FLAME: Free-Form Language-Based Motion Synthesis & Editing

Jihoon Kim[1,2], **Jiseob Kim**[2], **Sungjoon Choi**[1]

[1] Korea University
[2] Kakao Brain
jihoon-kim@korea.ac.kr, jiseob.kim@kakaobrain.com, sungjoon-choi@korea.ac.kr

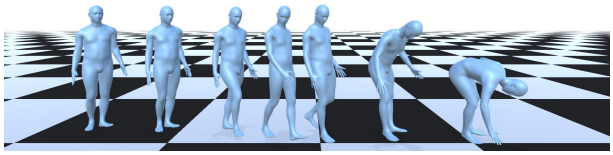
Abstract

Text-based motion generation models are drawing a surge of interest for their potential for automating the motion-making process in the game, animation, or robot industries. In this paper, we propose a diffusion-based motion synthesis and editing model named FLAME. Inspired by the recent successes in diffusion models, we integrate diffusion-based generative models into the motion domain. FLAME can generate high-fidelity motions well aligned with the given text. Also, it can edit the parts of the motion, both frame-wise and joint-wise, without any fine-tuning. FLAME involves a new transformer-based architecture we devise to better handle motion data, which is found to be crucial to manage variable-length motions and well attend to free-form text. In experiments, we show that FLAME achieves state-of-the-art generation performances on three text-motion datasets: HumanML3D, BABEL, and KIT. We also demonstrate that FLAME's editing capability can be extended to other tasks such as motion prediction or motion in-betweening, which have been previously covered by dedicated models.

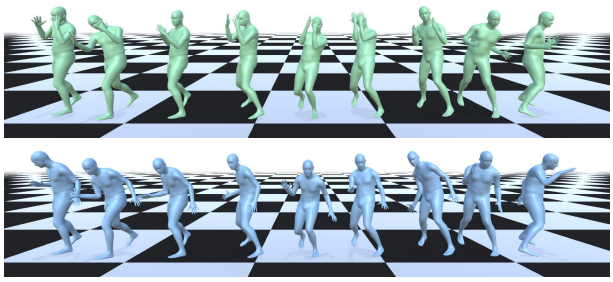
Introduction

Given the difficulty of 3D motion generation, automating text-based motion synthesis and editing has been considered a difficult problem despite their usefulness in industries. The process of text-to-motion synthesis and text-based motion editing motion should not only deliver the clear intention of motion but also be able to convey the naturalnesss of human motion at the same time. This effort can be reduced considerably if one can generate motion through language or edit existing motion to desirable motion simply by language. In this study, we present a method that can perform motion synthesis and editing using free-form texts.

Recently, research on generating motion from language has been actively conducted. Many previous studies (Guo et al. 2020; Petrovich, Black, and Varol 2021; Song et al. 2022) have explored methods to synthesize motion from behavioral labels, such as 'walking', 'jumping', or 'dancing' and demonstrated promising results on text-to-motion synthesis tasks. However, synthesizing motion from behavioral labels lacks descriptive power, which limits both diversity and controllability in motion synthesis. As large-scale pre-trained language models (PLMs) advance, there are studies (Ghosh et al. 2021; Petrovich, Black, and Varol 2022) take the advantage of using PLMs to generate motion from free-form texts, overcoming the limited expressiveness of simple labels. Although these methods present promising results in synthesizing motion from free-form texts, they lack capability in a flexible conditional generation.



(a) Text-to-motion synthesis result from FLAME with prompt: *"A person walks forward and bends down to pick up something."*



(b) (Green) Reference motion. (Blue) Text-based motion editing result from FLAME with prompt: *"A person dribbles a ball."*; The editing model is allowed to edit upper body parts while fixing lower body parts in this example.

Figure 1: Overview of text-to-motion synthesis and text-based motion editing. Motion flows from left to right.

In this paper, we introduce a versatile motion synthesis method that can generate motion well-aligned with the provided prompts and perform editing of a reference motion from textual descriptions. As we aim to present a method that can generate and edit motion, we employ the diffusion model (Ho, Jain, and Abbeel 2020; Dhariwal and Nichol 2021; Nichol et al. 2021), which has been presenting many successful results in image generation and in-

painting (Nichol et al. 2021; Ramesh et al. 2022) these days. With this, our proposed method can conduct text-to-motion synthesis and text conditional motion generation, including editing, forecasting, and in-betweening without any fine-tuning or modification of a trained model.

Architectures for diffusion models have been actively studied (Ho, Jain, and Abbeel 2020; Nichol and Dhariwal 2021; Dhariwal and Nichol 2021) in the image domain, but have not yet been explored in the motion domain. In order to introduce the diffusion model into the motion domain, we focus on the following differences between motion and image. First, unlike the image, which has spatial information without temporal information, motion is inherently spatio-temporal data. Secondly, the length of motion can vary from short motion to long-horizon motion, which requires a model that can handle arbitrary length. To this end, we introduce a transformer decoder-based architecture that can handle temporal aspects and variable lengths. Our proposed method takes diffusion time-step token, motion length token, language tokens, and motion tokens as inputs. Additional language-side information is extracted from PLM and fed to the transformer using cross-attention. Model is trained to learn the denoising process, which gradually reconstructs motion from isotropic Gaussian noise. To sample motion from free-form text, we use classifier-free guidance (Ho and Salimans 2021). We refer to our model as **FLAME**, which stands for **F**ree-form **LA**nguage-based **M**otion Synthesis and **E**diting. To the best of our knowledge, **FLAME** is the first to adopt a diffusion-based generative framework for synthesizing and editing motion.

