Text Residual Motion Encoder for 3D Human Motion Generation

Anonymous

Anonymous Affiliation

Abstract. In the domain of 3D human motion generation from textual descriptions, recent advancements in generating high-quality and diverse motions that accurately reflect nuanced textual inputs are limited primarily due to architectural flexibility and insufficient training data diversity. Addressing these challenges, we improve the generation of human motions from textual descriptions based on the text-to-motion generative pre-trained transformer framework by expanding the architecture and utilizing diverse datasets. We propose a Text Residual Motion Encoder (TRME) with an additional layer of residual block to the foundational architecture of the Vector quantized variational autoencoder. This improvement aims to capture more granular details in motion patterns, significantly increasing the diversity and complexity of generated motions. Simultaneously, we create a new Comprehensive Human Activity Dataset from the AMASS repository covering a wide range of human activities based on user specifications, allowing users to sample motions across different classes for model training. This expansion mitigates the diversity of the dataset through conditioned preprocessing and sampling of the different minority classes, eventually improving the model's reconstruction and generation capabilities of motion sequences. Extensive experimental results demonstrate that the proposed model significantly advances over the existing methods, particularly with diverse training datasets achieving promising results in human motion generation. The source code is available at https://github.com/undisclosed-rav/3DHumanMotion.

1 Introduction

The simulation of human motion from textual descriptions intersects critical domains such as virtual reality, gaming, and digital filmmaking, offering significant improvements over traditional motion capture systems. These traditional systems are often costly and involve complex setups. In recent years, various computational models have been developed to bridge the gap between textual descriptions and the generation of accurate human motions. However, despite progress, these models often struggle due to limited architectural flexibility and a lack of diverse training data, which are crucial for capturing the full spectrum of human motion dynamics.

In response to these challenges, our work introduces a novel architecture, Text Residual Motion Encoder (TRME), which is extended on the architecture of the Vector Quantized Variational Autoencoder (VQ-VAE) [49] mentioned in the text to motion generative pre-trained transformer (T2M-GPT) [52]. The TRME specifically targets enhancements in the model's capability to process and translate complex textual inputs into detailed motion sequences. By embedding an additional layer of residual blocks, the TRME refines the granularity with which motions are encoded and decoded, thus allowing for a more nuanced representation of human movements.

Moreover, the limitations of existing datasets have been a significant barrier to further advancements in this field. To address this, we have curated a new composite dataset that integrates the diverse movement data from MOYO [45, 45] and MM-FIT [41] alongside traditional datasets like HumanML3D [16, 17]. This integration aims to broaden the range of human activities the model can learn from, ensuring a more comprehensive coverage of potential motion types. The approach augments not only the dataset's volume but also its variety, with a specific focus on enhancing representation for minority motion classes through sophisticated preprocessing techniques. This introduction of advanced architectural modifications coupled with an enriched training dataset positions the TRME to advance the state-of-the-art in text-driven motion generation substantially. Our empirical analysis demonstrates the potential of TRME to outperform existing methodologies, particularly in generating more diverse, realistic, and contextually accurate human motions from textual descriptions.

2 Related Work

Foundational Models in Human Motion Synthesis Generative models have significantly advanced human motion synthesis, with pioneering techniques shaping the landscape. Recurrent Neural Networks (RNNs) [35], including bi-directional LSTM networks [4], have laid the groundwork by modeling temporal dynamics essential for capturing sequential data. Graph Convolutional Networks (GCNs) [23] and Generative Adversarial Networks (GANs) [13] have expanded capabilities by introducing methods for generating lifelike and spatially aware motion sequences. These foundational models set the stage for sophisticated techniques like Conditional VAEs (CVAEs) [38], which adapt the generative process to specific conditions, enhancing the diversity and realism of motion synthesis. The integration of natural language processing with motion synthesis has seen remarkable developments, beginning with systems like Text2Action [1] and Language2Pose [2], which convert textual descriptions into motion data. Innovations such as MotionCLIP [43], leveraging the capabilities of CLIP [33], have further enhanced text-to-motion alignment. Transformer-based architectures like ACTOR [30] and TEMOS [31] have advanced the generation of complex motion sequences from detailed textual narratives, demonstrating deep learning's growing influence in creative content generation. Further, the area of translating textual descriptions into human motions has shifted towards a stochastic generation of variable-length sequences, enabling the mapping of one text description to

multiple motion outputs [5]. This flexibility significantly enhances the realism of generated motions. Additionally, video synthesis techniques incorporate deep generative models like GANs and VAEs, with studies exploring recurrent GANs to differentiate between stationary and moving parts of images [7, 6], and attention mechanisms to improve alignment and contextual relevance in text-to-video generation and video captioning. These advancements provide a broader trend toward complex multimodal inputs to generate progressively detailed animations. The evolution of these technologies reflects a deeper understanding of human movements and the potential for generative models to significantly improve digital interactions and virtual realities.

