Noname manuscript No.
(will be inserted by the editor)

MotionDiffuse: Text-Driven Human Motion Generation with Diffusion Model

Mingyuan Zhang* · **Zhongang Cai*** · **Liang Pan** · **Fangzhou Hong** · **Xinying Guo** · **Lei Yang** · **Ziwei Liu**

Received: date / Accepted: date

Abstract Human motion modeling is important for many modern graphics applications, which typically require professional skills. In order to remove the skill barriers for laymen, recent motion generation methods can directly generate human motions conditioned on natural languages. However, it remains challenging to achieve diverse and fine-grained motion generation with various text inputs. To address this problem, we propose **MotionDiffuse**, the first diffusion model-based text-driven motion generation framework, which demonstrates several desired properties over existing methods. *1) Probabilistic Mapping.* Instead of a deterministic language-motion mapping, MotionDiffuse generates motions through a series of denoising steps in which variations are injected. *2) Realistic Synthesis.* MotionDiffuse excels at modeling complicated data distribution and generating vivid motion sequences. *3) Multi-Level Manipulation.* MotionDiffuse responds to fine-grained instructions on body parts, and arbitrary-length motion synthesis with time-varied text prompts. Our experiments show MotionDiffuse outperforms existing SoTA methods by convincing margins on text-driven motion generation and action-conditioned motion generation. A qualitative analysis further demonstrates MotionDiffuse's controllability for comprehensive motion generation. Homepage: https://mingyuan-zhang.github.io/projects/MotionDiffuse.html

Keywords Motion Synthesis · Conditional Motion Generation · Diffusion Model · Text-driven Generation

Mingyuan Zhang
S-Lab, Nanyang Technological University, Singapore
E-mail: mingyuan001@e.ntu.edu.sg

Zhongang Cai
S-Lab, Nanyang Technological University, Singapore
E-mail: caiz0023@e.ntu.edu.sg

Liang Pan
S-Lab, Nanyang Technological University, Singapore
E-mail: liang.pan@ntu.edu.sg

Fangzhou Hong
S-Lab, Nanyang Technological University, Singapore
E-mail: fangzhouhong820@gmail.com

Xinying Guo
Nanyang Technological University, Singapore
E-mail: XGUO012@e.ntu.edu.sg

Lei Yang
SenseTime, China
E-mail: yanglei@sensetime.com

Ziwei Liu
S-Lab, Nanyang Technological University, Singapore
E-mail: ziwei.liu@ntu.edu.sg

* Equal Contributions

1 Introduction

Human motion modeling is a critical component of animating virtual characters to imitate vivid and rich human movements, which has been a vital topic for many applications, such as film-making, game development, and virtual YouTuber animation. To mimic human motions, virtual characters should be capable of moving around naturally, reacting to environmental stimuli, and meanwhile expressing sophisticated emotions. Despite decades of exciting technological breakthroughs, it requires sophisticated equipment (e.g., expensive motion capture systems) and domain experts to produce lively and authentic body movements. In order to remove skill prerequisites for layman users and potentially scale to the mass audience, it is vital to create a versatile human motion generation model that could produce diverse, easily manipulable motion sequences.

Fig. 1 MotionDiffuse is a diffusion model-based text-driven motion generation method that features *1) Probabilistic Mapping 2) Realistic Synthesis* that results in highly diverse motions with high-fidelity shown in a)-d), and *3) Multi-Level Manipulation* that empowers comprehensive motion generation such as that in c) where multiple body parts are involved and d) where time-varied prompts are given.

Various condition signals, including pre-defined motion categories (Guo et al., 2020; Petrovich et al., 2021; Cervantes et al., 2022), music pieces (Huang et al., 2020; Li et al., 2020, 2021; Zhuang et al., 2020; Siyao et al., 2022), and natural language (Lin et al., 2018; Ahuja and Morency, 2019; Ghosh et al., 2021; Petrovich et al., 2022), have been leveraged in previous human motion generation methods. Among them, natural language is arguably the most user-friendly and convenient input format for motion sequence synthesis, and hence we focus on text-driven motion generation in this work. Recently, TEMOS (Petrovich et al., 2022) utilizes KIT Motion-Language MoCap dataset (Plappert et al., 2016) to demonstrate fine-grained trajectory synthesis. However, it does not support stylizing the generated motions and, therefore, could not achieve high diversity. MotionCLIP (Tevet et al., 2022) could generate stylized motions, but it is still limited to short text inputs and fails to handle complicated motion descriptions. In addition, they (Petrovich et al., 2022; Tevet et al., 2022) typically only accept a single text prompt, which greatly limits users' creativity.

To tackle the aforementioned challenges, we propose **MotionDiffuse**, a versatile and controllable motion generation framework that could generate diverse motions with comprehensive texts. Inspired by the recent progress of the text-conditioned image generation (Dhariwal and Nichol, 2021; Nichol and Dhariwal, 2021; Nichol et al., 2021; Ramesh et al., 2022), we propose to incorporate Denoising Diffusion Probabilistic Models (DDPM) (Ho et al., 2020) into motion generation. Unlike classical DDPM which is only capable of fixed-size generation, we propose a Cross-Modality Linear Transformer to achieve motion synthesis with an arbitrary length depending on the motion duration. Instead of learning a direct mapping between the text space and the motion space (Tevet et al., 2022), we propose to guide the generation pipeline with input texts softly, which could significantly increase the diversity of the generation results. To maintain the uncertainties in the denoising process, we process the noise terms conditioned on the input texts by several transformer decoder layers for each denoising step. In this way, the text conditions would not dominate the motion generation in a deterministic way, which facilitates generating diverse motion sequences from the driving texts.

Furthermore, MotionDiffuse can achieve body part-independent control with fine-grained texts. Specifically, to accommodate the human body structures, MotionDiffuse divides the whole-body motion into several near-independent parts(*e.g.* upper body and lower body). Based on the fine-grained body parts definition, we propose 'noise interpolation' to separately control different body parts while taking their correlations into consideration. Moreover, to synthesize arbitrary-length motion sequences, we propose a new sampling method to denoise several overlapped sequences simultaneously. Specifically, MotionDiffuse first gets results from each sequence independently and then mixes them with correction terms. Different from auto-regressive inference schemes that often require many long motion sequences for training, MotionDiffuse is capable of modeling the

correlations between continuous actions without introducing additional training costs.

We perform both extensive qualitative experiments on popular benchmarks, and quantitative evaluation on comprehensive motion generation. Firstly, we demonstrate significant improvements on text-driven motion generation over the current art on HumanML3D and KIT-ML. Secondly, to illustrate the superiority of MotionDiffuse, we directly apply it to action-conditioned motion generation tasks, and it outperforms all existing works on the HumanAct12 dataset and the UESTC datasets. Furthermore, we also demonstrate more possibilities of MotionDiffuse by conditioning the model on mixed control signals that allow body-part level manipulation and long motion generation.

In summary, our proposed MotionDiffuse has several desired properties over prior arts:

- *Probabilistic Mapping.* Benefiting from our new formulation of motion generation, where the DDPM is incorporated in, MotionDiffuse can be conditioned on text descriptions to generate motions in a probabilistic style, naturally leading to high diversity.
- *Realistic Synthesis.* The careful design of the architecture allows MotionDiffuse to synthesize high-fidelity motion sequences and achieves state-of-the-art on two conditional motion generation tasks.
- *Multi-Level Manipulation.* With the extended design, MotionDiffuse handles fine-grained text descriptions that mobilize the entire body (*e.g.* 'a person is drinking water while walking') and time-varied signals (*e.g.* 'a person is walking and then running').

2 Related Work

2.1 Motion Generative Model

Motion Generation has been studied for decades. Some early works focus on unconditional motion generation (Rose et al., 1998; Ikemoto et al., 2009; Mukai and Kuriyama, 2005). While some other works try to predict future motions given an initial pose or a starter motion sequence (Futrelle and Speckert, 1978; Gavrila, 1999; O'rourke and Badler, 1980). Statistical models such as PCA (Ormoneit et al., 2005), Motion Graph (Min and Chai, 2012) are applied for these purposes.

With the rapid development of Deep Learning (DL) techniques, more generative architectures occur and flourish. Previous works can be broadly divided into four groups: 1) Variational Auto Encoder (VAE); 2) Generative Adversarial Networks (GAN); 3) Normalization Flow Network; 4) Implicit Neural Representations.

VAE (Kingma and Welling, 2013) is one of the most commonly used generative models in motion synthesis. Yan et al. (2018) and Aliakbarian et al. (2020) regard the motion generation task as predicting a small future motion sequence with the given small current motion sequence. They use VAE to encode the pair of current sequence and future sequence and then reconstruct the future one. ACTOR (Petrovich et al., 2021) proposes a transformer-based encoder and decoder architecture. Transformer Encoder Layers and Transformer Decoder Layers (Vaswani et al., 2017) are the basic blocks to build up a motion encoder and a motion decoder. This architecture is also employed in later works (Tevet et al., 2022; Hong et al., 2022; Petrovich et al., 2022).

GAN (Goodfellow et al., 2014) introduces an auxiliary module, discriminator network, to justify the quality and validity of generated samples. Some works focus on proposing appropriate discriminator networks for motion generation to improve the synthesis quality (Barsoum et al., 2018; Harvey et al., 2020; Wang et al., 2020). HP-GAN (Barsoum et al., 2018) attempts to supervise the motion prediction results without the specific ground truth. Therefore, a data-driven discriminator is involved in learning a motion prior, which is used to justify the prediction quality. Harvey et al. (2020) target solving the blurriness of the predicted motion in the Motion In-between task and propose two discriminators for both short-term critic and long-term critic. Wang et al. (2020) build up a cyclic pipeline. With the help of a discriminator, the proposed pipeline can generate both class-specific and mixed-class motion sequences.

