the total loss applied to the encoder is:

$$\mathcal{L}^{I} = \mathcal{L}^{I}_{rec} + \lambda^{I}_{fcon}\mathcal{L}^{I}_{fcon} + \lambda_{root}\mathcal{L}^{I}_{root} + \lambda_{pos}\mathcal{L}^{I}_{pos}, \quad (8)$$

where we mostly use $\lambda_{fcon} = 100$, $\lambda_{root} = 2$, $\lambda_{pos} = 0.1$.

4. Applications

4.1. Latent Space Analysis and Applications

Latent Clusters We demonstrate that $\mathcal{W}$ is well-structured by clustering it into meaningful collections of motions. Recall that MoDi learns from datasets that cannot be semantically clustered due to their unstructured nature.

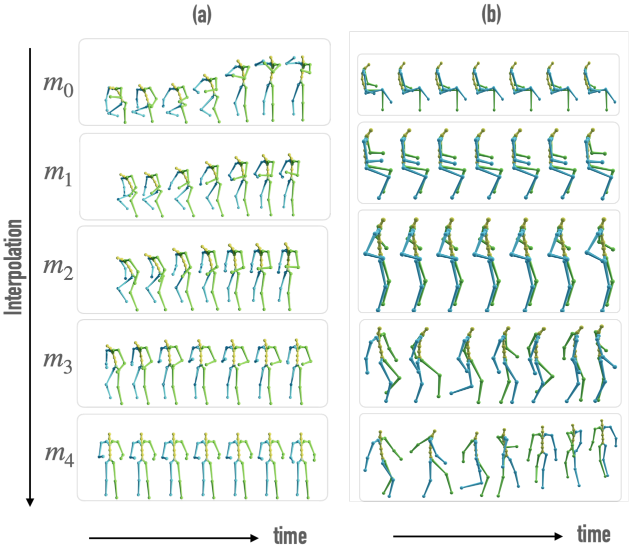

Figure 7. Interpolation in the latent space: (a) interpolation to the mean motion (truncation); (b) From sitting to walking: Note the gradual change from crossed legs sitting, to regular sitting, standing, small-step walking, and regular walking. Each motion is natural, despite interpolating between sitting and standing, which cannot be achieved by interpolating between joint values.

In Fig. 6 we observe the latent space $\mathcal{W}$, split into 8 clusters using K-means. The $\mathcal{W}$ values belong to 10,000 randomly synthesized motions. We randomly choose several motions from each cluster and depict them. Clearly, motions represented by different clusters are semantically different, and motions that share a cluster are semantically similar.

Latent interpolation The linearity of the latent space $\mathcal{W}$ is demonstrated by interpolating between the latent values and generating motions from the resulting interpolations, as depicted in Fig. 7. Further details, formal definitions, and analyses of our results are provided in the sup. mat.

Editing in the latent space If a latent space is sufficiently disentangled, it should be possible to find direction vectors that consistently correspond to individual factors of variation. Let a be an attribute related to motion. a can be any semantic attribute, such as movement speed, verticality measurement of parts in the body, or a motion style. Inspired by Shen *et al.* [46], we compute a score that measures a in a motion. For example, when measuring the verticality of a motion, a character doing a handstand would get a score of -1, lying down would get a score of 0, and standing up would get a score of 1. Using a given score, we train an SVM, yielding a hyperplane that serves as a separation boundary. Denote the unit normal of the hyperplane by n. Then $G(w+n)$ possesses an increased score of attribute a compared to $G(w)$. The only attribute that should change in such editing is a, preserving the rest of the motion intact.

Unlike image datasets, which hold labeling for various attributes (age, gender,...), there is not much labeling in motion datasets. We create our own simple classifiers and elaborate next regarding one of them, measuring *gradual right arm lifting*, denoted *gral*. *Gral* means that the right arm is lifted as time advances. Computing a score for the *gral* attribute is not straightforward, and is detailed in the sup. mat. Our results are visualized in Fig. 8, where we show that the *gral* attribute gets stronger while stepping in the latent space, and the naturalness of motions as well as their semantics are kept. In our video clip we show that when such an attribute is artificially applied via geometric interpolation, the results are unnatural. Obtaining manual natural results would require an artist's hard work.

