

# MoMask: Generative Masked Modeling of 3D Human Motions

Chuan Guo\*

Yuxuan Mu\*

Muhammad Gohar Javed\*

Sen Wang

Li Cheng

University of Alberta

{cguo2, ymu3, javed4, lcheng5}@ualberta.ca

<https://ericguo5513.github.io/momask/>

Figure 1. Our MoMask, when provided with a text input, generates high-quality 3D human motion with diversity and precise control over subtleties such as “two strides forward”, “pivot on left foot”, and “pivot swiftly”.

## Abstract

We introduce MoMask, a novel masked modeling framework for text-driven 3D human motion generation. In MoMask, a hierarchical quantization scheme is employed to represent human motion as multi-layer discrete motion tokens with high-fidelity details. Starting at the base layer, with a sequence of motion tokens obtained by vector quantization, the residual tokens of increasing orders are derived and stored at the subsequent layers of the hierarchy. This is consequently followed by two distinct bidirectional transformers. For the base-layer motion tokens, a Masked Transformer is designed to predict randomly masked motion tokens conditioned on text input at training stage. During generation (i.e. inference) stage, starting from an empty sequence, our Masked Transformer iteratively fills up the missing tokens; Subsequently, a Residual Transformer learns to progressively predict the next-layer tokens based on the results from current layer. Extensive experiments demonstrate that MoMask outperforms the state-

of-art methods on the text-to-motion generation task, with an FID of 0.045 (vs e.g. 0.141 of T2M-GPT) on the HumanML3D dataset, and 0.228 (vs 0.514) on KIT-ML, respectively. MoMask can also be seamlessly applied in related tasks without further model fine-tuning, such as text-guided temporal inpainting.

## 1. Introduction

Generating 3D human motions from textual descriptions, aka text-to-motion generation, is a relatively new task that may play an important role in a broad range of applications such as video games, metaverse, and virtual reality & augmented reality. In the past few years, it has generated intensive research interests [9, 15, 16, 21, 23, 36, 42, 49, 50]. Among them, it has become popular to engage generative transformers in modeling human motions [13, 16, 21, 49]. In this pipeline, motions are transformed into discrete tokens through vector quantization (VQ), then fed into e.g. an autoregressive model to generate the sequence of motion tokens in an unidirectional order. Though achieving impres-

\*These authors contributed equally to this work.

sive results, these methods shares two innate drawbacks. To begin with, the VQ process inevitably introduces approximation errors, which imposes undesired limit to the motion generation quality. Moreover, the unidirectional decoding may unnecessarily hinder the expressiveness of the generative models. For instance, consider the following scenario: at each time step, the motion content is generated by only considering the preceding (rather than global) context; furthermore, errors will often accumulate over the generation process. Though several recent efforts using discrete diffusion models [23, 30] have considered to decode the motion tokens bidirectionally, by relying on a cumbersome discrete diffusion process, they typically require hundreds of iterations to produce a motion sequence.

Motivated by these observations, we propose a novel framework, MoMask, for high-quality and efficient text-to-motion generation by leveraging the residual vector quantization (RVQ) techniques [4, 34, 48] and the recent generative masked transformers [7, 8, 24, 47]. Our approach builds on the following three components. First, an RVQ-VAE is learned to establish precise mappings between 3D motions and the corresponding sequences of discrete motion tokens. Unlike previous motion VQ tokenizers [13, 16, 49] that typically quantize latent codes in a single pass, our hierarchical RVQ employs iterative rounds of residual quantization to progressively reduce quantization errors. This results in multi-layer motion tokens, with the base layer serving to perform standard motion quantization, and the rest layers in the hierarchy capturing the residual coding errors of their respective orders, layer by layer. Our quantization-based hierarchical design is further facilitated by two distinct transformers, the Masked Transformer (i.e. M-Transformer) and Residual Transformer (R-Transformer), that are dedicated to generating motion tokens for the base VQ layer and the rest residual layers, respectively.

The M-Transformer, based on BERT [10], is trained to predict the randomly masked tokens at the base layer, conditioned on textual input. The ratio of masking, instead of being fixed [10, 18], is a scheduled variable that ranges from 0 to 1. During generation, starting from all tokens being masked out, M-Transformer produces a complete sequence of motion tokens within a small number of iterations. At each iteration, all masked tokens are predicted *simultaneously*. Predicted tokens with the highest confidence will remain unchanged, while the others are masked again and re-predicted in the next iteration. Once the base-layer tokens are generated, the R-Transformer ensues to progressively predict the residual tokens of the subsequent layer given the token sequence at current layer. Overall, the entire set of layered motion tokens can be efficiently generated within merely 15 iterations, regardless of the motion’s length.

Our main contributions can be summarized as follows: First, our MoMask is the first generative masked modeling

framework for the problem of text-to-motion generation. It comprises of a hierarchical quantization generative model and the dedicated mechanism for precise residual quantization, base token generation and residual token prediction. Second, our MoMask pipeline produces precise and efficient text-to-motion generation. Empirically, it achieves new state-of-the-art performance on text-to-motion generation task with an FID of 0.045 (vs. 0.141 in [49]) on HumanML3D and 0.204 (vs. 0.514 in [49]) on KIT-ML. Third, our MoMask also works well for related tasks, such as text-guided motion inpainting.

## 2. Related Work

**Human Motion Generation.** Recently, we have witnessed the surge of works for neural motion generation, with conditioning on various domains such as motion prefix [29, 33], action class [6, 14, 31, 35], audio [13, 40, 43, 53], texts [9, 15, 16, 36, 42]. Early works [1, 12, 20, 27, 38] commonly model motion generation deterministically, resulting in averaged and blurry motion results. This is properly addressed by stochastic models. GAN modeling is adopted in [5, 46] for action-conditioned motion generation. Meanwhile, temporal VAE framework and transformer architecture are exploited in the works of [17, 35]. In terms of text-to-motion generation, T2M [15] extended the temporal VAE to learn the probabilistic mapping between texts and motions. Similarly, TEMOS [36] takes advantage of Transformer VAE to optimize a joint variational space between natural language and motions, which is extended by TEACH [3] for long motion compositions. MotionCLIP [41] and ohMG [28] model text-to-motion in an unsupervised manner using the large pretrained CLIP [39] model. The emerging diffusion models and autoregressive models have significantly changed the field of motion generation. In diffusion methods, a network is learned to gradually denoise the motion sequence, supervised by a scheduled diffusion process [9, 22, 23, 30, 42, 43, 50]. Regarding autoregressive models [13, 16, 21, 49, 52], motions are firstly discretized as tokens via vector quantization [44], which are then modeled by the expressive transformers as in language model.