Our main contributions are summarized as follows:

- We propose FLAME, a unified model for motion synthesis and editing with free-form language description.
- Our model is the first attempt applying diffusion models to motion data; to handle the temporal nature of motion and variable-length, we devise a new architecture.
- We show FLAME can generate more diverse motions corresponding to the same text.
- We demonstrate FLAME can solve other classical tasks—prediction and in-betweening—through editing, without any fine-tuning.

Related Work

Diffusion Models & Text-conditional Generation

Diffusion models (Ho, Jain, and Abbeel 2020) are recently proposed generative models that are shown to be good at synthesizing highly-complex image datasets. Compared to GANs (Goodfellow et al. 2014; Karras, Laine, and Aila 2019; Karras et al. 2020; Sauer, Schwarz, and Geiger 2022) and VAEs (Kingma and Welling 2013; Van Den Oord, Vinyals et al. 2017), they have been presenting improved quality in generating multi-modal outputs, advantageous to text-to-image generation and text-to-motion generation of our interest—there can be various modes of images/motions corresponding to a single text description.

Diffusion models are originally proposed in Sohl-Dickstein et al. (2015) and developed in Ho, Jain, and Abbeel (2020) and Song, Meng, and Ermon (2020), showing high-quality image generation. After, they are extended to work on conditional generation settings, demonstrating even better performances. Class-conditional models are studied in Dhariwal and Nichol (2021), and text-conditional models are proposed by adapting the conditioning scheme for text (GLIDE; (Nichol et al. 2021)). unCLIP[1] (Ramesh et al. 2022) and Imagen (Saharia et al. 2022) further show that conditioning on pre-trained high-level embedding gives improved result. Our model shares some similarity with the GLIDE model, but has crucial differences in that it has a new design to handle temporal sequences of variable length.

To achieve the best performance, several techniques have been proposed and applied to the aforementioned models. Improved DDPM (Nichol and Dhariwal 2021) suggests learning the reverse-diffusion variances. Classifier-free guidance (Ho and Salimans 2021) has been introduced to enable conditional generation without the need for a separate classifier model. We employ both of these techniques in the proposed model.

3D Human Motion Generation

Motion prediction is a task to forecast subsequent frames from a given frame or multiple frames, and motion in-betweening is a boundary value problem that generates a natural motion sequence while satisfying the given starting and target poses. Motion prediction models (Fragkiadaki et al. 2015; Guo and Choi 2019) and in-betweening models (Harvey et al. 2020; Kim et al. 2022a; Duan et al. 2022) have been developed, but the models lack the ability to perform multiple tasks with a single model and cannot synthesize motion from textual descriptions.

In the text-to-motion domain, early models (Lin et al. 2018; Ahn et al. 2018) approach the text-to-motion synthesis task with the sequence-to-sequence model. After that, Guo et al. (2020) and Petrovich, Black, and Varol (2021) introduce a variational autoencoder (VAE) to create motion from behavioral labels to improve motion quality and produce a diversified range of motions. Recent models (Ghosh et al. 2021; Petrovich, Black, and Varol 2022) advance the text-to-motion task by composing 3D human motion from free-form texts, instead of simple action labels, to cover more expressive human motions. They demonstrate free-form language-based motion generation by taking advantage of pre-trained language model. However, these models have limitations in extensibility to conventional motion tasks or text-based motion editing.

In this study, we propose a model to perform high-quality *text-to-motion synthesis* with flexible editing capability, which can conduct *text-based motion editing* including traditional motion prediction and motion in-betweening without any fine-tuning or modification on a trained generative model.

Proposed Method: FLAME

We first review the diffusion-based modeling scheme. Then, we explain the model architecture, designed to handle mo-