Normalization Flow Network (Dinh et al., 2014) has a long history and has been studied extensively for image synthesis (Dinh et al., 2016; Kingma and Dhariwal, 2018). This kind of architecture builds up a reversible neural network and will map the input data into a multi-dimensional Gaussian distribution. Hence, we can generate an initially random vector from this distribution and feed them into the reversed network to generate motion samples. Inspired by the success of GLOW (Kingma and Dhariwal, 2018), MoGlow (Henter et al., 2020) proposes an auto-regressive normalization network to model motion sequences. History features from an LSTM model (Hochreiter and Schmidhuber, 1997) serve as the condition of the flow network, which predicts the next pose.

Recently, another generative model has attracted much attention with the considerable success achieved by NeRF (Mildenhall et al., 2020; Jain et al., 2021) in

rendering realistic images. Implicit Neural Representations (INR) are a series of neural networks that optimize their parameters to fit one sample instead of the whole distribution. One principal advantage is that this technique has superb generalization ability on spatial or temporal dimensions. For example, Cervantes et al. (2022) propose an implicit scheme, which simultaneously models action category and timestamp. Similar to the original NeRF, the timestamp is represented by sinusoidal values. After supervised training, the proposed method can generate a variable-length motion sequence for each action category.

This paper proposes a new motion generation pipeline based on the Denoising Diffusion Probabilistic Model (DDPM) (Ho et al., 2020). One of the principal advantages of DDPM is that the formation of the original motion sequence can be retained. It means that we can easily apply more constraints during the denoising process. In the later sections, we will explore more potential of DDPM in different types of conditions. Besides, benefiting from this nature, DDPM can generate more diverse samples.

2.2 Conditional Motion Generation

The increasing maturity of various generative models stimulates researchers' enthusiasm to study conditional motion generation. For example, some works (Guo et al., 2020; Petrovich et al., 2021; Cervantes et al., 2022) aim at synthesizing motion sequences of several specific categories. Action2Motion (Guo et al., 2020) builds up a recurrent conditional VAE for motion generation. Given history memory, this model predicts the next pose under the constraints of the action category. ACTOR (Petrovich et al., 2021) also uses VAE for random sampling. Unlike Action2Motion, ACTOR embeds the whole motion sequence into the latent space. This design avoids the accumulative error in the recurrent scheme. Besides, ACTOR proposes a Transformer-based motion encoder and decoder architecture. This structure significantly outperforms recurrent methods. Cervantes et al. (2022) attempt to model motion sequence with implicit functions, which can generate motion sequences with varied lengths.

Another significant conditional motion generation task is music to dance. This task requires that the generated motion has beat-wise connectivity, is a specific kind of dance, or can express similar content with the music. Many works attempt to embed the music feature and motion feature into a joint space (Lee et al., 2019; Sun et al., 2020; Li et al., 2020, 2021). Unlike direct feature embedding, Bailando (Siyao et al., 2022) proposes a two-stage dance generator. It first learns a quantized codebook of meaningful dance pieces and then attempts to generate the whole sequence with a series of elements from a codebook.

Similar to music-to-dance, text-driven motion generation can be regarded as learning a joint embedding of text feature space and motion feature space. There are two major differences. The first one is that language commands correlate more with the human body. Therefore, we expect to control each body part accurately. The second difference is that text-driven motion generation contains a vast range of motions. Some descriptions are direct commands to a specific body part, such as "touch head". Some describes arbitrary concepts like "playing the violin". Such huge complexity of motions brings many difficulties to the architecture design. Recently, many works have proposed text-driven motion generation pipelines. Most of them are deterministic generation (Ahuja and Morency, 2019; Ghosh et al., 2021; Tevet et al., 2022), which means they can only generate a single result from the given text. TEMOS (Petrovich et al., 2022) introduces the VAE architecture into this task. It can generate different motion sequences given one text description. However, these methods attempt to acquire a joint embedding space of motion and natural language. This design significantly compresses the information from text. Therefore, these works can hardly generate correct motion sequences from a detailed description. Guo et al. (2022) proposes an auto-regressive pipeline. It first encodes language descriptions into features and then auto-regressively generates motion frames conditioned on the text features. However, this method is hard to capture the global relation due to the auto-regressive scheme. Moreover, the generation quality is inferior. Instead, our proposed MotionDiffuse softly fuses text features into generation and can yield the whole sequence simultaneously. The experiment results prove the superiority of our design.

2.3 Motion Datasets

Human motion modeling has been a long-standing problem in computer vision and computer graphics. With the advent of deep learning, data has become increasingly important for training neural networks that perceive, understand, and generate human motions.

A common form of datasets containing videos of human subjects are recorded with annotations such as 2D keypoints (Jhuang et al., 2013; Andriluka et al., 2018), 3D keypoints (Ionescu et al., 2013; Joo et al., 2015; Mehta et al., 2017; Trumble et al., 2017; Li et al., 2021) and statistical model parameters (Yu et al., 2020; Patel et al., 2021; Cao et al., 2020; Cai et al., 2021, 2022). Action labels are also a popular attribute of datasets

for human action understanding that contains human-centric actions (Kuehne et al., 2011; Soomro et al., 2012; Karpathy et al., 2014; Gu et al., 2018; Shao et al., 2020; Chung et al., 2021), interaction (Carreira et al., 2019; Monfort et al., 2019; Zhao et al., 2019), fine-grained action understanding (Gu et al., 2018; Shao et al., 2020; Chung et al., 2021) and 3D data (Liu et al., 2019). For the action-conditioned motion generation task, HumanAct12 (Guo et al., 2020), UESTC (Ji et al., 2018), and NTU RGB+D (Liu et al., 2019) are three commonly used benchmarks. However, the above-mentioned datasets do not provide paired sophisticated semantic labels to the motion sequences.

KIT (Plappert et al., 2016) contains motion capture data annotated with detailed descriptions. Zhang et al. (2020) recruit actors and actresses to record body movements expressing emotions. Recently, BABEL (Punnakkal et al., 2021) and HumanML3D (Guo et al., 2022) re-annotates AMASS (Mahmood et al., 2019), a large scale motion capture dataset, with English language labels.

In this paper, we use the HumanML3D dataset and KIT dataset to evaluate the proposed methods for the text-driven motion generation task. HumanAct12 and UESTC are used to demonstrate the wide applicability of the proposed pipeline. Furthermore, we use the BABEL dataset for additional applications.

3 Methodology

We present a diffusion model-based framework, **MotionDiffuse**, for high-fidelity and controllable text-driven motion generation. We first give the problem definition, settings of the original text-driven motion generation in Section 3.1. After that, we provide an overall illustration of the proposed MotionDiffuse in Section 3.2, followed by introducing the diffusion model in Section 3.3 and the transformer-based architecture in Section 3.4. Finally, the inference strategy is illustrated for the fine-grained generation scenarios in Section 3.5.

3.1 Preliminaries

The motion sequence Θ is an array of (θ_i), $i \in \{1, 2, \ldots, F\}$, where $\theta_i \in \mathbb{R}^D$ represents the pose state in the i-th frame, and F is the number of frames. The representation of each pose state θ_i is distinct in different datasets. It generally contains joint rotation, joint position, joint velocity, and foot contact conditions. Our proposed MotionDiffuse is robust to the various motion representations. Therefore, we do not specify the components of θ_i in this section, and leave the details in Section 4.

For standard Text-driven Motion Generation, the training datasets consist of $(\theta_i, \text{text}_i)$ pairs, where text_i is the language description of motion sequence θ_i. During inference, given a set of descriptions $\{\text{text}_i\}$, we are requested to generate motion sequences conditioned on the given descriptions. This task can also be regarded as a text-to-motion translation (T2M) task. We will use this abbreviation below.

An related task is Action-conditioned Motion Generation. Given a pre-defined action category set, models are supposed to fit the data distribution and synthesize motion sequences of each category. Annotated data in this task can be represented as (y_i, Θ_i), where y_i is the category index of i-th data, Θ_i is the motion sequence of i-th data. In this paper, we replace y_i by its semantic description text_i. Then we can use the same pipeline as in the T2M task.

3.2 Pipeline Overview

Following the literature on the diffusion model in the image synthesis field (Ho et al., 2020), we first build up a text-conditioned motion generation pipeline using a denoising diffusion probabilistic model (DDPM). This model is the basis of our proposed MotionDiffuse. For the denoising process, we propose a Cross-Modality Linear Transformer to process input sequences conditioned on the given text prompts. Beyond the direct application of text-driven motion generation, we take one step further to explore methods that are conditioned on motion representation during denoising. Specifically, we experiment with two types of additional signals: part-aware text controlling and time-varied controlling. The former assigns different text conditions to different body parts so that we can accurately control each part of the body and generate more complicated motion sequences. The latter divides the whole sequence into several parts and assigns independent text conditions for each interval. Therefore, we can synthesize arbitrary-length motion sequences that incorporate several actions. These two kinds of conditions significantly expand the capability of MotionDiffuse. The overall pipeline is shown in Figure 2. We introduce each part of this architecture in the following subsections.