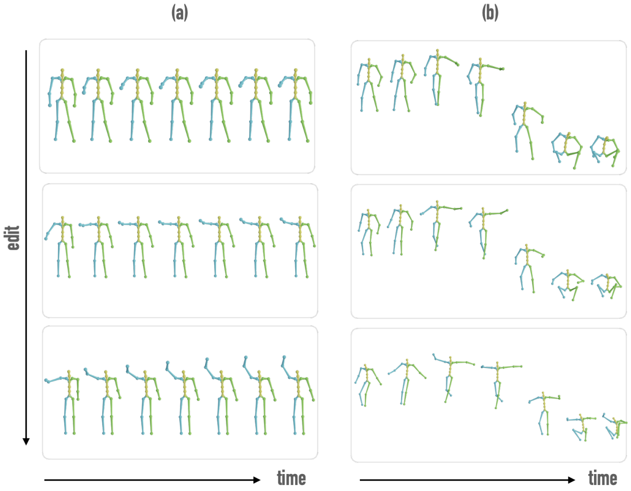

Figure 8. Editing in the latent space. The motion remains intact except for the edited attribute, *gradual right arm lifting* (*gral*). The *gral* attribute gets stronger as we go down in each column. The generative prior of MoDi keeps the jumping motion (b) natural, even at the expense of arm lifting.

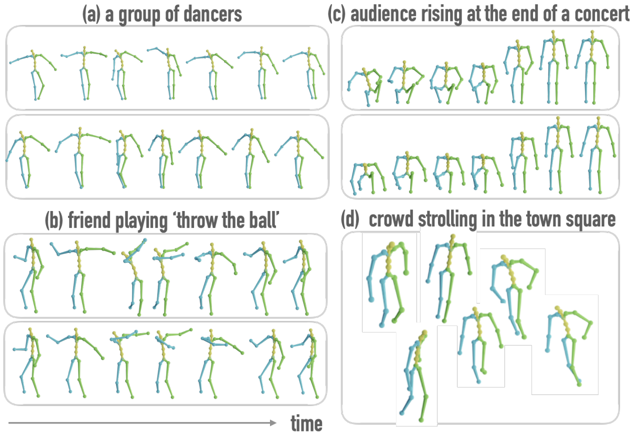
(a) a group of dancers
(c) audience rising at the end of a concert
(b) friend playing 'throw the ball'
(d) crowd strolling in the town square
time

Figure 9. Crowd animation. Blocks (a), (b) and (c) depict sequences of motion frames over time. The two sequences in each of these blocks visualise similar motions created using perturbation in the latent space. Block (d) depicts poses in *one* time frame extracted from six *distinct* motions.

