GranuMamba: A Multi-Granularity State Space Model for Co-Speech Gesture Generation: Supplementary Material

CONTENTS

1. RELATED WORK

A. Co-Speech Gesture Generation

The primary goal of co-speech gesture generation is to learn the complex mapping from speech input (audio, text) to continuous human motion. The field has progressed through several paradigms of deep learning models [1].

Probabilistic and Generative Models: Early deep learning approaches recognized the one-to-many nature of the problem—a single utterance can be accompanied by various plausible gestures. Variational Autoencoders (VAEs) [2] and Generative Adversarial Networks (GANs) [3] were instrumental in addressing this. VAEs learn a latent distribution of gestures to enable diverse sampling, while GANs employ a discriminator to improve the realism of generated motion, reducing the "over-smoothing" effect common in regression-based models. More recently, Vector Quantized VAEs (VQ-VAEs) have gained traction by learning a discrete codebook of "motion primitives" [4]. This allows for modeling gestures as a sequence of these learned codes, often using a subsequent autoregressive model, but introduces complexity through a multi-stage training pipeline[5].

Sequence-to-Sequence Models: With the success of Transformers in natural language processing, they were quickly adapted for gesture generation. Models like Gesticulator [6] leverage the self-attention mechanism to capture long-range dependencies between speech cues and motion dynamics. While effective for maintaining speech-motion alignment over long sequences, their attention mechanism's quadratic complexity in sequence length poses a significant bottleneck for training and real-time inference, especially with high-resolution motion data.

Diffusion Models: Currently representing the state-of-the-art in generation quality, diffusion models have produced highly realistic and nuanced gestures [7]. These models work by iteratively denoising a random signal, conditioned on speech features, to produce a clean gesture sequence. While their generation quality is unparalleled, this iterative sampling process makes inference computationally intensive and significantly slower than single-pass models, limiting their use in low-latency applications [8].

A common thread across these diverse architectures is that they typically process and generate motion at a single, uniform temporal scale. This can force the model into a trade-off between capturing fine-grained, rapid motions and maintaining long-term, coherent body posture [9].

B. State Space Models for Motion Synthesis

To mitigate the efficiency limitations of Transformers, State Space Models (SSMs), and particularly Mamba, have recently emerged as a powerful alternative for sequence modeling [10]. Mamba's selective mechanism allows it to model long-range dependencies with linear time complexity by dynamically focusing on relevant information in the input sequence [11].

This efficiency has made it an attractive choice for gesture generation. MambaTalk [12] was a pioneering work that demonstrated Mamba's effectiveness for holistic gesture synthesis, showing it could achieve performance comparable to Transformers at a fraction of the computational cost. Subsequent works like MambaGesture [13] have further explored its potential, for example, by integrating more sophisticated multi-modal fusion techniques to better align audio and motion representations. However, while these models successfully leverage Mamba's efficiency, they still largely operate within the single-scale processing paradigm [14]. They do not possess an explicit architectural bias to handle the inherently multi-scale nature of human gestures [15].

2. PRELIMINARIES

Selective State Spaces Model. In our approach, we adopt the Selective State Space (S3) model, specifically Mamba [10], which integrates a selection mechanism with a scan module (S6) for advanced sequence modeling. The key strength of this model is its ability to **dynamically select salient input segments for prediction, thereby enhancing its focus on pertinent information and improving overall performance**. Unlike the traditional S4 model, which relies on time-invariant matrices (A, B, C) and a scalar Δ, Mamba introduces a selection mechanism that learns these parameters directly from the input data using fully-connected layers. This adaptability enables the model to generalize more effectively and perform complex modeling tasks while maintaining computational efficiency and efficient data storage.

For each input x_t, the model processes it to update a hidden state h_t and produce an output y_t. When $t > 0$, the model's formulation is as follows:

$$\begin{aligned} h_t &= \tilde{A}_t h_{t-1} + \tilde{B}_t x_t \\ y_t &= C_t h_t \end{aligned} \tag{S1}$$

where $\tilde{A}_t$, $\tilde{B}_t$, and C_t are matrices and vectors that are updated at each time step, allowing the model to adapt to the temporal dynamics of the input sequence. For discrete data, the continuous parameters are transformed using a sampling interval Δ. The discretized matrices are derived as follows:

$$\begin{aligned} \tilde{A} &= \exp(\Delta A) \\ \tilde{B} &= (\Delta A)^{-1}(\exp(\Delta A) - I) \cdot \Delta B \end{aligned} \tag{S2}$$

where I denotes the identity matrix. The scan module within Mamba is designed to efficiently capture temporal patterns and dependencies across multiple time steps by applying a set of trainable operations to each segment of the input sequence.