Diffusion Models Recent work on diffusion models has gained prominence in the generative model landscape [44], characterized by their ability to transform a noise distribution into a complex data distribution through controlled, step-wise refinement. These models operate by gradually converting a simple noise distribution into a complex data distribution through a series of learned reverse diffusion steps, effectively inverting the diffusion process initially described by Sohl-Dickstein et al. [37]. This capability makes them particularly well-suited for detailed and nuanced tasks such as human motion synthesis. In the context of human motions, pioneering studies on diffusion models like Ho et al. [20] and Nichol et al. [28] have demonstrated the efficacy of diffusion models in generating high-fidelity animations. These models employ a conditioning mechanism that intricately shapes the motion by iterating toward a less noisy and more defined state. This approach allows the model to capture the subtle nuances of human movement, resulting in animations that are not only realistic but also dynamically consistent [39]. Furthermore, Song and Ermon [40] explored the extension of diffusion processes to include conditional generation, enhancing the model's ability to tailor outputs to specific inputs such as textual descriptions or control signals. The flexibility of diffusion models in handling complex conditioning information has led to their increased adoption in other areas of generative modeling as well. For instance, their use in tasks like image synthesis [10] and audio generation [9] parallels their application in motion synthesis, where the key to success lies in the model's ability to iteratively refine its outputs to match the high standards required by contemporary applications. Building on this foundation, diffusion models like Motion Diffuse [53], and MDM [44] have set new benchmarks in the domain by integrating advanced conditioning techniques. These techniques leverage additional contextual data to guide the synthesis process, ensuring that the generated motions are not only visually appealing but also precisely align the intended motion dynamics with the input texts.

Comprehensive Motion Datasets The efficacy of generative models in human motion synthesis largely depends on the diversity and quality of the training datasets utilized. Initial datasets such as the Carnegie Mellon University Motion Capture Database (CMU MoCap) [8] and the Human3.6M dataset [22] set the foundation by providing extensive motion capture data that has been pivotal for training predictive models. Building upon these, the AMASS dataset [25] offers a comprehensive aggregation of multiple motion capture sources. In addition, the KIT Motion-Language Dataset [32] introduced linguistic descriptions aligned with motion sequences, significantly enhancing the applicability of models in text-to-motion tasks by facilitating the training of models that understand nuanced human activities in a contextual manner.

The ACCAD dataset [29] offers a focused collection of athletic motions, capturing the dynamic and precise movements of trained

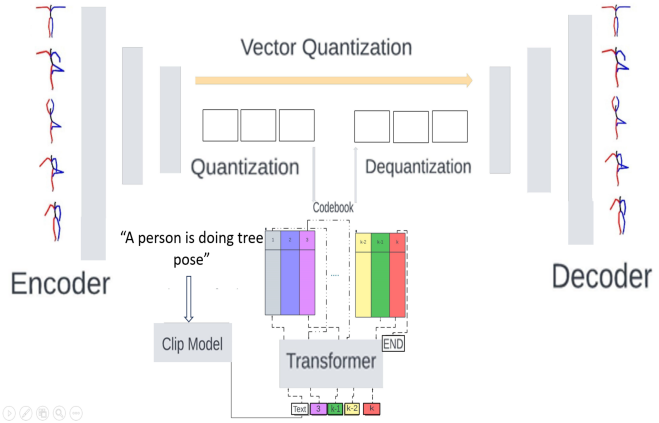

Figure 1: *Illustration of our TRME architecture for text-driven motion generation. The architecture, inspired by recent advancements in neural discrete representation learning, combines convolutional layers, residual blocks, and strategic downsampling and upsampling mechanisms to encode and decode motion sequences efficiently. The model learns to map input motion sequences X to discrete indices S using a learned codebook, enabling high-fidelity representation of diverse motion patterns. The discrete indices are then fed into an autoregressive decoder to reconstruct the motion sequences. During inference, the model generates code indexes in an autoregressive fashion, producing coherent and contextually aligned motion sequences based on textual descriptions.*

individuals. The HDM05 dataset [27], renowned for its structured curation of human dynamics, complements this by providing a broad range of standard motions meticulously documented. In tandem, the TCDHands dataset [21] contributes granular hand and finger motion data, an essential aspect of nuanced gesture synthesis. Further augmenting the scope of standard movements, the SFU dataset [12] introduces a plethora of daily and athletic activities. The BML-movi [11] and BMLrub [46] datasets extend this landscape with expressive locomotion and object manipulation sequences. Technical datasets such as MoSh [24], EKUT [26], and TotalCapture [47] provide advanced motion data that help in understanding the subtleties of human postures under varying conditions. For sports and handball movements, BMLhandball [18] [19] offers a unique dataset that captures the quick and intricate motions specific to the sport. The Transitions dataset [25] contributes to the study of movement transitions, offering insights into how humans naturally progress from one action to another. Similarly, PosePrior [3], and HumanEva [36] contribute to the understanding of pose estimation and are instrumental in refining the generation of human figures in motion. Further developments led to the creation of more specialized and diverse datasets like MOYO [45] and MM-FIT [41], which offer a broader range of human activities, from everyday actions to complex exercise routines like Yoga. These datasets have been instrumental in pushing the boundaries of what generative models can achieve, enabling them to produce animations that are not only realistic but also contextually accurate. The HumanML3D dataset [17] provides a comprehensive dataset that combines the diversity of actions and the depth of annotations necessary for developing advanced motion synthesis models. Our methodology specifically leverages these rich datasets to enhance the generative capabilities of our models, ensuring that the motions generated are not just diverse but truly reflective of the complexities of human movements.

Our proposed TRME model architecture consists of two main components: the Motion VQ-VAE [49] and the Text-to-Motion Generative Pre-trained Transformer (T2M-GPT) [52]. The Motion VQ-VAE is designed to learn a complex mapping between motion sequences and discrete codebook vectors within a latent space, effectively capturing the intricate dynamics of human movements. We build upon this rich representation of features by introducing the Text Residual Motion Encoder (TRME), a model designed to navigate the nuanced space between textual narratives and the subtleties of human movement. Our methodology converts motion data into motion snippet codes, which are reconstructed into motion sequences during training. This process involves a text encoder paired with VAE networks [51], employing a triad of prior, posterior, and generator networks to ensure high-quality pose reconstruction. During inference, motion is generated from the text by determining the desired motion length and processing text features to produce motion snippet codes, culminating in a dynamic pose sequence [30]. In terms of data resources, the KIT Motion-Language Dataset [32] remains a crucial but somewhat limited resource due to its focus on locomotion.