3.3 Diffusion Model for Motion Generation

Generative Adversarial Networks (GANs) involve a discriminator to improve the generation quality in an adversarial manner. GANs are typically challenging

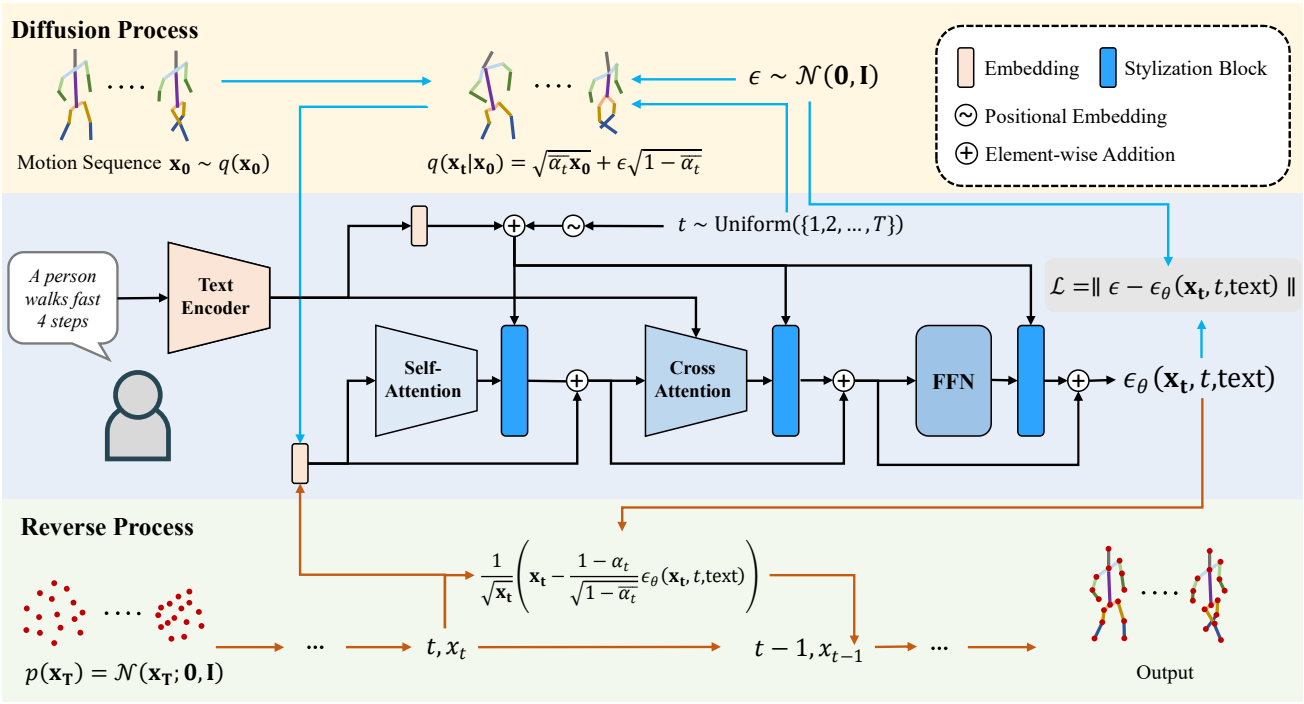

Fig. 2 Overall Pipeline of the proposed MotionDiffuse. The colors of the arrows indicate different stages: blue for training, red for inference, and black for both training and inference.

to train, especially for conditional motion generation tasks. Implicit Functions use Multi-Layer Perceptron (MLP) to fit motion sequences. This neat architecture is easily trained on a small number of data but tends to be less generalizable when it is subjected to complicated conditions. Auto-Encoder (AE) and Variational Auto-Encoder (VAE) are the most widely used approaches in text-driven motion generation (Ghosh et al., 2021; Petrovich et al., 2022). Previous works learn a joint embedding of motion sequences and languages that explicitly apply the text condition in the deterministic language-motion mapping. However, high-level text features typically contain insufficient fine-grained details to guide the generation of subtly different motion. Hence, directly linking text embedding to motion embedding results in the limited diversity of the generated motions.

To tackle the problem, we build our text-driven motion generation pipeline based on diffusion models. Diffusion Models (Ho et al., 2020; Dhariwal and Nichol, 2021; Nichol and Dhariwal, 2021; Nichol et al., 2021) are a new class of generative models. A probabilistic model is learned to gradually denoises a Gaussian noise to generate a target output, such as a 2D image or 3D point cloud. Formally, diffusion models are formulated as $p_\theta(\mathbf{x}_0) := \int p_\theta(\mathbf{x}_{0:T})\, d\mathbf{x}_{1:T}$, where $\mathbf{x}_0 \sim q(\mathbf{x}_0)$ is the real data, and $\mathbf{x}_1, \cdots, \mathbf{x}_T$ are the latent data. They generally have a diffusion process and a reverse process. To approximate posterior $q(\mathbf{x}_{1:T}|\mathbf{x}_0)$, the diffusion process follows a Markov chain to gradually add Gaussian noise to the data until its distribution is close to the latent distribution $\mathcal{N}(\mathbf{0}, \mathbf{I})$, according to variance schedules given by β_t:

$$\begin{aligned} q(\mathbf{x}_{1:T}|\mathbf{x}_0) &:= \prod_{t=1}^{T} q(\mathbf{x}_t|\mathbf{x}_{t-1}), \\ q(\mathbf{x}_t|\mathbf{x}_{t-1}) &:= \mathcal{N}(\mathbf{x}_t; \sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t\mathbf{I}). \end{aligned} \tag{1}$$

The reverse process $p_\theta(\mathbf{x}_{0:T})$ is also a Markov chain that predicts and eliminates the noise with learned Gaussian transitions starting at $p(\mathbf{x}_T) = \mathcal{N}(\mathbf{x}_T; \mathbf{0}, \mathbf{I})$:

$$\begin{aligned} p_\theta(\mathbf{x}_{0:T}) &:= p(\mathbf{x}_T) \prod_{t=1}^{T} p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t), \\ p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) &:= \mathcal{N}(\mathbf{x}_{t-1}; \mu_\theta(\mathbf{x}_t, t), \Sigma_\theta(\mathbf{x}_t, t)). \end{aligned} \tag{2}$$

To accomplish the reverse process of the diffusion model, we need to construct and optimize a neural network. During training, first we uniformly sample steps t for each ground truth motion $\mathbf{x}_0$ and then generate a sample from $q(\mathbf{x}_t|\mathbf{x}_0)$. Instead of repeatedly adding noises on $\mathbf{x}_0$, Ho et al. (2020) formulate the diffusion process as

$$q(\mathbf{x}_t|\mathbf{x}_0) = \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \epsilon\sqrt{1-\bar{\alpha}_t}, \epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \tag{3}$$

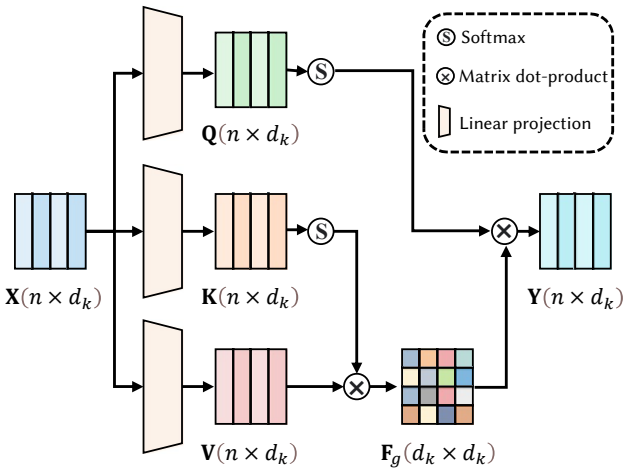

Fig. 3 Architecture of our Linear Self-attention.

where $\alpha_t = 1-\beta_t$, $\bar{\alpha}_t = \prod_{s=0}^{t} \alpha_s$. Hence, we can simply sample a noise ϵ and then directly generate $\mathbf{x}_t$ by this formulation. Instead of predicting $\mathbf{x}_{t-1}$, here we follow GLIDE (Nichol et al., 2021) and predict the noise term ϵ. It means that we need to construct a network to fit $\epsilon_\theta(\mathbf{x}_t, t, \text{text})$. We optimize the model parameters to decrease a mean squared error as

$$\mathcal{L} = \mathrm{E}_{t\in[1,T],\mathbf{x}_0\sim q(\mathbf{x}_0),\epsilon\sim\mathcal{N}(\mathbf{0},\mathbf{I})}[\| \epsilon - \epsilon_\theta(\mathbf{x}_t, t, \text{text}) \|]. \quad (4)$$

This is the only loss we used in model training. To generate samples from the given text description, we denoise the sequence from $p(\mathbf{x}_T) = \mathcal{N}(\mathbf{x}_T; \mathbf{0}, \mathbf{I})$. Equation 2 shows that we need to estimate $\mu_\theta(\mathbf{x}_t, t, \text{text})$ and $\Sigma_\theta(\mathbf{x}_t, t, \text{text})$. To simplify the problem, we set $\Sigma_\theta(\mathbf{x}_t, t, \text{text})$ as a constant number β_t. $\mu_\theta(\mathbf{x}_t, t, \text{text})$ can be estimated as

$$\mu_\theta(\mathbf{x}_t, t, \text{text}) = \frac{1}{\sqrt{\mathbf{x}_t}}(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(\mathbf{x}_t, t, \text{text})). \quad (5)$$

Therefore we can denoise the motion sequence step by step and finally get a clean motion sequence, which is conditioned on the given text.

3.4 Cross-Modality Linear Transformer

In Section 3.3, we illustrate diffusion models as motion generators and a neural network $\epsilon_\theta(\mathbf{x}_t, t, \text{text})$ is essential for denoising steps. In this section, we will introduce the design of $\epsilon_\theta(\mathbf{x}_t, t, \text{text})$ in our proposed MotionDiffuse.

Previous works (Ho et al., 2020; Dhariwal and Nichol, 2021; Nichol and Dhariwal, 2021; Nichol et al., 2021) mainly utilize UNet-like structure as the denoising model. However, the target motion sequences are variable-length in the motion generation task, making convolution-based networks unsuitable. Therefore, we propose a Cross-Modality Linear Transformer, as shown in Figure 2. Similar to the machine translation task, our proposed model includes a text encoder and a motion decoder. To meet the requirement of the diffusion models, we further customize each layer of the motion decoder.

