LiveGesture: Streamable Co-Speech Gesture Generation Model

Muhammad Usama Saleem[1] Mayur Jagdishbhai Patel[1] Ekkasit Pinyoanuntapong[1] Zhongxing Qin[1]
Li Yang[1] Hongfei Xue[1] Ahmed Helmy[1] Chen Chen[2] Pu Wang[1]
[1]University of North Carolina, [2]University of Central Florida
msaleem2@charlotte.edu

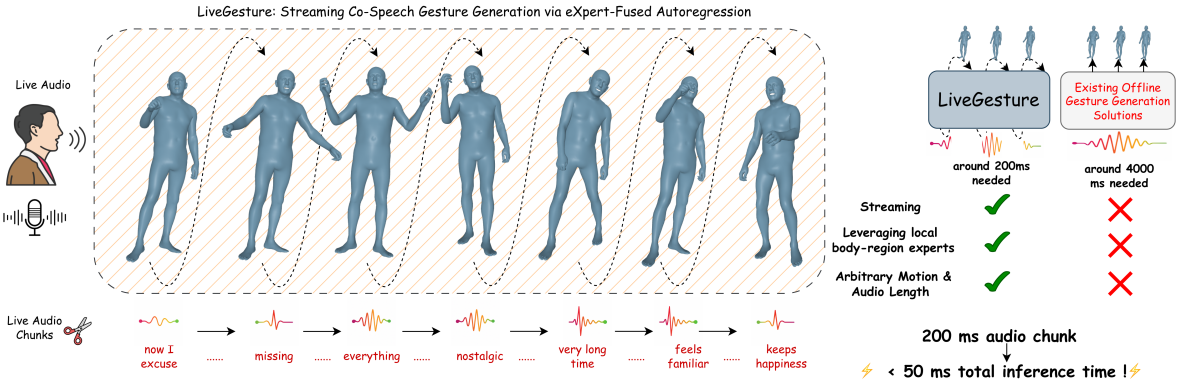

Figure 1. LiveGesture overview. Given live audio chunks, our framework generates full-body SMPL-X motion online with zero look-ahead. A streamable SVQ motion tokenizer and a hierarchical eXpert-fused autoregressive model (region-wise AR eXperts plus causal spatial–temporal fusion) enable low-latency (< 50 ms per 200 ms chunk) generation of diverse, beat-synchronous gestures over arbitrary-length speech, in contrast to prior offline gesture methods that require full utterances and incur much higher latency.

Abstract

We propose LiveGesture, the first fully streamable, speech-driven full-body gesture generation framework that operates with zero look-ahead and supports arbitrary sequence length. Unlike existing co-speech gesture methods—which are designed for offline generation and either treat body regions independently or entangle all joints within a single model—LiveGesture is built from the ground up for causal, region-coordinated motion generation. LiveGesture *consists of two main modules: the Streamable Vector-Quantized Motion Tokenizer (SVQ) and the Hierarchical Autoregressive Transformer (HAR). The SVQ tokenizer converts the motion sequence of each body region into causal, discrete motion tokens, enabling real-time, streamable token decoding. On top of SVQ, HAR employs region-eXpert autoregressive (xAR) transformers to model expressive, fine-grained motion dynamics for each body region. A causal spatio-temporal fusion module (xAR-Fusion) then captures and integrates correlated motion dynamics across regions. Both xAR and xAR-Fusion are conditioned on live, continuously arriving audio signals encoded by a streamable causal audio encoder. To enhance robustness under streaming noise and prediction errors, we introduce autoregressive masking training, which leverages uncertainty-guided token masking and random region masking to expose the model to imperfect, partially erroneous histories during training. Experiments on the BEAT2 dataset demonstrate that LiveGesture produces coherent, diverse, and beat-synchronous full-body gestures in real time, matching or surpassing state-of-the-art offline methods under true zero–look-ahead conditions.*

1. Introduction

Gestures naturally emerge alongside speech, emphasizing key ideas, conveying intent, and grounding abstract concepts in physical space. Far from being cosmetic, these nonverbal behaviors are central to how humans communicate and strongly influence perceived engagement and conversational naturalness. As virtual humans and embodied AI agents become more common in assistants, VR/AR avatars,

telepresence, and content creation [12, 32, 34–36, 38], generating believable, real-time co-speech gestures is becoming a core requirement.

Recent co-speech gesture models [18, 23, 44, 45] synthesize full-body motion using either continuous trajectories [11] or discrete motion tokens [18, 45]. While they produce locally plausible movements, their decoupled region representations often fail to capture fine-grained interdependencies across body parts, leading to gestures that lack holistic full-body coordination. Most systems rely on *offline* diffusion-based [3, 4] or autoregressive [18, 45] architectures that assume access to complete utterances or long audio/text contexts, incurring high latency and preventing truly interactive use. GestureLSM [22] moves toward fast inference with a lightweight architecture, but remains *non-streamable*, requiring full speech segments and preventing incremental updates as audio arrives.

In this work, we introduce *LiveGesture*, to our knowledge the first *fully streamable*, zero–look-ahead co-speech full-body gesture generation framework. LiveGesture comprises two core components: the Streamable Vector-Quantized Motion Tokenizer (SVQ) and the Hierarchical Autoregressive Transformer (HAR). The SVQ tokenizer encodes each body region's motion sequence into causal, discrete motion tokens, enabling real-time, low-latency decoding. Built on top of SVQ, HAR employs region-eXpert autoregressive (xAR) transformers to model expressive and fine-grained motion dynamics for individual regions. A causal spatiotemporal fusion module (xAR-Fusion) further integrates inter-region dependencies, capturing coherent, whole-body coordination. Both xAR and xAR-Fusion operate under strict causality, conditioned on continuously arriving audio features extracted by a streamable causal audio encoder. To improve robustness against streaming noise and prediction drift, we introduce autoregressive masking training, which applies uncertainty-guided token masking and random region masking to expose the model to imperfect and partially corrupted histories. Evaluated on the BEAT dataset, *LiveGesture* generates coherent, diverse, and beat-synchronous full-body SMPL-X gestures in real time—achieving state-of-the-art performance under true zero–look-ahead conditions. Our contributions are summarized as follows:

- We propose *LiveGesture*, the first zero–look-ahead, fully streamable speech-driven full-body gesture generation framework supporting arbitrary sequence lengths, designed for real-time human–AI interaction.
- We introduce a streamable motion tokenizer that converts continuous SMPL-X regional motion sequences into discrete latent motion tokens through an asymmetric architecture of bidirectional encoding, causal decoding, and token quantization, trained via a two-stage pretraining and quantization strategy to ensure strict causality, low latency, and stable representation learning.
- We design a Hierarchical Autoregressive Transformer that consists of region-eXpert autoregressive (xAR) transformers that model expressive, fine-grained motion dynamics for each body region, a causal spatiotemporal fusion module (xAR-Fusion) that captures and integrates correlated motion dynamics across regions, and a streamable causal audio encoder that processes continuously arriving audio signals for live gesture generation.
- Experiments on the **BEAT** dataset demonstrate that *LiveGesture* produces coherent, diverse, and beat-synchronous SMPL-X gestures in real time, matching or surpassing state-of-the-art offline methods under true zero–look-ahead conditions.

2. Related Works

Autoregressive Motion Generation. Autoregressive models have shown remarkable success in language and image generation, as demonstrated by GPT [1], DALL-E [31], and VQ-GAN [7, 41, 47]. Following this, Masked Motion Models [10, 30] have adopted discrete motion token sequences from pretrained codebooks in human motion generation by decoding them in parallel. However, these models require generating all sequences at once and require a predefined motion length. In contrast, methods such as T2M-GPT [48], AttT2M [50], and BAMM [29] also rely on codebook tokens but train their models autoregressively to predict motion token sequences token by token. Similarly, MotionGPT [13] generates motion tokens autoregressively, but integrates both language and motion tokens into a unified encoder, enabling a broader range of motion-related tasks. However, these approaches cannot achieve streamable generation because their motion-token decoders rely on a bidirectional decoder, future tokens influence the decoding of current ones.

Co-speech Gesture Generation. Early work by [9] employs an adversarial model to predict hand and arm poses directly from audio and uses conditional video generation techniques based on pix2pixHD [40] and EverybodyDanceNow [2]. More recent studies [5, 19–21, 23, 24, 33, 43, 49] explore generative models to derive audio-conditioned gesture generations. For example, HA2G [24] constructs both high-level and low-level audio–motion embeddings to guide gesture synthesis. TalkShow [45] models joint body and hand motion dynamics for talk-show videos. CaMN [17] and EMAGE [18] leverage GPT-like decoders for unified face and body modeling. Several methods focus on improving generation quality or efficiency. MambaTalk [44] accelerates gesture synthesis through the state space model architecture. DiffSHEG [4], SynTalker [3], and GestureLSM [22] adopt diffusion-based pipelines for co-speech gesture synthesis. However, none of the aforementioned methods support live streamable gesture gener-

ation because they require access to the complete text and speech input before generating gestures.

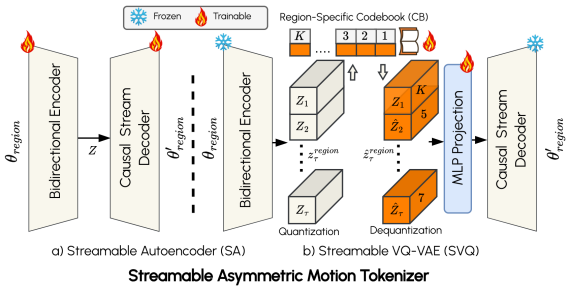

Figure 2. Overview of the Streamable Asymmetric Motion Tokenizer. (a) For each body region, a bidirectional encoder and causal stream decoder learn low-rate latent sequences. (b) Freezing this autoencoder, the Streamable VQ-VAE (SVQ) adds a region-specific codebook and projection head to quantize latents into discrete, time-synchronous motion tokens that remain compatible with strictly causal decoding for streaming.

3. Proposed Method: LiveGesture

Problem Formulation. We study *streaming* co-speech full-body gesture generation with *zero look-ahead*. At time t, the model receives only the recent motion history $\mathbf{S}_t = [\mathbf{q}_{t-H+1}, \ldots, \mathbf{q}_{t-1}]$, the current audio token a_t, and optional online transcript tokens w_t, and must predict the next full-body pose gesture $\mathbf{q}_t$. Formally, the gesture generation f_Θ is strictly causal, $\hat{\mathbf{q}}_t = f_\Theta(\mathbf{S}_t, a_t, w_t)$,. Unlike offline formulations that access future audio or motion, this setup enforces real-time multimodal conditioning and requires region-coordinated, temporally smooth motion aligned with speech rhythm and content under strict latency constraints.