Table 1: Details of our created CHAD Dataset.

Dataset	Subjects	Motions	Minutes
ACCAD	20	252	26.74
HDM05 (MPI_HDM05)	4	215	144.54
TCD (TCD_handMocap)	1	62	8.37
SFU	7	44	15.23
BMLmovi	89	1864	174.39
CMU	96	1983	543.49
Mosh (MPI_mosh)	19	77	16.53
EKUT	4	349	30.74
KIT	55	4232	661.84
EyesJapanDataset	12	750	397.04
BMLhandball	10	649	101.98
Transitions (Transitions_mocap)	1	110	15.10
PosePrior (MPI_Limits)	3	35	20.82
HumanEva	3	28	8.47
SSM (SSM_synced)	3	30	1.87
DFaust (DFaust_67)	10	139	5.72
TotalCapture	5	37	41.10
BMLrub (BioMotionLab_NTroje)	111	3061	522.69
HumanML3D	453	10901	2736.34
DanceDB	20	173	203.38
MOYO	1	181	-
MM-FIT	1	342	800
CNRS	2	79	9.61
GRAB	10	1340	226.59

3 Dataset Creation

The expansion of the Comprehensive Human Activity Dataset (CHAD) incorporates new data sources from AMASS, including DanceDB [48], MOYO [45], CNRS [34], and GRAB [42]. This systematic integration aimed to increase the variety of motion sequences available for model training within our research framework. To meet the diverse needs of researchers and practitioners, we adopted a user-friendly approach, giving users the flexibility to customize their data selection. This approach enables individuals to define the relative importance of various motion classes, enabling them to generate datasets that align with specific research goals and application domains. We began by categorizing the motions in the dataset into specific labels for each data source and implemented a sampling algorithm that assigns class weightage based on the number of existing classes.

The GUI in Figure 2 allows users to sample data according to their preferences, offering a unique method for creating custom datasets. Users can enter the desired number of motion files from each class to create their tailored dataset. After data sampling, the dataset undergoes normalization and is partitioned into training, testing, and validation subsets. This segmentation follows a standard split configuration, allocating 70% of the data for model training, 20% for testing model performance, and 10% for validation to ensure robustness across various evaluation scenarios.

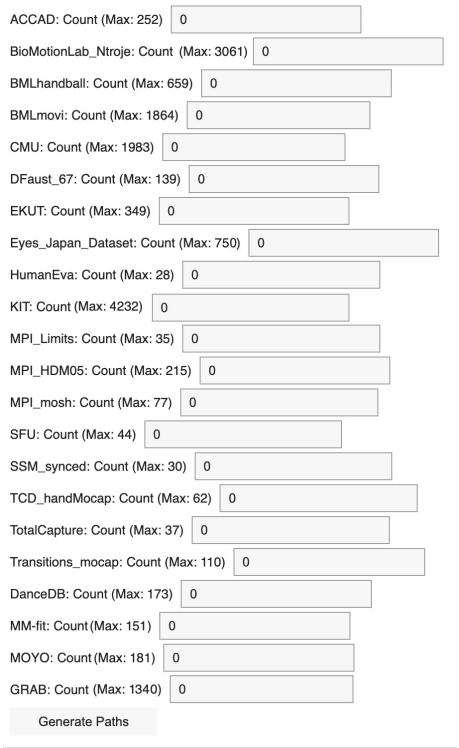

Figure 2: Sampling GUI: Users can sample custom datasets according to specific use cases.

3.1 Data Augmentation

Our primary focus involves expanding the HumanML3D [16] dataset, which serves as a foundational collection of motion data for our study. In our dataset CHAD, augmentation includes the integration of additional sequences from the MOYO [45] dataset, published by AMASS, to address class imbalances and enrich the dataset's representational capacity.

3.1.1 Addressing Class Imbalance and Data Augmentation Strategy

Class imbalance, particularly prevalent in yoga poses within the MOYO [45] dataset, poses significant challenges to training effective motion generation models. To address this issue, we developed a novel data augmentation strategy that leverages the inherent characteristics of yoga movements.

Motion Sequence Analysis We observed that yoga poses of ten begin and end in a standing position, with the practitioner typically returning to the initial stance. This retracing pattern occurs in approximately 95% of the sequences observed.

Normalization and Augmentation Technique We designed a data augmentation technique to enhance the dataset's representational balance and normalized the motion data by appending the starting sequence in reverse order, following the core yoga pose. This approach simulates the natural retracing motion to the original standing position, thereby enhancing the realism and continuity of the motion sequence. Mathematically, the augmentation process can be defined as follows.

Let $S = (s_1, s_2, ..., s_n)$ represent the starting sequence, where s_i denotes the i-th pose in the sequence. Let C denote the core yoga position sequence. The reverse of S, denoted as R, is given by $R = (s_n, s_{n-1}, ..., s_1)$. The augmented sequence A can be formulated by concatenating S, C, and R:

$$A = S \oplus C \oplus R \tag{1}$$

where $\oplus$ denotes the concatenation operation. This augmentation effectively doubles the number of sequences available for each yoga pose, enhancing the dataset's diversity and providing a more balanced representation of motion classes.