Text Encoder Here we directly use classical transformer Vaswani et al. (2017) to extract text features. Specifically, the input data first passes through an embedding layer to get the embedding feature from raw text and then is further processed by a series of transformer blocks. Each block contains two components: a multi-head attention module (MHA) and a feed-forward network (FFN). Suppose the input feature is $\mathbf{X} \in \mathbb{R}^{n\times d}$ (n denotes the number of elements and d denotes the element feature dimension), MHA extracts query feature vectors $\mathbf{Q} \in \mathbb{R}^{n\times d}$, key feature vectors $\mathbf{K} \in \mathbb{R}^{n\times d}$, and value feature vectors $\mathbf{V} \in \mathbb{R}^{n\times d}$ as:

$$\mathbf{Q} = W_q\,\mathbf{X}, \quad \mathbf{K} = W_k\,\mathbf{X}, \quad \mathbf{V} = W_v\,\mathbf{X}, \quad (6)$$

where W_q, W_k and W_v are the corresponding linear projections to generate $\mathbf{Q}$, $\mathbf{K}$ and $\mathbf{V}$, respectively. The value features are then aggregated with attention weights $\mathbf{A} \in \mathbb{R}^{n\times n}$:

$$\mathbf{A} = \text{softmax}(\frac{\mathbf{Q}\otimes\mathbf{K}^\top}{\sqrt{d}}), \quad \mathbf{Y} = \mathbf{A}\otimes\mathbf{V}, \quad (7)$$

where $\mathbf{Y} \in \mathbb{R}^{n\times d}$ is the output of MHA modules, d is the dimension of each element in $\mathbf{X}$, and $\otimes$ denotes the matrix multiplication. The multi-head mechanism divides the input vector into several parts, which pass through the process in Equation 6 and 7 independently. The outputs are concatenated so that the dimension remains unchanged. A residual connection is applied between input and output of the MHA modules. This feature is further processed by FFN, which contains three linear transformations and two GELU (Hendrycks and Gimpel, 2016) layers between them.

To enhance the generalization ability, we use parameter weights in CLIP (Radford et al., 2021) to initialize the first several layers. This part of the parameters is frozen and will not be optimized in the later training.

Linear Self-attention. This module aims at enhancing motion features by modeling correlations between different frames. The principal advantage of self-attention is to get an overview of the input sequence and is thus beneficial to estimating the injected noise

ϵ. However, the time complexity of calculating attention weight $\mathbf{A} \in \mathbb{R}^{n \times n}$ is $\mathcal{O}(n^2 d)$. When the target motion sequence length increases, the time cost increases quadratically. In the T2M task, the length can be several hundred, which leads to low speed. Hence, we adopt Efficient Attention (Shen et al., 2021) to speed up the self-attention module. Instead of calculating pair-wise attention weights, efficient attention generates global feature map $\mathbf{F}_g \in \mathbb{R}^{d_k \times d_k}$, where d_k is the dimension of feature after multi-head split:

$$\mathbf{F}_g = \text{softmax}(\mathbf{K}^\top) \otimes \mathbf{V}, \quad \mathbf{Y} = \text{softmax}(\mathbf{Q}) \otimes \mathbf{F}_g. \quad (8)$$

The time complexity of these two steps is $\mathcal{O}(d_k^2 nk) = \mathcal{O}(dd_k n)$, where n is the number of the elements in the sequences, k is the number of heads of self-attention.

Another advantage of efficient attention for the diffusion model is that feature map $\mathbf{F}_g$ explicitly aggregates global information while classical self-attention focus more on pair-wise relation. Global information gives more clues about the semantic meaning of the motion sequence than pair-wise one. The experiment results also prove this conclusion.

Linear Cross-attention. Cross-attention replaces $\mathbf{X}$ in $\mathbf{K}$ and $\mathbf{V}$ calculation by the text feature. Other formulations are the same as Linear Self-attention. Text features are injected into motion sequences in this process to generate motion conditioned on the given text.

Stylization Block. In each denoising step, the output is conditioned on the given text and the timestamp t. Linear Cross-attention fuses the text features into motion sequences. We need another Stylization Block component to bring timestamp t to the generation process. This block is applied after each Linear Self-attention block, Linear Cross-attention block, and FFN block.

Similar to GLIDE (Nichol et al., 2021), we first get a text embedding e_{text} by a linear transformation on the text features and a timestamp embedding $\mathbf{e_t}$ by positional embedding (Vaswani et al., 2017). These two terms are summed together into one single vector $\mathbf{e}$. Given the original output $\mathbf{Y}$ from other blocks, the Stylization block will process the feature as:

$$\mathbf{B} = \psi_b(\phi(\mathbf{e})), \quad \mathbf{W} = \psi_w(\phi(\mathbf{e})), \quad \mathbf{Y}' = \mathbf{Y} \cdot \mathbf{W} + \mathbf{B}, \quad (9)$$

where $(\cdot)$ denotes Hadamard product, $\mathbf{Y}'$ is the output of stylization blocks. ψ_b, ψ_w, ϕ denote three different linear projections. In classical transformers, the output from each block is added to the original input as a residual connection, as shown in Figure 2. In MotionDiffuse, these outputs pass through stylization blocks and are added to the information. This modification enables our proposed method to know the timestamp t.

3.5 Fine-grained Controlling

To enrich the capability of MotionDiffuse, we explore the properties of both the motion representation and the denoising process of DDPM. Unlike VAE, the generated motion sequence is in its explicit form instead of being compressed in the latent space. This characteristic of DDPM-based motion generation allows more operations to be applied to this motion sequence to increase the manipulability.

Body Part-independent Controlling. Due to the lack of diversity in text descriptions, we cannot achieve accurate motion control for each body part from text descriptions only. For example, the prompt 'a person is running and waving left hand' is highly challenging to the model because the expected motion sequence is significantly far from the training distribution. Even if we manually split the original description into two independent ones: 'a person is running' for lower limbs, and 'a person is waving left hand' for upper limbs, it is still difficult for the model to generate correct motions. An intuitive solution for this situation is to separately generate two motion sequences and combine the upper-limb motion of the first sequence and the lower-limb motion of the second sequence. This simple solution mitigates the problem to some extent. However, it ignores the correlation between these two parts. Specifically for 'running and waving left hand', the frequencies of the two motions should match. Otherwise, the motion generated by this naive method appears unnatural. To better solve this problem, we propose a body part-independent controlling scheme.

Recall that, during the denoising process, our diffusion model predicts the noise term $\epsilon_\theta(\mathbf{x}_t, t, \text{text}) \in \mathbb{R}^{F \times D}$, where F represents the number of frames, D denotes the dimension of each pose state, which includes translation and rotations of body joints. This noise term determines the denoising direction of the whole body.

Inspired by the application of the latent code interpolation, here we propose 'noise interpolation' to separately control the different parts of the human body. Suppose we have n text descriptions $\{\text{text}_i\}$ for different body parts $\{s_i\}$. We want to calculate the noise term $\epsilon = \{\epsilon_i^{\text{joint}}\}, i \in [1, m]$, where $\epsilon_i^{\text{joint}}$ represents the noise term for the i-th body part, m denotes the number of partition. We first estimate the noise $\epsilon_i^{\text{part}} = \epsilon_\theta(\mathbf{x}_t, t, \text{text}_i), \epsilon_i^{\text{part}} \in \mathbb{R}^{F \times D}$. An intuitive method for combining these terms is $\epsilon^{\text{part}} = \sum_{i=1}^{m} \epsilon_i^{\text{part}} \cdot M_i$, where $M_i \in \{0, 1\}^D$ is a binary vector to show which body part should we focus. $(\cdot)$ denotes the Hadamard product, and here we ignore the broadcast in computation for simplicity. Although this method succeeds to some extent, the direct ignoring of some parts in ϵ_i^{part} will in-

crease the combination difficulty and lead to low-quality generation results. Therefore, we add a correction item for smoothing interpolation:

$$\bar{\epsilon}^{\text{part}} = \sum_{i=1}^{m} \epsilon_i^{\text{part}} \cdot M_i + \lambda_1 \cdot \nabla\Big(\sum_{1 \le i,j \le m} \|\epsilon_i^{\text{part}} - \epsilon_j^{\text{part}}\| \Big), \quad (10)$$

where ∇ denotes gradient calculation, λ_1 is a hyperparameter to balance these two items. This correction item enforces a smoother denoising direction so that the motion of different body parts will be more natural.

Time-varied Controlling Long-term motion generation plays a vital role in real-world applications. Previous works mainly focus on motion generation with a single type of motion. Auto-regressive methods (Henter et al., 2020; Guo et al., 2022) have solved this problem with satisfactory performance. However, none of them are capable of synthesizing different actions in a continuous manner. Benefiting from the nature of DDPM, here we propose another sampling method to meet this requirement.

Recall that we are given an array $\{\text{text}_{i,j}, [l_{i,j}, r_{i,j}]\}$, $i \in [1, m]$, where m is the number of intervals. Similar to the method we proposed in the previous paragraph, we first estimate the noise term ϵ_i^{time} for i-th interval independently. Suppose the overall length of the target sequence is F'. By padding zeros, we extend each noise term into the same dimension $F' \times D$. Then we interpolate these noises with a correcting term:

$$\bar{\epsilon}^{\text{time}} = \sum_{i=1}^{m} \bar{\epsilon}_i^{\text{time}} + \lambda_2 \cdot \nabla\Big(\sum_{1 \le i,j \le m} \|\bar{\epsilon}_i^{\text{time}} - \bar{\epsilon}_j^{\text{time}}\| \Big), \quad (11)$$

where $\bar{\epsilon}_j^{\text{time}}$ is the padded term from ϵ_i^{time}, λ_2 is a hyperparameter.

4 Experiments

We evaluate MotionDiffuse with three categories of experiments: text-driven motion generation (Section 4.1), action-conditioned motion generation (Section 4.2), and motion manipulation (Section 4.3). In all the evaluated benchmarks, MotionDiffuse could significantly outperform previous SoTA methods.

4.1 Text-driven Motion Generation

Datasets. KIT Motion Language datset (Plappert et al., 2016) provides 3911 motion sequences and 6353 sequence-level natural language descriptions. HumanML3D (Guo et al., 2022) re-annotates the AMASS dataset (Mahmood et al., 2019) and the HumanAct12 dataset (Guo et al., 2020). It provides 44970 annotations on 14616 motion sequences. KIT and HumanML3D are two important benchmarks for text-driven motion generation tasks. Following Guo et al. (2022), we utilize the pretrained text-motion contrastive model.