Method Overview. As illustrated in Figure 2 and Figure 3, *LiveGesture* comprises two main modules, *Streamable Vector-Quantized Motion Tokenizer (SVQ)* and the *Hierarchical Autoregressive Transformer (HAR)*, for strictly causal, zero–look-ahead inference. First, the motion tokenizer converts the motion sequence of each body region into causal and discrete motion tokens to support real-time streamable token decoding. On top of SVQ, HAR first employs region-eXpert autoregressive (xAR) transformers to learn expressive and fine-grained motion dynamics of each body region. Then, the causal spatial-temporal fusion (xAR-fusion) model captures and fuses correlated motion dynamics among different body regions. Both xAR and xAR-fusion models are conditioned on the live-continuously arriving audio signals encoded by a streamable audio encoder.

3.1. Streamable Vector-Quantized Motion Tokenizer

We propose a per-region streamable motion tokenizer that converts the continuous motion sequence of each SMPL-X body region (lower body, upper body, hands, and head) [28] into discrete latent motion tokens while preserving strict causality and low latency. The tokenizer adopts an asymmetric architecture comprising three key components: (1) a bidirectional encoder that projects raw motion sequence into a latent space capturing rich bidirectional dependencies; (2) a causal decoder that reconstructs motion in a strictly streamable and causal manner from the latent representation; and (3) a token quantizer that discretizes continuous motion embeddings into compact latent tokens, enabling autoregressive motion generation through cross-entropy training. Previous research shows that simultaneous training of encoder and quantizer can cause token and embedding collapses, leading to degradation in the quality of learned representations. To mitigate this problem, we adopt a two-stage training strategy: Asymmetric Autoencoder Pretraining and Quantization Learning.

Asymmetric Autoencoder Pretraining. In the first stage, we train separate *Streamable Autoencoders* (SA) for each region's SMPL-X parameters. Let $\boldsymbol{\theta}_t^{\text{SMPL-X}} \in \mathbb{R}^D$ denote the full SMPL-X vector at original motion frame time $t \in \{1, \ldots, T_\text{f}\}$, partitioned into region-specific subsets $\boldsymbol{\theta}_t^{\text{region}} \in \mathbb{R}^{d_{\text{region}}}$. For each region, the input sequence $\{\boldsymbol{\theta}_t^{\text{region}}\}_{t=1}^{T_\text{f}}$ is encoded by a 1D convolutional encoder E into a downsampled latent sequence

$$z^{\text{region}} = \{z_\tau\}_{\tau=1}^{T} = E(\{\boldsymbol{\theta}_t^{\text{region}}\}_{t=1}^{T_\text{f}}), \quad T = T_\text{f}/4,$$

using strided convolutions to reduce the frame rate by a factor of 4 and obtain a compact representation for fast streaming inference. The encoder is bidirectional, aggregating past and future context, while the corresponding *Causal Stream Decoder* D_{CS} is strictly causal and reconstructs $\hat{\boldsymbol{\theta}}_t^{\text{region}}$ from the latent sequence z^{region}. This asymmetric design yields a globally coherent latent space while enforcing the causal decoding required for streaming. Each SA is trained with a reconstruction loss

$$\mathcal{L}_{\text{AE}} = \lambda_{\text{AE}} \mathcal{L}_{\text{recon}}.$$

Quantization Learning. In the second stage, we convert the continuous latents $z^{\text{region}} = \{z_\tau\}_{\tau=1}^{T}$ into discrete tokens, while keeping the encoder E and decoder D_{CS} frozen so that the temporal geometry and streamability learned in Stage 1 are preserved. For each region, we introduce a region-specific codebook

$$C^{\text{region}} = \{c_k\}_{k=1}^{K},$$

and vector-quantize z^{region} via nearest-neighbor assignment, obtaining dequantized embeddings

$$\hat{z}^{\text{region}} = \{\hat{z}_\tau\}_{\tau=1}^{T},$$

whose indices define time-synchronous motion tokens at the downsampled rate. A lightweight projection head W^{region},

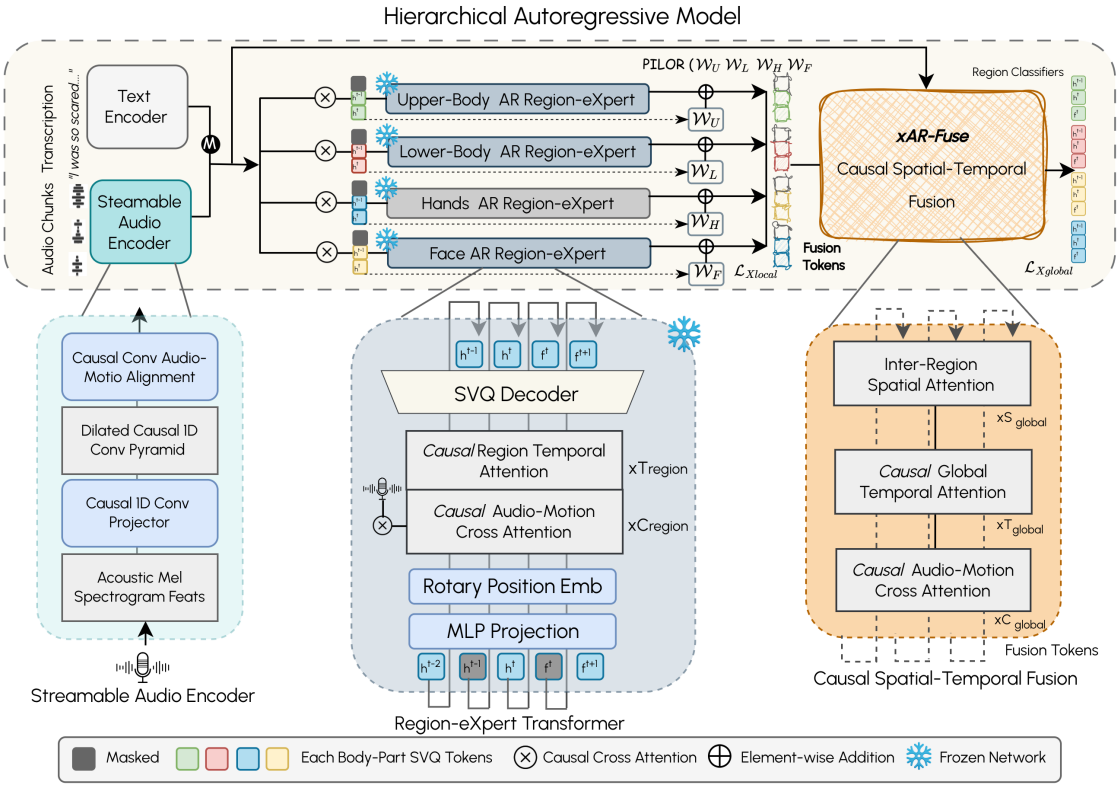

Figure 3. Overview of the Hierarchical Autoregressive Model in *LiveGesture*. A streamable audio encoder and optional text encoder provide causal audio/text tokens to four local AR Region-eXperts (upper body, lower body, hands, face), each modeling its own SVQ motion token stream. Their frozen states are adapted by per-region Pre-Infusion Adapters (PILOR) and fused by xAR-Fuse, a causal spatial–temporal transformer with audio–motion cross-attention, global temporal attention, and inter-region spatial attention that predicts next-step SVQ tokens for zero–look-ahead streaming full-body gesture generation.

implemented as a small MLP, is applied after quantization to map each $\hat{z}_\tau$ into the latent space expected by the frozen decoder, yielding $\tilde{z}^{\text{region}} = \{\tilde{z}_\tau\}_{\tau=1}^{T}, \quad \tilde{z}_\tau = W^{\text{region}}(\hat{z}_\tau)$

This projection serves as a learned adapter between the discrete codebook space and the original autoencoder latent space: it allows the codebook to remain compact and region-specific, while the decoder operates in a richer, globally learned space, and it absorbs quantization artifacts so that codebook updates do not disrupt the temporal structure encoded in D_{CS}. The projected sequence $\tilde{z}^{\text{region}}$ is then decoded causally to reconstructed SMPL-X parameters, $\hat{\boldsymbol{\theta}}^{\text{region}} = D_{\text{CS}}(\tilde{z}^{\text{region}})$. During this stage, only the codebook C^{region} and projection head W^{region} are updated. We use an L1 reconstruction term on SMPL-X region parameters and a standard VQ codebook loss $\mathcal{L}_{\text{cb}}(z^{\text{region}}, e^{\text{region}})$ [39], where e^{region} denotes the selected codebook embeddings, leading to

$$\mathcal{L}_{\text{stage2}} = \lambda_{\text{rec}} \left\| \boldsymbol{\theta}^{\text{region}} - D_{\text{CS}}(W^{\text{region}}(\hat{z}^{\text{region}})) \right\|_1 + \lambda_{\text{cb}}\, \mathcal{L}_{\text{cb}}(z^{\text{region}}, e^{\text{region}}).$$

EMA-style codebook updates and occasional resets [8] keep the codebook well conditioned, and the resulting SVQ tokens are discrete, time-synchronous, region-wise representations that remain compatible with the strictly causal decoder and are well suited for downstream streaming autoregressive SMPL-X gesture generation.

3.2. Hierarchical Autoregressive Model

The hierarchical autoregressive model consists of (1) region-eXperts, which are temporal causal transformers that learn expressive and fine-grained motion dynamics of each body region, (2) a global fusion model, which is a spatial-temporal causal transformer that captures the coherent and dependent dynamics among different body regions, and (3) a streamable audio encoder, which is a causal convolution network that enables live gesture generation conditioned on continuously arriving audio signals.