In our framework, Mamba serves as a sequence modeling tool for decoding gesture actions across different parts of the body. Our primary adaptation involves **modifying the decoder's input and the range of features**. This approach allows us to utilize Mamba to separately model the global motion features and the local motion features of different body parts. These modifications are implemented as learnable operations within the network, which are optimized during training to assist the model in processing sequential gesture data.

3. STAGE 1 TRAINING LOSS

Training Objectives. To optimize the discrete motion prior, we adopt a combination of reconstruction, commitment, and codebook losses, following the standard VQ-VAE paradigm.

First, the reconstruction loss ensures that the decoded motion $\hat{B}$ remains close to the original input sequence B:

$$\mathcal{L}_{\text{rec}} = \|B - \hat{B}\|_2^2 \tag{S3}$$

Second, a codebook loss aligns the quantized embeddings with their corresponding latent features, encouraging effective codebook utilization:

$$\mathcal{L}_{\text{cb}} = \|\text{sg}[z_t] - e_k\|_2^2 \tag{S4}$$

where sg[$\cdot$] denotes the stop-gradient operator.

Third, a commitment loss prevents latent features from fluctuating excessively and stabilizes training:

$$\mathcal{L}_{\text{com}} = \beta\|z_t - \text{sg}[e_k]\|_2^2 \tag{S5}$$

where β is a weighting factor.

The overall loss for this stage is the weighted sum:

$$\mathcal{L}_{\text{prior}} = \mathcal{L}_{\text{rec}} + \mathcal{L}_{\text{cb}} + \mathcal{L}_{\text{com}} \tag{S6}$$

This objective encourages the discrete latent space to be both semantically expressive and structurally consistent, enabling the codebook tokens to serve as effective motion priors.

4. EXPERIMENTS

A. Implementation Details

All models are implemented in PyTorch and trained on a single NVIDIA A100 GPU. In the first stage, GraphConv-VQ-VAE was pretrained with separate VQ-VAE models for facial features, upper-body, hands, and lower-body motions. In the second stage, the upper-body motion representation is trained using features extracted from the first stage. Upper-body joint poses are represented using SMPL-X FLAME with 78 dimensions, while facial features have 100 dimensions. The audio-to-motion model uses word-level embeddings with a vocabulary size of 11,195 and an embedding dimension of 300, optionally pre-encoded with FastText.

In the second stage, full-body SMPL-X representations with 330 joint dimensions are used, with sequence lengths of 64 frames and a stride of 20 frames. Audio inputs combine amplitude, CTC, and raw audio features at a 16 kHz sampling rate, and speaker identity is encoded as a one-hot vector. The MambaTalk model consists of a single-layer Transformer with a hidden size of 768. The training batch size is 64, with a base learning rate of 3e-4 for the first-stage upper-body representation training and 5e-4 for the second-stage full-body motion generation. Reconstruction loss is weighted by 1, and gradients are clipped with a norm of 0.99. All experiments, including ablation studies and evaluations, are conducted on a single A100 GPU, with an average training time of approximately 5 hours per run.

B. Evaluation Metrics

To assess the realism and fidelity of generated videos, we adopt the Fréchet Video Distance (FVD), an extension of the widely used Fréchet Inception Distance (FID) to the video domain. FVD quantifies the similarity between real and synthesized video distributions in a learned feature space. Specifically, video features are extracted using a pretrained Inflated 3D ConvNet (I3D) model, and their distributions are approximated as multivariate Gaussians. The FVD is then defined as the Fréchet distance between the Gaussian distributions of real and generated video features:

$$\text{FVD} = \|\mu_r - \mu_g\|^2 + \text{Tr}\left(\Sigma_r + \Sigma_g - 2(\Sigma_r\Sigma_g)^{\frac{1}{2}}\right) \tag{S7}$$

where (μ_r, Σ_r) and (μ_g, Σ_g) represent the mean and covariance of the real and generated video feature distributions, respectively. A lower FVD indicates that the generated videos are closer to real ones in terms of temporal dynamics and visual quality.