Evaluation Metrics. We evaluate all methods with five different metrics. 1) *Frechet Inception Distance (FID).* Features are extracted from both the generated results and ground truth motion sequences by the pretrained motion encoder. FID is calculated between these two distributions to measure the similarity. 2) *R Precision.* For each pair of generated sequence and description text, 31 other prompts are randomly selected from the test set. The pretrained contrastive model calculates the average top-k accuracy. This section reports the top-1, top-2, and top-3 accuracies. 3) *Diversity.* The generated sequences from all test texts are randomly split into pairs. Then the average joint differences are calculated in each pair, which serves as the diversity metric. 4) *Multimodality.* As for a single text description, we randomly generate 32 motion sequences. Multimodality measures the differences in joint positions between these homogeneous motion sequences. 5) *Multimodal Distance.* Assisted by the pretrained contrastive model, we can calculate the difference between the text feature from the given description and the motion feature from the generated results, called multimodal distance.

In this section, R Precision and FID are the principal metrics from which we make the most conclusions. Besides, for a fair comparison, we run each evaluation 20 times and report the statistic interval with 95% confidence.

Implementation Details. For both HumanML3D and KIT-ML datasets, we build up an 8-layer transformer as the motion decoder. As for the text encoder, we first directly use the text encoder in the CLIP ViT-B/32 (Radford et al., 2021), and then add four more transformer encoder layers. The latent dimension of the text encoder and the motion decoder are 256 and 512, respectively. As for the diffusion model, the number of diffusion steps is 1000, and the variances β_t are linearly from 0.0001 to 0.02. We opt for Adam as the optimizer to train the model with a 0.0002 learning rate. We use 8 Tesla V100 for the training, and there are 128 samples on each GPU, so the total batch size is 1024. The total number of iterations is 40K for KIT-ML and 100K for HumanML3D.

Following Guo et al. (2022), pose states in this series of experiments mainly contain seven different parts: $(r^{va}, r^{vx}, r^{vz}, r^h, \mathbf{j}^p, \mathbf{j}^v, \mathbf{j}^r)$. Here Y-axis is perpendicular

Table 1 Quantitative results on the HumanML3D test set. All methods use the real motion length from the ground truth. '→' means results are better if the metric is closer to the real motions. We run all the evaluation 20 times and ± indicates the 95% confidence interval. The best results are in **bold**.

Methods	R Precision↑			FID↓	MultiModal Dist↓	Diversity→	MultiModality
	Top 1	Top 2	Top 3				
Real motions	$0.511^{\pm.003}$	$0.703^{\pm.003}$	$0.797^{\pm.002}$	$0.002^{\pm.000}$	$2.974^{\pm.008}$	$9.503^{\pm.065}$	-
Language2Pose	$0.246^{\pm.002}$	$0.387^{\pm.002}$	$0.486^{\pm.002}$	$11.02^{\pm.046}$	$5.296^{\pm.008}$	$7.676^{\pm.058}$	-
Text2Gesture	$0.165^{\pm.001}$	$0.267^{\pm.002}$	$0.345^{\pm.002}$	$7.664^{\pm.030}$	$6.030^{\pm.008}$	$6.409^{\pm.071}$	-
MoCoGAN	$0.037^{\pm.000}$	$0.072^{\pm.001}$	$0.106^{\pm.001}$	$94.41^{\pm.021}$	$9.643^{\pm.006}$	$0.462^{\pm.008}$	$0.019^{\pm.000}$
Dance2Music	$0.033^{\pm.000}$	$0.065^{\pm.001}$	$0.097^{\pm.001}$	$66.98^{\pm.016}$	$8.116^{\pm.006}$	$0.725^{\pm.011}$	$0.043^{\pm.001}$
Guo et al.	$0.457^{\pm.002}$	$0.639^{\pm.003}$	$0.740^{\pm.003}$	$1.067^{\pm.002}$	$3.340^{\pm.008}$	$9.188^{\pm.002}$	$2.090^{\pm.083}$
Ours	$\mathbf{0.491^{\pm.001}}$	$\mathbf{0.681^{\pm.001}}$	$\mathbf{0.782^{\pm.001}}$	$\mathbf{0.630^{\pm.001}}$	$\mathbf{3.113^{\pm.001}}$	$\mathbf{9.410^{\pm.049}}$	$1.553^{\pm.042}$

Table 2 Quantitative results on the KIT-ML test set. All methods use the real motion length from the ground truth.

Methods	R Precision↑			FID↓	MultiModal Dist↓	Diversity→	MultiModality
	Top 1	Top 2	Top 3				
Real motions	$0.424^{\pm.005}$	$0.649^{\pm.006}$	$0.779^{\pm.006}$	$0.031^{\pm.004}$	$2.788^{\pm.012}$	$11.08^{\pm.097}$	-
Language2Pose	$0.221^{\pm.005}$	$0.373^{\pm.004}$	$0.483^{\pm.005}$	$6.545^{\pm.072}$	$5.147^{\pm.030}$	$9.073^{\pm.100}$	-
Text2Gesture	$0.156^{\pm.004}$	$0.255^{\pm.004}$	$0.338^{\pm.005}$	$12.12^{\pm.183}$	$6.964^{\pm.029}$	$9.334^{\pm.079}$	-
MoCoGAN	$0.022^{\pm.002}$	$0.042^{\pm.003}$	$0.063^{\pm.003}$	$82.69^{\pm.242}$	$10.47^{\pm.012}$	$3.091^{\pm.043}$	$0.250^{\pm.009}$
Dance2Music	$0.031^{\pm.002}$	$0.058^{\pm.002}$	$0.086^{\pm.003}$	$115.4^{\pm.240}$	$10.40^{\pm.016}$	$0.241^{\pm.004}$	$0.062^{\pm.002}$
Guo et al.	$0.370^{\pm.005}$	$0.569^{\pm.007}$	$0.693^{\pm.007}$	$2.770^{\pm.109}$	$3.401^{\pm.008}$	$10.91^{\pm.119}$	$1.482^{\pm.065}$
Ours	$\mathbf{0.417^{\pm.004}}$	$\mathbf{0.621^{\pm.004}}$	$\mathbf{0.739^{\pm.004}}$	$\mathbf{1.954^{\pm.062}}$	$\mathbf{2.958^{\pm.005}}$	$\mathbf{11.10^{\pm.143}}$	$0.730^{\pm.013}$

to the ground. $r^{va}, r^{vx}, r^{vz} \in \mathbb{R}$ denotes the root joint's angular velocity along Y-axis, linear velocity along X-axis and Z-axis, respectively. $r^h \in \mathbb{R}$ is the height of the root joint. $\mathbf{j}^p, \mathbf{j}^v \in \mathbb{R}^{J\times 3}$ are the position and linear velocity of each joint, where J is the number of joints. $\mathbf{j}^r \in \mathbb{R}^{J\times 6}$ is the 6D rotation (Zhou et al., 2019) of each joint. Specifically, J is 22 in HumanML3D and 21 in KIT-ML.

Quantitative Results. We compare our proposed MotionDiffuse with five baseline models: Language2Pose (Ahuja and Morency, 2019), Text2Gesture (Bhattacharya et al., 2021), MoCoGAN (Tulyakov et al., 2018), Dance2Music (Lee et al., 2019), and Guo et al. (2022). All baselines' performances are quoted from Guo et al. (2022). Table 1 and Table 2 show the quantitative comparison on the HumanML3D dataset and the KIT-ML dataset. Our proposed MotionDiffuse outperforms all existing works with a remarkable margin in aspects of precision, FID, MultiDodal Distance, and Diversity. The precision of MotionDiffuse is even close to that of real motions, which suggests that our generated motion sequences are satisfyingly high-fidelity and realistic.

Guo et al. (2022) states that the results on the MultiModality metric should be larger whenever possible. However, the literature in action-conditioned motion generation task (Guo et al., 2020; Petrovich et al., 2021; Cervantes et al., 2022) argue that this metric should be close to the real motion. In the T2M task, it is difficult to calculate this metric of real motions. Therefore, we only report these results without comparison.

Table 3 Ablation of the pretrained CLIP and the efficient attention technique. All results are reported on the KIT-ML test set.

CLIP	EFF	R Precision↑		
		Top 1	Top 2	Top 3
N	N	$0.288^{\pm.004}$	$0.440^{\pm.004}$	$0.539^{\pm.004}$
N	Y	$0.136^{\pm.003}$	$0.233^{\pm.003}$	$0.309^{\pm.003}$
Y	N	$0.357^{\pm.004}$	$0.555^{\pm.004}$	$0.679^{\pm.005}$
Y	Y	$\mathbf{0.417^{\pm.004}}$	$\mathbf{0.621^{\pm.004}}$	$\mathbf{0.739^{\pm.004}}$

To further understand the function of CLIP initialization and efficient attention, we report ablation results in Table 3. The models without pretrained CLIP suffer from severe performance drops, which indicates the necessity of a pretrained language model for the T2M task. As for efficient attention, it is significantly beneficial when we use CLIP simultaneously. However, this module also limits the model's performance when without CLIP. A possible explanation for this phenomenon is that the global relation in efficient attention is misleading when the semantic information from given text is insufficient.

We explore how the size of architecture influences the performance. Table 4 suggests that the latent dimension plays a more important role. The models with 512 latent dimension significantly outperform the models with 256 latent dimension. On the contrary, the increase of the number of layers improves the performance when the latent dimension is either 128 or 256, but has little effect when the dimension is 512.