**Generative Masked Modeling.** BERT [10] introduces masked modeling for language tasks that word tokens are randomly masked out with a fixed ratio, and then the bi-directional transformer learns to predict the masked tokens. Despite being a decent pre-trained text encoder, BERT cannot synthesize novel samples. In this regard, [7] proposes to mask the tokens with a variable and traceable rate that is controlled by a scheduling function. Therefore, new samples can be synthesized iteratively following the scheduled masking. MAGE [24] unifies representation learning and image synthesis using the masked generative encoder. Muse [8] extends this paradigm for text-to-image generation and editing. Magvit [47] suggests a versatile masking

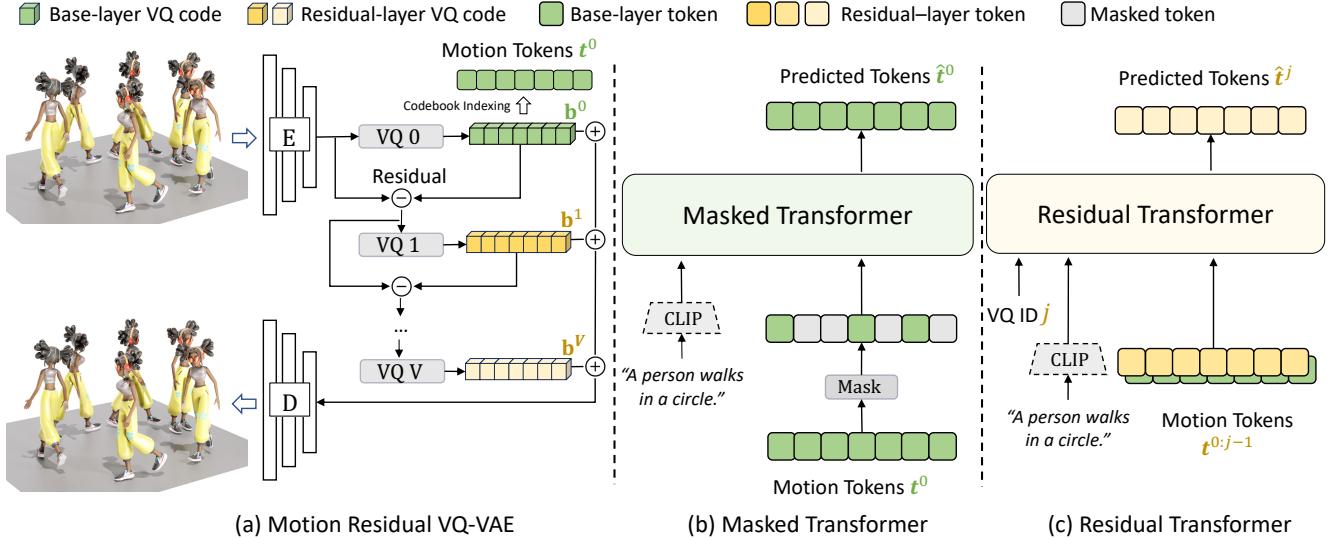


Figure 2. **Approach overview.** (a) Motion sequence is tokenized through vector quantization (VQ), also referred to as the base quantization layer, as well as a hierarchy of multiple layers for residual quantization. (b) *Parallel* prediction by the Masked Transformer: the tokens in the base layer  $t^0$  are randomly masked out with a variable rate, and then a text-conditioned masked transformer is trained to predict the masked tokens in the sequence simultaneously. (c) Layer-by-layer *progressive* prediction by the Residual Transformer. A text-conditioned residual transformer learns to progressively predict the residual tokens  $t^{j>0}$  from the tokens in previous layers,  $t^{0:j-1}$ .

strategy for multi-task video generation. Inspired by these successes, we first introduce generative masked modeling for human motion synthesis in this paper.

**Deep Motion Quantization and RVQ.** [2] learns semantically meaningful discrete motif words leveraging triplet contrastive learning. TM2T [16] starts applying vector quantized-VAE [44] to learn the mutual mapping between human motions and discrete tokens, where the autoencoding latent codes are replaced with the selected entries from a codebook. T2M-GPT [49] further enhances the performance using EMA and code reset techniques. Nevertheless, the quantization process inevitably introduces errors, leading to suboptimal motion reconstruction. In this work, we adapt residual quantization [4, 34, 48], a technique used in neural network compression [11, 25, 26] and audio quantization [4, 45] which iteratively quantizes a vector and its residuals. This approach represents the vector as a stack of codes, enabling high-precision motion discretization.

### 3. Approach

Our goal is to generate a 3D human pose sequence  $\mathbf{m}_{1:N}$  of length  $N$  guided by a textual description  $c$ , where  $\mathbf{m}_i \in \mathbb{R}^D$  with  $D$  denoting the dimension of pose features. As illustrated in Fig. 2, our approach consists of three principle components: a residual-based quantizer that tokenizes motion sequence into multi-layer discrete tokens (Sec. 3.1), a masked transformer that generates motion tokens in the base layer (Sec. 3.2), and a residual transformer (Sec. 3.3) that predicts the tokens in the subsequent residual layers.

The inference process of generation is detailed in Sec. 3.4.

#### 3.1. Training: Motion Residual VQ-VAE

Conventional motion VQ-VAEs [16, 21, 49, 52] transform a motion sequence into one tuple of discrete motion tokens. Specifically, the motion sequence  $\mathbf{m}_{1:N} \in \mathbb{R}^{N \times D}$  is firstly encoded into a latent vector sequence  $\tilde{\mathbf{b}}_{1:n} \in \mathbb{R}^{n \times d}$  with downsampling ratio of  $n/N$  and latent dimension  $d$ , using 1D convolutional encoder  $E$ ; each vector is subsequently replaced with its nearest code entry in a preset codebook  $\mathcal{C} = \{\mathbf{c}_k\}_{k=1}^K \subset \mathbb{R}^d$ , known as quantization  $Q(\cdot)$ . Then the quantized code sequence  $\mathbf{b}_{1:n} = Q(\tilde{\mathbf{b}}_{1:n}) \in \mathbb{R}^{n \times d}$  is projected back to motion space for reconstructing the motion  $\hat{\mathbf{m}} = D(\mathbf{b})$ . After all, the indices of the selected codebook entries (namely *motion tokens*) are used as the alternative discrete representation of input motion. Though effective, the quantization operation  $Q(\cdot)$  inevitably leads to information loss, which further limits the quality of reconstruction.

To address this issue, we introduce residual quantization (RQ) as described in Fig. 2(a). In particular, RQ represents a motion latent sequence  $\tilde{\mathbf{b}}$  as  $V + 1$  ordered code sequences, using  $V + 1$  quantization layers. Formally, this is defined as  $RQ(\tilde{\mathbf{b}}_{1:n}) = [\mathbf{b}_{1:n}^v]_{v=0}^V$ , with  $\mathbf{b}_{1:n}^v \in \mathbb{R}^{n \times d}$  denoting the code sequence at the  $v$ -th quantization layer. Concretely, starting from 0-th residual  $\mathbf{r}^0 = \tilde{\mathbf{b}}$ , RQ recursively calculates  $\mathbf{b}^v$  as the approximation of residual  $\mathbf{r}^v$ , and then the next residual  $\mathbf{r}^{v+1}$  as

$$\mathbf{b}^v = Q(\mathbf{r}^v), \quad \mathbf{r}^{v+1} = \mathbf{r}^v - \mathbf{b}^v, \quad (1)$$