[1] branded as DALL-E 2 to the public

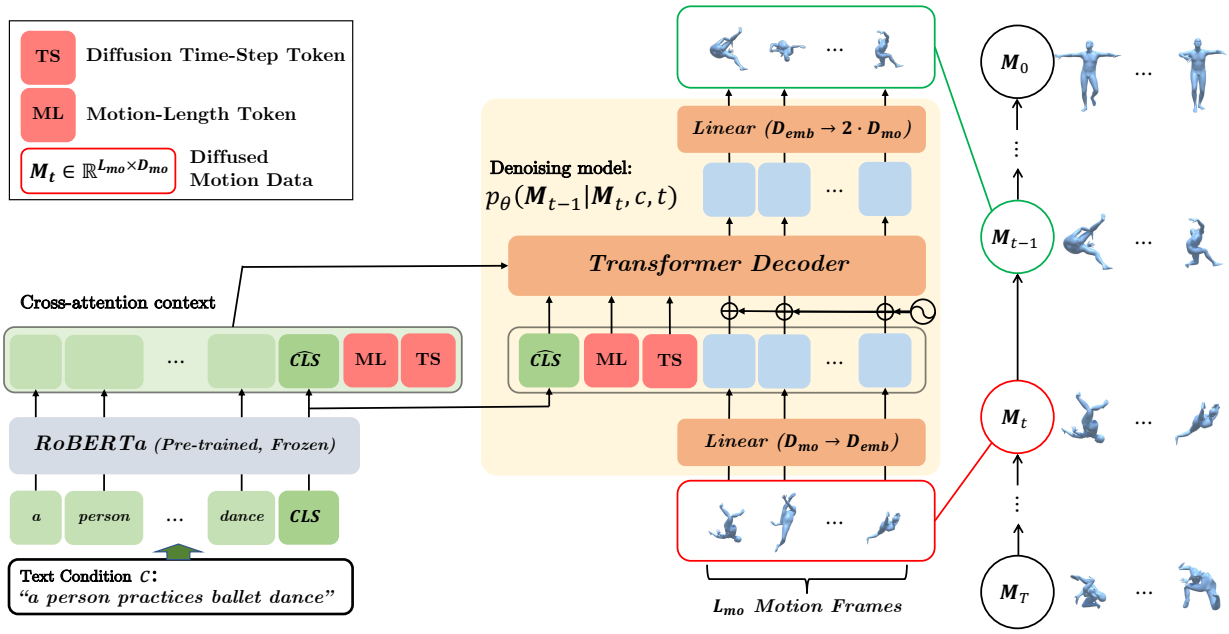


Figure 2: Overview of the architecture: FLAME learns the denoising process p_θ from M_t to M_{t-1} at diffusion time-step t. Input motion is projected and concatenated with language pooler token (CLS), motion-length token (ML), and diffusion time-step token (TS) as input tokens for the transformer decoder. Additional language-side information is fed from a pre-trained frozen language encoder as a cross-attention context. FLAME outputs a $2 \cdot D_{mo}$-dimensional sequence of vectors as it predicts both the mean and variance of noise at each diffusion time-steps.

tion data. In the last two subsections, we explain how to do the inference in the synthesis and the editing scenarios using the trained FLAME model.

Diffusion-Based Model

The generative modeling scheme of FLAME is inspired by the denoising diffusion probabilistic model (DDPM) (Ho, Jain, and Abbeel 2020) and its extension (Nichol and Dhariwal 2021). The general idea of DDPM is to design a diffusion process that gradually adds small amounts of noise to the data and train the model to reverse each of these diffusion steps. The diffusion process eventually converts the data into isotropic Gaussian noise, and thus the fully-trained model would generate samples by repeating denoising steps, starting from pure noise. Essentially, this scheme divides a complex distribution-modeling problem into a set of simple denoising problems.

In details, DDPM defines the diffusion process with the following conditional distribution:

$$q(\boldsymbol{M}_t|\boldsymbol{M}_{t-1}) = \mathcal{N}(\boldsymbol{M}_t; \sqrt{1-\beta_t}\boldsymbol{M}_{t-1}, \beta_t \boldsymbol{I}), \quad (1)$$

where $\boldsymbol{M}_t$ denotes the diffused data at time-step[2] $t \in \{0, 1, \cdots, T\}$; $\boldsymbol{M}_0$ and $\boldsymbol{M}_T$ denote the original data and the fully-diffused Gaussian noise, respectively. In case of motion data, $\boldsymbol{M}_0$ is the set of all the joint values of the entire frames. $\beta_t \in (0, 1)$ are hyperparameters with respect to the *variance schedule*, which is set to the *cosine* in FLAME, following (Nichol and Dhariwal 2021). With a sufficiently large T (usually 1,000), this design guarantees $\boldsymbol{M}_T \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$. Also, it gives the marginal distribution in a closed form: $q(\boldsymbol{M}_t|\boldsymbol{M}_0) = \mathcal{N}(\boldsymbol{M}_t; \sqrt{\bar{\alpha}_t}\boldsymbol{M}_0, (1-\bar{\alpha}_t)\boldsymbol{I})$, where $\bar{\alpha}_t = \prod_{i=1}^{T} \alpha_i$ and $\alpha_t = 1 - \beta_t$. This allows connecting $\boldsymbol{M}_t$ and $\boldsymbol{M}_0$ with a single Gaussian noise, which is to be used in formulating Eq. 3.

The model considers the reverse of the diffusion process with the following parameterization:

$$p_\theta(\boldsymbol{M}_{t-1}|\boldsymbol{M}_t, c) = \mathcal{N}(\boldsymbol{M}_{t-1}; \boldsymbol{\mu}_\theta(\boldsymbol{M}_t, c, t), \boldsymbol{\Sigma}_\theta(\boldsymbol{M}_t, c, t)), \quad (2)$$

where c is an optional conditioning variable, language description in our problem. Once the model learns this distribution, inference is done by first sampling $\boldsymbol{M}_T \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$ and then sampling from $p_\theta(\boldsymbol{M}_{t-1}|\boldsymbol{M}_t, c)$, from $t = T$ to $t = 1$.

Training & Loss Functions Training the model parameters, θ, is mostly the same as the VAE (Kingma and Welling 2013). Treating $\boldsymbol{M}_t$ and $\boldsymbol{M}_{t-1}$ as the latent and the data in VAEs, respectively, training is done by maximizing the evidence lower bound (ELBO) for every t. In DDPM, it is shown that the core terms in the ELBO loss can be far simplified to the following after a proper re-weighting and reparameterization (see Appendix B for the details):

$$L_{\text{simple}} = \mathbb{E}_{t, \boldsymbol{M}_0, \boldsymbol{\epsilon}_t} \left[\|\boldsymbol{\epsilon}_t - \boldsymbol{\epsilon}_\theta(\boldsymbol{M}_t(\boldsymbol{M}_0, \boldsymbol{\epsilon}_t), c, t)\|^2 \right], \quad (3)$$

where $\boldsymbol{\epsilon}_t \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$ is the noise used to diffuse $\boldsymbol{M}_0$ to make $\boldsymbol{M}_t$. Now, the model is parameterized to sort out the

[2]Time-steps in this paper denote the diffusion steps. To avoid confusion, the times in motion are deliberately denoted as frames.

noise component $\boldsymbol{\epsilon}_t$ from $\boldsymbol{M}_t$ instead of considering the mean $\boldsymbol{\mu}_\theta$. Performing better in practice, this parameterization still allows computing the mean indirectly from the predicted noise[3], and we can sample from $p_\theta(\boldsymbol{M}_{t-1}|\boldsymbol{M}_t, c)$. Note, however, the reverse-process variance $\boldsymbol{\Sigma}_\theta$ cannot be learned with this loss as it has been deliberately removed.