Qualitative Results Figure 4 shows a comparison between our method and Guo et al. (2022) as a base-

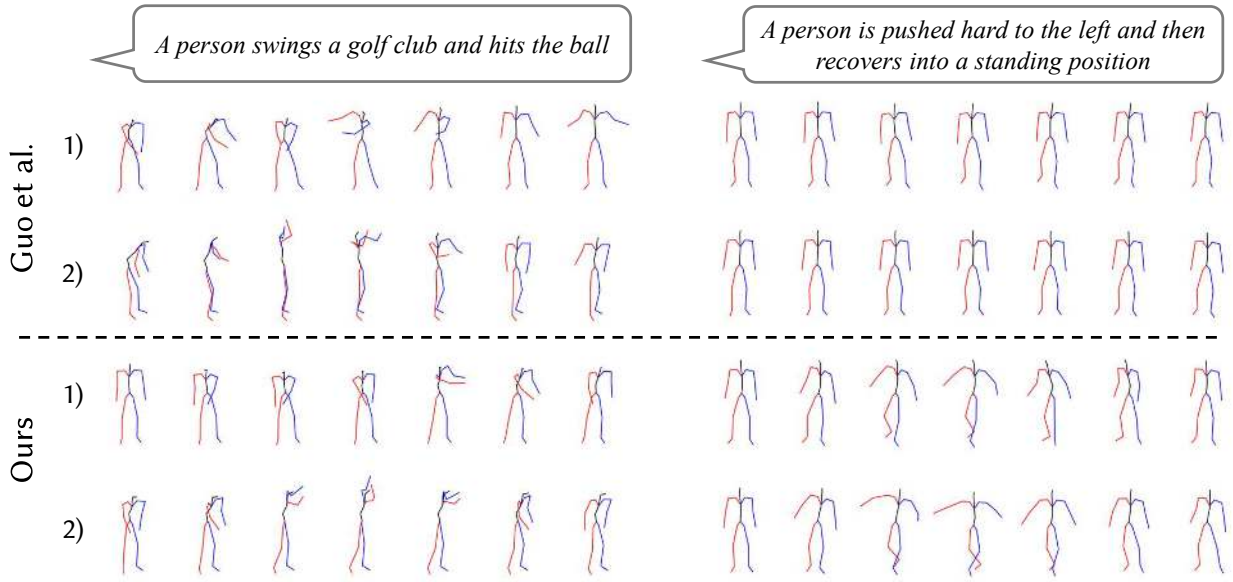

Fig. 4 Qualitative results on the HumanML3D dataset. We compare our method with Guo et al. (2022) and visualize two examples for each given prompt. MotionDiffuse is able to achieve both accuracy and diversity.

Table 4 Ablation of the latent dimension and the number of transformer layers. All results are reported on the KIT-ML test set.

#layers	Dim	R Precision↑		
		Top 1	Top 2	Top 3
4	128	$0.033^{\pm.002}$	$0.066^{\pm.003}$	$0.097^{\pm.003}$
4	256	$0.095^{\pm.002}$	$0.166^{\pm.003}$	$0.227^{\pm.003}$
4	512	$0.405^{\pm.005}$	$0.620^{\pm.005}$	$\mathbf{0.743}^{\pm.004}$
8	128	$0.025^{\pm.002}$	$0.053^{\pm.002}$	$0.086^{\pm.002}$
8	256	$0.198^{\pm.003}$	$0.335^{\pm.004}$	$0.441^{\pm.004}$
8	512	$\mathbf{0.417}^{\pm.004}$	$\mathbf{0.621}^{\pm.004}$	$0.739^{\pm.004}$
12	128	$0.031^{\pm.002}$	$0.063^{\pm.003}$	$0.091^{\pm.002}$
12	256	$0.209^{\pm.003}$	$0.348^{\pm.004}$	$0.452^{\pm.003}$
12	512	$0.412^{\pm.006}$	$0.616^{\pm.004}$	$0.741^{\pm.004}$

line. We highlight that MotionDiffuse achieves a balance between diversity and realness. For example, for prompt 'A person swings a golf club and hits the ball', our generated motions portraits the described motion more faithfully. In contrast, the baseline method has high multi-modality at the expenses of accuracy. In addition, given a complicated prompt such as "A person is pushed hard to the left and then recovers into a standing position", MotionDiffuse is able to generate high-quality motions that reflects the detailed description whereas the baseline method fails to produce any meaningful movement.

4.2 Action-conditioned Motion Generation

Datasets. HumanAct12 dataset (Guo et al., 2020) provides 12 kinds of motion sequences. This dataset is adapted from PHSPD dataset (Zou et al., 2020), which contains 1191 videos. HumanAct12 further arranges these videos into trimmed motion clips. UESTC dataset (Ji et al., 2018) is also a significant benchmark for action-conditioned motion generation tasks, which includes 25K motion sequences across 40 different action categories. Petrovich et al. (2021) further uses pre-trained VIBE (Kocabas et al., 2020) to extract SMPL (Loper et al., 2015) sequences from the UESTC dataset and provides pretrained action recognition model for evaluation.

Evaluation Metrics. Four evaluation metrics are applied for this task: FID, Accuracy, Diversity, and Multimodality. The pretrained action recognition module can directly calculate the average accuracy for all action categories without arranging mini-batches. This metric has a similar function to R Precision. The other three metrics have been introduced in Section 4.1. HumanAct12 has no official split, and we report the FID on the whole dataset. UESTC has a test split, so we report the FID on it, which is more representative than the train split. In this section, FID and Accuracy are two principal metrics. Our conclusion are mainly based on them.

Table 5 Quantitative results for Action-conditioned Motion Generation. As for UESTC dataset, we report FID on the test split. MM: MultiModality.

Methods	HumanAct12				UESTC			
	FID↓	Accuracy↑	Diversity→	MM→	FID↓	Accuracy↑	Diversity→	MM→
Real motions	$0.020^{\pm.010}$	$0.997^{\pm.001}$	$6.850^{\pm.050}$	$2.450^{\pm.040}$	$2.79^{\pm.29}$	$0.988^{\pm.001}$	$33.34^{\pm.320}$	$14.16^{\pm.06}$
Action2Motion	$0.338^{\pm.015}$	$0.917^{\pm.003}$	$6.879^{\pm.066}$	$2.511^{\pm.023}$	-	-	-	-
ACTOR	$0.12^{\pm.00}$	$0.955^{\pm.008}$	$6.84^{\pm.03}$	$2.53^{\pm.02}$	$23.43^{\pm2.20}$	$0.911^{\pm.003}$	$31.96^{\pm.33}$	$\mathbf{14.52^{\pm.09}}$
INR	$0.088^{\pm.004}$	$0.973^{\pm.001}$	$6.881^{\pm.048}$	$2.569^{\pm.040}$	$15.00^{\pm.09}$	$0.941^{\pm.001}$	$31.59^{\pm.19}$	$14.68^{\pm.07}$
Ours	$\mathbf{0.07^{\pm.00}}$	$\mathbf{0.992^{\pm.13}}$	$\mathbf{6.85^{\pm.02}}$	$\mathbf{2.46^{\pm.02}}$	$\mathbf{9.10^{\pm.437}}$	$\mathbf{0.950^{\pm.000}}$	$\mathbf{32.42^{\pm.214}}$	$14.74^{\pm.07}$

Implementation Details. All the setting are the same to those for text-driven motion generation tasks except for the learning rate, the number of iterations and the motion representation. In this series of expreiments, we train 100K iterations for the HumanAct12 dataset and 500K for the UESTC dataset, both with a 0.0001 learning rate.

Motion represention in this task is slightly different from the T2M task. As for the HumanAct12 dataset, each pose state can be representated as $(\mathbf{j}^x, \mathbf{j}^y, \mathbf{j}^z)$, where $\mathbf{j}^x, \mathbf{j}^y, \mathbf{j}^z \in \mathbb{R}^{24\times 3}$ are the coordinates of 24 joints. We use $(r^x, r^y, r^z, \mathbf{j}^r)$ as the pose representation for the UESTC dataset, where $r^x, r^y, r^z \in \mathbb{R}$ are the coordinates of the root joint, and $\mathbf{j}^r \in \mathbb{R}^{24\times 6}$ is the rotation angle of each joint in 6D representation.

Quantitative Results. Following Cervantes et al. (2022), three baseline models are selected: Action2Motion (Guo et al., 2020), ACTOR (Petrovich et al., 2021), INR (Cervantes et al., 2022). Table 5 shows the quantitative results on the HumanAct12 dataset and the UESTC datasets. Our proposed MotionDiffuse achieves the best performance in aspects of FID and Accuracy when compared to other existing works. We want to highlight that our results of the HumanAct12 dataset are notably close to real motions on all four metrics.

Table 6 Ablation of the latent dimension and the number of transformer layers. All results are reported on the HumanAct12 dataset.

#layers	Dim	FID↓	Accuracy↑
4	128	$0.29^{\pm0.00}$	$0.892^{\pm1.97}$
4	256	$0.14^{\pm0.00}$	$0.958^{\pm0.51}$
4	512	$0.09^{\pm0.00}$	$0.984^{\pm0.21}$
8	128	$0.22^{\pm0.00}$	$0.929^{\pm1.04}$
8	256	$0.09^{\pm0.00}$	$0.983^{\pm0.23}$
8	512	$\mathbf{0.07^{\pm0.00}}$	$0.992^{\pm0.13}$
12	128	$0.11^{\pm0.00}$	$0.954^{\pm0.67}$
12	256	$0.10^{\pm0.00}$	$0.988^{\pm0.21}$
12	512	$0.08^{\pm0.00}$	$\mathbf{0.996^{\pm0.08}}$

Here we also try different combination of latent dimension and the number of layers, as shown in Table 6. Similar to the conclusions we found in Section 4.1, latent dimension is more important than the number of layers.