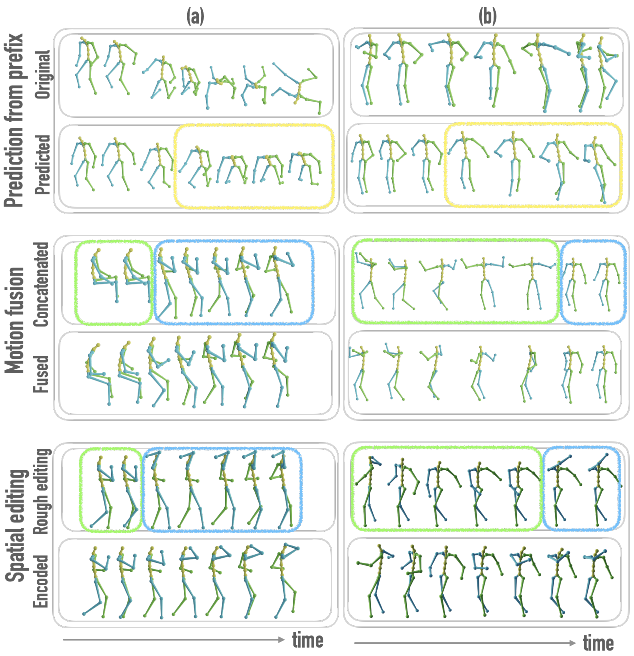

Figure 10. Our encoder enables coping with ill-posed tasks. Prediction from prefix: the top row is the original motion, from which only the prefix is seen by the encoder. The predicted output is in the second row. Notice that the encoder synthesises a coherent suffix (in yellow), without overfitting the original one. Motion fusion: (a) sitting (green) and walking (blue). Notice the smooth fused sequence versus the concatenated one; (b) dancing (green) and standing (blue) where the last green frame looks backwards, while the first blue frame looks forward. The encoder mitigates this challenging concatenation by gradually rotating the character. Spatial editing: green and blue rectangles encircle frames left intact or amateurishly edited, respectively. Here we edit the *gradual right arm lifting (gral)* attribute using a very different approach. Instead of going through the tedious work of finding an editing direction in the latent space (Sec. 4.1), we spatially edit the right arms and let the encoder turn our rough edit into a natural one.

Crowd animation Given an input motion m, we can sample variations in our latent space $\mathcal{W}$ by simply sampling the neighborhood of the w that corresponds to m. We sample in a Gaussian $\sim \mathcal{N}(m, \sigma^2)$, with σ in the range 0.1-0.8. This way, one can simulate, for instance, people walking in the town square, a group of dancers, or a group of friends jumping from a bench. See Fig. 9 and our video for examples.

4.2. Solving Ill-posed Tasks with the Encoder

Many tasks in the motion domain are ill-posed. That is, there is more than one single solution to the task. We present several examples, each of which uses an encoder to achieve high-quality results for an ill-posed task. The tasks in this section are conditioned and show that a generator that has been trained in an unconditional setting can be used for a variety of downstream tasks, as illustrated in Fig. 10.

For all applications except the first, the encoder is used as is without further training. The main concept behind these applications is to invert a given motion m into the healthy regions of the latent space and then generate an output motion. The result $G(I(m))$ remains faithful to the input motion while being projected into the manifold of natural motions, thanks to the well-behaved nature of the latent space and the generator's ability to produce only natural motions. In the following paragraphs, we denote the motion frame index by $t \in [1 \dots T]$.

Prediction from Prefix Given a motion prefix of length t (namely, t frames that represent the beginning of a motion sequence), complete the motion into length T. Unlike the other tasks, for this task we re-train our encoder such that the input is zeroed in frames $[(t+1)\dots T]$. The encoder is supervised by the full motion, using its full set of losses. Generalizing this solution to another ill-posed task, in-betweening, is straightforward.

Motion Fusion Given two input motions, take [1...t] frames from the first and $[(t+1)\dots T]$ frames from the second, and output one motion whose prefix is the first motion and suffix is the second one. A naïve concatenation of the frames would lead to a non-continuous motion at the concatenation frame. Denote the concatenated motion by m. Applying $G(I(m))$ smooths the frames around the concatenation area and yields a natural-looking motion, while fidelity to the prefix and suffix motions is kept. Generalization to a variable number of motions is simple.

Denoising Given a noisy input motion m, output a natural looking, smooth motion. Similar to motion fusion, we apply $G(I(m))$ and obtain a denoised output. Generalization to more types of corrupted motions is straightforward.

Spatial Editing Given an input motion m, with several frames manually edited, output a natural-looking, coherent motion. Often animators are interested in spatial changes of existing motions, *e.g.*, raising a hand. Manually editing several frames in the spatial domain is exhaustive and requires a professional. We tackle this task by performing a bulky manual edit, yielding a non-natural motion, and running it through $G \circ I$ to get a natural and coherent output.

A qualitative and quantitative comparison to applications in other works can be found in the sup. mat.

5. Experiments

Datasets We use Mixamo [5] and HumanAct12 [21], as elaborated in our supplementary materials.

5.1. Quantitative Results

Metrics We use the metrics FID, KID, precision-recall, and diversity, and describe them in the sup. mat. The metrics build upon the latent features of an action recognition model. However, training such a model on Mixamo is challenging, as there is no action labeling in it. We resort to a creative solution, as detailed in the sup. mat.