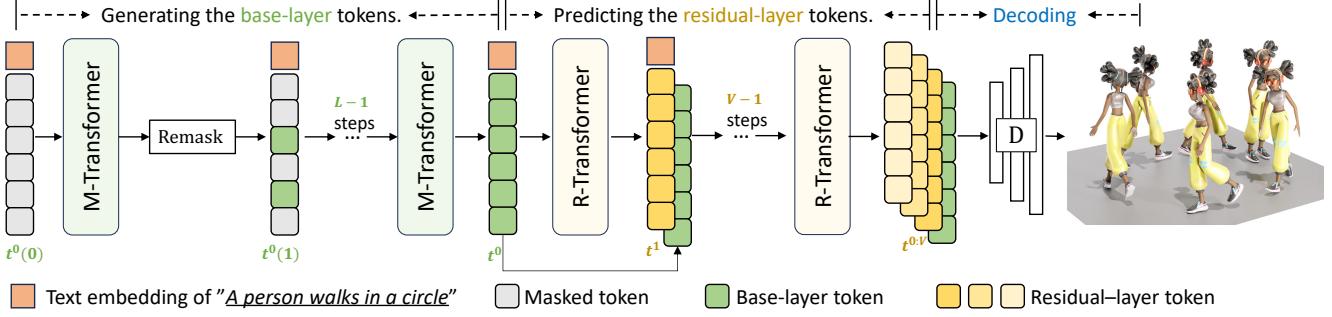


Figure 3. **Inference process.** Starting from an empty sequence  $t^0(0)$ , the M-Transformer generates the base-layer token sequence  $t^0$  in  $L$  iterations. Following this, the R-Transformer progressively predicts the rest-layer token sequences  $t^{2:V}$  within  $V - 1$  steps.

for  $v = 0, \dots, V$ . After RQ, the final approximation of latent sequence  $\tilde{\mathbf{b}}$  is the sum of all quantized sequences  $\sum_{v=0}^V \mathbf{b}^v$ , which is then fed into decoder D for motion reconstruction.

Overall, the residual VQ-VAE is trained via a motion reconstruction loss combined with a latent embedding loss at each quantization layer:

$$\mathcal{L}_{rvq} = \|\mathbf{m} - \hat{\mathbf{m}}\|_1 + \beta \sum_{v=1}^V \|\mathbf{r}^v - \text{sg}[\mathbf{b}^v]\|_2^2, \quad (2)$$

where  $\text{sg}[\cdot]$  denotes the stop-gradient operation, and  $\beta$  a weighting factor for embedding constraint. This framework is optimized with straight-through gradient estimator [44], and our codebooks are updated via exponential moving average and codebook reset following T2M-GPT [49].

**Quantization Dropout.** Ideally, the early quantization layers are expected to restore the input motion as much as possible; then the later layers add up the missing finer details. To exploit the capacity of each quantizer, we adopt a *quantization dropout* strategy, which randomly disables the last 0 to  $V$  layers with probability  $q \in [0, 1]$  during training.

After training, each motion sequence  $\mathbf{m}$  can be represented as  $V + 1$  discrete motion token sequences  $T = [t_{1:n}^v]_{v=0}^V$  where each token sequence  $t_{1:n}^v \in \{1, \dots, |\mathcal{C}^v|\}^n$  is the ordered codebook-indices of quantized embedding vectors  $\mathbf{b}_{1:n}^v$ , such that  $\mathbf{b}_i^v = \mathcal{C}_{t_i^v}^v$  for  $i \in [1, n]$ . Among these  $V + 1$  sequences, the first (i.e. base) sequence possesses the most prominent information, while the impact of subsequent layers gradually diminishes.

### 3.2. Training: Masked Transformer

Our bidirectional masked transformer is designed to model the base-layer motion tokens  $t_{1:n}^0 \in \mathbb{R}^n$ , as illustrated in Figure 2(b). We first randomly masked out a varying fraction of sequence elements, by replacing the tokens with a special [MASK] token. With  $\tilde{t}^0$  denoting the sequence after masking, the goal is to predict the masked tokens given text  $c$  and  $\tilde{t}^0$ . We use CLIP [39] for extracting text features. Mathematically, our masked transformer  $p_\theta$  is optimized to minimize the negative log-likelihood of target predictions:

$$\mathcal{L}_{mask} = \sum_{\tilde{t}_k^0=[\text{MASK}]} -\log p_\theta(t_k^0 | \tilde{t}^0, c). \quad (3)$$

**Mask Ratio Schedule.** We adopt a cosine function  $\gamma(\cdot)$  for scheduling the masking ratio following [7, 8]. Practically, the mask ratio is obtained by  $\gamma(\tau) = \cos(\frac{\pi\tau}{2}) \in [0, 1]$ , where  $\tau \in [0, 1]$  that  $\tau = 0$  means the sequence is completely corrupted. During training, the  $\tau \sim \mathcal{U}(0, 1)$  is randomly sampled, and then  $m = \lceil \gamma(\tau) \cdot n \rceil$  sequence entries are uniformly selected to be masked with  $n$  denoting the length of sequence.

**Replacing and Remasking.** To enhance the contextual reasoning of the masked transformer, we adopt the remasking strategy used in BERT pretraining [10]. If a token is selected for masking, we replace this token with (1) [MASK] token 80% of the time; (2) a random token 10% of the time; and (3) an unchanged token 10% of the time.

### 3.3. Training: Residual Transformer

We learn a single residual transformer to model the tokens from the other  $V$  residual quantization layers. The residual transformer has a similar architecture to the masked transformer (Sec. 3.2), except that it contains  $V$  separate embedding layers. During training, we randomly select a quantizer layer  $j \in [1, V]$  to learn. All the tokens in the preceding layers  $t^{0:j-1}$  are embedded and summed up as the token embedding input. Taking the token embedding, text embedding, and RQ layer indicator  $j$  as input, the residual transformer  $p_\phi$  is trained to predict the  $j$ -th layer tokens in parallel. Overall, the training objective is:

$$\mathcal{L}_{res} = \sum_{j=1}^V \sum_{i=1}^n -\log p_\phi(t_i^j | t_i^{1:j-1}, c, j). \quad (4)$$

We also share the parameters of the  $j$ -th prediction layer and the  $(j+1)$ -th motion token embedding layer for more efficient learning.