Uniform Skeleton Conversion Different datasets might have varying skeleton structures and joint offsets. This process ensures consistency by aligning the joint offsets of all source data to a common target skeleton. Forward kinematics is then employed to obtain the corresponding joint positions, guaranteeing a uniform representation across the data. In essence, this augmentation technique standardizes the skeletal representation of the motion data from various sources, facilitating consistent processing and analysis.

Integration with Extended Feature Sets In preparation for future studies, we also considered integrating additional datasets that contain different numbers of joints compared to HumanML3D [16]. These datasets have been preprocessed to align with the existing framework and will be included after further augmentation, enhancing the HumanML3D dataset with extended feature motion representations.

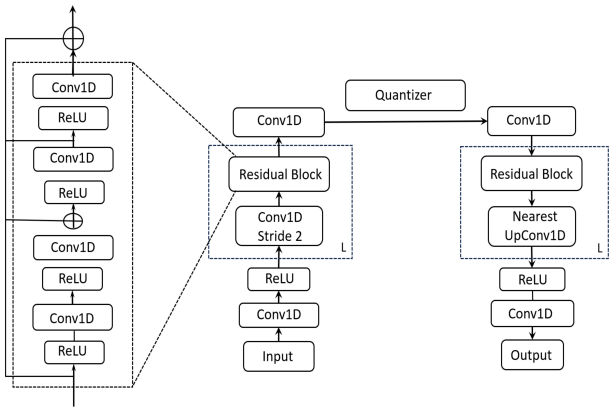

Figure 3: *Block diagram of the TRME architecture. The architecture consists of an encoder and a decoder. The encoder uses Conv1D to extract features from the input motion sequence. Deep Residual blocks are used to help the network learn long-range dependencies in the sequence. Stride 2 downsampling is used to reduce the dimensionality of the data, which helps capture higher-level features. Quantization (Quantizer) is performed to map the continuous latent representation to a discrete code. The decoder uses nearest neighbor upsampling (Nearest L UpConv1D) and Conv1D layers to reconstruct the motion sequence from the discrete latent code.*

4 Methods

4.1 Vector Quantized Variational Autoencoder (VQ-VAE)

Introduced by van den Oord et al. [49], the VQ-VAE enables learning of discrete representations crucial for handling high-dimensional data like motion sequences. Consider a motion sequence $X = [x_1, x_2, ..., x_T]$ where x_t represents the motion frame in $\mathbb{R}^d$, T denotes the number of frames, and d is the dimensionality of each frame.

Model Architecture and Quantization Process The VQ-VAE [49] consists of an encoder E and a decoder D, with a learnable codebook $C = \{c_k\}_{k=1}^{K}$, where each $c_k \in \mathbb{R}^{d_c}$ represents a code vector in the discrete latent space

$$Z = E(X) \quad \text{where} \quad Z = [z_1, z_2, ..., z_{T/l}] \quad \text{and} \quad z_i \in \mathbb{R}^{d_c}. \tag{2}$$

Here, l indicates the temporal downsampling rate of the encoder, effectively reducing the temporal dimension from T to T/l. The quantization step maps each latent feature z_i to the nearest code c_k in the codebook:

$$\hat{z}_i = \underset{k}{\operatorname{argmin}} \|z_i - c_k\|^2. \tag{3}$$

4.2 Optimization and Loss Functions

The training of TRME involves multiple loss components, each designed to optimize a specific aspect of the model:

- **Reconstruction Loss ($\mathcal{L}_{re}$):** Measures the fidelity of the reconstructed motion sequence compared to the original, employing the L_1 smooth loss:

$$\mathcal{L}_{re} = \sum_t \|x_t - D(\hat{z}_t)\|_1. \tag{4}$$

  This loss helps in minimizing the discrepancies between the original and reconstructed sequences, ensuring high-quality motion generation.
- **Embedding Loss ($\mathcal{L}_{embed}$):** Ensures that the latent embeddings align closely with the nearest codebook vectors:

$$\mathcal{L}_{embed} = \|\text{sg}[Z] - \hat{Z}\|_2^2. \tag{5}$$

  Here, sg represents the stop-gradient operator, which prevents gradients from flowing into the encoder during the optimization of the embedding loss.
- **Commitment Loss ($\mathcal{L}_{commit}$):** Encourages consistency between the encoder outputs and the chosen codebook vectors, penalizing large deviations:

$$\mathcal{L}_{commit} = \beta\|Z - \text{sg}[\hat{Z}]\|_2^2. \tag{6}$$

  The hyper-parameter β controls the weighting of the commitment loss, balancing the influence of this term in the overall optimization process.

Overall Loss Function The composite loss function for training the TRME integrates these components

$$\mathcal{L}_{vq} = \mathcal{L}_{re} + \mathcal{L}_{embed} + \mathcal{L}_{commit}. \tag{7}$$

This formulation ensures a robust learning process, promoting the generation of coherent and diverse motion sequences from textual descriptions.