Model	FID ↓	KID ↓	Precision ↑ Recall ↑	Diversity ↑
ACTOR [43]	48.8	0.53	**0.72**, 0.74	14.1
MDM [51]	31.92	0.96	0.66, 0.62	17.00
MoDi (ours) with mixing	15.55	0.14	**0.72**, 0.75	17.36
MoDi (ours) without mixing	**13.03**	**0.12**	0.71, **0.81**	**17.57**

Table 1. Quantitative results for state-of-the-art works on the HumanAct12 dataset. The grayed line shows our original algorithm, without the changes that make it comparable. Note that our model leads in all the variations. Best scores are emphasized in **bold.**

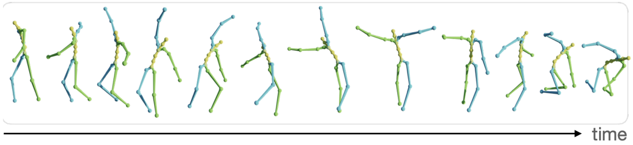

Figure 11. Qualitative result. More are in the video and sup.

Results We compare our model with state-of-the-art synthesis networks, ACTOR [43] and MDM [51], on the HumanAct12 [21] dataset, and show the results in Tab. 1. The compared works use contemporary state-of-the-art techniques, namely transformers and diffusion models. Yet, in the challenging setting of unconstrained synthesis, MoDi outperforms them by a large margin. To enable comparison in an unconstrained setting, we re-train ACTOR with one fixed label and use MDM's unconstrained variation. In the sup. mat. we further provide a comparative discussion of unconditional works. For the sole purpose of comparison with other works, we provide a version of MoDi, that skips style-mixing [34], as mixing may change the distribution of synthesized motions, yielding degradation in the metric score. Both versions are shown in Tab. 1. Moreover, our model facilitates multiple short fine-tunings for various conditions, based on a one-time unconditional training. An example of the action-to-motion task, which includes quantitative and qualitative results, can be found in the sup. mat.

5.2. Qualitative Results

The reader is encouraged to watch our supplementary video in order to get a full impression of the quality of our results. For completeness, we show one special motion in Fig. 11, and several more in the sup. mat.

5.3. Ablation

Table 2 presents the results of an extensive study on various architectures. The first variation shows metric scores for a non-skeleton-aware architecture using pseudo images to represent motion. The drawbacks of such images are discussed in Sec. 2. The second variation employs joint locations instead of rotations, which we explain in the sup. mat. as being inferior. Our third variation refrains from using our new convolutional scaler, and uses skeleton-aware pooling [2], testifying that our new filter improves the results. Next, we examine the effect of removing one in-place convolution from each hierarchical layer. Finally, we measure the scores for our final architecture and conclude that our architectural choices outperform other alternatives.

Architecture variation	FID ↓	KID ↓	Precision ↑ Recall ↑	Diversity ↑
non skel.-aware	$23.0^{\pm 0.3}$	$0.17^{\pm 0.02}$	$0.46^{\pm 0.01}$ $0.41^{\pm 0.01}$	$13^{\pm 0.08}$
joint loc. rather than rot.	$17.3^{\pm 0.06}$	$0.2^{\pm 0.03}$	$0.46^{\pm 0.02}$ $0.58^{\pm 0.01}$	$14.0^{\pm 0.3}$
pool rather than conv. scaler	$14.9^{\pm 0.7}$	$0.16^{\pm 0.02}$	$\mathbf{0.49^{\pm 0.01}}$ $0.58^{\pm 0.03}$	$15.3^{\pm 0.02}$
remove one in-place conv. per hierarchy	$14.1^{\pm 1.4}$	$0.15^{\pm 0.02}$	$0.46^{\pm 0.02}$ $0.66^{\pm 0.1}$	$\mathbf{15.4^{\pm 0.9}}$
final architecture	$\mathbf{11.5^{\pm 0.9}}$	$\mathbf{0.1^{\pm 0.01}}$	$0.46^{\pm 0.02}$ $\mathbf{0.69^{\pm 0.02}}$	$\mathbf{15.4^{\pm 0.2}}$

Table 2. Quantitative results for various generator designs, on the Mixamo dataset. Best scores are emphasized in **bold**.

Every training configuration is run 3 times, and every evaluation (per configuration) 5 times. The numbers in Tab. 2 are of the form $mean^{\pm std}$. Further ablation studies and analysis of overfitting are available in the sup. mat.