To further evaluate the realism of generated gestures, we employ the Fréchet Gesture Distance (FGD) [9], which measures the distributional similarity between real and synthesized body gestures. FGD adopts the same Fréchet distance formulation as FVD, but operates in a gesture feature space:

$$\text{FGD}(\mathbf{g}, \hat{\mathbf{g}}) = \left\|\mu_r - \mu_g\right\|^2 + \text{Tr}\left(\Sigma_r + \Sigma_g - 2\left(\Sigma_r\Sigma_g\right)^{1/2}\right) \tag{S8}$$

where (μ_r, Σ_r) denote the mean and covariance of the latent feature distribution z_r for real gestures $\mathbf{g}$, while (μ_g, Σ_g) correspond to those of the synthesized gestures $\hat{\mathbf{g}}$.

Subsequently, Diversity[16] is quantified by computing the average L1 distance across multiple body gesture clips. Higher Diversity signifies greater variance within the gesture clips. We compute the average L1 distance across various N motion clips using the following equation:

$$\text{Diversity} = \frac{1}{2N(N-1)} \sum_{t=1}^{N} \sum_{j=1}^{N} \left\| p_t^i - \hat{p}_t^j \right\|_1 \tag{S9}$$

where p_t denotes the positions of joints in frame t. We assess diversity across the entire test dataset. Moreover, when calculating joint positions, translation is zeroed, indicating that L1 Diversity is exclusively concentrated on local motion dynamics.

The synchronization between the speech and motion is conducted using Beat Alignment Score (BAS) [17]. BC indicates a more precise synchronization between the rhythm of gestures and the audio's beat. We define the onset of speech as the audio's beat and identify the local minima of the upper body joints' velocity (excluding fingers) as the motion's beat. The synchronization between audio and gesture is determined using the following equation:

$$\text{BeatAlignScore} = \frac{1}{m} \sum_{i=1}^{m} \exp\left(-\frac{\min_{t_j^y \in B^y} \| t_i^x - t_j^y \|^2}{2\sigma^2} \right) \tag{S10}$$

where $B^x = \{t_i^x\}$ is the kinematic beats, $B^y = \{t_j^y\}$ is the audio beats and σ is a parameter to normalize sequences with different FPS. We set $\sigma = 3$ in all our experiments as the FPS of all our experiment sequences is 60.

To assess whether the generated videos preserve the identity of the target person, we employ the Cosine Similarity Identity Metric (CSIM) [18]. CSIM evaluates the similarity between the facial identity features extracted from real and generated frames using a pretrained face recognition model (e.g., ArcFace). Specifically, let $\mathbf{f}_r$ and $\mathbf{f}_g$ denote the feature embeddings of the real and generated frames, respectively. The CSIM is then defined as the cosine similarity between the two embeddings:

$$\text{CSIM} = \frac{\mathbf{f}_r \cdot \mathbf{f}_g}{\|\mathbf{f}_r\| \, \|\mathbf{f}_g\|} \tag{S11}$$

A higher CSIM score indicates that the generated video better preserves the identity of the target person, ensuring consistency between the synthesized content and the reference subject.

To further evaluate the accuracy of motion generation, we adopt the Percent of Correct Motion parameters (PCM) [17]. PCM measures the proportion of generated motion parameters that lie within a predefined error threshold relative to the ground truth. Formally, let $\mathbf{p} = \{p_t\}_{t=1}^{T}$ denote the ground-truth motion parameters and $\hat{\mathbf{p}} = \{\hat{p}_t\}_{t=1}^{T}$ the generated motion parameters. The frame-wise error is defined as