4.3 Motion Manipulation

To better evaluate the capability of text-driven motion generation models, we design two task variants. First, *Spatially-diverse T2M task (T2M-S)*. T2M-S requires the generated motion sequence to contain multiple actions on different body parts (*e.g.* 'a person is running and drinking water simultaneously'). Specifically, i-th test sample in T2M-S task can be represented by a set of text-mask pairs $\{(\text{text}_{i,j}, \text{M}_{i,j})\}$, where $\text{M}_{i,j} \in \{0,1\}^D$ is a D-dimension binary vector. It indicates which body part we should focus on when given the text description $\text{text}_{i,j}$. Second, *Temporally-diverse T2M task (T2M-T)*. T2M-T expects models to generate a long motion sequence, which includes multiple actions in a specific order spanning over different time intervals (*e.g.* 'a person is walking and then running'). The i-th test sample is an array of text-duration pairs $\{\text{text}_{i,j}, [l_{i,j}, r_{i,j}]\}, l_{i,j} < r_{i,j}$. It means that the motion clip from $l_{i,j}$-th frame to $r_{i,j}$ frame is supposed to contain the action $\text{text}_{i,j}$.

Implementation Details. We train our proposed MotionDiffuse on the BABEL dataset (Punnakkal et al., 2021) with 50K iterations. Each pose state is represented by $(r^x, r^y, r^z, \mathbf{j}^r)$, which is same to the setting for the UESTC dataset. Other settings remain unchanged. $\lambda_1 = \lambda_2 = 0.01$ are used for the visualization.

Qualitative Results. As shown in Fig 5, MotionDiffuse has the capability to handle highly comprehensive prompts that assign motions to multiple body parts (such as "Kicking and punching" and "Jumping and raising arms" that require coordination of the upper and lower body). Moreover, MotionDiffuse is able to generate long sequences according to a complex instruction that includes multiple actions (such as "Tying the shoe, standing up and then walking forward" that includes a series of vastly different motions).

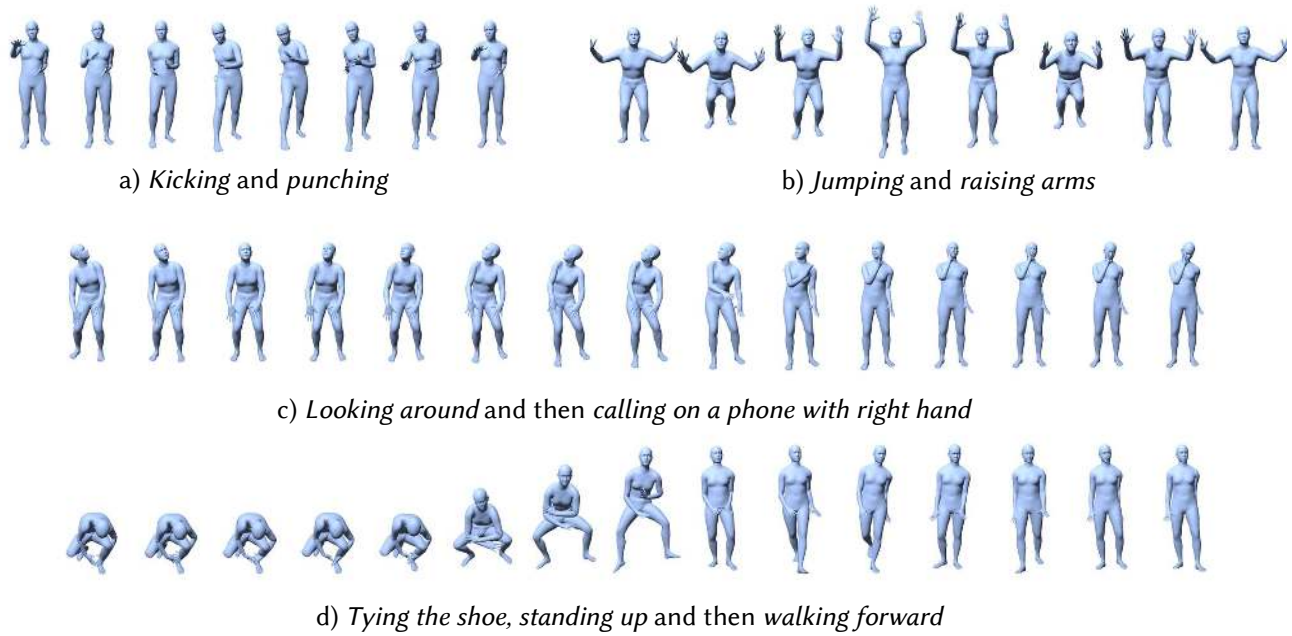

Fig. 5 Qualitative results on the BABEL dataset. MotionDiffuse is able to generate dynamic sequences according to complicated prompt that involves multiple body parts or actions.

5 Conclusion, Limitations and Future Work

We propose MotionDiffuse, the first diffusion model-based method for text-driven motion generation. MotionDiffuse demonstrates three major strengths: Probabilistic Mapping that enhances diversity, Realistic Synthesis that ensures plausibility of motion sequences, and Multi-Level Manipulation that allows for per-part manipulation and long sequence generation. Both quantitative and qualitative evaluations show that MotionDiffuse outperforms existing arts on various tasks such as text-driven motion generation and action-conditioned motion generation, and demonstrates remarkable motion manipulation capabilities.

Although MotionDiffuse has pushed forward the performance boundary of motion generation tasks, there still exist some problems. First, diffusion models require a large amount of diffusion steps during inference and it is challenging to generate motion sequences in real-time. Second, current pipeline only accepts a single form of motion representation. A more generalized pipeline that adapts concurrently to all datasets would be more versatile for various scenarios.

Acknowledgements This work is supported by NTU NAP, MOE AcRF Tier 2 (T2EP20221-0033), and under the RIE2020 Industry Alignment Fund - Industry Collaboration Projects (IAF-ICP) Funding Initiative, as well as cash and in-kind contribution from the industry partner(s). Corresponding author: Ziwei Liu (ziwei.liu@ntu.edu.sg).

References

Ahuja C, Morency LP (2019) Language2pose: Natural language grounded pose forecasting. In: 2019 International Conference on 3D Vision (3DV), IEEE, pp 719–728

Aliakbarian S, Saleh FS, Salzmann M, Petersson L, Gould S (2020) A stochastic conditioning scheme for diverse human motion prediction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 5223–5232

Andriluka M, Iqbal U, Insafutdinov E, Pishchulin L, Milan A, Gall J, Schiele B (2018) Posetrack: A benchmark for human pose estimation and tracking. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 5167–5176

Barsoum E, Kender J, Liu Z (2018) Hp-gan: Probabilistic 3d human motion prediction via gan. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops, pp 1418–1427

Bhattacharya U, Rewkowski N, Banerjee A, Guhan P, Bera A, Manocha D (2021) Text2gestures: A transformer-based network for generating emotive body gestures for virtual agents. In: 2021 IEEE Virtual Reality and 3D User Interfaces (VR), IEEE, pp 1–10

Cai Z, Zhang M, Ren J, Wei C, Ren D, Li J, Lin Z, Zhao H, Yi S, Yang L, et al. (2021) Playing for 3d human recovery. arXiv preprint arXiv:211007588

Cai Z, Ren D, Zeng A, Lin Z, Yu T, Wang W, Fan X, Gao Y, Yu Y, Pan L, et al. (2022) Humman: Multi-modal 4d human dataset for versatile sensing and modeling. arXiv preprint arXiv:220413686

Cao Z, Gao H, Mangalam K, Cai QZ, Vo M, Malik J (2020) Long-term human motion prediction with scene context. In: European Conference on Computer Vision, Springer, pp 387–404

Carreira J, Noland E, Hillier C, Zisserman A (2019) A short note on the kinetics-700 human action dataset. arXiv preprint arXiv:190706987

Cervantes P, Sekikawa Y, Sato I, Shinoda K (2022) Implicit neural representations for variable length human motion generation. arXiv preprint arXiv:220313694

Chung J, Wuu Ch, Yang Hr, Tai YW, Tang CK (2021) Haa500: Human-centric atomic action dataset with curated videos. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 13465–13474

Dhariwal P, Nichol A (2021) Diffusion models beat gans on image synthesis. Advances in Neural Information Processing Systems 34

Dinh L, Krueger D, Bengio Y (2014) Nice: Non-linear independent components estimation. arXiv preprint arXiv:14108516

Dinh L, Sohl-Dickstein J, Bengio S (2016) Density estimation using real nvp. arXiv preprint arXiv:160508803

Futrelle R, Speckert G (1978) Extraction of motion data by interactive processing. In: Proc. 1978 IEEE Conf. Pattern Recog. and Image Processing, Chicago, IL, pp 405–408

Gavrila DM (1999) The visual analysis of human movement: A survey. Computer vision and image understanding 73(1):82–98

Ghosh A, Cheema N, Oguz C, Theobalt C, Slusallek P (2021) Synthesis of compositional animations from textual descriptions. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 1396–1406

Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y (2014) Generative adversarial nets. Advances in neural information processing systems 27

Gu C, Sun C, Ross DA, Vondrick C, Pantofaru C, Li Y, Vijayanarasimhan S, Toderici G, Ricco S, Sukthankar R, et al. (2018) Ava: A video dataset of spatio-temporally localized atomic visual actions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 6047–6056

Guo C, Zuo X, Wang S, Zou S, Sun Q, Deng A, Gong M, Cheng L (2020) Action2motion: Conditioned generation of 3d human motions. In: Proceedings of the 28th ACM International Conference on Multimedia, pp 2021–2029

Guo C, Zou S, Zuo X, Wang S, Ji W, Li X, Cheng L (2022) Generating diverse and natural 3d human motions from text. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 5152–5161

Harvey FG, Yurick M, Nowrouzezahrai D, Pal C (2020) Robust motion in-betweening. ACM Transactions on Graphics (TOG) 39(4):60–1

Hendrycks D, Gimpel K (2016) Gaussian error linear units (gelus). arXiv preprint arXiv:160608415

Henter GE, Alexanderson S, Beskow J (2020) Moglow: Probabilistic and controllable motion synthesis using normalising flows. ACM Transactions on Graphics (TOG) 39(6):1–14

Ho J, Jain A, Abbeel P (2020) Denoising diffusion probabilistic models. Advances in Neural Information Processing Systems 33:6840–6851

Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural computation 9(8):1735–1780