6. Conclusion

One of the most fascinating phenomena of deep learning is that it can gain knowledge, and even learn semantics, from unsupervised data. In this work, we have presented a deep neural architecture that learns motion prior in a completely unsupervised setting. The main challenge has been to learn a generic prior from a diverse, unstructured, and unlabeled motion dataset. This has necessarily required a careful design of a neural architecture to process the unlabeled data. We have presented MoDi, an architecture that distills a powerful, well-behaved latent space, which then facilitates downstream latent-based motion manipulations.

Like any data-driven method, the quality of the generalization power of MoDi is a direct function of the training data, which, at least compared to image datasets, is still lacking. Another limitation is that skeleton-aware kernels, with dedicated kernels per joint, occupy large volumes, resulting in relatively large running time.

In the future, we would like to address the challenging problem of learning motion priors from video. Building upon networks like MoDi, with proper inductive bias, may open the way towards it.

Acknowledgments Supported in part by the Israel Science Foundation (grants no. 2492/20 and 3441/21).

References

[1] Rameen Abdal, Yipeng Qin, and Peter Wonka. Image2stylegan: How to embed images into the stylegan latent space? In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4432–4441, 2019. 4

[2] Kfir Aberman, Peizhuo Li, Dani Lischinski, Olga Sorkine-Hornung, Daniel Cohen-Or, and Baoquan Chen. Skeleton-aware networks for deep motion retargeting. *ACM Transactions on Graphics (TOG)*, 39(4):62–1, 2020. 2, 3, 4, 8

[3] Kfir Aberman, Yijia Weng, Dani Lischinski, Daniel Cohen-Or, and Baoquan Chen. Unpaired motion style transfer from video to animation. *ACM Transactions on Graphics (TOG)*, 39(4):64–1, 2020. 2

[4] Kfir Aberman, Rundi Wu, Dani Lischinski, Baoquan Chen, and Daniel Cohen-Or. Learning character-agnostic motion for motion retargeting in 2d. *arXiv preprint arXiv:1905.01680*, 2019. 2

[5] Adobe Systems Inc. Mixamo, 2021. Accessed: 2021-12-25. 2, 7

[6] Chaitanya Ahuja and Louis-Philippe Morency. Language2pose: Natural language grounded pose forecasting. In *2019 International Conference on 3D Vision (3DV)*, pages 719–728. IEEE, 2019. 2

[7] 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. IEEE Computer Society, 2019. 2

[8] Andreas Aristidou, Anastasios Yiannakidis, Kfir Aberman, Daniel Cohen-Or, Ariel Shamir, and Yiorgos Chrysanthou. Rhythm is a dancer: Music-driven motion synthesis with global structure. *arXiv preprint arXiv:2111.12159*, 2021. 2

[9] 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

[10] Amit H Bermano, Rinon Gal, Yuval Alaluf, Ron Mokady, Yotam Nitzan, Omer Tov, Oren Patashnik, and Daniel Cohen-Or. State-of-the-art in the architecture, methods and applications of stylegan. In *Computer Graphics Forum*, volume 41, pages 591–611. Wiley Online Library, 2022. 1

[11] Uttaran Bhattacharya, Nicholas Rewkowski, Abhishek Banerjee, Pooja Guhan, Aniket Bera, and Dinesh Manocha. Text2gestures: A transformer-based network for generating emotive body gestures for virtual agents. In *2021 IEEE Virtual Reality and 3D User Interfaces (VR)*, pages 1–10. IEEE, 2021. 2

[12] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018. 3

[13] Pablo Cervantes, Yusuke Sekikawa, Ikuro Sato, and Koichi Shinoda. Implicit neural representations for variable length human motion generation. *arXiv preprint arXiv:2203.13694*, 2022. 2

[14] Bruno Degardin, João Neves, Vasco Lopes, João Brito, Ehsan Yaghoubi, and Hugo Proença. Generative adversarial graph convolutional networks for human action synthesis. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1150–1159, Los Alamitos, CA, USA, 2022. IEEE Computer Society. 2, 3, 4

[15] Yinglin Duan, Tianyang Shi, Zhengxia Zou, Yenan Lin, Zhehui Qian, Bohan Zhang, and Yi Yuan. Single-shot motion completion with transformer. *arXiv preprint arXiv:2103.00776*, 2021. 2

[16] 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. IEEE Computer Society, 2015. 2