$$e_t = \|p_t - \hat{p}_t\|_2 \tag{S12}$$

and the PCM score is computed as

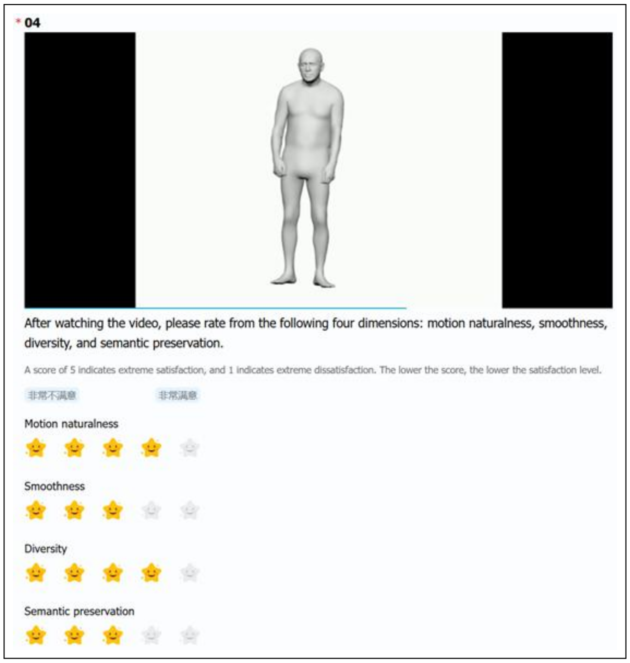
$$\text{PCM} = \frac{1}{T} \sum_{t=1}^{T} \mathbb{I}(e_t < \delta) \tag{S13}$$

where δ is a tolerance threshold and $\mathbb{I}(\cdot)$ is the indicator function. A higher PCM value indicates that the generated motion more closely follows the ground-truth parameters, reflecting higher motion accuracy.

C. User Study Details

In practice, objective measures may not always align with subjective human perception, particularly in novel contexts of collaborative voice and video generation. To gain deeper insights into the visual performance of our methods, we evaluated the visual performance of our generated videos through a user study.

To evaluate the overall quality of the body movements generated, we conducted a user study using 16 randomly sampled videos from the BEAT2 test set, each video is approximately one minute long. According to the standards established by the International Telecommunication Union (ITU) [19]. We invited 30 participants to rate the videos based on four dimensions: Natural, Diversity, Smootha, and Semantic preservation. Each criterion was rated on a scale from 1 to 5, with 5 representing the highest quality.


* 04

After watching the video, please rate from the following four dimensions: motion naturalness, smoothness, diversity, and semantic preservation.

A score of 5 indicates extreme satisfaction, and 1 indicates extreme dissatisfaction. The lower the score, the lower the satisfaction level.

非常不满意 非常满意

Motion naturalness

★ ★ ★ ★ ☆

Smoothness

★ ★ ★ ☆ ☆

Diversity

★ ★ ★ ★ ☆

Semantic preservation

★ ★ ★ ☆ ☆

Fig. S1. Screenshot of user study website.

We created a Tencent questionnaire, as shown in the Figure S1, which includes test videos and four rating dimensions. We recorded the scores from all participants, cleaned the non-compliant data, and then calculated the average score for each dimension. Prior to participation, we provided training to the participants to ensure that their ratings were reasonable.

D. Efficiency Analysis

To assess the efficiency of our framework, we conducted a detailed comparison of inference speed across different methods. Leveraging the lightweight temporal modeling of Mamba and the discrete representation learning of VQ-VAE, our model reduces computational overhead while maintaining high-quality generation. We measured inference time on an NVIDIA GeForce RTX 3090 GPU for generating a 40-second video at 30 FPS, averaging over three runs, with results summarized in Table S1. As shown, EMAGE and MambaTalk require 171.43 s and 69.04 s, respectively, to generate the same sequence. Our method achieves an inference time of 41.15 s, significantly faster than EMAGE and comparable to MambaTalk, while still slightly slower than GestureLSM (31.75 s). These results indicate that our framework is suitable for real-time or interactive applications, providing low-latency gesture generation with competitive efficiency.

Table S1. Time consumption comparison of inference (1 NVIDIA GeForce RTX 3090 GPU).