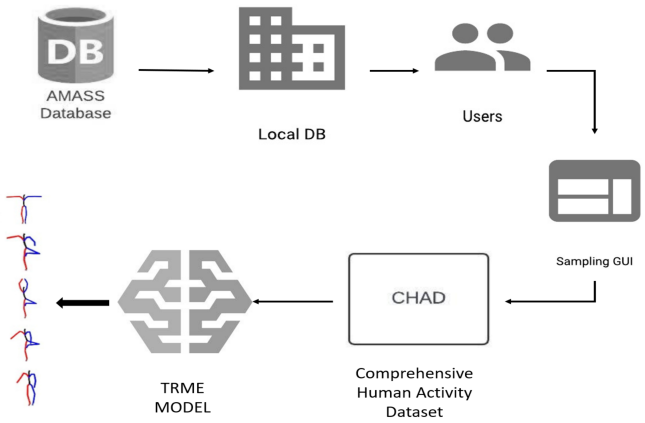

Figure 4: *Data flow diagram for the TRME model, highlighting the progression from the AMASS database to the creation of the CHAD dataset and subsequent motion generation.*

4.3 Quantization Strategy

Training a TRME effectively involves addressing potential issues such as codebook collapse, where certain codes in the codebook become redundant and are rarely or never used. This phenomenon, often observed in naive implementations, leads to inefficient learning and underutilization of the model's capacity. To mitigate this, two prominent strategies are employed: the exponential moving average (EMA) and codebook reset.

4.3.1 Exponential Moving Average (EMA)

The EMA approach ensures that updates to the codebook are made gradually, allowing the code vectors to evolve smoothly over time. This method helps prevent abrupt changes that could destabilize the learning process. Mathematically, the update rule for the codebook using EMA can be expressed as:

$$C_t \leftarrow \lambda C_{t-1} + (1-\lambda)C_t, \tag{8}$$

where C_t is the codebook at iteration t, λ is the decay parameter (exponential moving constant) that determines the rate at which older observations are dampened. A higher value of λ places more emphasis on past values, leading to smoother updates.

4.3.2 Codebook Reset

Codebook reset tackles the problem of inactive codes by periodically scanning the codebook for unused vectors. During this process, any code that is found to be inactive is reassigned to a value based on the current input data. This dynamic reallocation helps maintain the diversity of the codebook and ensures all codes have the potential to contribute to the model's learning. The codebook reset can be particularly effective in scenarios where the data distribution exhibits significant changes over the training period.

4.4 TRME Architecture

The architecture of our TRME model is thoughtfully designed to efficiently capture the temporal dynamics of motion sequences using a convolutional approach. It integrates several key components optimized for handling 1D sequential data, specifically tailored for motion data processing.

Convolutional Layers At the core of our TRME architecture are 1D convolutional layers. These layers are instrumental in extracting local and temporal features from the sequential motion data. We employ convolutions with a stride of 2, which serves the dual purpose of reducing the dimensionality of the input data and capturing longer-range dependencies between the motion frames. This striding mechanism effectively doubles the receptive field with each subsequent layer, allowing the network to integrate information over increasingly larger temporal windows.

Residual Blocks Following the initial convolutional layers, our architecture incorporates multiple residual blocks, as suggested in Figure 3. Each residual block consists of four convolutional layers with ReLU activations in between and a skip connection that adds the input of the block to its output this is implemented 2 times for each pair of 2 convolution and relu blocks. This design helps in alleviating the vanishing gradient problem by allowing gradients to flow through the skip connections, making the network easier to train and enabling deeper architectures. The number of these blocks, denoted by L, directly influences the depth of the network and the abstraction level of the features extracted.

ReLU Activation ReLU (Rectified Linear Unit) activation functions are used throughout the network following each convolutional layer (excluding the last layer in each residual block where the skip connection is applied). ReLU helps introduce non-linearities into the model, which is crucial for learning more complex patterns in the data without significantly increasing the computational burden.

Temporal Downsampling and Upsampling As mentioned previously for downsampling, we use convolutional layers with stride 2 combined with nearest interpolation. This approach reduces the temporal resolution of the motion sequence, condensing the information and reducing the computational load for subsequent layers. The downsampling rate is defined as $l = 2^L$, where L is the number of residual blocks. This exponential relationship highlights the rapid increase in receptive field and decrease in temporal resolution as we progress deeper into the network. In the upsampling phase, nearest neighbor interpolation is employed to restore the temporal resolution of the motion sequences to their original size.

4.4.1 Overall Architecture Illustration

Our TRME architecture, inspired by [52], is depicted in Figure 1. The combination of convolutional layers, residual blocks, and strategic downsampling and upsampling mechanisms ensures that our model efficiently learns to encode and decode motion sequences, optimizing both performance and computational efficiency. This design ensures that our TRME is not only robust in handling diverse motion data but also scalable and efficient in terms of training and inference, making it suitable for a wide range of applications in motion analysis and synthesis.

4.4.2 Motion Sequence Encoding and Decoding with TRME

With a proficiently trained motion TRME, a motion sequence $X = [x_1, x_2, \ldots, x_T]$ can be effectively transformed into a sequence of discrete indices $S = [s_1, s_2, \ldots, s_{T/l}, \text{End}]$, where each s_i represents an index from the learned codebook. The indices correspond to the most representative codeword in the latent space for segments of the input motion sequence. Notably, we append a special *End* token to the sequence to denote the termination of the motion. This design choice diverges from approaches like [15], which utilize

an additional module to predict the length of the motion sequence, thereby simplifying the model architecture and reducing potential error sources related to motion length estimation.

Projecting S back to their corresponding codebook entries yields $\hat{Z} = [\hat{z}_1, \hat{z}_2, \ldots, \hat{z}_{T/l}]$, with $\hat{z}_i = c_{s_i}$. These quantized vectors are then fed into the decoder D, reconstructing the motion sequence X_{re}. This process ensures that the motion data, although compressed and encoded through discrete representation, retains essential dynamic characteristics when reconstructed.