3.2.1. Region-eXpert Autoregressive Transformer (xAR)

Local region experts learn audio-driven motion dynamics for individual body parts, as different regions exhibit distinct motion distributions. For example, the upper body often produces large-scale movements, while the

hands perform fine-grained, high-frequency gestures. In particular, we divide full-body into four regions $\mathcal{R} = \{\text{upper body}, \text{lower body}, \text{hands}, \text{face}\}$. For each region $r \in \mathcal{R}$, we maintain a stream of motion tokens $\{x_t^r\}_{t=1}^T$ and aligned audio tokens $\{a_t\}_{t=1}^T$, with optional text tokens $\{w_t\}_{t=1}^T$ when available. In this stage, each eXpert is trained independently, but all share the same audio encoder so they learn to respond to a common audio representation. At time step t, the eXpert for region r receives a causal window of past region tokens $\{x_{t-h}^r, \ldots, x_{t-1}^r\}$ and audio tokens $\{a_{t-h}, \ldots, a_t\}$. Motion tokens are embedded by an MLP projection and augmented with rotary positional embeddings, yielding a continuous sequence that is stable under streaming shifts. This sequence is processed by a small stack of causal Transformer blocks that interleave (i) causal audio–motion cross attention, where region tokens attend only to past and current audio (and text) tokens, which enables rhythm-and-gesture alignment, and (ii) causal temporal self-attention, which captures region-specific inter-token correlations. The output motion embedding h_t^r is fed into a token classifier that models audio-condition categorical motion distribution.

3.2.2. Causal Spatial-Temporal Fusion (xAR-Fuse)

Local eXperts learn rich region-specific motion distributions but do not explicitly enforce full-body coordination. The causal fusion transformer operates on top of the frozen eXperts to model inter-region spatiotemporal correlations while preserving strict causality. At each time step t, we collect the hidden embeddings $\{h_t^r\}_{r\in\mathcal{R}}$ from all region eXperts. Since these embeddings are generated by independently trained networks, they are not naturally aligned for joint fusion. To bridge this gap, we introduce a lightweight residual adapter for each region, i.e., $\Delta h_t^r = \mathcal{W}_r h_t^r, \quad \tilde{h}_t^r = h_t^r + \Delta h_t^r$. This adapter gently aligns the output embeddings from different eXperts into a shared fusion space with minimal parameters and computational overhead. The adapted embeddings $\{\tilde{h}_t^r\}_{r\in\mathcal{R}}$ are passed through the fusion transformer, implemented as a stack of causal Transformer blocks. Each block is factorized into three attention layers that separately address audio-motion coupling, temporal refinement, and spatial coordination. A causal audio–motion cross-attention layer allows each region to query the current audio (and text) tokens, reinforcing beat and semantic alignment at every step. The inter-region spatial-attention layer, coupled with the causal global temporal-attention layer, explicitly and efficiently captures whole-body spatiotemporal coordination such as arm–torso coupling or mirrored hand gestures. Instead of sharing a single MLP token classifier, the output embeddings from the fusion transformer are finally fed into region-specific token classifiers to enhance fine-grained regional expressiveness.

3.2.3. Streamable Audio Encoder

To enable live gesture generation conditioned on continuously arriving audio signals, we design a causal audio encoder. First, an acoustic mel-spectrogram feature extraction module converts raw waveforms into log-mel frames using a short analysis window and small hop, chosen to match the latency budget of downstream streaming. These features are processed by a causal 1D convolutional projection module with left-only padding that enforces causality from the first layer. A dilated causal convolutional pyramid then captures multi-scale temporal structures in rhythm and prosody, expanding the receptive field without increasing latency. Finally, a causal audio–motion alignment module aggregates pyramid activations within each motion step using temporal striding, producing audio tokens a_t at the same frame rate as motion tokens. Each audio token depends only on past and current acoustic evidence, yielding multi-scale, low-latency, and temporally aligned representations ideal for zero–look-ahead streaming gesture generation.

3.3. Autoregressive Masked Training

Our model is trained in two stages: first learn audio-driven autoregressive motion dynamics for each region, and then train the fusion transformer with a hybrid masking strategy.

Stage 1: Local autoregressive modeling.. For each region $r \in \mathcal{R}$, the local eXpert defines a causal token-level model

$$p_\phi^r(x_{1:T}^r \mid a_{1:T}, w_{1:T}) = \prod_{t=1}^{T} p_\phi^r\big(x_t^r \mid x_{1:t-1}^r, a_{1:t}, w_{1:t}\big),$$

where x_t^r is the motion token for region r at time t, a_t is the audio token, and w_t is the optional text token. During training, we teacher-force the ground-truth history and optimize a standard autoregressive cross-entropy loss

$$\mathcal{L}_{\text{local}} = -\sum_{r\in\mathcal{R}}\sum_{t=1}^{T} \log p_\phi^r\big(x_t^r \mid x_{1:t-1}^r, a_{1:t}, w_{1:t}\big),$$

optionally combined with a small reconstruction loss in pose space. To mitigate *exposure bias*, i.e., the mismatch between teacher-forced training histories and self-generated histories at inference time, we also inject small Gaussian noise into the embedded history tokens with a small probability p_{noise}, so that local eXperts learn to cope with slightly corrupted context rather than relying on perfectly clean ground-truth sequences. This stage yields frozen local eXperts that provide well-calibrated next-token distributions and hidden states h_t^r for each region.

Stage 2: Hybrid Masking for Robust Fusion. To capture the inherent motion dynamics dependency among different body regions, a certain number of motion tokens from different regions at different time instances are masked out. The fusion transformer is trained to predict the next token

based on partially corrupted motion sequence, audio embeddings, and optional text transcript tokens. The reconstruction probability of each motion token under motion corruptions is

$$p_\theta\big(x_t^r \mid \tilde{x}_{1:t}^{1:|\mathcal{R}|}, a_{1:t}, \tilde{w}_{1:t}\big), \quad r \in \mathcal{M}_t,$$

where $\tilde{x}_{1:t}^{1:|\mathcal{R}|}$ represent corrupted motion sequence up to time t and $\tilde{w}_{1:t}$ are the (possibly masked) text tokens. The training objective is to minimize the negative log-likelihood of the next motion token prediction

$$\mathcal{L}_{\text{fuse}} = -\sum_{t=1}^{T} \sum_{r \in \mathcal{M}_t} \log p_\theta\big(x_t^r \mid \tilde{x}_{1:t}^{1:|\mathcal{R}|}, a_{1:t}, \tilde{w}_{1:t}\big).$$

We adopt two types of masking strategies: uncertainty-guided token masking (UGM) and random region masking (RM). Together, they expose the fusion model to the imperfect, partially erroneous histories it will see at inference time, further reducing exposure bias. For token masking, the masking ratio (i.e., the percentage of tokens is masked) is controlled by a cosine scheduler over training steps. Concretely, let $\lambda_{\text{UGR}}(s) \in [0, 1]$ be a cosine-annealed masking ratio at training step s. The $M_{\text{eff}}(s) = \lfloor \lambda_{\text{UGR}}(s)\, M_{\max} \rfloor$ least confident tokens based on their prediction probabilities are masked out. Early in training, $\lambda_{\text{UGR}}(s) \approx 0$, so masking ratio is low and the fusion transformer learns to inter-eXpert dependency under mostly clean inputs. As training progresses, $\lambda_{\text{UGR}}(s)$ increases toward 1, and the model is gradually exposed to more aggressively corrupted motion sequences. In addition to token-level masking, we apply a *region masking* scheme: with a small probability p_{drop} per training sequence, we select a region $r_{\text{drop}} \in \mathcal{R}$ and treat its entire token trajectory $\{x_t^{r_{\text{drop}}}\}_{t=1}^{T}$ as masked, forcing the fusion transformer to reconstruct the masked body region purely from audio and the remaining unmasked regions

.

Classifier-Free Logits Guidance. To further enhance audio/text–motion alignment, we apply classifier-free guidance to the discrete logits of the token classifiers. During training, audio, text, or both modalities are randomly dropped with a small probability to enable the model to learn an unconditional prior. At inference, the conditional and unconditional logits for each region $r \in \mathcal{R}$ are combined as $\ell^r_{\text{guided}} = \ell^r_{\text{uncond}} + \gamma\big(\ell^r_{\text{cond}} - \ell^r_{\text{uncond}}\big)$, where $\gamma \geq 1$ denotes the guidance scale. The resulting guided logits are then passed through a softmax function to produce the token distribution used for sampling during inference.

4. Experiments

Datasets. We train and evaluate LiveGesture on the BEAT2 corpus introduced by EMAGE [18], which contains 60 hours of SMPL-X full-body motion aligned with

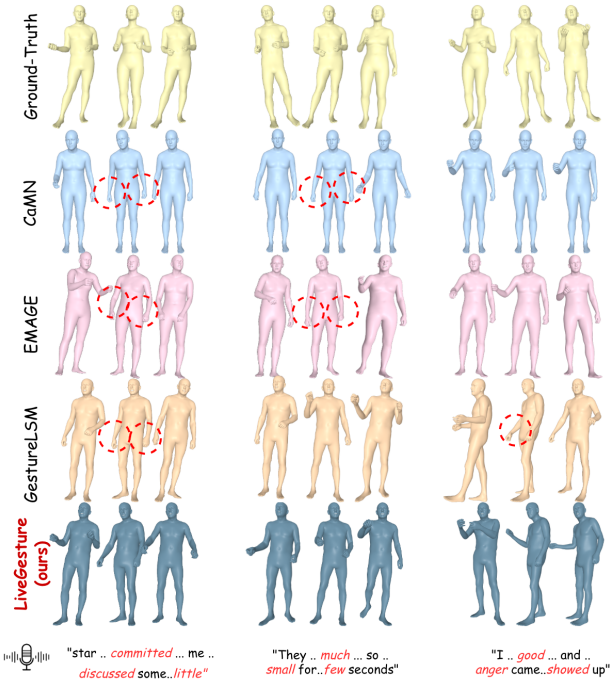

Figure 4. Qualitative comparison with state-of-the-art methods on BEAT. LiveGesture generates more diverse, expressive full-body gestures that better follow speech rhythm, while red circles mark representative failure cases of prior methods.

speech from 25 speakers (1,762 conversational clips, average length 65.66 s). The data cover diverse speaking styles and gesture behaviors. All experiments follow the official train/test split of EMAGE [18] for fair comparison.

Evaluation Metrics. We assess gesture quality in terms of realism, diversity, audio–motion alignment, and facial accuracy. Realism is measured by Fréchet Gesture Distance (FGD) [46]; diversity (Div.) [14] is the average L1 distance between generated clips for the same audio; audio–motion synchronization is evaluated by Beat Constancy (BC) [15]; and facial accuracy is measured by vertex MSE between predicted and ground-truth SMPL-X face vertices [42]. For readability, FGD and BC are scaled by 10^{-1} and MSE by 10^{-7} in all tables.