Datasets	Methods	R Precision↑			FID↓	MultiModal Dist↓	MultiModality↑
		Top 1	Top 2	Top 3			
Human ML3D	TM2T [16]	0.424 $\pm$ .003	0.618 $\pm$ .003	0.729 $\pm$ .002	1.501 $\pm$ .017	3.467 $\pm$ .011	<u>2.424</u> $\pm$ .093
	T2M [15]	0.455 $\pm$ .003	0.636 $\pm$ .003	0.736 $\pm$ .002	1.087 $\pm$ .021	3.347 $\pm$ .008	2.219 $\pm$ .074
	MDM [42]	-	-	0.611 $\pm$ .007	0.544 $\pm$ .044	5.566 $\pm$ .027	<b>2.799</b> $\pm$ .072
	MLD [9]	0.481 $\pm$ .003	0.673 $\pm$ .003	0.772 $\pm$ .002	0.473 $\pm$ .013	3.196 $\pm$ .010	2.413 $\pm$ .079
	MotionDiffuse [50]	0.491 $\pm$ .001	0.681 $\pm$ .001	0.782 $\pm$ .001	0.630 $\pm$ .001	3.113 $\pm$ .001	1.553 $\pm$ .042
	T2M-GPT [49]	0.492 $\pm$ .003	0.679 $\pm$ .002	0.775 $\pm$ .002	0.141 $\pm$ .005	3.121 $\pm$ .009	1.831 $\pm$ .048
	ReMoDiffuse [51]	0.510 $\pm$ .005	0.698 $\pm$ .006	0.795 $\pm$ .004	0.103 $\pm$ .004	<u>2.974</u> $\pm$ .016	1.795 $\pm$ .043
	MoMask (base)	0.504 $\pm$ .004	0.699 $\pm$ .006	0.797 $\pm$ .004	0.082 $\pm$ .008	3.050 $\pm$ .013	1.050 $\pm$ .061
	<b>MoMask</b>	<b>0.521</b> $\pm$ .002	<b>0.713</b> $\pm$ .002	<b>0.807</b> $\pm$ .002	<b>0.045</b> $\pm$ .002	<b>2.958</b> $\pm$ .008	1.241 $\pm$ .040
KIT- ML	TM2T [16]	0.280 $\pm$ .005	0.463 $\pm$ .006	0.587 $\pm$ .005	3.599 $\pm$ .153	4.591 $\pm$ .026	<b>3.292</b> $\pm$ .081
	T2M [15]	0.361 $\pm$ .005	0.559 $\pm$ .007	0.681 $\pm$ .007	3.022 $\pm$ .107	3.488 $\pm$ .028	2.052 $\pm$ .107
	MDM [42]	-	-	0.396 $\pm$ .004	0.497 $\pm$ .021	9.191 $\pm$ .022	1.907 $\pm$ .214
	MLD [9]	0.390 $\pm$ .008	0.609 $\pm$ .008	0.734 $\pm$ .007	0.404 $\pm$ .027	3.204 $\pm$ .027	<u>2.192</u> $\pm$ .071
	MotionDiffuse [50]	0.417 $\pm$ .004	0.621 $\pm$ .004	0.739 $\pm$ .004	1.954 $\pm$ .062	2.958 $\pm$ .005	0.730 $\pm$ .013
	T2M-GPT [49]	0.416 $\pm$ .006	0.627 $\pm$ .006	0.745 $\pm$ .006	0.514 $\pm$ .029	3.007 $\pm$ .023	1.570 $\pm$ .039
	ReMoDiffuse [51]	0.427 $\pm$ .014	0.641 $\pm$ .004	0.763 $\pm$ .055	<b>0.155</b> $\pm$ .006	<u>2.814</u> $\pm$ .012	1.239 $\pm$ .028
	MoMask (base)	0.415 $\pm$ .010	0.634 $\pm$ .011	0.760 $\pm$ .005	0.372 $\pm$ .020	2.931 $\pm$ .041	1.097 $\pm$ .054
	<b>MoMask</b>	<b>0.433</b> $\pm$ .007	<b>0.656</b> $\pm$ .005	<b>0.781</b> $\pm$ .005	0.204 $\pm$ .011	<b>2.779</b> $\pm$ .022	1.131 $\pm$ .043

Table 1. **Quantitative evaluation on the HumanML3D and KIT-ML test set.**  $\pm$  indicates a 95% confidence interval. MoMask (base) means that MoMask only uses base-layer tokens. **Bold** face indicates the best result, while underscore refers to the second best.

### 3.4. Inference

As presented in Figure 3, there are three stages in inference. Firstly, starting from an empty sequence  $t^0(0)$  that all tokens are masked out, we expect to generate the base-layer token sequence  $t^0$  of length  $n$  in  $L$  iterations. Given the masked token sequence at  $l$ -th iteration  $t^0(l)$ , M-Transformer first predicts the probability distribution of tokens at the masked locations, and samples motion tokens with the probability. Then the sampled tokens with the lowest  $\lceil \gamma(\frac{l}{L}) \cdot n \rceil$  confidences are masked again, and the other tokens will remain unchanged for the rest iterations. This new token sequence  $t^0(l+1)$  is used to predict the token sequence at the next iteration until  $l$  reaches  $L$ . Once the base-layer tokens are completely generated, the R-Transformer progressively predicts the token sequence in the rest quantization layers. Finally, all tokens are decoded and projected back to motion sequences through the RVQ-VAE decoder.

**Classifier Free Guidance.** We adopt classifier-free guidance (CFG) [8, 19] for the prediction of both M-Transformer and R-Transformer. During training, we train the transformers unconditionally  $c = \emptyset$  with probability of 10%. During inference, CFG takes place at the final linear projection layer before softmax, where the final logits  $\omega_g$  are computed by moving the conditional logits  $\omega_c$  away from the unconditional logits  $\omega_u$  with guidance scale  $s$ :

$$\omega_g = (1 + s) \cdot \omega_c - s \cdot \omega_u. \quad (5)$$

## 4. Experiments

Empirical evaluations are conducted on two widely used motion-language benchmarks, HumanML3D [15] and KIT-ML [37]. **HumanML3D** dataset collects 14,616 motions from AMASS [32] and HumanAct12 [14] datasets, with each motion described by 3 textual scripts, totaling 44,970 descriptions. This diverse motion-language dataset contains a variety of actions, including exercising, dancing, and acrobatics. **KIT-ML** dataset consists of 3,911 motions and 6,278 text descriptions, offering a small-scale evaluation benchmark. For both motion datasets, we adopt the pose representation from the work of T2M [15]. The datasets are augmented by mirroring, and divided into training, testing, and validation sets with the ratio of **0.8:0.15:0.05**.

**Evaluation metrics** from T2M [15] are also adopted throughout our experiments including: (1) *Frechet Inception Distance* (FID), which evaluates the overall motion quality by measuring the distributional difference between the high-level features of the generated motions and those of real motions; (2) *R-Precision* and *multimodal distance*, which gauge the semantic alignment between input text and generated motions; and (3) *Multimodality* for assessing the diversity of motions generated from the same text.

Though *multimodality* is indeed important, we stress its role as a secondary metric that should be assessed in the conjunction with primary performance metrics such as FID and RPrecision. Emphasizing multimodality without considering the overall quality of generated results could lead to optimization of models that produce random outputs for