Text-to-Motion: Autoregressive Prediction The task of generating motion from textual descriptions is formulated as an autoregressive prediction of the next index in the sequence of codebook indices. Specifically, given the preceding indices up to index $i-1$, denoted as $S_{<i}$, and a textual condition c, the model aims to predict the distribution of potential next indices $p(S_i|c, S_{<i})$. This prediction task is managed using a transformer model [50], which excels in handling sequential data due to its attention mechanisms that consider the entire sequence context. The transformer processes the concatenated input of textual conditions and previously generated indices, learning to infer the next index that best continues the motion sequence according to the textual description.

Figure Overview: The structure and workflow of our transformer-based text-to-motion model are illustrated in Figure 4. This visualization highlights the interplay between the text inputs, the transformer's predictive capabilities, and the final motion output through the TRME decoder, providing a comprehensive view of the generative process. These enhancements to our model architecture not only improve the fidelity and variety of generated motions but also streamline the training and inference processes.

4.5 Optimization Strategy and Causal Self-Attention Mechanism

Our model aims to enhance text-to-motion synthesis by directly maximizing the log-likelihood of the data distribution under a transformer architecture. This is achieved by optimizing the following objective:

$$\mathcal{L}_{trans} = \mathbb{E}_{S\sim p(S)}[-\log p(S|c)], \tag{9}$$

where $p(S|c) = \prod_{i=1}^{|S|} p(S_i|c, S_{<i})$ represents the likelihood of the sequence S given the text context c. This formulation stresses the importance of each token's dependency within the sequence, governed by its preceding tokens, thus facilitating the generation of coherent and contextually accurate motion sequences.

Text Embedding Extraction with CLIP To extract robust and contextually enriched text embeddings c, we leverage the capabilities of CLIP [43], a model known for its effectiveness in various multimedia tasks. CLIP's dual-modality approach, which concurrently trains on text and images, allows it to generate embeddings that are highly adaptable and semantically rich, making it ideal for our task where textual descriptions directly influence motion synthesis.

Implementation of Causal Self-Attention The core mechanism enabling our model's autoregressive nature is causal self-attention [50], which is implemented to ensure that the prediction for each index in the sequence is contingent only on the preceding indices. The computation of the causal self-attention is detailed as follows:

$$\text{Attention} = \text{Softmax}\left(\frac{QK^T \times \text{mask}}{\sqrt{d_k}}\right), \tag{10}$$

where Q and K are the query and key matrices respectively, both in $\mathbb{R}^{T\times d_k}$, where T is the sequence length and d_k is the dimensionality of the keys and queries. The *mask* is a triangular matrix where the value is $-\infty$ for positions $i < j$ to prevent future positions from influencing the current position's output, effectively enforcing the autoregressive property.

Motion Generation and Diversity At the inference stage, the model initiates sequence generation with text embeddings and proceeds autoregressively, crafting indices until the 'End' token is predicted, indicating a complete motion sequence. The model's predictive capabilities allow it to draw from the transformer's output, ensuring a rich variety of motion sequences that are in tune with the specific context of the given text. This balance of accuracy and diversity not only increases the model's relevance but also broadens its practical use in fields such as animation and virtual reality. The efficiency of this approach sheds light on the model's intricate architecture, which effectively marries text interpretation with motion synthesis, illustrating the nuanced interplay of its internal mechanisms.

5 Evaluation and Results

This section presents a detailed comparison of the performance of TRME (Ours) compared to the baseline T2M-GPT and MDM architectures on different datasets, primarily focusing on metrics such as Fréchet Inception Distance (FID), Diversity, R-Precision (Top 3), and MM-Dist. These metrics collectively evaluate the quality, variability, relevance, and distance metrics of the generated motions compared to real human movements.

5.1 Performance Metrics

- **FID (Fréchet Inception Distance)** ↓: Measures dissimilarity between the generated and ground truth distributions. Lower values indicate that the distributions of generated motions are closer to the real data distributions, implying higher quality.

$$FID(x, g) = ||\mu_x - \mu_g||^2 + Tr(\Sigma_x + \Sigma_g - 2(\Sigma_x\Sigma_g)^{1/2}) \tag{11}$$

  where μ_x, Σ_x are the mean and covariance of the real data, and μ_g, Σ_g are those of the generated data.
- **Diversity** ↑: Higher values suggest greater variability in the generated motions, which is crucial for the model's ability to produce a range of different movements.

$$Diversity(g) = \frac{1}{N(N-1)} \sum_{i=1}^{N} \sum_{j=i+1}^{N} d(g_i, g_j) \tag{12}$$

  where $d(g_i, g_j)$ measures the difference between two generated motions g_i and g_j, and N is the number of generated motions.
- **R-Precision (Top 3)** ↑: This metric measures the relevance of the top three generated motions to the given text, aiming for higher alignment accuracy.
- **MM-Dist (Multimodal Distance)** ↓: Measure the relevancy of the generated motions to the input prompts. Lower values indicate a closer match between the generated motion and the expected motion, suggesting higher accuracy in motion generation.

Table 2: Comparison with the state-of-the-art methods on HumanML3D [16], MOYO [45] & MM-FIT [41] dataset. We compute standard metrics following Guo et al. [16].