Table 1 compares LiveGesture with recent state-of-the-art co-speech gesture models on BEAT2. Although it is the *only* zero–look-ahead streaming method, LiveGesture remains competitive with or superior to offline systems: it achieves near-best Fréchet Gesture Distance (FGD; 4.57 vs. 4.25 for GestureLSM), the best Beat Constancy (BC = 0.794), the highest Diversity (13.91), and the second-lowest facial MSE (1.241 vs. 1.021), indicating tight rhythm alignment, rich motion variation, and high-fidelity full-body and facial motion under strict streaming constraints. We attribute this to three factors: (i) the streamable asymmet-

Table 1. State-of-the-art comparison on BEAT. Best results are shown in **bold** and second best are underlined. The *Streaming* column indicates whether the method supports zero-look-ahead streaming (✓) or is offline-only (✗); Our method is the only streaming model while remaining superior in most of important metrics.

Methods	Venue	Streaming	FGD (↓)	BC (→)	Diversity (↑)	MSE (↓)
Offline Solutions						
HA2G [24]	CVPR'22	✗	12.32	0.677	8.626	–
DisCo [16]	MM'22	✗	9.417	0.643	9.912	–
CaMN [17]	ECCV'22	✗	6.644	0.676	10.86	–
TalkShow [45]	CVPR'23	✗	6.209	0.695	13.47	7.791
DiffSHEG [4]	CVPR'24	✗	7.141	0.743	8.21	9.571
ProbTalk [25]	CVPR'24	✗	5.040	0.771	13.27	8.614
EMAGE [18]	CVPR'24	✗	5.512	0.772	13.06	7.680
MambaTalk [44]	NeurIPS'24	✗	5.366	0.781	13.05	7.680
SynTalker [3]	MM'24	✗	4.687	0.736	12.43	–
RAG-Gesture [27]	CVPR'25	✗	9.110	0.727	12.62	-
RAG-Gesture (w/ Discourse) [27]	CVPR'25	✗	8.790	0.739	12.62	-
GestureLSM [22]	ICCV'25	✗	**4.247**	0.729	13.76	**1.021**
Streaming Solution (Ours)						
LiveGesture (ours)	Ours	✓	4.57	**0.794**	**13.91**	1.241

ric motion tokenizer (SVQ), which learns a globally coherent latent space while decoding strictly causally, providing clean low-rate motion tokens; (ii) region-wise AR eXperts plus xAR-Fuse, which align frozen eXpert states and apply causal spatio-temporal attention for coherent whole-body coordination; and (iii) hybrid masked training on top of a strictly causal audio encoder, which exposes the model to realistic corrupted histories and yields robust audio–motion coupling. Together, these choices allow LiveGesture to achieve extremely low First-Token Latency (250 ms), the amount of time a model takes to generate the first motion frame in its response after receiving the first audio token.

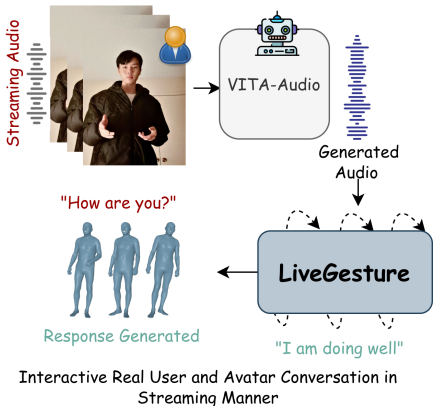

Figure 5. Interactive human–avatar conversation enabled by *LiveGesture*. User speech is converted into a spoken reply by VITA-Audio, while our LiveGesture streaming gesture model simultaneously generates synchronized full-body SMPL-X motions from live audio, allowing the avatar to respond in real time.

4.1. Ablation Studies

We ablate the key components of LiveGesture to understand which design choices allow a strictly causal, zero–look-ahead model to surpass offline methods. We vary (i) fusion architecture and region conditioning, (ii) motion tokenization and audio encoding, and (iii) streaming-aware training objectives and masking strategies. Quantitative results are summarized in Table 2. More

Effectiveness of LiveGesture Components. Table 2a shows that fusion-level temporal attention is essential: removing it causes the largest drop across all metrics (FGD 4.57→15.52, BC 0.794→0.712, Div. 13.97→10.40), confirming the need for causal temporal modeling beyond local eXperts. Spatial attention also clearly improves FGD and diversity (6.64 / 11.56 without it), highlighting the importance of inter-region coordination for full-body coherence. PILOR removal degrades FGD and BC (4.89 / 0.774), indicating that low-rank pre-infusion alignment stabilizes fusion over frozen eXperts. Disabling UGR yields the worst BC (0.723) and higher FGD (4.98), showing that uncertainty-guided refinement mitigates exposure bias under streaming errors. Text removal leaves FGD and diversity almost unchanged and slightly improves BC (0.796), suggesting that audio is the primary driver of rhythm, with text providing modest semantic/style refinements.

Region-eXperts vs. Full xAR-Fuse. Table 2b compares local region-eXperts alone to the full architecture. eXperts alone achieve reasonable quality (FGD 6.458, BC 0.762, Div. 12.844) but underperform LiveGesture. Adding xAR-Fuse on top of frozen eXperts improves all metrics (FGD 4.57, BC 0.794, Div. 13.91), validating our hierarchical design: specialized region-eXperts provide strong local priors,

Table 2. Ablation studies of LiveGesture design choices, including core components, architecture, tokenization, audio encoders, loss weights, and UGR.

Configuration	FGD↓	BC→	Div.↑
w/o PILOR	4.89	0.774	13.41
w/o UGR	4.98	0.723	13.64
w/o spatial attn.	6.64	0.732	11.56
w/o temporal attn.	15.52	0.712	10.40
w/o text cues	4.60	**0.796**	13.96
LiveGesture (ours)	**4.57**	0.794	**13.97**

(a) Effectiveness of LiveGesture components

Architecture	FGD↓	BC→	Div.↑
Local region eXperts only	6.458	0.762	12.844
LiveGesture (ours)	**4.57**	**0.794**	**13.91**

(b) Local experts vs. full LiveGesture

Stage-2 trainable components	FGD↓
1-stage (Enc+Dec+Quant)	4.691
AE + 2-stage (Dec+Quant)	4.633
AE + 2-stage (Dec+Quant+MLP)	4.610
AE + 2-stage (Quant+MLP)	**4.557**

(c) Effect of which SVQ components are trainable in Stage 2 after autoencoder pretraining on xAR-Fuse performance.

Model Type	FGD↓	BC→	Params (M)↓
Magic-Codec [37]	**4.51**	0.793	103
FocalCodec [6]	4.95	0.773	88
ours	4.57	**0.794**	**0.5**

(d) Streamable audio encoders

λ_{local}	λ_{fuse}	FGD↓	BC→
0.0	1.0	4.89	**0.795**
0.3	1.0	**4.57**	0.794
0.5	1.0	4.67	0.781
0.7	1.0	4.74	0.783
1.0	1.0	4.85	0.774

(e) Role of Local Expert Loss in $\mathcal{L}_{\text{AR}}$.

UGM masking schedule	FGD↓	BC→	Div.↑
$\gamma(\tau \in \mathcal{U}(0,1))$	4.63	0.775	12.95
$\gamma(\tau \in \mathcal{U}(0,0.3))$	4.84	0.792	13.24
$\gamma(\tau \in \mathcal{U}(0,0.5))$	**4.57**	**0.794**	**13.91**
$\gamma(\tau \in \mathcal{U}(0,0.7))$	4.67	0.773	13.60

(f) Effect of masking schedule in uncertainty-guided masking (UGM).

while a dedicated causal fusion stage is needed to turn them into globally coherent streaming motion.

Effect of SVQ on xAR-Fuse. Table 2c examines which SVQ components are trainable in Stage 2. Jointly training encoder, decoder, and quantizer (1-stage) yields the worst FGD (4.691), indicating that learning compression and causal decoding simultaneously is suboptimal. In the two-stage setting, performance improves as we freeze more of the autoencoder, with the best FGD when only the Quantizer+MLP are updated and the decoder is fully frozen (4.557). This supports our SVQ design (Sec. 3.1): first learn a stable, streamable latent space, then adapt a lightweight quantizer and projection on top without altering the decoder.

Streamable Audio Encoder. Table 2d compares our lightweight Steamable Audio Encoder with heavier pre-trained neural codecs. Magic-Codec slightly improves FGD (4.51 vs. 4.57) but uses 103M params, whereas our audio encoder attains near-identical FGD, the best BC (0.794), and only 0.5M parameters; FocalCodec is both larger and less accurate. This shows that a lightweight, task-specific, strictly causal encoder aligned to the motion token rate is sufficient to generalize well and deliver high-quality, low-latency gesture streaming without the overhead of generic codecs.

Role of Local Expert Loss. Table 2e ablates the weights in $\mathcal{L}_{\text{AR}} = \lambda_{\text{local}}\mathcal{L}_{\text{local}} + \lambda_{\text{fuse}}\mathcal{L}_{\text{fuse}}$. Using only fusion loss ($\lambda_{\text{local}} = 0$) yields good BC but worse FGD, so xAR-Fuse alone under-regularizes regional dynamics. A small local weight ($\lambda_{\text{local}} = 0.3$) gives the best FGD while preserving BC, whereas larger values hurt both metrics, indicating that region-eXperts work best as lightly guided priors with xAR-Fuse handling final coordination.

Masking Ratio in Uncertainty-Guided Refinement (UGR). Table 2f evaluates different UGR masking schedules. Very broad or very narrow masking ranges lead to worse FGD or BC, while our default schedule $\gamma(\tau \in \mathcal{U}(0, 0.5))$ achieves the best trade-off (FGD 4.57, BC 0.794, Div. 13.91). This suggests that UGR is most effective when it injects moderate corruption that mimics streaming errors without making refinement unstable.

4.2. Application: Interactive Human–Avatar Conversation

LiveGesture can be plugged into streaming speech agents such as VITA-Audio [26] to enable fully interactive human–avatar conversations. In our prototype, user speech is converted by VITA-Audio into a spoken response whose waveform is streamed into our xAR-based gesture model, driving synchronized full-body SMPL-X motion in real time. The same pipeline can be extended with virtual clothing or character skins over the generated motion, enabling expressive avatars for games, VTubers, and telepresence.