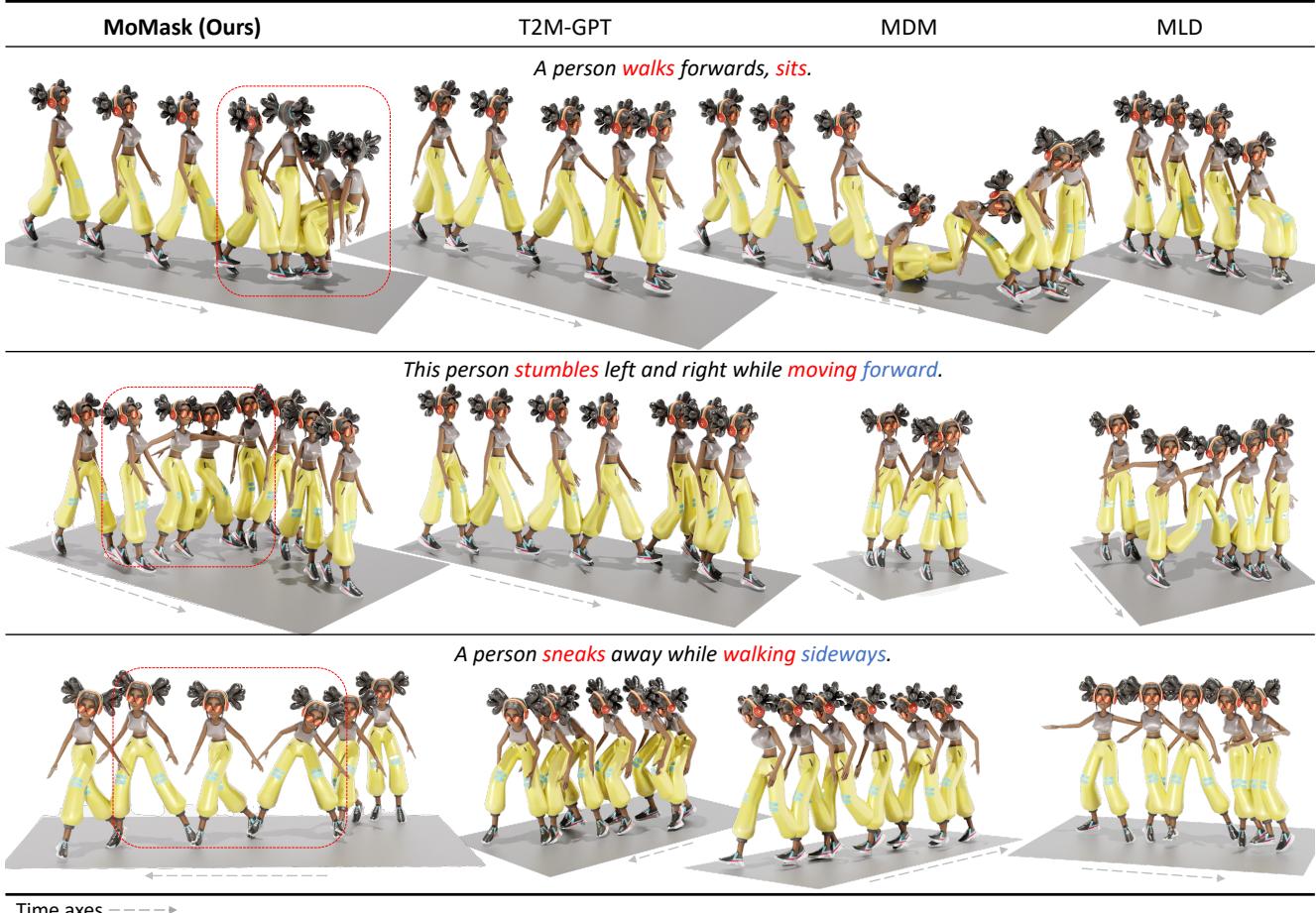


Figure 4. **Visual comparisons** between the different methods given three distinct text descriptions from HumanML3D testset. Only key frames are displayed. Compared to previous methods, MoMask generates motions with higher quality and better understanding of the subtle language concepts such as “stumble”, “sneak”, “walk sideways”. Please refer to the demo video for complete motion clips.

any given input.

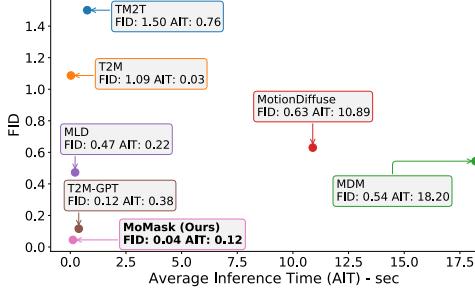
**Implementation Details.** Our models are implemented using PyTorch. For the motion residual VQ-VAE, we employ resblocks for both the encoder and decoder, with a down-scale factor of 4. The RVQ consists of 6 quantization layers, where each layer’s codebook contains 512 512-dimensional codes. The quantization dropout ratio  $q$  is set to 0.2. Both the masked transformer and residual transformer are composed of 6 transformer layers, with 6 heads and a latent dimension of 384, applied to the HumanML3D and KIT-ML datasets. The learning rate reaches 2e-4 after 2000 iterations with a linear warm-up schedule for the training of all models. The mini-batch size is uniformly set to 512 for training RVQ-VAE and 64, 32 for training transformers on HumanML3D and KIT-ML, respectively. During inference, we use the CFG scale of 4 and 5 for M-Transformer and R-Transformer on HumanML3D, and (2, 5) on KIT-ML. Meanwhile,  $L$  is set to 10 on both datasets.

#### 4.1. Comparison to state-of-the-art approaches

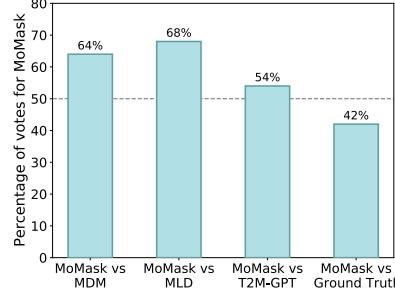
We compare our approach to a set of existing state-of-the-art works ranging from VAE [15], diffusion-based models [9, 42, 51], to autoregressive models [16, 49].

**Quantitative Comparisons.** Following previous practices [15, 42], each experiment is repeated 20 times, and the reported metric values represent the mean with a 95% statistical confidence interval. Additionally, we conduct experiments with MoMask exclusively generating the base-layer motion tokens, denoted as MoMask (base). Quantitative results for the HumanML3D and KIT-ML datasets are presented in Table 1.

Overall, MoMask attains state-of-the-art performance on both datasets, demonstrating substantial improvements in metrics such as FID, R-Precision, and multimodal distance. For the suboptimal performance on KIT-ML dataset, we would like to point out that the leading model, ReMoDiffuse [51], involves more intricate data retrieval from a large



(a) Comparisons on FID and Inference Cost



(b) HumanML3D User Study

Figure 5. (a) Comparison of inference time costs. All tests are conducted on the same Nvidia2080Ti. The closer the model is to the origin, the better. (b) User study results on the HumanML3D dataset. Each bar represents the preference rate of MoMask over the compared model. Overall, MoMask is preferred over the other models most of the time. The dashed line marks 50%.

database to achieve high-quality motion generation. Additionally, we observe that MoMask, even with the base-layer tokens alone, already achieves competitive performance compared to baselines, and the inclusion of residual tokens further elevates the results to a higher level.

In Figure 5(a), we evaluate the efficiency and quality of motion generation using various methods. The inference cost is calculated as the average inference time over 100 samples on one Nvidia2080Ti device. Comparing to baseline methods, MoMask positions itself more favorably between generation quality and efficiency.

**User Study.** We further conduct a user study on Amazon Mechanical Turk to validate our previous conclusions. This user study involves 42 AMT users with *master* recognition, with the side-by-side comparisons between MoMask and each of the state-of-the-art methods including MDM [42], MLD [9] and T2M-GPT [49]. We generate the 50 motions for each method using the same text pool from HumanML3D test set, and collect feedback from 3 distinct users for each comparison. As shown in Fig. 5(b), MoMask is preferred by users in most of the time, and even earns 42% of preference on par with ground truth motions.

**Qualitative Comparisons.** Figure 4 displays qualitative comparisons of our approach and MDM[42], MLD [9], and T2M-GPT [49]. MDM [42] usually generates overall semantically correct motions but fails to capture nuanced concepts such as “*sneak*” and “*sideways*”. Though T2M-GPT [49] and MLD [9] have improved performance in this aspect, they still find it difficult to generate motions accurately aligned with the textual description. For example, in the bottom row, the motions from these two methods either forget to *walk sideways* (T2M-GPT [49]) or to *sneak away* (MLD [9]). Moreover, MLD [9] sometimes produces lifeless motions where the character slides around, as shown in the top row. In comparison, our method is able to generate high-quality motions faithful to the input texts. Please refer to supplementary videos for dynamic visualizations.