Model Name	Dataset	Steps	FID ↓	Diversity ↑	R-Precision ↑ (Top3)	MM-Dist ↓
T2M-GPT [52]	HumanML3D [17]	300K	0.141	9.722	0.775	2.280
MDM[44]	HumanML3D [17]	383K	2.945	7.879	0.353	5.925
T2M [14]	HumanML3D [17]	344 (epochs)	1.073	9.183	0.736	2.113
TRME (Ours)	HumanML3D [17]	100K	0.1372	9.793	0.751	2.275
T2M-GPT [52]	CHAD	100K	0.625	9.235	0.589	2.897
MDM[44]	CHAD	100K	7.235	8.878	0.385	6.536
T2M [14]	CHAD	100 (epochs)	2.073	10.781	0.567	4.582
TRME (Ours)	CHAD	100K	0.170	9.981	0.698	2.452

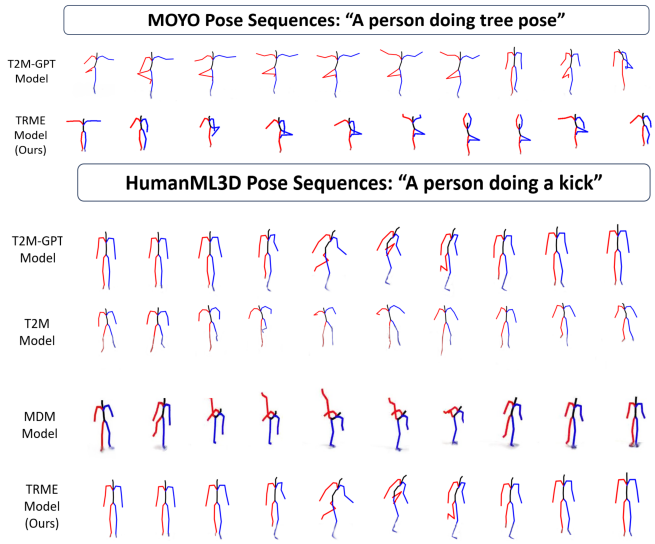

Figure 5: *Generated Sequences for MOYO [45] and HumanML3D [16] from T2M-GPT and TRME. The figure provides detailed visualizations of motion sequences generated by our models. The generated motions correspond to two different captions extracted from the HumanML3D [16] and MOYO [45] datasets. Notably, TRME demonstrates superior performance in capturing dependencies across diverse motion classes compared to state-of-the-art models.*

5.2 Comparative Analysis

Our findings highlight that the TRME model surpasses current leading methods, such as T2M-GPT[52] and MDM[44], in both FID and Diversity metrics on the HumanML3D dataset, signaling enhanced quality and range in the motions it generates. Specifically, with training on the CHAD dataset, our model shows superior performance in capturing a broader spectrum of human activities, reflecting its effectiveness in dealing with complex data. Nevertheless, the broader dataset scope slightly impacts the R-Precision score, indicating that while the model benefits in terms of motion variety, there's room to improve the alignment between the generated motions and their textual descriptions.

5.3 Final Results

In our method for generating 3D human motions from textual descriptions, we performed detailed empirical assessments on the enriched dataset CHAD, which combines datasets from the AMASS database as indicated in Table 2. Our sampling methodology enables users to construct custom datasets tailored to specific use cases. This targeted evaluation approach guarantees that the performance of our model is specifically optimized for the datasets most pertinent to our study, offering a precise evaluation of its effectiveness. Notably, on the CHAD dataset, our TRME model achieved a Frechet Inception Distance (FID) of 0.170, a Diversity score of 9.981, and an R-Precision (Top3) of 0.698, underscoring its robust capability in diverse settings.

The results highlights the progress made in text-to-motion generation and highlight our model's nuanced understanding of human motion. Through meticulous dataset augmentation and extensive training, the model demonstrates an exceptional ability to capture the essence of the described actions and translate them into realistic motion sequences.

6 Conclusion

In conclusion, the Text Residual Motion Encoder (TRME) substantially advances 3D human motion generation from textual descriptions. By incorporating residual blocks and training on the new CHAD dataset in the VQ-VAE architecture, TRME excels in capturing complex motion dynamics. It consistently outperforms state-of-the-art (SOTA) models in key metrics such as FID, Diversity, and R-Precision. This demonstrates TRME's effectiveness in 3D motion synthesis, marking a significant step forward in the field.

References

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

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

[3] I. Akhter and M. J. Black. Pose-conditioned joint angle limits for 3d human pose reconstruction. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1446–1455, 2015.

[4] S. Bengio, O. Vinyals, N. Jaitly, and N. Shazeer. Scheduled sampling for sequence prediction with recurrent neural networks. *NeurIPS*, pages 1171–1179, 2015.

[5] Y. Bengio, J. Louradour, R. Collobert, and J. Weston. Curriculum learning. In *Proceedings of the 26th annual International Conference on Machine Learning*, pages 41–48. ACM, 2009.

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

[7] H. Cai, C. Bai, Y.-W. Tai, and C.-K. 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.

[8] Carnegie Mellon University. Carnegie mellon motion capture database. http://mocap.cs.cmu.edu/, 2009. Accessed: 04/01/2025.

[9] N. Chen, Y. Zhang, H. Zen, R. J. Weiss, M. Norouzi, and W. Chan. Wavegrad: Estimating gradients for waveform generation. *arXiv preprint arXiv:2009.00713*, 2020.

[10] P. Dhariwal and A. Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

[11] S. Ghorbani, K. Mahdaviani, A. Thaler, K. Kording, D. J. Cook, G. Blohm, and N. F. Troje. Movi: A large multi-purpose human motion and video dataset. *Plos one*, 16(6):e0253157, 2021.

[12] S. Ghorbani et al. Movi: A large multipurpose motion and video dataset. https://arxiv.org/abs/2003.01888, 2020.

[13] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.

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

[15] C. Guo, S. Zou, X. Zuo, S. Wang, W. Ji, X. Li, and L. Cheng. Generating diverse and natural 3d human motions from text. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5142–5151, 2022. doi: 10.1109/CVPR52688.2022.00509.