5. Conclusion

We introduce *LiveGesture*, a zero–look-ahead, fully streamable co-speech full-body gesture framework that integrates a streamable asymmetric motion tokenizer with hierarchical region-eXpert autoregression. A per-region SVQ tokenizer first produces causal motion tokens; local audio-conditioned eXperts model region-specific dynamics, while xAR-Fuse performs causal spatio-temporal fusion across regions. Hybrid uncertainty-guided masking, along with random region masking strategies, trains the model to refine low-confidence predictions under realistic streaming noise. Together, these components enable diverse, coherent, and beat-synchronous gesture generation over arbitrarily long sequences, making *LiveGesture* well suited for interactive applications such as live avatars, VTubers, telepresence agents, social robots, and AR/VR characters

References

[1] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, T. J. Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeff Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. *ArXiv*, abs/2005.14165, 2020. 2

[2] Caroline Chan, Shiry Ginosar, Tinghui Zhou, and Alexei A Efros. Everybody Dance Now. In *ICCV*, 2019. 2

[3] Bohong Chen, Yumeng Li, Yao-Xiang Ding, Tianjia Shao, and Kun Zhou. Enabling synergistic full-body control in prompt-based co-speech motion generation. In *Proceedings of the 32nd ACM International Conference on Multimedia*, page 10, New York, NY, USA, 2024. ACM. 2, 7

[4] Junming Chen, Yunfei Liu, Jianan Wang, Ailing Zeng, Yu Li, and Qifeng Chen. DiffSHEG: A Diffusion-Based Approach for Real-Time Speech-driven Holistic 3D Expression and Gesture Generation, 2024. 2, 7

[5] Anna Deichler, Shivam Mehta, Simon Alexanderson, and Jonas Beskow. Diffusion-based co-speech gesture generation using joint text and audio representation. In *INTERNATIONAL CONFERENCE ON MULTIMODAL INTERACTION*. ACM, 2023. 2

[6] Luca Della Libera, Cem Subakan, and Mirco Ravanelli. Focalcodec-stream: Streaming low-bitrate speech coding via causal distillation. *arXiv preprint arXiv:2509.16195*, 2025. 8

[7] Patrick Esser, Robin Rombach, and Björn Ommer. Taming transformers for high-resolution image synthesis. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12868–12878, 2020. 2

[8] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12873–12883, 2021. 4

[9] S. Ginosar, A. Bar, G. Kohavi, C. Chan, A. Owens, and J. Malik. Learning Individual Styles of Conversational Gesture. In *CVPR*. IEEE, 2019. 2

[10] Chuan Guo, Yuxuan Mu, Muhammad Gohar Javed, Sen Wang, and Li Cheng. Momask: Generative masked modeling of 3d human motions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1900–1910, 2024. 2

[11] Ikhsanul Habibie, Weipeng Xu, Dushyant Mehta, Lingjie Liu, Hans-Peter Seidel, Gerard Pons-Moll, Mohamed Elgharib, and Christian Theobalt. Learning speech-driven 3d conversational gestures from video. *arXiv preprint arXiv:2102.06837*, 2021. 2

[12] Chao Huang, Dejan Markovic, Chenliang Xu, and Alexander Richard. Modeling and driving human body soundfields through acoustic primitives, 2024. 2

[13] Biao Jiang, Xin Chen, Wen Liu, Jingyi Yu, Gang Yu, and Tao Chen. Motiongpt: Human motion as a foreign language. *ArXiv*, abs/2306.14795, 2023. 2

[14] Jing Li, Di Kang, Wenjie Pei, Xuefei Zhe, Ying Zhang, Zhenyu He, and Linchao Bao. Audio2gestures: Generating diverse gestures from speech audio with conditional variational autoencoders. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11293–11302, 2021. 6, 13

[15] Ruilong Li, Shan Yang, David A Ross, and Angjoo Kanazawa. AI Choreographer: Music Conditioned 3D Dance Generation with AIST++. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13401–13412, 2021. 6, 14

[16] Haiyang Liu, Naoya Iwamoto, Zihao Zhu, Zhengqing Li, You Zhou, Elif Bozkurt, and Bo Zheng. DisCo: Disentangled Implicit Content and Rhythm Learning for Diverse Co-Speech Gestures Synthesis. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 3764–3773, 2022. 7

[17] Haiyang Liu, Zihao Zhu, Naoya Iwamoto, Yichen Peng, Zhengqing Li, You Zhou, Elif Bozkurt, and Bo Zheng. BEAT: A Large-Scale Semantic and Emotional Multi-Modal Dataset for Conversational Gestures Synthesis. *arXiv preprint arXiv:2203.05297*, 2022. 2, 7, 14, 15

[18] Haiyang Liu, Zihao Zhu, Giorgio Becherini, Yichen Peng, Mingyang Su, You Zhou, Naoya Iwamoto, Bo Zheng, and Michael J Black. EMAGE: Towards Unified Holistic Co-Speech Gesture Generation via Masked Audio Gesture Modeling. *arXiv preprint arXiv:2401.00374*, 2023. 2, 6, 7, 14, 15

[19] Haiyang Liu, Xingchao Yang, Tomoya Akiyama, Yuantian Huang, Qiaoge Li, Shigeru Kuriyama, and Takafumi Taketomi. Tango: Co-speech gesture video reenactment with hierarchical audio motion embedding and diffusion interpolation. *arXiv preprint arXiv:2410.04221*, 2024. 2

[20] Lanmiao Liu, Esam Ghaleb, Aslı Özyürek, and Zerrin Yumak. Semges: Semantics-aware co-speech gesture generation using semantic coherence and relevance learning, 2025.

[21] Pinxin Liu, Haiyang Liu, Luchuan Song, and Chenliang Xu. Intentional gesture: Deliver your intentions with gestures for speech, 2025. 2

[22] Pinxin Liu, Luchuan Song, Junhua Huang, and Chenliang Xu. Gesturelsm: Latent shortcut based co-speech gesture generation with spatial-temporal modeling. In *IEEE/CVF International Conference on Computer Vision*, 2025. 2, 7, 14, 15

[23] Pinxin Liu, Pengfei Zhang, Hyeongwoo Kim, Pablo Garrido, Ari Sharpio, and Kyle Olszewski. Contextual gesture: Co-speech gesture video generation through context-aware gesture representation, 2025. 2

[24] Xian Liu, Qianyi Wu, Hang Zhou, Yinghao Xu, Rui Qian, Xinyi Lin, Xiaowei Zhou, Wayne Wu, Bo Dai, and Bolei Zhou. Learning Hierarchical Cross-Modal Association for Co-Speech Gesture Generation. In *CVPR*, pages 10462–10472, 2022. 2, 7

[25] Yifei Liu, Qiong Cao, Yandong Wen, Huaiguang Jiang, and Changxing Ding. Towards variable and coordinated

holistic co-speech motion generation. *arXiv preprint arXiv:2404.00368*, 2024. 7

[26] Zuwei Long, Yunhang Shen, Chaoyou Fu, Heting Gao, Lijiang Li, Peixian Chen, Mengdan Zhang, Hang Shao, Jian Li, Jinlong Peng, et al. Vita-audio: Fast interleaved cross-modal token generation for efficient large speech-language model. *arXiv preprint arXiv:2505.03739*, 2025. 8

[27] M Hamza Mughal, Rishabh Dabral, Merel CJ Scholman, Vera Demberg, and Christian Theobalt. Retrieving semantics from the deep: an rag solution for gesture synthesis. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 16578–16588, 2025. 7

[28] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed AA Osman, Dimitrios Tzionas, and Michael J Black. Expressive body capture: 3d hands, face, and body from a single image. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10975–10985, 2019. 3

[29] Ekkasit Pinyoanuntapong, Muhammad Usama Saleem, Pu Wang, Minwoo Lee, Srijan Das, and Chen Chen. Bamm: Bidirectional autoregressive motion model. In *Computer Vision – ECCV 2024*, 2024. 2

[30] Ekkasit Pinyoanuntapong, Pu Wang, Minwoo Lee, and Chen Chen. Mmm: Generative masked motion model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 2

[31] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. *ArXiv*, abs/2102.12092, 2021. 2

[32] Luchuan Song, Bin Liu, and Nenghai Yu. Talking face video generation with editable expression. In *Image and Graphics: 11th International Conference, ICIG 2021, Haikou, China, August 6–8, 2021, Proceedings, Part III 11*, pages 753–764. Springer, 2021. 2

[33] Luchuan Song, Guojun Yin, Bin Liu, Yuhui Zhang, and Nenghai Yu. Fsft-net: face transfer video generation with few-shot views. In *2021 IEEE international conference on image processing (ICIP)*, pages 3582–3586. IEEE, 2021. 2

[34] Luchuan Song, Lele Chen, Celong Liu, Pinxin Liu, and Chenliang Xu. Texttoon: Real-time text toonify head avatar from single video. In *SIGGRAPH Asia 2024 Conference Papers*, pages 1–11, 2024. 2

[35] Luchuan Song, Pinxin Liu, Lele Chen, Guojun Yin, and Chenliang Xu. Tri 2-plane: Thinking head avatar via feature pyramid. In *European Conference on Computer Vision*, pages 1–20. Springer, 2024.

[36] Luchuan Song, Pinxin Liu, Guojun Yin, and Chenliang Xu. Adaptive super resolution for one-shot talking-head generation. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4115–4119, 2024. 2

[37] Yakun Song, Jiawei Chen, Xiaobin Zhuang, Chenpeng Du, Ziyang Ma, Jian Wu, Jian Cong, Dongya Jia, Zhuo Chen, Yuping Wang, et al. Magicodec: Simple masked gaussian-injected codec for high-fidelity reconstruction and generation. *arXiv preprint arXiv:2506.00385*, 2025. 8

[38] Yunlong Tang, Junjia Guo, Pinxin Liu, Zhiyuan Wang, Hang Hua, Jia-Xing Zhong, Yunzhong Xiao, Chao Huang, Luchuan Song, Susan Liang, et al. Generative ai for cel-animation: A survey. *arXiv preprint arXiv:2501.06250*, 2025. 2

[39] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017. 4

[40] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-Resolution Image Synthesis and Semantic Manipulation with Conditional GANs. In *CVPR*, 2018. 2

[41] Will Williams, Sam Ringer, Tom Ash, John Hughes, David Macleod, and Jamie Dougherty. Hierarchical quantized autoencoders. *ArXiv*, abs/2002.08111, 2020. 2

[42] Jinbo Xing, Menghan Xia, Yuechen Zhang, Xiaodong Cun, Jue Wang, and Tien-Tsin Wong. Codetalker: Speech-driven 3d facial animation with discrete motion prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12780–12790, 2023. 6, 14

[43] Zunnan Xu, Yachao Zhang, Sicheng Yang, Ronghui Li, and Xiu Li. Chain of generation: Multi-modal gesture synthesis via cascaded conditional control, 2023. 2