[17] Saeed Ghorbani, Calden Wloka, Ali Etemad, Marcus A. Brubaker, and Nikolaus F. Troje. Probabilistic character motion synthesis using a hierarchical deep latent variable model. *Computer Graphics Forum*, 2020. 2

[18] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014. 4

[19] Brian Gordon, Sigal Raab, Guy Azov, Raja Giryes, and Daniel Cohen-Or. Flex: Extrinsic parameters-free multi-view 3d human motion reconstruction. In *European Conference on Computer Vision (ECCV)*, pages 176–196. Springer, 2022. 4

[20] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5152–5161, 2022. 2

[21] 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 *Proceedings of the 28th ACM International Conference on Multimedia*, pages 2021–2029, 2020. 2, 7, 8

[22] 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

[23] Félix G Harvey and Christopher Pal. Recurrent transition networks for character locomotion. In *SIGGRAPH Asia 2018 Technical Briefs*, pages 1–4, 2018. 2

[24] Félix G Harvey, Mike Yurick, Derek Nowrouzezahrai, and Christopher Pal. Robust motion in-betweening. *ACM Transactions on Graphics (TOG)*, 39(4):60–1, 2020. 2

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

[26] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020. 3

[27] Daniel Holden, Ikhsanul Habibie, Ikuo Kusajima, and Taku Komura. Fast neural style transfer for motion data. *IEEE computer graphics and applications*, 37(4):42–49, 2017. 2

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

[29] 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

[30] 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

[31] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE international conference on computer vision*, pages 1501–1510, 2017. 4

[32] 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

[33] Tero Karras, Miika Aittala, Samuli Laine, Erik Härkönen, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Alias-free generative adversarial networks. *Advances in Neural Information Processing Systems*, 34, 2021. 3

[34] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8110–8119, 2020. 1, 2, 3, 4, 8

[35] Manuel Kaufmann, Emre Aksan, Jie Song, Fabrizio Pece, Remo Ziegler, and Otmar Hilliges. Convolutional autoencoders for human motion infilling. In *2020 International Conference on 3D Vision (3DV)*, pages 918–927. IEEE, 2020. 2

[36] Durk P Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. *Advances in neural information processing systems*, 31, 2018. 3

[37] Juheon Lee, Seohyun Kim, and Kyogu Lee. Listen to dance: Music-driven choreography generation using autoregressive encoder-decoder network. *arXiv preprint arXiv:1811.00818*, 2018. 2

[38] Maosen Li, Siheng Chen, Zijing Zhang, Lingxi Xie, Qi Tian, and Ya Zhang. Skeleton-parted graph scattering networks for 3d human motion prediction. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part VI*, pages 18–36. Springer, 2022. 2

[39] Peizhuo Li, Kfir Aberman, Zihan Zhang, Rana Hanocka, and Olga Sorkine-Hornung. Ganimator: Neural motion synthesis from a single sequence. *ACM Transactions on Graphics (TOG)*, 41(4):138, 2022. 4

[40] Ruilong Li, Shan Yang, David A Ross, and Angjoo Kanazawa. Learn to dance with aist++: Music conditioned 3d dance generation. *arXiv e-prints*, pages arXiv–2101, 2021. 2

[41] Shubh Maheshwari, Debtanu Gupta, and Ravi Kiran Sarvadevabhatla. Mugl: Large scale multi person conditional action generation with locomotion. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 257–265, Los Alamitos, CA, USA, 2022. IEEE Computer Society. 2

[42] Yotam Nitzan, Rinon Gal, Ofir Brenner, and Daniel Cohen-Or. Large: Latent-based regression through gan semantics. *arXiv preprint arXiv:2107.11186*, 2021. 4

[43] Mathis Petrovich, Michael J Black, and Gül Varol. Action-conditioned 3d human motion synthesis with transformer vae. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10985–10995. IEEE Computer Society, 2021. 2, 8

[44] Mathis Petrovich, Michael J. Black, and Gül Varol. Temos: Generating diverse human motions from textual descriptions, 2022. 2

[45] Elad Richardson, Yuval Alaluf, Or Patashnik, Yotam Nitzan, Yaniv Azar, Stav Shapiro, and Daniel Cohen-Or. Encoding in style: a stylegan encoder for image-to-image translation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2287–2296, 2021. 2