Methods	Inference(video of $\sim$ 40 sec)
EMAGE	$\sim$ 171.43 sec
MambaTalk	$\sim$ 69.04 sec
GestureLSM	$\sim$ **31.75** sec
Ours	$\sim$ 41.15 sec

5. LIMITATIONS

Although GranuMamba achieves strong performance across objective and subjective evaluations, several limitations remain. First, the model is trained and validated only on the BEAT2 dataset, which may constrain its generalization to broader conversational or emotional scenarios. Second, while the multi-granularity design improves local-global motion coherence, it may still struggle with highly subtle hand articulations. Third, our current framework focuses on 3D motion parameters without explicitly modeling physical constraints such as collision avoidance, which may occasionally lead to implausible poses. Addressing these limitations by incorporating larger and more diverse datasets, extending to multi-party interactions, and integrating physical priors will be important directions for future work.

REFERENCES

1. S. Nyatsanga, T. Kucherenko, C. Ahuja, *et al.*, "A comprehensive review of data-driven co-speech gesture generation," in *Computer Graphics Forum,* vol. 42 (Wiley Online Library, 2023), pp. 569–596.
2. D. P. Kingma and M. Welling, "Auto-encoding variational bayes," arXiv preprint arXiv:1312.6114 (2013).
3. I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, *et al.*, "Generative adversarial nets," Adv. neural information processing systems **27** (2014).
4. A. Van Den Oord, O. Vinyals *et al.*, "Neural discrete representation learning," Adv. neural information processing systems **30** (2017).
5. Y.-J. Zhang, S. Pan, L. He, and Z.-H. Ling, "Learning latent representations for style control and transfer in end-to-end speech synthesis," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP),* (IEEE, 2019), pp. 6945–6949.
6. T. Kucherenko, P. Jonell, S. Van Waveren, *et al.*, "Gesticulator: A framework for semantically-aware speech-driven gesture generation," in *Proceedings of the 2020 international conference on multimodal interaction,* (2020), pp. 242–250.
7. J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," Adv. neural information processing systems **33**, 6840–6851 (2020).
8. S. Alexanderson, R. Nagy, J. Beskow, and G. E. Henter, "Listen, denoise, action! audio-driven motion synthesis with diffusion models," ACM Transactions on Graph. (TOG) **42**, 1–20 (2023).
9. Y. Yoon, B. Cha, J.-H. Lee, *et al.*, "Speech gesture generation from the trimodal context of text, audio, and speaker identity," ACM Transactions on Graph. (TOG) **39**, 1–16 (2020).
10. A. Gu and T. Dao, "Mamba: Linear-time sequence modeling with selective state spaces," arXiv preprint arXiv:2312.00752 (2023).
11. A. Gu, I. Johnson, K. Goel, *et al.*, "Combining recurrent, convolutional, and continuous-time models with linear state space layers," Adv. neural information processing systems **34**, 572–585 (2021).
12. Z. Xu, Y. Lin, H. Han, *et al.*, "Mambatalk: Efficient holistic gesture synthesis with selective state space models," Adv. Neural Inf. Process. Syst. **37**, 20055–20080 (2024).
13. C. Fu, Y. Wang, J. Zhang, *et al.*, "Mambagesture: Enhancing co-speech gesture generation with mamba and disentangled multi-modality fusion," in *Proceedings of the 32nd ACM International Conference on Multimedia,* (2024), pp. 10794–10803.
14. L. Wang, K. Hu, L. Bai, *et al.*, "Multi-scale control signal-aware transformer for motion synthesis without phase," in *Proceedings of the AAAI Conference on Artificial Intelligence,* vol. 37 (2023), pp. 6092–6100.
15. Z. Yin, Y. H. Tsui, and P. Hui, "M3g: Multi-granular gesture generator for audio-driven full-body human motion synthesis," arXiv preprint arXiv:2505.08293 (2025).
16. X. Liu, Q. Wu, H. Zhou, *et al.*, "Learning hierarchical cross-modal association for co-speech gesture generation," in *Proceedings of the IEEE/CVF CVPR,* (2022), pp. 10462–10472.
17. R. Li, S. Yang, D. A. Ross, and A. Kanazawa, "Ai choreographer: Music conditioned 3d dance generation with aist++," in *Proceedings of the IEEE/CVF CVPR,* (2021), pp. 13401–13412.
18. J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," in *Proceedings of the IEEE/CVF CVPR,* (2019), pp. 4690–4699.
19. R. BT, "Methodology for the subjective assessment of the quality of television pictures," Int. Telecommun. Union **4**, 19 (2002).