[44] Zunnan Xu, Yukang Lin, Haonan Han, Sicheng Yang, Ronghui Li, Yachao Zhang, and Xiu Li. Mambatalk: Efficient holistic gesture synthesis with selective state space models, 2024. 2, 7

[45] Hongwei Yi, Hualin Liang, Yifei Liu, Qiong Cao, Yandong Wen, Timo Bolkart, Dacheng Tao, and Michael J Black. Generating Holistic 3D Human Motion from Speech. In *CVPR*, 2023. 2, 7

[46] Youngwoo Yoon, Bok Cha, Joo-Haeng Lee, Minsu Jang, Jaeyeon Lee, Jaehong Kim, and Geehyuk Lee. Speech Gesture Generation from the Trimodal Context of Text, Audio, and Speaker Identity. *ACM TOG*, 39(6), 2020. 6, 13

[47] Jiahui Yu, Xin Li, Jing Yu Koh, Han Zhang, Ruoming Pang, James Qin, Alexander Ku, Yuanzhong Xu, Jason Baldridge, and Yonghui Wu. Vector-quantized image modeling with improved vqgan. *ArXiv*, abs/2110.04627, 2021. 2

[48] Jianrong Zhang, Yangsong Zhang, Xiaodong Cun, Shaoli Huang, Yong Zhang, Hongwei Zhao, Hongtao Lu, and Xiaodong Shen. Generating human motion from textual descriptions with discrete representations. *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14730–14740, 2023. 2

[49] Pengfei Zhang, Pinxin Liu, Hyeongwoo Kim, Pablo Garrido, and Bindita Chaudhuri. Kinmo: Kinematic-aware human motion understanding and generation, 2024. 2

[50] Chongyang Zhong, Lei Hu, Zihao Zhang, and Shihong Xia. Attt2m: Text-driven human motion generation with multi-perspective attention mechanism. *ArXiv*, abs/2309.00796, 2023. 2

1. Supplementary Material

A. Overview

The supplementary material is organized as follows:

- Section B: Implementation Details of the Streamable Vector-Quantized Motion Tokenizer (SVQ) and the Hierarchical Autoregressive Transformer (HAR)
- Section C: Evaluation Metrics
- Section D: SOTA Comparison: Quantitative Results Without Face Module
- Section E: SOTA Comparison: User Study
- Section F: Impact of Codebook Size in Streamable Vector-Quantized Motion Tokenizer
- Section G: Causal Attention Design in xAR and xAR-Fuse
- Section H: Role of Classifier-Free Guidance in *LiveGesture*
- Section I: Role of Compositional vs. Full-Body SVQ Tokenization
- Section J: Ablation on Role of Region Masking in Fusion Training
- Section K: Impact of Noise Injection in Local Region-eXperts
- Section L: Ablation on Attention Depth in xAR and xAR-Fuse

B. Implementation Details

LiveGesture is implemented in PyTorch and trained in two major phases: (i) the per-region *Streamable Vector-Quantized Motion Tokenizer* (SVQ), and (ii) the *Hierarchical Autoregressive Transformer* (HAR).

B.1. Streamable Vector-Quantized Motion Tokenizer

Asymmetric Autoencoder Pretraining.. We train one *Streamable Autoencoder* (SA) per SMPL-X body region, $\mathcal{R} = \{\text{upper body}, \text{lower body}, \text{hands}, \text{face}\}$. The input parameterization for each region is as follows: 13 upper-body joints (78-D Rot6D), 30 hand joints (180-D Rot6D), 100-D FLAME expression parameters plus 3-D jaw rotation for the face, and 9 lower-body joints (54-D Rot6D) augmented with global translation (3-D) and four binary foot-contact indicators. Global translation and contact flags are included only in the lower-body region, as leg motion and foot contacts provide strong supervision for root displacement and help reduce foot-sliding artifacts.

For training, each SA processes motion in fixed sliding windows of length $T_w = 16$ frames, denoted $\{\boldsymbol{\theta}_t^{\text{region}}\}_{t=1}^{T_w}$ for a given region. The bidirectional encoder E begins with a temporal convolutional stem that lifts framewise SMPL-X inputs into a higher-dimensional feature space. It then applies two downsampling stages, each consisting of a stride-2 temporal convolution (halving the frame rate) followed by a `ResNet1D` block composed of dilated temporal residual layers. These residual layers jointly capture local kinematic patterns and longer-range bidirectional dependencies within the window. After two stages, the encoder outputs a latent sequence

$$z^{\text{region}} = \{z_\tau\}_{\tau=1}^{T_w/4},$$

i.e., a compact motion representation at one-quarter of the original frame rate. This asymmetric design allows E to exploit both past and future frames to build a globally coherent latent space, while the decoder remains strictly causal.

The Causal Stream Decoder D_{CS} mirrors the encoder hierarchy but enforces strict causality: all convolutions use left-only padding, so the reconstruction at frame t depends only on latent tokens $\{z_{1:\tau}^{\text{region}}\}$ up to the corresponding downsampled index. Each upsampling stage performs nearest-neighbor temporal upsampling by 2, followed by a causal `ResNet1D` block that refines the feature sequence using only past context. After two stages, the decoder reconstructs the full-resolution regional motion window

$$\hat{\boldsymbol{\theta}}^{\text{region}} = D_{\text{CS}}(z^{\text{region}}).$$

Because D_{CS} is strictly causal and operates at a fixed downsampling factor, the same architecture can be run convolutionally over arbitrarily long sequences at inference time without violating the zero–look-ahead constraint. In the final streaming system, only the causal decoder D_{CS} is used at inference; the bidirectional encoder E is employed solely during offline training to shape the latent space.

Each SA is trained independently with per-region normalization and an L1 reconstruction loss

$$\mathcal{L}_{\text{AE}} = \lambda_{\text{AE}} \left\| \hat{\boldsymbol{\theta}}^{\text{region}} - \boldsymbol{\theta}^{\text{region}} \right\|_1, \quad \lambda_{\text{AE}} = 1.$$

This stage focuses solely on learning a temporally coherent, streamable latent space and does not involve any vector quantization.

Quantization Learning.. In Stage 2, we convert the continuous latents z^{region} into discrete, time-synchronous SVQ motion tokens while preserving the temporal geometry and causal decoding behavior learned in Stage 1. To this end, we freeze both the bidirectional encoder E and the causal decoder D_{CS} and learn a region-specific vector-quantized tokenizer on top of the SA latents.

For each region $r \in \mathcal{R}$, we introduce a codebook

$$C^{\text{region}} = \{c_k\}_{k=1}^{K} \in \mathbb{R}^{K\times 128},$$

with $K = 2048$ entries, updated using EMA with decay 0.99. The encoder latents z_τ^{region} are first linearly projected into the 128-D code space. Each projected latent is then assigned to its nearest codebook vector c_k, producing a discrete token index and the corresponding dequantized embedding sequence

$$\hat{z}^{\text{region}} = \{\hat{z}_\tau\}_{\tau=1}^{T_w/4}.$$

To remain compatible with the frozen decoder latent space, each region employs a lightweight projection head W^{region}, implemented as a small MLP. This head maps dequantized latents back into the latent space expected by D_{CS}:

$$\tilde{z}^{\text{region}} = \{\tilde{z}_\tau\}_{\tau=1}^{T_w/4}, \quad \tilde{z}_\tau = W^{\text{region}}(\hat{z}_\tau).$$

The projected sequence is then decoded causally to reconstruct the regional motion window,

$$\hat{\boldsymbol{\theta}}^{\text{region}} = D_{\text{CS}}(\tilde{z}^{\text{region}}).$$

In this stage, only the codebook C^{region} and projection head W^{region} are trainable; E and D_{CS} remain fixed. The codebook is maintained with EMA and we periodically reset rarely used entries to avoid codebook collapse and encourage effective usage of the discrete space. The Stage 2 objective combines an L1 reconstruction term on SMPL-X region parameters with a standard VQ codebook loss:

$$\mathcal{L}_{\text{stage2}} = \lambda_{\text{rec}} \left\| \hat{\boldsymbol{\theta}}^{\text{region}} - \boldsymbol{\theta}^{\text{region}} \right\|_1 + \lambda_{\text{cb}}\, \mathcal{L}_{\text{cb}}(z^{\text{region}}, e^{\text{region}}),$$

where e^{region} denotes the selected codebook embeddings, $\lambda_{\text{rec}} = 1$, and $\lambda_{\text{cb}} = 0.2$. Gradients pass through the quantizer via a straight-through estimator. This two-stage asymmetric design ensures that (i) the latent space and causal decoder remain stable and well-conditioned for streaming, and (ii) the SVQ tokens are compact, region-specific, and time-synchronous at one-quarter of the original motion frame rate, making them ideal discrete inputs for the downstream region-eXpert autoregressive transformers in HAR.

B.2. Hierarchical Autoregressive Transformer (HAR)

Region-eXpert Autoregressive Transformer (xAR).. Each *region-eXpert* xAR module performs causal autoregressive modeling on the SVQ motion tokens produced at one-quarter of the original motion frame rate. For every region $r \in \mathcal{R} = \{\text{upper body}, \text{lower body}, \text{hands}, \text{face}\}$, we maintain an independent stream of SVQ tokens together with aligned audio tokens from the streamable audio encoder and optional text tokens. The model operates with a maximum history of 32 past motion tokens, each mapped to a 128-D codebook embedding ($K = 2048$ entries) and projected to a 256-D representation through a small MLP; rotary positional embeddings are added for stable temporal alignment during streaming. Each region-eXpert is implemented as a lightweight causal Transformer with $L_{\text{xAR}} = 2$ blocks, where every block contains three *causal audio–motion cross-attention* layers that attend only to past and current audio/text tokens, followed by three *causal temporal self-attention* layers applied over the region's token history; all attention layers use strict lower-triangular masks to ensure zero–look-ahead. The final hidden state at time t is passed through a region-specific linear classifier to produce logits over the 2048-entry codebook vocabulary. To improve robustness, we inject Gaussian noise with standard deviation $\sigma_{\text{noise}} = 0.1$ into the embedded motion history with probability $p_{\text{noise}} = 0.2$, and apply classifier-free dropout to audio/text inputs with probability $p_{\text{cf}} = 0.1$, enabling classifier-free guidance at inference with scale $\gamma = 1.25$. All four region-eXperts share the same streamable audio encoder but maintain independent Transformer and classifier weights, and are trained using Adam (learning rate 1×10^{-4}, batch size 128) under the standard autoregressive objective.