Hong F, Zhang M, Pan L, Cai Z, Yang L, Liu Z (2022) Avatarclip: Zero-shot text-driven generation and animation of 3d avatars. arXiv preprint arXiv:220508535

Huang R, Hu H, Wu W, Sawada K, Zhang M, Jiang D (2020) Dance revolution: Long-term dance generation with music via curriculum learning. arXiv preprint arXiv:200606119

Ikemoto L, Arikan O, Forsyth D (2009) Generalizing motion edits with gaussian processes. ACM Transactions on Graphics (TOG) 28(1):1–12

Ionescu C, Papava D, Olaru V, Sminchisescu C (2013) Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. IEEE transactions on pattern analysis and machine intelligence 36(7):1325–1339

Jain A, Tancik M, Abbeel P (2021) Putting nerf on a diet: Semantically consistent few-shot view synthesis. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 5885–5894

Jhuang H, Gall J, Zuffi S, Schmid C, Black MJ (2013) Towards understanding action recognition. In: Proceedings of the IEEE international conference on computer vision, pp 3192–3199

Ji Y, Xu F, Yang Y, Shen F, Shen HT, Zheng WS (2018) A large-scale rgb-d database for arbitrary-view human action recognition. In: Proceedings of the 26th ACM international Conference on Multimedia, pp 1510–1518

Joo H, Liu H, Tan L, Gui L, Nabbe B, Matthews I, Kanade T, Nobuhara S, Sheikh Y (2015) Panoptic studio: A massively multiview system for social mo-

tion capture. In: Proceedings of the IEEE International Conference on Computer Vision, pp 3334–3342

Karpathy A, Toderici G, Shetty S, Leung T, Sukthankar R, Fei-Fei L (2014) Large-scale video classification with convolutional neural networks. In: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, pp 1725–1732

Kingma DP, Dhariwal P (2018) Glow: Generative flow with invertible 1x1 convolutions. Advances in neural information processing systems 31

Kingma DP, Welling M (2013) Auto-encoding variational bayes. arXiv preprint arXiv:13126114

Kocabas M, Athanasiou N, Black MJ (2020) Vibe: Video inference for human body pose and shape estimation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 5253–5263

Kuehne H, Jhuang H, Garrote E, Poggio T, Serre T (2011) Hmdb: a large video database for human motion recognition. In: 2011 International conference on computer vision, IEEE, pp 2556–2563

Lee HY, Yang X, Liu MY, Wang TC, Lu YD, Yang MH, Kautz J (2019) Dancing to music. Advances in Neural Information Processing Systems 32

Li J, Yin Y, Chu H, Zhou Y, Wang T, Fidler S, Li H (2020) Learning to generate diverse dance motions with transformer. arXiv preprint arXiv:200808171

Li R, Yang S, Ross DA, Kanazawa A (2021) Ai choreographer: Music conditioned 3d dance generation with aist++. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 13401–13412

Lin AS, Wu L, Corona R, Tai K, Huang Q, Mooney RJ (2018) Generating animated videos of human activities from natural language descriptions. In: NeurIPS Workshop

Liu J, Shahroudy A, Perez M, Wang G, Duan LY, Kot AC (2019) Ntu rgb+ d 120: A large-scale benchmark for 3d human activity understanding. IEEE transactions on pattern analysis and machine intelligence 42(10):2684–2701

Loper M, Mahmood N, Romero J, Pons-Moll G, Black MJ (2015) Smpl: A skinned multi-person linear model. ACM transactions on graphics (TOG) 34(6):1–16

Mahmood N, Ghorbani N, Troje NF, Pons-Moll G, Black MJ (2019) Amass: Archive of motion capture as surface shapes. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 5442–5451

Mehta D, Rhodin H, Casas D, Fua P, Sotnychenko O, Xu W, Theobalt C (2017) Monocular 3d human pose estimation in the wild using improved cnn supervision. In: 2017 international conference on 3D vision (3DV), IEEE, pp 506–516

Mildenhall B, Srinivasan PP, Tancik M, Barron JT, Ramamoorthi R, Ng R (2020) Nerf: Representing scenes as neural radiance fields for view synthesis. In: European conference on computer vision, Springer, pp 405–421

Min J, Chai J (2012) Motion graphs++ a compact generative model for semantic motion analysis and synthesis. ACM Transactions on Graphics (TOG) 31(6):1–12

Monfort M, Andonian A, Zhou B, Ramakrishnan K, Bargal SA, Yan T, Brown L, Fan Q, Gutfreund D, Vondrick C, et al. (2019) Moments in time dataset: one million videos for event understanding. IEEE transactions on pattern analysis and machine intelligence 42(2):502–508

Mukai T, Kuriyama S (2005) Geostatistical motion interpolation. In: ACM SIGGRAPH 2005 Papers, pp 1062–1070

Nichol A, Dhariwal P, Ramesh A, Shyam P, Mishkin P, McGrew B, Sutskever I, Chen M (2021) Glide: Towards photorealistic image generation and editing with text-guided diffusion models. arXiv preprint arXiv:211210741

Nichol AQ, Dhariwal P (2021) Improved denoising diffusion probabilistic models. In: International Conference on Machine Learning, PMLR, pp 8162–8171

Ormoneit D, Black MJ, Hastie T, Kjellström H (2005) Representing cyclic human motion using functional analysis. Image and Vision Computing 23(14):1264–1276

O'rourke J, Badler NI (1980) Model-based image analysis of human motion using constraint propagation. IEEE Transactions on Pattern Analysis and Machine Intelligence (6):522–536

Patel P, Huang CHP, Tesch J, Hoffmann DT, Tripathi S, Black MJ (2021) AGORA: Avatars in geography optimized for regression analysis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 13468–13478

Petrovich M, Black MJ, Varol G (2021) Action-conditioned 3d human motion synthesis with transformer vae. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 10985–10995

Petrovich M, Black MJ, Varol G (2022) Temos: Generating diverse human motions from textual descriptions. arXiv preprint arXiv:220414109

Plappert M, Mandery C, Asfour T (2016) The kit motion-language dataset. Big data 4(4):236–252

Punnakkal AR, Chandrasekaran A, Athanasiou N, Quiros-Ramirez A, Black MJ (2021) Babel: Bodies,

action and behavior with english labels. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 722–731

Radford A, Kim JW, Hallacy C, Ramesh A, Goh G, Agarwal S, Sastry G, Askell A, Mishkin P, Clark J, et al. (2021) Learning transferable visual models from natural language supervision. arXiv preprint arXiv:210300020

Ramesh A, Dhariwal P, Nichol A, Chu C, Chen M (2022) Hierarchical text-conditional image generation with clip latents. arXiv preprint arXiv:220406125

Rose C, Cohen MF, Bodenheimer B (1998) Verbs and adverbs: Multidimensional motion interpolation. IEEE Computer Graphics and Applications 18(5):32–40

Shao D, Zhao Y, Dai B, Lin D (2020) Finegym: A hierarchical video dataset for fine-grained action understanding. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 2616–2625

Shen Z, Zhang M, Zhao H, Yi S, Li H (2021) Efficient attention: Attention with linear complexities. In: Proceedings of the IEEE/CVF winter conference on applications of computer vision, pp 3531–3539

Siyao L, Yu W, Gu T, Lin C, Wang Q, Qian C, Loy CC, Liu Z (2022) Bailando: 3d dance generation by actor-critic gpt with choreographic memory. arXiv preprint arXiv:220313055

Soomro K, Zamir AR, Shah M (2012) Ucf101: A dataset of 101 human actions classes from videos in the wild. arXiv preprint arXiv:12120402

Sun G, Wong Y, Cheng Z, Kankanhalli MS, Geng W, Li X (2020) Deepdance: music-to-dance motion choreography with adversarial learning. IEEE Transactions on Multimedia 23:497–509

Tevet G, Gordon B, Hertz A, Bermano AH, Cohen-Or D (2022) Motionclip: Exposing human motion generation to clip space. arXiv preprint arXiv:220308063

Trumble M, Gilbert A, Malleson C, Hilton A, Collomosse JP (2017) Total capture: 3d human pose estimation fusing video and inertial sensors. In: BMVC, vol 2, pp 1–13

Tulyakov S, Liu MY, Yang X, Kautz J (2018) Mocogan: Decomposing motion and content for video generation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1526–1535

Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I (2017) Attention is all you need. Advances in neural information processing systems 30

Wang Z, Yu P, Zhao Y, Zhang R, Zhou Y, Yuan J, Chen C (2020) Learning diverse stochastic human-action generators by learning smooth latent transitions. In: Proceedings of the AAAI conference on artificial intelligence, vol 34, pp 12281–12288

Yan X, Rastogi A, Villegas R, Sunkavalli K, Shechtman E, Hadap S, Yumer E, Lee H (2018) Mt-vae: Learning motion transformations to generate multimodal human dynamics. In: Proceedings of the European conference on computer vision (ECCV), pp 265–281

Yu Z, Yoon JS, Lee I, Venkatesh P, Park J, Yu J, Park H (2020) Humbi: A large multiview dataset of human body expressions. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) pp 2987–2997

Zhang M, Yu L, Zhang K, Du B, Zhan B, Chen S, Jiang X, Guo S, Zhao J, Wang Y, et al. (2020) Kinematic dataset of actors expressing emotions. Scientific data 7(1):1–8

Zhao H, Torralba A, Torresani L, Yan Z (2019) Hacs: Human action clips and segments dataset for recognition and temporal localization. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 8668–8678

Zhou Y, Barnes C, Lu J, Yang J, Li H (2019) On the continuity of rotation representations in neural networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 5745–5753

Zhuang W, Wang C, Xia S, Chai J, Wang Y (2020) Music2dance: Dancenet for music-driven dance generation. arXiv preprint arXiv:200203761

Zou S, Zuo X, Qian Y, Wang S, Xu C, Gong M, Cheng L (2020) 3d human shape reconstruction from a polarization image. In: European Conference on Computer Vision, Springer, pp 351–368