To train the variance, Nichol and Dhariwal (2021) proposes a hybrid loss and demonstrates better performance:

$$L_{\text{hybrid}} = L_{\text{simple}} + \lambda L_{\text{vlb}}, \tag{4}$$

where L_{vlb} is the original ELBO loss without the reweighting (see Appendix B for the details). We also use this loss for training FLAME.

Model Architecture for Motion Data

Unlike images, motion involves temporal as well as spatial patterns, and its length varies by samples. Thus, we cannot use the U-Net-based architectures widely adopted in the previous diffusion models; we instead propose a new transformer-based architecture (see Figure 2).

Transformer Decoder As explained in the previous section, the model learns the denoising distribution, $p_\theta(\boldsymbol{M}_{t-1}|\boldsymbol{M}_t, c)$, and we use transformer decoder to implement this. As an input to the transformer, the diffused motion $\boldsymbol{M}_t$ is presented as a sequence of L_{mo} frames, where each frame consists of D_{mo}-dimensional joint angle values. To be used as tokens, each frame passes through a linear layer, converted to be D_{emb}-dimensional (see Fig. 2). The conditioning with a language description c is implemented as a cross-attention context. The context is a sequence of token embeddings computed from a pre-trained language model (to be explained in the next paragraph). The output frames are collected at positions where the motion tokens are processed. Then they pass through a linear layer, but this time converted to $2{\cdot}D_{emb}$-dimensional vectors to learn both mean and variance. The vectors are concatenations of ϵ and Σ, which fully parameterize $p_\theta(\boldsymbol{M}_{t-1}|\boldsymbol{M}_t, c)$ together. Variable length is handled by masking in the transformer decoder.

Pre-trained Language Model (PLM) We use the pre-trained RoBERTa model (Liu et al. 2019) to encode the textual description into a sequence of high-level token embeddings. PLMs have been showing their language understanding capability can be transferred to a variety of tasks. Hence, we also adopt the PLM to extract language features.

Time-Step (TS) and Motion-Length (ML) Tokens In addition to the language and the motion tokens of length L_{lang} and L_{mo}, respectively, we introduce two special embedding tokens as inputs to the transformer. The time-step token (TS) is to give the time-step information t, and the motion-length token (ML) is to give the motion-length information L_{mo} to the model. Since FLAME generates the entire motion at once, unlike the autoregressive generation using causal masks, these tokens can explicitly inform the network about motions to be generated.

[3] $\boldsymbol{\mu}_\theta(\mathbf{M}_t, c, t) = \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{M}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(\mathbf{M}_t, c, t)\right)$

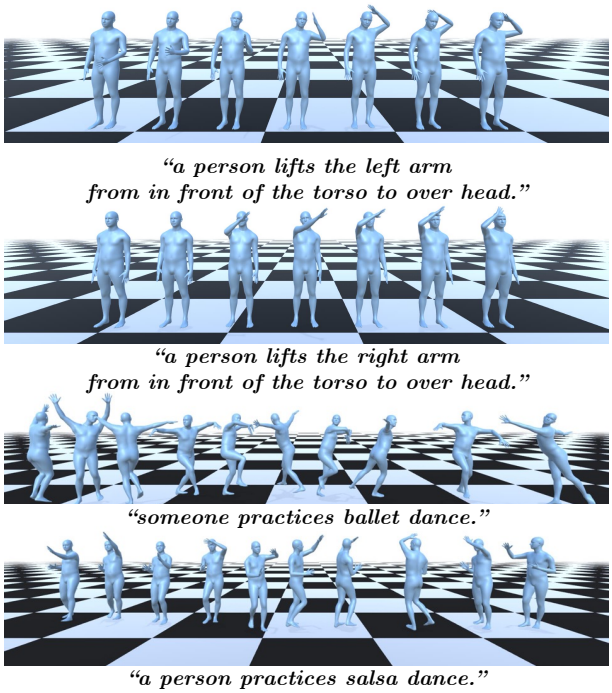


Figure 3: Qualitative results on text-to-motion synthesis task. FLAME is able to synthesize motion from detailed textual descriptions. Motion sequences flow from left to right.

Inference for Motion Synthesis

When synthesizing motion from text, we use the classifier-free guidance (Ho and Salimans 2021) technique for better semantic alignment. While the guidance trades off the sample diversity by little, it uplifts the precision by large and is used in many text-to-image generation models (Nichol et al. 2021; Ramesh et al. 2022).

In details, the guidance amplifies the effect of the conditioning variable c when predicting the noise:

$$\hat{\epsilon}_\theta(\mathbf{M}_t \mid c) = \epsilon_\theta(\mathbf{M}_t \mid \emptyset) + s \cdot (\epsilon_\theta(\mathbf{M}_t \mid c) - \epsilon_\theta(\mathbf{M}_t \mid \emptyset)) . \tag{5}$$

At each denoising step, a guided version $\hat{\epsilon}_\theta(\mathbf{M}_t \mid c)$ is used instead of the original prediction $\epsilon_\theta(\mathbf{M}_t \mid c)$, which amplifies the conditioning effect by a scalar amount $s > 1$. To get an unconditioned prediction as well from the model, $\epsilon_\theta(\mathbf{M}_t \mid \emptyset)$, we randomly replace the text with an empty string $\emptyset$ during training.

Inference for Motion Editing

In motion editing, we want to manipulate parts of data, either frame-wise, joint-wise, or both. Similarly to image inpainting (Banitalebi-Dehkordi and Zhang 2021), we take a "diffuse then conditionally denoise" strategy to in-fill the editable parts with the given language condition. This way, we can make a bridge between the unedited data and the edited data distributions.