Causal Spatial–Temporal Fusion (xAR-Fuse)..

xAR-Fuse enforces whole-body coordination by operating on top of the frozen hidden states produced by the region-eXpert xAR modules. At each time step t, we gather the region-wise features $\{h_t^r\}_{r\in\mathcal{R}}$ and align them using lightweight per-region PILOR adapters, each implemented as a single linear layer with a residual connection. These adapters add only a small number of parameters per region while reliably mapping independently learned region features into a shared fusion space. The aligned features are then processed by a causal fusion Transformer with $L_{\text{fuse}} = 3$ blocks. Each fusion block contains three components: (i) *inter-region spatial attention* across regions at the current time step, (ii) *causal global temporal attention* with key–value caching for long-horizon streaming, and (iii) *causal audio–motion cross-attention* that conditions fused region tokens on past and current audio/text input. The resulting fused representations are fed into the same region-specific token classifiers used in the local xAR stage, now predicting SVQ tokens from joint multi-region context.

Training uses the hybrid masking strategy described in the main method: uncertainty-guided token masking (UGM) corrupts a subset of the lowest-confidence tokens according to a cosine schedule $\lambda_{\text{UGR}}(s)$, where the effective masking ratio increases from 0 to a maximum of 0.5 over the course of training; for each batch, a masking ratio is sampled uniformly in $[0, \lambda_{\text{UGR}}(s)]$ and applied to the selected tokens. Random region masking (RM) is implemented in an analogous way: we gradually increase a region-drop probability p_{drop} from 0 to 0.5, and for each batch sample a value of p_{drop} within the current range and use it as the probability of dropping an entire region's motion-token sequence, encouraging cross-region reasoning under missing modalities. We also apply classifier-free dropout with probability $p_{\text{cf}} = 0.1$, identical to xAR, to retain compatibility with inference-time classifier-free guidance. During xAR-Fuse training, the SVQ tokenizer, causal audio encoder, and all xAR eXperts are frozen; only the PILOR adapters, fusion Transformer blocks, and token classifiers are updated. We train xAR-Fuse using Adam (learning rate 1×10^{-4}, batch size 128). Because all attention operations are strictly causal, the resulting model runs directly in real-time streaming mode without any architectural changes. In the overall training objective, we weight the local xAR loss and the fusion loss with coefficients $\lambda_{\text{local}} = 0.3$ and $\lambda_{\text{fuse}} = 1.0$, respectively.

B.3. Application: Interactive Human–Avatar Conversation

For deployment, *LiveGesture* connects directly to a streaming speech system such as VITA-Audio. Audio is emitted in small chunks (200 ms hop) and immediately passed to the causal audio encoder, which updates the audio tokens at the SVQ rate. HAR advances synchronously, generating one SVQ token per region at each audio step. The SVQ decoder reconstructs full SMPL-X poses in real time, enabling fully synchronized speech–gesture behavior. Because the entire HAR stack is strictly causal, the same implementation supports real-time human–avatar interaction without look-ahead or buffering.

C. Evaluation Metrics

We evaluate *LiveGesture* using four standard metrics that measure realism, variability, speech–motion synchrony, and facial accuracy, following the official BEAT2 protocol.

Fréchet Gesture Distance (FGD).. FGD [46] measures the distributional similarity between real and generated full-body motion in the feature space of a pretrained gesture encoder. Let feature sets extracted from real and generated gestures be $G = \{\mathbf{g}_i\}$ and $\hat{G} = \{\hat{\mathbf{g}}_i\}$, with means $\boldsymbol{\mu}_G,\ \boldsymbol{\mu}_{\hat{G}}$ and covariances $\Sigma_G,\ \Sigma_{\hat{G}}$. FGD is defined as

$$\text{FGD}(G, \hat{G}) = \|\boldsymbol{\mu}_G - \boldsymbol{\mu}_{\hat{G}}\|_2^2 + \text{Tr}\Big(\Sigma_G + \Sigma_{\hat{G}} - 2(\Sigma_G\Sigma_{\hat{G}})^{1/2}\Big)\,. \tag{1}$$

Lower FGD indicates that generated gestures follow natural human-motion statistics, which is critical for strictly causal, zero–look-ahead generation.

L1 Diversity.. L1 Diversity [14] measures variability across multiple gesture realizations produced for the same audio. Given N generated sequences with corresponding joint positions $\{\mathbf{p}_t^{(i)}\}_{t=1}^T$ for $i = 1, \ldots, N$, the diversity score is

$$\text{Div.} = \frac{1}{2N(N-1)T} \sum_{i=1}^{N} \sum_{\substack{j=1 \\ j\neq i}}^{N} \sum_{t=1}^{T} \big\|\mathbf{p}_t^{(i)} - \mathbf{p}_t^{(j)}\big\|_1. \tag{2}$$

Global translation of the SMPL-X body is removed prior to evaluation. Higher values indicate more expressive and varied motion under causal token-by-token prediction.

Beat Constancy (BC).. Beat Constancy [15] evaluates the synchrony between motion beats and prosodic beats in the audio. Motion beats are detected from local minima of upper-body joint velocity, while audio beats correspond to peaks in prosodic intensity. Let g_{mot} and a_{aud} denote the sets of detected motion and audio beat times. BC is computed as

$$\text{BC} = \frac{1}{|g_{\text{mot}}|} \sum_{b_g \in g_{\text{mot}}} \exp\left(-\frac{\min_{b_a \in a_{\text{aud}}} \|b_g - b_a\|^2}{2\sigma^2}\right), \tag{3}$$

where σ is a temporal tolerance parameter. Higher BC indicates stronger speech–gesture alignment. Because *LiveGesture* is fully causal with zero–look-ahead, BC directly reflects its ability to track prosody in real time.

Facial Vertex MSE.. For SMPL-X facial motion accuracy, we compute the mean squared error between ground-truth and predicted mesh vertices following [42]. Let $\mathbf{V} = \{\mathbf{v}_i\}$ and $\hat{\mathbf{V}} = \{\hat{\mathbf{v}}_i\}$ be the sets of ground-truth and predicted facial vertices. The metric is

$$\text{MSE}_{\text{face}} = \frac{1}{|\mathbf{V}|} \sum_{i=1}^{|\mathbf{V}|} \|\mathbf{v}_i - \hat{\mathbf{v}}_i\|_2^2. \tag{4}$$

This complements body-motion metrics by evaluating fine-grained facial deformation fidelity.

D. SOTA Comparison: Quantitative Results Without Face Module

Table 1. Comparison with state-of-the-art methods on BEAT2 without the facial motion module. *LiveGesture* remains the only zero–look-ahead streaming model while achieving competitive or superior performance in BC and Diversity.

Methods	Venue	Streaming	FGD (↓)	BC (→)	Diversity (↑)
Offline Solutions					
GestureLSM [22]	ICCV'25	✗	**4.088**	0.714	<u>13.24</u>
Streaming Solution (Ours)					
LiveGesture (ours)	Ours	✓	<u>4.51</u>	**0.783**	**13.31**

Table 1 reports quantitative results when evaluating only full-body motion without the facial module. GestureLSM, an offline model with full future context, achieves the lowest FGD by leveraging non-causal temporal information. In contrast, *LiveGesture* operates under strict zero–look-ahead constraints, yet obtains the best BC and highest Diversity, indicating superior rhythm alignment and richer motion variability. This behavior reflects the strength of *LiveGesture*'s causal audio conditioning and hierarchical xAR–xAR-Fuse architecture, which jointly enable expressive, beat-synchronous gestures even without access to future frames. Although the absence of facial cues slightly increases the distributional distance (FGD) for our model, the strong improvements in synchrony and diversity demonstrate that *LiveGesture* maintains high perceptual quality while remaining the only fully streamable solution.

E. SOTA Comparison: User Study

Protocol. We recruit fifteen participants and present them with a mixed set of gesture clips generated by CaMN [17], EMAGE [18], GestureLSM [22], and *LiveGesture*. Each clip depicts a speaking subject with full-body SMPL-X motion driven by BEAT2 test utterances. All videos are anonymized and randomly ordered so that participants cannot identify which method produced which sequence, reducing bias toward any specific model.

For every clip, participants provide ratings on a five-point Likert scale (1 = lowest, 5 = highest) along three criteria. *Realness* evaluates the naturalness and plausibility of the produced motion. *Speech–gesture synchrony* measures how well the gestures align with the rhythm and prosodic structure of the audio. *Smoothness* assesses temporal continuity and penalizes jitter, discontinuities, or incoherent regional coordination. We report Mean Opinion Scores (MOS) averaged across all participants and clips for each method.

Results. Table 2 summarizes the MOS results. Both *LiveGesture* and GestureLSM are clearly preferred over CaMN and EMAGE across all criteria, indicating that recent models produce noticeably more convincing co-speech motion. GestureLSM, an offline method with full-sequence access, achieves slightly higher Realness (4.2 vs. 4.1) and Smoothness (4.3

Table 2. Mean Opinion Scores (MOS, 1–5, higher is better) from the user study on BEAT2 test clips. The *Streaming* column indicates whether the method supports zero–look-ahead streaming (✓) or is offline-only (✗). *LiveGesture* is the only strictly streaming model and is preferred on speech–gesture synchrony while remaining competitive in realness and smoothness.

Method	Streaming	Realness ↑	Synchrony ↑	Smoothness ↑
CaMN [17]	✗	3.3	3.2	3.6
EMAGE [18]	✗	3.5	3.5	3.7
GestureLSM [22]	✗	**4.2**	4.1	**4.3**
LiveGesture (ours)	✓	4.1	**4.3**	3.9

vs. 3.9) than *LiveGesture*, reflecting its advantage in long-range temporal refinement when future frames are available. In contrast, *LiveGesture* attains the best speech–gesture synchrony score (4.3), surpassing all offline baselines and aligning with its superior BC metric in the quantitative evaluation. This pattern suggests that our strictly causal xAR–xAR-Fuse architecture, together with the streamable audio encoder, is particularly effective at tracking prosody and timing under zero–look-ahead constraints, while still delivering realism and smoothness comparable to the strongest offline system. Overall, the user study confirms that a fully streaming model can match or nearly match the perceptual quality of state-of-the-art offline methods, while providing better perceived synchrony with speech.

F. Impact of Codebook Size in Streamable Vector-Quantized Motion Tokenizer

Table 3. Effect of codebook size in the SVQ motion tokenizer on full-body gesture generation. Larger codebooks increase representational capacity and yield consistent gains across all metrics.