[46] Yujun Shen, Ceyuan Yang, Xiaoou Tang, and Bolei Zhou. InterFaceGAN: interpreting the disentangled face representation learned by GANs. *arXiv preprint arXiv:2005.09635*, 2020. 4, 6

[47] Mingyi Shi, Kfir Aberman, Andreas Aristidou, Taku Komura, Dani Lischinski, Daniel Cohen-Or, and Baoquan Chen. Motionet: 3d human motion reconstruction from monocular video with skeleton consistency. *ACM Transactions on Graphics (TOG)*, 40(1):1–15, 2020. 4, 5

[48] Theodoros Sofianos, Alessio Sampieri, Luca Franco, and Fabio Galasso. Space-time-separable graph convolutional network for pose forecasting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11209–11218, 2021. 2

[49] Guofei Sun, Yongkang Wong, Zhiyong Cheng, Mohan S Kankanhalli, Weidong Geng, and Xiangdong Li. Deepdance: music-to-dance motion choreography with adversarial learning. *IEEE Transactions on Multimedia*, 23:497–509, 2020. 2

[50] Guy Tevet, Brian Gordon, Amir Hertz, Amit H Bermano, and Daniel Cohen-Or. Motionclip: Exposing human motion generation to clip space. *arXiv preprint arXiv:2203.08063*, 2022. 2

[51] Guy Tevet, Sigal Raab, Brian Gordon, Yonatan Shafir, Amit H Bermano, and Daniel Cohen-Or. Human motion diffusion model. *arXiv preprint arXiv:2209.14916*, 2022. 1, 2, 8

[52] Ruben Villegas, Jimei Yang, Duygu Ceylan, and Honglak Lee. Neural kinematic networks for unsupervised motion retargetting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8639–8648, 2018. 2

[53] Qi Wang, Thierry Artières, Mickael Chen, and Ludovic Denoyer. Adversarial learning for modeling human motion. *The Visual Computer*, 36(1):141–160, 2020. 2

[54] Zhenyi Wang, Ping Yu, Yang Zhao, Ruiyi Zhang, Yufan Zhou, Junsong Yuan, and Changyou Chen. Learning diverse stochastic human-action generators by learning smooth latent transitions. In *Proceedings of the AAAI conference on*

artificial intelligence, volume 34, pages 12281–12288, 2020. 2

[55] Sirui Xu, Yu-Xiong Wang, and Liang-Yan Gui. Diverse human motion prediction guided by multi-level spatial-temporal anchors. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXII*, pages 251–269. Springer, 2022. 2

[56] Sijie Yan, Zhizhong Li, Yuanjun Xiong, Huahan Yan, and Dahua Lin. Convolutional sequence generation for skeleton-based action synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4394–4402. IEEE Computer Society, 2019. 2, 4

[57] Zhengyuan Yang, Yuncheng Li, Jianchao Yang, and Jiebo Luo. Action recognition with spatio–temporal visual attention on skeleton image sequences. *IEEE Transactions on Circuits and Systems for Video Technology*, 29(8):2405–2415, 2018. 2

[58] Ping Yu, Yang Zhao, Chunyuan Li, Junsong Yuan, and Changyou Chen. Structure-aware human-action generation. In *European Conference on Computer Vision*, pages 18–34. Springer, 2020. 2, 3

[59] Ye Yuan and Kris Kitani. Dlow: Diversifying latent flows for diverse human motion prediction. In *European Conference on Computer Vision*, pages 346–364. Springer, 2020. 2

[60] Jia-Qi Zhang, Xiang Xu, Zhi-Meng Shen, Ze-Huan Huang, Yang Zhao, Yan-Pei Cao, Pengfei Wan, and Miao Wang. Write-an-animation: High-level text-based animation editing with character-scene interaction. In *Computer Graphics Forum*, volume 40, pages 217–228. Wiley Online Library, 2021. 2

[61] Yan Zhang, Michael J Black, and Siyu Tang. We are more than our joints: Predicting how 3d bodies move. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3372–3382, 2021. 2

[62] Yi Zhou, Zimo Li, Shuangjiu Xiao, Chong He, Zeng Huang, and Hao Li. Auto-conditioned recurrent networks for extended complex human motion synthesis. In *International Conference on Learning Representations*, 2018. 1, 2