In details, we are given with data $\boldsymbol{M}_0^{\text{ref}}$ to edit and a binary mask m that designates the parts for editing with zeros, and

Method	HumanML3D						BABEL					
				R-Precision ↑						R-Precision ↑		
	mCLIP ↑	FD ↓	MID ↑	Top-1	Top-2	Top-3	mCLIP ↑	FD ↓	MID ↑	Top-1	Top-2	Top-3
Lin et al. (2018)	0.142	58.694	18.141	0.225	0.298	0.363	0.183	51.873	33.967	0.678	0.709	0.754
Language2Pose	0.145	55.365	18.982	0.233	0.305	0.381	0.199	42.209	39.360	0.685	0.713	0.788
Ghosh et al. (2021)	0.106	109.778	14.643	0.122	0.151	0.204	0.150	75.316	28.363	0.505	0.591	0.653
TEMOS	0.254	49.142	28.570	0.355	0.481	0.589	0.273	38.679	46.953	0.786	0.835	0.893
Guo et al. (2022)	0.281	27.950	27.744	0.452	0.611	0.675	0.301	24.882	44.758	0.832	0.894	0.911
FLAME (Ours)	**0.297**	**21.152**	**29.935**	**0.513**	**0.673**	**0.749**	**0.318**	**18.234**	**53.003**	**0.888**	**0.926**	**0.939**

Table 1: Text-to-motion benchmark on the HumanML3D and BABEL.

Method	Average Positional Error ↓				Average Variance Error ↓			
	root joint	global traj	mean local	mean global	root joint	global traj	mean local	mean global
Lin et al. (2018)	1.966	1.956	0.105	1.969	0.790	0.789	0.007	0.791
Language2Pose	1.622	1.616	**0.097**	1.630	0.669	0.669	0.006	0.672
Ghosh et al. (2021)	1.291	1.242	0.206	1.294	0.564	0.548	0.024	0.563
TEMOS	0.963	0.955	0.104	0.976	**0.445**	**0.445**	**0.005**	**0.448**
Guo et al. (2022)	0.949	0.937	0.108	0.940	0.510	0.507	0.007	0.552
FLAME (Ours)	**0.881**	**0.869**	0.110	**0.899**	0.497	0.495	0.007	0.500

Table 2: APE and AVE benchmark on the KIT dataset.

	mCLIP↑	Joint Variance ↑	Multimodality ↑
TEMOS	0.252	0.017	11.901
FLAME (Ours)	0.298	0.072	31.500

Table 3: Diversity evaluation on HumanML3D. Each model generates 10 samples per text in the test set for this evaluation.

ones otherwise. We first diffuse $\boldsymbol{M}_0^{\text{ref}}$ in the same way as done in training, obtaining $\boldsymbol{M}_t^{\text{ref}}$ for every t. Then, the fully diffused data $\boldsymbol{M}_T^{\text{ref}}$ is denoised step-by-step using the trained model; however, the masked parts (where $m = 1$) are now overwritten with the unedited ground truth, or the reference $\boldsymbol{M}_{t-1}^{\text{ref}}$:

$$\boldsymbol{M}_{t-1}^{\text{edit}} = (1 - m) \odot \boldsymbol{M}_{t-1}^{\text{pred}} + m \odot \boldsymbol{M}_{t-1}^{\text{ref}}. \quad (6)$$

Here, $\boldsymbol{M}_{t-1}^{\text{pred}} \sim p_\theta(\boldsymbol{M}_{t-1}|\boldsymbol{M}_t^{\text{edit}}, c^{\text{edit}})$ is a predicted denoised sample by the model with a new condition c^{edit} (note $\boldsymbol{M}_T^{\text{edit}} := \boldsymbol{M}_T^{\text{ref}}$).

Experiments

Datasets

In the experiments, we train and evaluate our model on the following datasets. Detailed preprocessing is described in Appendix A.

1. **HumanML3D$_{\text{SMPL}}$ (Guo et al. 2022)** is a recently proposed large motion-text pair dataset containing 44,970 full-sentence text descriptions for 14,616 motions from AMASS (Mahmood et al. 2019) and HumanAct12 (Guo et al. 2020). We use SMPL motion data from AMASS directly for the annotation set.
2. **BABEL (Punnakkal et al. 2021)** provides a language description for AMASS. We use 63,353 frame-level annotations to precisely represent the semantics of motion.

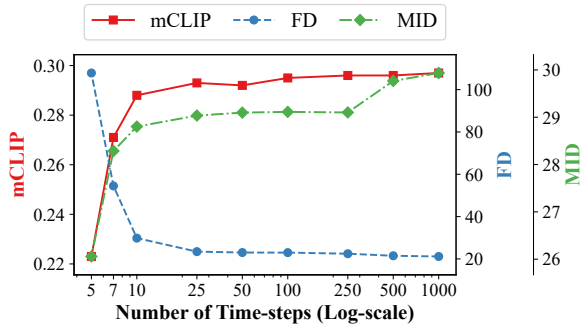


Figure 4: Quantitative results with different numbers of reduced sampling steps. The same trained model with $T = 1000$ diffusion time-steps is used.

3. **KIT (Plappert, Mandery, and Asfour 2016)** consists of 3,911 motion sequences paired with 6,353 textual descriptions. We follow the evaluation protocol used by TEMOS (Petrovich, Black, and Varol 2022).