Methods	Reconstruction		Generation	
	FID↓	MPJPE↓	FID↓	MM-Dist↓
<b>Evaluation on KIT-ML dataset</b>				
M2DM [23]	0.413 <sup>±.009</sup>	-	0.515 <sup>±.029</sup>	3.015 <sup>±.017</sup>
T2M-GPT [49]	0.472 <sup>±.011</sup>	-	0.514 <sup>±.029</sup>	3.007 <sup>±.023</sup>
<b>MoMask</b>	<b>0.112<sup>±.002</sup></b>	<b>37.2</b>	<b>0.228<sup>±.011</sup></b>	<b>2.774<sup>±.022</sup></b>
<b>Evaluation on HumanML3D dataset</b>				
TM2T [16]	0.307 <sup>±.002</sup>	230.1	1.501 <sup>±.017</sup>	3.467 <sup>±.011</sup>
M2DM [23]	0.063 <sup>±.001</sup>	-	0.352 <sup>±.005</sup>	3.116 <sup>±.008</sup>
T2M-GPT [49]	0.070 <sup>±.001</sup>	58.0	0.141 <sup>±.005</sup>	3.121 <sup>±.009</sup>
<b>MoMask</b>	<b>0.019<sup>±.001</sup></b>	<b>29.5</b>	<b>0.051<sup>±.002</sup></b>	<b>2.957<sup>±.008</sup></b>
w/o RQ	0.091 <sup>±.001</sup>	58.7	0.093 <sup>±.004</sup>	3.031 <sup>±.009</sup>
w/o QDropout	0.077 <sup>±.000</sup>	39.3	0.091 <sup>±.003</sup>	2.959 <sup>±.008</sup>
w/o RRmask	-	-	0.063 <sup>±.003</sup>	3.049 <sup>±.006</sup>
MoMask ( <i>V</i> , 0)	0.091 <sup>±.001</sup>	58.7	0.093 <sup>±.004</sup>	3.031 <sup>±.009</sup>
MoMask ( <i>V</i> , 1)	0.069 <sup>±.001</sup>	54.6	0.073 <sup>±.003</sup>	3.031 <sup>±.008</sup>
MoMask ( <i>V</i> , 2)	0.049 <sup>±.002</sup>	46.0	0.079 <sup>±.003</sup>	2.978 <sup>±.006</sup>
MoMask ( <i>V</i> , 3)	0.037 <sup>±.001</sup>	42.5	0.064 <sup>±.003</sup>	2.970 <sup>±.007</sup>
MoMask ( <i>V</i> , 4)	0.027 <sup>±.001</sup>	35.3	0.069 <sup>±.003</sup>	2.987 <sup>±.007</sup>
MoMask ( <i>V</i> , 5)	0.019 <sup>±.001</sup>	29.5	<b>0.051<sup>±.002</sup></b>	<b>2.962<sup>±.008</sup></b>
MoMask ( <i>V</i> , 6)	0.014 <sup>±.001</sup>	26.7	0.076 <sup>±.003</sup>	2.994 <sup>±.007</sup>
MoMask ( <i>V</i> , 7)	<b>0.014<sup>±.000</sup></b>	<b>25.3</b>	0.084 <sup>±.004</sup>	2.968 <sup>±.007</sup>
MoMask ( <i>q</i> , 0)	0.077 <sup>±.000</sup>	39.3	0.091 <sup>±.003</sup>	2.959 <sup>±.008</sup>
MoMask ( <i>q</i> , 0.2)	<b>0.019<sup>±.001</sup></b>	<b>29.5</b>	<b>0.051<sup>±.002</sup></b>	<b>2.957<sup>±.008</sup></b>
MoMask ( <i>q</i> , 0.4)	0.021 <sup>±.000</sup>	30.2	0.082 <sup>±.003</sup>	3.006 <sup>±.007</sup>
MoMask ( <i>q</i> , 0.6)	0.024 <sup>±.000</sup>	33.2	0.053 <sup>±.003</sup>	2.946 <sup>±.006</sup>
MoMask ( <i>q</i> , 0.8)	0.023 <sup>±.000</sup>	33.4	0.083 <sup>±.004</sup>	3.002 <sup>±.008</sup>

Table 2. Comparison of our RVQ design vs. motion VQs from previous works [16, 23, 49], and further analysis on residual quantization (RQ), quantization dropout (QDropout), and replacing & remasking (RRmask). *V* and *q* are the number of RQ and QDropout ratio, respectively. MPJPE is measured in millimeters.

## 4.2. Component Analysis

In Table 2, we comprehensively evaluate the impact of different design components in MoMask through various comparisons, showcasing the performance in both motion reconstruction and generation. Initially, we compare our approach with previous VQ-based motion generation methods [16, 23, 49] on the HumanML3D and KIT-ML datasets. Notably, M2DM [23] incorporates orthogonality constraints

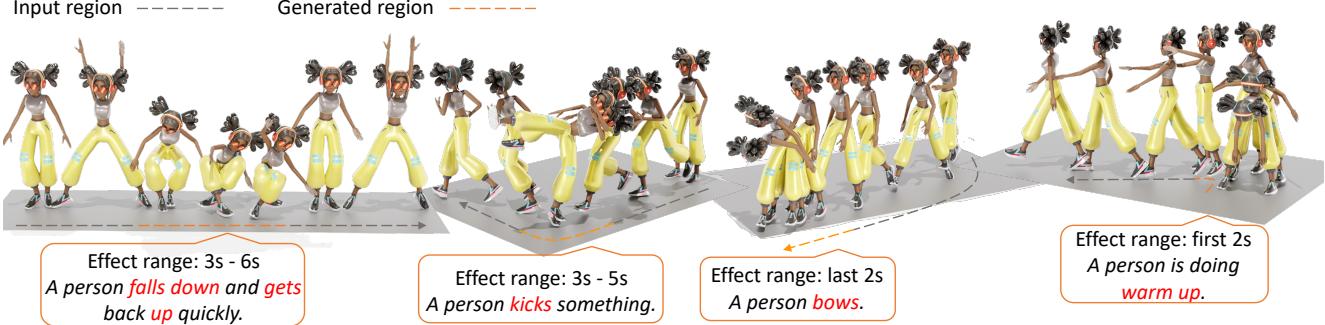


Figure 6. **Examples of temporal inpainting.** Dark dash line indicates the range(s) where the motion content(s) is given by the reference sequence. Orange dash line indicates the range of motion content generated by MoMask, conditioned on the text prompt below.

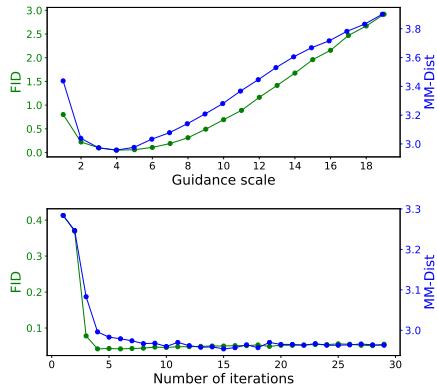


Figure 7. Evaluation sweep over guidance scale  $s$  (top) and iteration numbers  $L$  (bottom) in inference. We find a accuracy-fidelity sweep spot around  $s = 4$ , meanwhile 10 iterations ( $L = 10$ ) for masked decoding yield sufficiently good results.

among all codebook entries to enhance VQ performance. Our residual design shows clearly superior performance when comparing with these single VQ-based approaches.

**Ablation.** In the ablation experiments, we observe that both residual quantization (RQ) and quantization dropout (QDropout) effectively contribute to the enhancement of motion quality in terms of both reconstruction and generation. Additionally, replacing-and-resmasking strategy, as well as RQ, facilitates more faithful motion generation.

**Number of Residual Layers ( $V$ ).** In Tab. 2, we investigate RVQ with different numbers of quantization layers. Generally, more residual VQ layers result in more precise reconstruction, but they also increase the burden on the R-Transformer for residual token generation. We particularly observe that the generation performance starts to degrade with more than 5 residual layers. This finding emphasizes the importance of striking a balance in the number of residual layers for optimal performance.