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

[17] C. Guo, S. Zou, X. Zuo, S. Wang, W. Ji, X. Li, and L. 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.

[18] F. Helm, N. F. Troje, and J. Munzert. Motion database of disguised and non-disguised team handball penalty throws by novice and expert performers. *Data in brief*, 15:981–986, 2017.

[19] F. Helm, R. Cañal-Bruland, D. L. Mann, N. F. Troje, and J. Munzert. Integrating situational probability and kinematic information when anticipating disguised movements. *Psychology of Sport and Exercise*, 46: 101607, 2020.

[20] J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[21] L. Hoyet, K. Ryall, R. McDonnell, and C. O'Sullivan. Sleight of hand: perception of finger motion from reduced marker sets. In *Proceedings of the ACM SIGGRAPH symposium on interactive 3D graphics and games*, pages 79–86, 2012.

[22] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu. 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, 2013.

[23] T. N. Kipf and M. Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

[24] M. Loper, N. Mahmood, and M. J. Black. Mosh: motion and shape capture from sparse markers. *ACM Trans. Graph.*, 33(6):220–1, 2014.

[25] N. Mahmood, N. Ghorbani, N. F. Troje, G. Pons-Moll, and M. 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.

[26] C. Mandery, Ö. Terlemez, M. Do, N. Vahrenkamp, and T. Asfour. The kit whole-body human motion database. In *2015 International Conference on Advanced Robotics (ICAR)*, pages 329–336. IEEE, 2015.

[27] M. Müller, A. Baak, and H.-P. Seidel. Efficient and robust annotation of motion capture data. In *Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 17–26, 2009.

[28] A. Q. Nichol and P. Dhariwal. Improved denoising diffusion probabilistic models. In *International conference on machine learning*, pages 8162–8171. PMLR, 2021.

[29] OSU ACCAD. Accad. https://accad.osu.edu/ research/motion-lab/system-data, 2024. Accessed: 04/01/2025.

[30] M. Petrovich, M. J. Black, and G. 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.

[31] M. Petrovich, M. J. Black, and G. Varol. Temos: Generating diverse human motions from textual descriptions. In *European Conference on Computer Vision*, pages 480–497. Springer, 2022.

[32] M. Plappert, C. Mandery, and T. Asfour. The kit motion-language dataset. *Big Data*, 4(4):236–252, 2016.

[33] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[34] C. Research. Cnrs research data. https://entrepot.recherche.data.gouv.fr/dataverse/cnrs, 2022.

[35] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning internal representations by error propagation, parallel distributed processing, explorations in the microstructure of cognition, ed. de rumelhart and j. mcclelland. vol. 1. 1986. *Biometrika*, 71:599–607, 1986.

[36] L. Sigal, A. O. Balan, and M. J. Black. Humaneva: Synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion. *International journal of computer vision*, 87 (1):4–27, 2010.

[37] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pages 2256–2265. PMLR, 2015.

[38] K. Sohn, H. Lee, and X. Yan. Learning structured output representation using deep conditional generative models. *Advances in neural information processing systems*, 28, 2015.

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

[40] Y. Song and S. Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019.

[41] D. Strömbäck, S. Huang, and V. Radu. Mm-fit: Multimodal deep learning for automatic exercise logging across sensing devices. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 4(4):1–22, 2020.

[42] O. Taheri, N. Ghorbani, M. J. Black, and D. Tzionas. Grab: A dataset of whole-body human grasping of objects. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IV 16*, pages 581–600. Springer, 2020.

[43] G. Tevet, B. Gordon, A. Hertz, A. H. Bermano, and D. Cohen-Or. Motionclip: Exposing human motion generation to clip space. In *European Conference on Computer Vision*, pages 358–374. Springer, 2022.

[44] G. Tevet, S. Raab, B. Gordon, Y. Shafir, D. Cohen-Or, and A. H. Bermano. Human motion diffusion model. *arXiv preprint arXiv:2209.14916*, 2022.

[45] S. Tripathi, L. Müller, C.-H. P. Huang, O. Taheri, M. J. Black, and D. Tzionas. 3d human pose estimation via intuitive physics. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4713–4725, 2023.

[46] N. F. Troje. Decomposing biological motion: A framework for analysis and synthesis of human gait patterns. *Journal of vision*, 2(5):2–2, 2002.

[47] M. Trumble, A. Gilbert, C. Malleson, A. Hilton, and J. Collomosse. Total capture: 3d human pose estimation fusing video and inertial sensors. In *Proceedings of 28th British Machine Vision Conference*, pages 1–13, 2017.

[48] S. Tsuchida, S. Fukayama, M. Hamasaki, and M. Goto. Aist dance video database: Multi-genre, multi-dancer, and multi-camera database for dance information processing. In *ISMIR*, volume 1, page 6, 2019.

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

[50] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[51] S. Venugopalan, M. Rohrbach, J. Donahue, R. Mooney, T. Darrell, and K. Saenko. Sequence to sequence-video to text. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4534–4542, 2015.

[52] J. Zhang, Y. Zhang, X. Cun, S. Huang, Y. Zhang, H. Zhao, H. Lu, and X. Shen. T2m-gpt: Generating human motion from textual descriptions with discrete representations. *arXiv preprint arXiv:2301.06052*, 2023.

[53] M. Zhang, Z. Cai, L. Pan, F. Hong, X. Guo, L. Yang, and Z. Liu. Motiondiffuse: Text-driven human motion generation with diffusion model. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.