Motion Representation

HumanML3D$_{\text{SMPL}}$ and BABEL To represent motion, we use the coordinates of the root joint $\boldsymbol{r}_{\text{root}} \in \mathbb{R}^3$ and the rotations of 24 SMPL-joints (Loper et al. 2015) with respect to their parent joints. We use the SMPL pose parameters directly, instead of employing a customized skeleton for simplicity and compatibility. We adopt 6D representation (Zhou et al. 2019) to describe rotations rather than the axis-angle format. In total, a single pose $\boldsymbol{p}$ is represented with a 147-dimensional vector $\boldsymbol{p} \in \mathbb{R}^{147=3+24\times6}$, and motion is represented with a sequence of pose vectors $\boldsymbol{M} = [\boldsymbol{p}_1, \boldsymbol{p}_2, \cdots, \boldsymbol{p}_{L_{mo}}] \in \mathbb{R}^{L_{mo}\times147}$.

Self-Attn	ML Token	X-Attn	PLM Freeze	mCLIP ↑	FD ↓	MID ↑	R-Precision ↑		
							Top-1	Top-2	Top-3
✓				0.239	60.142	20.980	0.405	0.445	0.460
✓	✓			0.239	59.254	21.301	0.410	0.441	0.465
✓	✓	✓		0.290	27.157	29.010	0.439	0.580	0.654
✓	✓	✓	✓	0.297	21.152	29.935	0.513	0.673	0.749

Table 4: Ablation study on four components of FLAME on the HumanML3D.

Sampling Steps	5	25	50	100	500	1,000
Time Elapsed (s)	0.72	1.31	2.17	3.70	16.66	32.81

Table 5: Elapsed time for sampling a motion. Performance is recorded on a single NVIDIA's Tesla V100 SXM2 32GB machine.

KIT For consistency with the prior work, we follow the motion representation used by TEMOS on the KIT dataset. The human pose is encoded with a 64-dimensional feature vector $\boldsymbol{p} \in \mathbb{R}^{64}$ composed of coordinates for 20 joints, an angle between the local and global coordinate system, and translation.

Evaluation Metrics

APE and AVE The Average Position Error (APE) measures the mean positional difference for a generated motion against the ground-truth motion, and the Average Variance Error (AVE) measures the difference of variances between the generated and ground-truth motion. Ahuja and Morency (2019), Ghosh et al. (2021), and Petrovich, Black, and Varol (2022) used the APE and AVE as quantitative metrics for text-to-motion task evaluation on the KIT dataset (see Appendix C for the details).

Feature Extractor Although APE and AVE are used in the previous work, the metrics have a limitation in that they only rely on the joint values of the reference motion instead of high-level semantics. This problem has been complemented by using CLIP (Radford et al. 2021) score or FID in image-text domain. In a similar vein, we separately train a motion and text encoder in a contrastive manner using InfoNCE loss (Oord, Li, and Vinyals 2018). This model is used to compute motion-text alignment (mCLIP), Fréchet distance (FD), mutual information divergence (MID) (Kim et al. 2022a), and R-Precision.

Motion CLIP Score (mCLIP) Motion CLIP score (mCLIP) computes motion-text alignment by computing the cosine similarity between motion and text embeddings from the separately trained motion CLIP model and denote the similarity as *mCLIP*.

Fréchet Distance (FD) FID (Heusel et al. 2017) has been used in the image generation domain combined with the Inception model (Szegedy et al. 2016) to measure the distance between the real and generated image feature vectors. We use the same concept in this work by replacing image feature vectors with motion feature vectors.

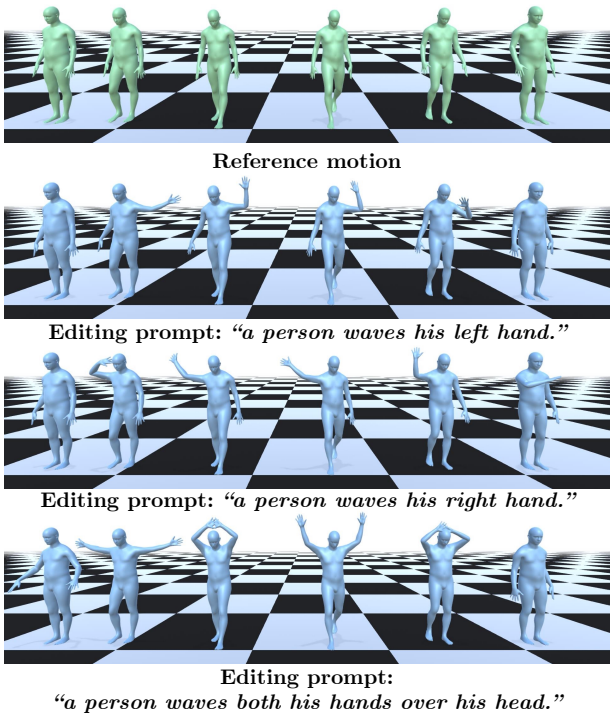


Figure 5: Qualitative results on text-based motion editing. FLAME edits reference motion with given prompts. The model is allowed to edit from both shoulders to hands in this motion. Motion flows from left to right.

Mutual Information Divergence (MID) Similar to mCLIP, the metric measures the alignment between different modalities, but it measures the alignment based on cross-mutual information instead of cosine similarity. MID (Kim et al. 2022b) is recently proposed as a unified metric to evaluate multimodal generative models.

R-Precision R-precision is a metric to measure the alignment between the generated motion and prompt, proposed by Guo et al. (2022). During sampling, it generates 32 textual descriptions composed of one ground truth and randomly sampled 31 texts from the test annotation pool. R-precision counts the average retrieval accuracy by ranking the motion feature and text feature by Euclidean distance.