**Quantization Dropout ( $q$ ).** We also analyze the impact of quantization dropout ratio  $q$  in Tab. 2. As we increase

dropout probability from 0.2, the performance gains become marginal, or even converse. We speculate that frequent disabling quantization layers may disturb the learning of quantization models.

**Inference Hyper-parameters.** The CFG scale  $s$  and the number of iterations  $L$  are two crucial hyperparameters during the inference of masked modeling. In Fig. 7, we present the performance curves of FID and multimodality distance by sweeping over different values of  $s$  and  $L$ . Several key observations emerge. Firstly, an optimal guidance scale  $s$  for M-Transformer inference is identified around  $s = 4$ . Over-guided decoding may even inversely deteriorate the performance. Secondly, more iterations are not necessarily better. As  $L$  increases, the FID and multimodality distance converge to the minima quickly, typically within around 10 iterations. Beyond 10 iterations, there are no further performance gains in both FID and multimodal distance. In this regard, our MoMask requires fewer inference steps compared to most autoregressive and diffusion models.

### 4.3. Application: Temporal Inpainting

In Fig. 6, we showcase the capability of MoMask in temporally inpainting a specific region in a motion sequence. The region can be freely located in the middle, suffix, or prefix. Specifically, we mask out all the tokens in the region of interest and then follow the same inference procedure described in Sec. 3.4. For both tasks, our approach generates smooth motions in coherence with the given text descriptions. Additionally, we conduct a user study to quantitatively compare our inpainting results with those of MDM [42]. In this study, 40 samples are generated from both methods using the same motion and text input, and presented to users side-by-side. With 6 users involved, 68% of the results from MoMask are preferred over MDM.

## 5. Discussion and Conclusion

**Limitations.** We acknowledge certain limitations of MoMask. Firstly, while MoMask excels in fidelity and faith-

fulness for text-to-motion synthesis, its diversity is relatively limited. We plan to delve into the underlying causes of this limitation in future work. Secondly, MoMask requires the target length as input for motion generation. This could be properly addressed by applying the text2length sampling [15] beforehand. Thirdly, akin to most VQ-based methods, MoMask may face challenges when generating motions with fast-changing root motions, such as *spinning*. Exemplar cases are presented in the supplementary videos.

In conclusion, we introduce MoMask, a novel generative masked modeling framework for text-driven 3D human motion generation. MoMask features three advanced techniques: residual quantization for precise motion quantization, masked transformer and residual transformer for high-quality and faithful motion generation. MoMask is efficient and flexible, achieving superior performance without extra inference burden, and effortlessly supporting temporal motion inpainting in multiple contexts.

## References

- [1] Chaitanya Ahuja and Louis-Philippe Morency. Language2pose: Natural language grounded pose forecasting. In *2019 International Conference on 3D Vision (3DV)*, pages 719–728. IEEE, 2019. [2](#)
- [2] Andreas Aristidou, Daniel Cohen-Or, Jessica K Hodgins, Yiorgos Chrysanthou, and Ariel Shamir. Deep motifs and motion signatures. *ACM Transactions on Graphics (TOG)*, 37(6):1–13, 2018. [3](#)
- [3] Nikos Athanasiou, Mathis Petrovich, Michael J Black, and GÜl Varol. Teach: Temporal action composition for 3d humans. In *2022 International Conference on 3D Vision (3DV)*, pages 414–423. IEEE, 2022. [2](#)
- [4] Zalán Borsos, Raphaël Marinier, Damien Vincent, Eugene Kharitonov, Olivier Pietquin, Matt Sharifi, Dominik Roblek, Olivier Teboul, David Grangier, Marco Tagliasacchi, et al. Audiolm: a language modeling approach to audio generation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2023. [2, 3](#)
- [5] Haoye Cai, Chunyan Bai, Yu-Wing Tai, and Chi-Keung Tang. Deep video generation, prediction and completion of human action sequences. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 366–382, 2018. [2](#)
- [6] Pablo Cervantes, Yusuke Sekikawa, Ikuro Sato, and Koichi Shinoda. Implicit neural representations for variable length human motion generation. In *European Conference on Computer Vision*, pages 356–372. Springer, 2022. [2](#)
- [7] Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T Freeman. Maskgit: Masked generative image transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11315–11325, 2022. [2, 4](#)
- [8] Huiwen Chang, Han Zhang, Jarred Barber, AJ Maschinot, Jose Lezama, Lu Jiang, Ming-Hsuan Yang, Kevin Murphy, William T Freeman, Michael Rubinstein, et al. Muse: Text-to-image generation via masked generative transformers. *arXiv preprint arXiv:2301.00704*, 2023. [2, 4, 5](#)
- [9] Xin Chen, Biao Jiang, Wen Liu, Zilong Huang, Bin Fu, Tao Chen, and Gang Yu. Executing your commands via motion diffusion in latent space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18000–18010, 2023. [1, 2, 5, 6, 7](#)
- [10] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. [2, 4](#)
- [11] Sohrab Ferdowsi, Slava Voloshynovskiy, and Dimche Kostadinov. Regularized residual quantization: a multi-layer sparse dictionary learning approach. *arXiv preprint arXiv:1705.00522*, 2017. [3](#)
- [12] Anindita Ghosh, Noshaba Cheema, Cennet Oguz, Christian Theobalt, and Philipp Slusallek. Synthesis of compositional animations from textual descriptions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1396–1406, 2021. [2](#)
- [13] Kehong Gong, Dongze Lian, Heng Chang, Chuan Guo, Xinxin Zuo, Zihang Jiang, and Xinchao Wang. Tm2d: Bi-modality driven 3d dance generation via music-text integration. *arXiv preprint arXiv:2304.02419*, 2023. [1, 2](#)
- [14] Chuan Guo, Xinxin Zuo, Sen Wang, Shihao Zou, Qingyao Sun, Annan Deng, Minglun Gong, and Li Cheng. Action2motion: Conditioned generation of 3d human motions. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 2021–2029, 2020. [2, 5](#)
- [15] Chuan Guo, Shihao Zou, Xinxin Zuo, Sen Wang, Wei Ji, Xingyu Li, and Li Cheng. Generating diverse and natural 3d human motions from text. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5152–5161, 2022. [1, 2, 5, 6, 9](#)
- [16] Chuan Guo, Xinxin Zuo, Sen Wang, and Li Cheng. Tm2t: Stochastic and tokenized modeling for the reciprocal generation of 3d human motions and texts. In *European Conference on Computer Vision*, pages 580–597. Springer, 2022. [1, 2, 3, 5, 6, 7](#)
- [17] Chuan Guo, Xinxin Zuo, Sen Wang, Xinshuang Liu, Shihao Zou, Minglun Gong, and Li Cheng. Action2video: Generating videos of human 3d actions. *International Journal of Computer Vision*, 130(2):285–315, 2022. [2](#)
- [18] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16000–16009, 2022. [2](#)
- [19] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. [5](#)
- [20] Ruozhi Huang, Huang Hu, Wei Wu, Kei Sawada, Mi Zhang, and Dixin Jiang. Dance revolution: Long-term dance generation with music via curriculum learning. *arXiv preprint arXiv:2006.06119*, 2020. [2](#)
- [21] Biao Jiang, Xin Chen, Wen Liu, Jingyi Yu, Gang Yu, and Tao Chen. Motiongpt: Human motion as a foreign language. *arXiv preprint arXiv:2306.14795*, 2023. [1, 2, 3](#)