Training Details

Our FLAME model uses 1,000 diffusion time steps to learn the reverse process with cosine beta scheduling (Nichol and

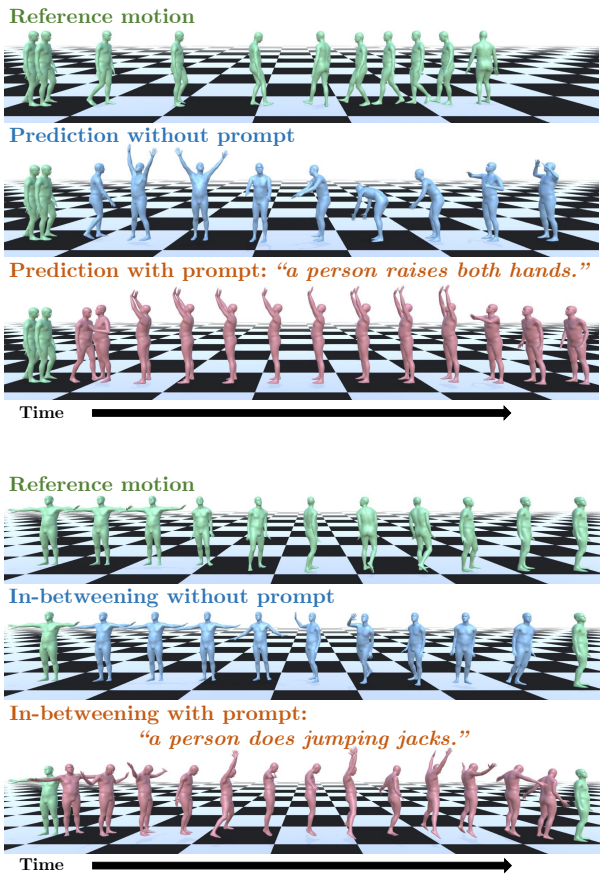


Figure 6: Application of FLAME on motion prediction and motion in-betweening. Green poses are conditioning frames.

Dhariwal 2021). AdamW (Loshchilov and Hutter 2017) is used for experiments with learning rate of 0.0001 and weight decay of 0.0001. For classifier-free guidance, 25% of texts are replaced with empty strings during training, and the classifier guidance scale of 8.0 is used for sampling. FLAME is backed by 8 transformer decoder layers with 8 heads, 2,048 feedforward dimensions, and 768 embedding dimensions (D_{emb}) for both motion and text features (total 64M trainable parameters, see Appendix D for the details). We train the FLAME model using 4 $\times$ NVIDIA Tesla V100 SXM2 32GB for 600K steps on the HumanML3D, 1M steps on the BABEL, and 200K steps on the KIT dataset.

Quantitative Results on Text-to-Motion

We compare our method to four state-of-the-art models: Lin et al. (2018), Language2Pose (Ahuja and Morency 2019), Ghosh et al. (2021), TEMOS (Petrovich, Black, and Varol 2022), and Guo et al. (2022). In case of comparing models using PLM, we replace the PLM with the same model, RoBERTa, to prevent the selection of PLM from influencing the benchmark. Table 1 and Table 2 present the benchmark results on the three datasets. For fair comparison on KIT dataset, we evaluate our model on the same pipeline used in TEMOS. FLAME outperforms other models on all metrics except for the variance metrics in Table 2. However, large variance in motion does not necessarily mean low quality, for example, a prompt "a person dribbles a ball." can correspond to a diverse set of motions rather than a single corresponding ground-truth motion. To validate this, we further compare generated motions on three metrics. First, we sample 10 motions per text annotation in the test set, then we average the variance of joints for the 10 generated motions (Joint Variance). Multimodality of motion (Guo et al. (2020)) is employed to measure the diversity of generated motions. We also compute the average mCLIP score to support that generated motions are not only diverse but also well-aligned to the prompt. Table 3 summarizes the results. All reported metrics are averaged after three trials.

Ablation Study

An ablation study is conducted to validate the four components in our proposed architecture: self-attention block, motion length token, cross-attention block, and freezing of the language model. We start our model from transformer encoder architecture, which uses self-attention only. Next, we add a motion length token to input tokens to explicitly feed the model the number of frames to be generated. On top of these, we employ the cross-attention mechanism, using the transformer decoder architecture. To make the cross-attention context more expressive, we include the first 20 tokens from the PLM output along with the CLS token output. Lastly, we freeze the PLM during the training stage, which results in considerable improvement in performance. These are provided in Table 4.

One of the major drawbacks of diffusion-based models is the slow sampling speed. As provided in Table 5, sampling using 1,000 diffusion steps takes more than 30 seconds per sample, which hinders its use in practical application. To improve sampling speed, we reduced sampling steps from the same trained model and empirically observed the reduced sampling steps can maintain sample quality unless the diffusion step is extremely reduced (Figure 4).

Application on Other Motion Tasks

The proposed motion editing method can be extended to other motion tasks: motion prediction and in-betweening. Unlike most previous task-specific methods, FLAME can perform various motion tasks due to its flexible conditional generation capability (Figure 6).

Conclusion

In this study, we explored a unified model to perform text-to-motion generation and text-based motion editing. To achieve the objective, we proposed a diffusion-based motion generative model FLAME, which is distinguished from previous work in terms of sample quality and flexibility in conditional generations. We expect our proposed model can greatly streamline the laborious motion generation process and lower the barrier to 3D motion synthesis. In the future, we would like to improve the sampling strategy to enable real-time application and make use of features learned in other domains such as the image-vision domain.

Acknowledgments

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. 2019-0-00079 and No. 2022-0-00612).

References

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

Ahuja, C.; and Morency, L.-P. 2019. Language2pose: Natural language grounded pose forecasting. In *2019 International Conference on 3D Vision (3DV)*, 719–728. IEEE.

Banitalebi-Dehkordi, A.; and Zhang, Y. 2021. Repaint: Improving the Generalization of Down-Stream Visual Tasks by Generating Multiple Instances of Training Examples. *arXiv preprint arXiv:2110.10366*.

Dhariwal, P.; and Nichol, A. 2021. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34: 8780–8794.

Duan, Y.; Lin, Y.; Zou, Z.; Yuan, Y.; Qian, Z.; and Zhang, B. 2022. A Unified Framework for Real Time Motion Completion. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(4): 4459–4467.

Fragkiadaki, K.; Levine, S.; Felsen, P.; and Malik, J. 2015. Recurrent network models for human dynamics. In *Proceedings of the IEEE International Conference on Computer Vision*, 4346–4354.

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

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

Guo, C.; Zou, S.; Zuo, X.; Wang, S.; Ji, W.; Li, X.; and 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*, 5152–5161.

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

Guo, X.; and Choi, J. 2019. Human motion prediction via learning local structure representations and temporal dependencies. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 2580–2587.

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

Heusel, M.; Ramsauer, H.; Unterthiner, T.; Nessler, B.; and Hochreiter, S. 2017. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30.

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

Ho, J.; and Salimans, T. 2021. Classifier-free diffusion guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*.

Karras, T.; Laine, S.; and Aila, T. 2019. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 4401–4410.

Karras, T.; Laine, S.; Aittala, M.; Hellsten, J.; Lehtinen, J.; and Aila, T. 2020. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 8110–8119.

Kim, J.; Byun, T.; Shin, S.; Won, J.; and Choi, S. 2022a. Conditional Motion In-betweening. *Pattern Recognition*, 108894.

Kim, J.-H.; Kim, Y.; Lee, J.; Yoo, K. M.; and Lee, S.-W. 2022b. Mutual Information Divergence: A Unified Metric for Multimodal Generative Models. *arXiv preprint arXiv:2205.13445*.

Kingma, D. P.; and Welling, M. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Lin, A. S.; Wu, L.; Corona, R.; Tai, K.; Huang, Q.; and Mooney, R. J. 2018. Generating Animated Videos of Human Activities from Natural Language Descriptions. In *Proceedings of the Visually Grounded Interaction and Language Workshop at NeurIPS 2018*.

Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; and Stoyanov, V. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Loper, M.; Mahmood, N.; Romero, J.; Pons-Moll, G.; and Black, M. J. 2015. SMPL: A Skinned Multi-Person Linear Model. *ACM Trans. Graphics (Proc. SIGGRAPH Asia)*, 34(6): 248:1–248:16.

Loshchilov, I.; and Hutter, F. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Mahmood, N.; Ghorbani, N.; Troje, N. F.; Pons-Moll, G.; and Black, M. J. 2019. AMASS: Archive of Motion Capture as Surface Shapes. In *International Conference on Computer Vision*, 5442–5451.

Nichol, A.; Dhariwal, P.; Ramesh, A.; Shyam, P.; Mishkin, P.; McGrew, B.; Sutskever, I.; and Chen, M. 2021. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*.

Nichol, A. Q.; and Dhariwal, P. 2021. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, 8162–8171. PMLR.

Oord, A. v. d.; Li, Y.; and Vinyals, O. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

Petrovich, M.; Black, M. J.; and Varol, G. 2021. Action-conditioned 3d human motion synthesis with transformer vae. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 10985–10995.

Petrovich, M.; Black, M. J.; and Varol, G. 2022. TEMOS: Generating diverse human motions from textual descriptions. In *European Conference on Computer Vision (ECCV)*.

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

Punnakkal, A. R.; Chandrasekaran, A.; Athanasiou, N.; Quiros-Ramirez, A.; and Black, M. J. 2021. BABEL: Bodies, Action and Behavior with English Labels. In *Proceedings IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 722–731.

Radford, A.; Kim, J. W.; 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. In *International Conference on Machine Learning*, 8748–8763. PMLR.

Ramesh, A.; Dhariwal, P.; Nichol, A.; Chu, C.; and Chen, M. 2022. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*.

Saharia, C.; Chan, W.; Saxena, S.; Li, L.; Whang, J.; Denton, E.; Ghasemipour, S. K. S.; Ayan, B. K.; Mahdavi, S. S.; Lopes, R. G.; et al. 2022. Photorealistic Text-to-Image Diffusion Models with Deep Language Understanding. *arXiv preprint arXiv:2205.11487*.

Sauer, A.; Schwarz, K.; and Geiger, A. 2022. Stylegan-xl: Scaling stylegan to large diverse datasets. In *Special Interest Group on Computer Graphics and Interactive Techniques Conference Proceedings*, 1–10.

Sohl-Dickstein, J.; Weiss, E.; Maheswaranathan, N.; and Ganguli, S. 2015. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, 2256–2265. PMLR.

Song, J.; Meng, C.; and Ermon, S. 2020. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*.

Song, Z.; Wang, D.; Jiang, N.; Fang, Z.; Ding, C.; Gan, W.; and Wu, W. 2022. ActFormer: A GAN Transformer Framework towards General Action-Conditioned 3D Human Motion Generation. *arXiv preprint arXiv:2203.07706*.

Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; and Wojna, Z. 2016. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2818–2826.

Van Den Oord, A.; Vinyals, O.; et al. 2017. Neural discrete representation learning. *Advances in neural information processing systems*, 30.

Zhou, Y.; Barnes, C.; Lu, J.; Yang, J.; and 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*, 5745–5753.