- [22] Jihoon Kim, Jiseob Kim, and Sungjoon Choi. Flame: Free-form language-based motion synthesis & editing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 8255–8263, 2023. 2
- [23] Hanyang Kong, Kehong Gong, Dongze Lian, Michael Bi Mi, and Xinchao Wang. Priority-centric human motion generation in discrete latent space. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14806–14816, 2023. 1, 2, 7
- [24] Tianhong Li, Huiwen Chang, Shlok Mishra, Han Zhang, Dina Katabi, and Dilip Krishnan. Mage: Masked generative encoder to unify representation learning and image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2142–2152, 2023. 2
- [25] Yue Li, Wenrui Ding, Chunlei Liu, Baochang Zhang, and Guodong Guo. Trq: Ternary neural networks with residual quantization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 8538–8546, 2021. 3
- [26] Zefan Li, Bingbing Ni, Wenjun Zhang, Xiaokang Yang, and Wen Gao. Performance guaranteed network acceleration via high-order residual quantization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2584–2592, 2017. 3
- [27] Angela S Lin, Lemeng Wu, Rodolfo Corona, Kevin Tai, Qixing Huang, and Raymond J Mooney. Generating animated videos of human activities from natural language descriptions. *Learning*, 2018(1), 2018. 2
- [28] Junfan Lin, Jianlong Chang, Lingbo Liu, Guanbin Li, Liang Lin, Qi Tian, and Chang-wen Chen. Being comes from not-being: Open-vocabulary text-to-motion generation with wordless training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23222–23231, 2023. 2
- [29] Zhenguang Liu, Shuang Wu, Shuyuan Jin, Shouling Ji, Qi Liu, Shijian Lu, and Li Cheng. Investigating pose representations and motion contexts modeling for 3d motion prediction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(1):681–697, 2022. 2
- [30] Yunhong Lou, Linchao Zhu, Yaxiong Wang, Xiaohan Wang, and Yi Yang. Diversemotion: Towards diverse human motion generation via discrete diffusion. *arXiv preprint arXiv:2309.01372*, 2023. 2
- [31] Thomas Lucas, Fabien Baradel, Philippe Weinzaepfel, and Grégory Rogez. Posegpt: Quantization-based 3d human motion generation and forecasting. In *European Conference on Computer Vision*, pages 417–435. Springer, 2022. 2
- [32] Naureen Mahmood, Nima Ghorbani, Nikolaus F Troje, Gerard Pons-Moll, and Michael J Black. Amass: Archive of motion capture as surface shapes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5442–5451, 2019. 5
- [33] Wei Mao, Miaomiao Liu, Mathieu Salzmann, and Hongdong Li. Learning trajectory dependencies for human motion prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9489–9497, 2019. 2
- [34] Julieta Martinez, Holger H Hoos, and James J Little. Stacked quantizers for compositional vector compression. *arXiv preprint arXiv:1411.2173*, 2014. 2, 3
- [35] Mathis Petrovich, Michael J Black, and Gü̈l Varol. Action-conditioned 3d human motion synthesis with transformer vae. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10985–10995, 2021. 2
- [36] Mathis Petrovich, Michael J Black, and Gü̈l Varol. Temos: Generating diverse human motions from textual descriptions. In *European Conference on Computer Vision*, pages 480–497. Springer, 2022. 1, 2
- [37] Matthias Plappert, Christian Mandery, and Tamim Asfour. The kit motion-language dataset. *Big data*, 4(4):236–252, 2016. 5
- [38] Matthias Plappert, Christian Mandery, and Tamim Asfour. Learning a bidirectional mapping between human whole-body motion and natural language using deep recurrent neural networks. *Robotics and Autonomous Systems*, 109:13–26, 2018. 2
- [39] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 2, 4
- [40] Li Siyao, Weijiang Yu, Tianpei Gu, Chunze Lin, Quan Wang, Chen Qian, Chen Change Loy, and Ziwei Liu. Bailando: 3d dance generation by actor-critic gpt with choreographic memory. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11050–11059, 2022. 2
- [41] Guy Tevet, Brian Gordon, Amir Hertz, Amit H Bermano, and Daniel Cohen-Or. Motionclip: Exposing human motion generation to clip space. In *European Conference on Computer Vision*, pages 358–374. Springer, 2022. 2
- [42] Guy Tevet, Sigal Raab, Brian Gordon, Yonatan Shafir, Daniel Cohen-Or, and Amit H Bermano. Human motion diffusion model. *arXiv preprint arXiv:2209.14916*, 2022. 1, 2, 5, 6, 7, 8
- [43] Jonathan Tseng, Rodrigo Castellon, and Karen Liu. Edge: Editable dance generation from music. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 448–458, 2023. 2
- [44] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in Neural Information Processing Systems*, 30, 2017. 2, 3, 4
- [45] Chengyi Wang, Sanyuan Chen, Yu Wu, Ziqiang Zhang, Long Zhou, Shujie Liu, Zhuo Chen, Yanqing Liu, Huaming Wang, Jinyu Li, et al. Neural codec language models are zero-shot text to speech synthesizers. *arXiv preprint arXiv:2301.02111*, 2023. 3
- [46] Zhenyi Wang, Ping Yu, Yang Zhao, Ruiyi Zhang, Yufan Zhou, Junsong Yuan, and Changyou Chen. Learning diverse stochastic human-action generators by learning smooth latent transitions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 12281–12288, 2020. 2
- [47] Lijun Yu, Yong Cheng, Kihyuk Sohn, José Lezama, Han Zhang, Huiwen Chang, Alexander G Hauptmann, Ming-Hsuan Yang, Yuan Hao, Irfan Essa, et al. Magvit:

- Masked generative video transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10459–10469, 2023. 2
- [48] Neil Zeghidour, Alejandro Luebs, Ahmed Omran, Jan Skoglund, and Marco Tagliasacchi. Soundstream: An end-to-end neural audio codec. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30:495–507, 2021. 2, 3
- [49] Jianrong Zhang, Yangsong Zhang, Xiaodong Cun, Shaoli Huang, Yong Zhang, Hongwei Zhao, Hongtao Lu, and Xi Shen. T2m-gpt: Generating human motion from textual descriptions with discrete representations. *arXiv preprint arXiv:2301.06052*, 2023. 1, 2, 3, 4, 5, 6, 7
- [50] Mingyuan Zhang, Zhongang Cai, Liang Pan, Fangzhou Hong, Xinying Guo, Lei Yang, and Ziwei Liu. Motiondiffuse: Text-driven human motion generation with diffusion model. *arXiv preprint arXiv:2208.15001*, 2022. 1, 2, 5
- [51] Mingyuan Zhang, Xinying Guo, Liang Pan, Zhongang Cai, Fangzhou Hong, Huirong Li, Lei Yang, and Ziwei Liu. Remodiffuse: Retrieval-augmented motion diffusion model. *arXiv preprint arXiv:2304.01116*, 2023. 5, 6
- [52] Yaqi Zhang, Di Huang, Bin Liu, Shixiang Tang, Yan Lu, Lu Chen, Lei Bai, Qi Chu, Nenghai Yu, and Wanli Ouyang. Motiongpt: Finetuned llms are general-purpose motion generators. *arXiv preprint arXiv:2306.10900*, 2023. 2, 3
- [53] Zixiang Zhou and Baoyuan Wang. Ude: A unified driving engine for human motion generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5632–5641, 2023. 2