Codebook size	FGD (↓)	BC (→)	Diversity (↑)
512 entries	6.63	0.734	12.14
1024 entries	5.13	0.774	13.21
2048 entries	4.51	0.794	13.91

Table 3 examines how the size of the region-specific codebook $C^{\text{region}} = \{c_k\}_{k=1}^{K}$ in the SVQ motion tokenizer affects downstream gesture generation. With a small codebook (K=512), many continuous latents z_τ^{region} are mapped to the same discrete code, leading to quantization collisions that restrict the expressiveness of the motion token sequence $\{x_t^r\}$. This loss of granularity degrades realism (higher FGD) and weakens prosodic alignment (lower BC), as subtle variations in timing and amplitude cannot be preserved. Increasing the codebook to K=1024 reduces collisions and provides a richer set of motion primitives, improving both synchrony and diversity. The best performance is obtained with K=2048, where the token inventory is sufficiently large to capture finer-grained spatial and temporal variations while still remaining learnable under causal decoding. The resulting discrete representation enables xAR and xAR-Fuse to model expressive region-level dynamics more faithfully, yielding improved realism (FGD), stronger rhythm alignment (BC), and greater motion variability (Diversity).

Table 4. Comparison of causal self-attention and causal audio–motion cross-attention in the Hierarchical Autoregressive Transformer (HAR). All models maintain strictly aligned tokenization rates between audio tokens $\{a_t\}$ and motion tokens $\{x_t^r\}$ for fair comparison.

Method	FGD (↓)	BC (→)	Diversity (↑)
Causal self-attention	4.63	0.784	12.53
Causal cross-attention	**4.57**	**0.794**	**13.91**

G. Causal Attention Design in xAR and xAR-Fuse

Table 4 compares two causal attention strategies used in the hierarchical autoregressive transformer. In the "causal self-attention" variant, audio and motion embeddings at each time step are combined by elementwise addition,

$$u_t = x_t^r + a_t,$$

and the transformer attends only over the fused sequence $u_{1:t}$. While this preserves causality and keeps audio and motion aligned at the same token rate, the additive fusion collapses modality structure: the model cannot distinguish which features originate from audio and which from motion, and it cannot form directed cross-modal queries. As a result, the network must implicitly disentangle prosodic cues from the blended representation, which weakens rhythm conditioning and reduces expressive variability, leading to higher FGD (4.63), lower BC (0.784), and reduced Diversity (12.53).

In contrast, the "causal cross-attention" variant maintains separate motion queries and audio keys/values, enabling each motion token x_t^r to directly attend to synchronized audio cues $\{a_{1:t}\}$ through an explicit cross-modal pathway. This structured alignment allows the model to selectively extract energy changes, prosodic beats, and local speech dynamics essential for gesture timing and expressiveness. Consequently, causal cross-attention yields systematically better results (FGD = 4.57, BC = 0.794, Div. = 13.91), showing that explicit causal audio–motion conditioning is more effective than additive fusion for real-time gesture generation.

H. Role of Classifier-Free Guidance in LiveGesture

Table 5. Effect of classifier-free guidance scale γ on streaming gesture generation.

γ	FGD↓	BC→	Div.↑
1.00	4.61	0.781	12.90
1.25	4.57	0.794	13.91
1.35	5.23	0.763	12.80
2.00	6.42	0.756	11.74

Table 5 examines how the classifier-free guidance scale γ affects the token prediction distribution in *LiveGesture*. During inference, the guided logits for each region r are computed as

$$\ell^r_{\text{guided}} = \ell^r_{\text{uncond}} + \gamma \left(\ell^r_{\text{cond}} - \ell^r_{\text{uncond}}\right),$$

which amplifies the influence of audio-conditioned predictions while preserving causal decoding. When γ is too small ($\gamma = 1.00$), the model underutilizes modality conditioning, resulting in weaker synchronization and reduced expressiveness. Moderate guidance ($\gamma = 1.25$) provides the best balance, sharpening the conditional distribution enough to strengthen prosodic alignment (highest BC) and support rich gesture variability (highest Diversity) without oversuppressing natural motion variability, leading to the lowest FGD. Increasing γ further ($\gamma \geq 1.35$) over-amplifies conditional logits, causing overly deterministic predictions that reduce Diversity and destabilize temporal dynamics, which ultimately harms realism and synchrony under streaming constraints. These results demonstrate that *LiveGesture* benefits from moderate classifier-free guidance, which reinforces audio–motion coupling while maintaining the flexibility needed for expressive full-body gestures.

I. Role of Compositional vs. Full-Body SVQ Tokenization

Table 6. Comparison between a single full-body SVQ tokenizer and compositional per-region SVQ tokenizers. Both variants use the same total codebook capacity (2048 entries with 128-d embeddings).

Method	FGD↓	BC→	Div.↑
Full-body SVQ (1 tokenizer)	6.84	0.753	11.23
Per-region SVQ (4 tokenizers)	4.57	0.794	13.91

Table 6 compares two approaches for streamable motion tokenization under identical codebook capacity (2048×128). A single full-body SVQ must quantize the entire SMPL-X pose vector into a single latent stream z_τ, forcing one codebook C to represent heterogeneous motion patterns spanning upper body, lower body, hands, and face. This produces severe quantization interference: high-frequency regions (e.g., hands) and low-frequency regions (e.g., torso) compete for the same discrete codes, leading to token collisions and loss of fine-grained structure. As a result, the downstream autoregressive models receive less informative tokens x_t, which degrades realism (higher FGD), weakens prosodic synchronization (lower BC), and suppresses expressive variability (lower Diversity).

In contrast, the compositional design factorizes the motion stream into region-specific latent sequences z_τ^r with their own codebooks C^{region}. This specialization enables each SVQ to capture the appropriate temporal and spatial scale of its region without cross-region interference. The resulting tokens x_t^r preserve fine-grained dynamics and yield much richer conditioning signals for xAR and xAR-Fuse, improving FGD, BC, and Diversity as shown in Table 6.

J. Ablation on Role of Region Masking in Fusion Training

Table 7. Effect of random region masking (RM) on fusion training. $p_{\text{drop}} \sim \mathcal{U}(0, a)$ denotes the range of probabilities used to fully mask a region's token trajectory.

p_{drop} range	FGD↓	BC→	Div.↑
$\mathcal{U}(0, 0.2)$	4.57	0.794	13.91
$\mathcal{U}(0, 0.3)$	4.85	0.753	13.14
$\mathcal{U}(0, 0.5)$	5.30	0.770	12.82

Table 7 analyzes the effect of region-level masking in Stage 2 of fusion training, where an entire region's token history $\{x_{1:t}^r\}$ is removed with probability $p_{\text{drop}} \sim \mathcal{U}(0, a)$. Moderate masking ($a = 0.2$) yields the best performance because it gently exposes the fusion transformer to incomplete cross-region cues while preserving enough valid context to learn stable spatio–temporal dependencies among regions. As a increases, masking removes critical information from multiple regions, forcing xAR-Fuse to infer missing motion solely from $\{a_{1:t}, w_{1:t}\}$ and the remaining regions. This degrades the quality of the fused representations $\tilde{h}_t^r$ and weakens whole-body coordination, leading to reduced synchrony (lower BC), reduced variability (lower Diversity), and larger distribution drift (higher FGD). These results confirm that region masking is beneficial only when corruption remains mild, allowing the fusion model to learn robustness without collapsing inter-region dynamics.

K. Impact of Noise Injection in Local Region-eXperts

Table 8. Effect of noise injection in local region-eXperts during Stage 1 training. $p_{\text{noise}} \sim \mathcal{U}(0, a)$ determines the probability of adding Gaussian noise to embedded history tokens.

p_{noise} range	FGD↓	BC→	Div.↑
$\mathcal{U}(0, 0.2)$	4.57	0.794	13.91
$\mathcal{U}(0, 0.3)$	4.67	0.773	13.01
$\mathcal{U}(0, 0.5)$	5.30	0.760	12.63

Table 8 evaluates the effect of noise injection into the embedded history tokens of each region-eXpert, where noise is applied with probability $p_{\text{noise}} \sim \mathcal{U}(0, a)$. Light noise ($a = 0.2$) improves robustness by preventing overreliance on perfectly clean token histories, which rarely occur during causal autoregressive inference. This helps each local eXpert learn stable conditional distributions

$$p_\phi^r\big(x_t^r \mid x_{1:t-1}^r, a_{1:t}, w_{1:t}\big),$$

resulting in better synchrony and variance during downstream fusion. However, larger a introduces excessive corruption early in training, degrading the temporal structure in the latent sequences and disrupting the mapping between region tokens and audio cues. This harms both BC and Diversity, and increases FGD due to over-regularization. These results show that mild stochastic perturbation is sufficient to improve streaming robustness, whereas heavy noise erodes the fine-grained temporal patterns essential for expressive gesture generation.

L. Ablation on Attention Depth in xAR and xAR-Fuse

Tables 9 and 10 show the effect of increasing the depth of causal attention in both the local Region-eXpert transformers (xAR) and the global fusion transformer (xAR-Fuse).

In xAR, raising the number of causal audio–motion cross-attention layers and causal temporal self-attention layers from two to three consistently improves FGD, BC, and Diversity. Under strict zero–look-ahead constraints, deeper causal modeling allows each expert to more effectively capture fine-grained dependencies in the joint sequence $(x_{1:t-1}^r, a_{1:t})$, improving rhythm sensitivity and region-specific temporal expressiveness.

Table 9. Effect of causal audio–motion cross-attention depth and causal temporal self-attention depth in the Region-eXpert Autoregressive Transformer (xAR).

Cross-attn layers	Temporal-attn layers	FGD↓	BC→	Div.↑
2	2	4.60	0.791	13.42
3	3	**4.57**	**0.794**	**13.91**

Table 10. Effect of causal audio–motion cross-attention depth and causal spatial–temporal attention depth in the fusion transformer (xAR-Fuse).

Cross-attn layers	Spatio–temporal layers	FGD↓	BC→	Div.↑
2	2	4.65	0.784	13.35
3	3	**4.57**	**0.794**	**13.91**

A similar trend is observed for xAR-Fuse. Increasing the number of causal cross-attention layers and spatio–temporal fusion layers from two to three enhances FGD, BC, and Diversity. The deeper configuration performs more rounds of causal spatial reasoning over region embeddings $\{h_t^r\}$ and more extensive temporal reasoning over global motion history, resulting in stronger whole-body coordination and improved audio–motion synchrony.

Overall, the deeper (3+3)-layer configuration yields the best performance in both xAR and xAR-Fuse. Under strictly causal conditions, both local and global modules benefit from increased attention depth, compensating for the absence of future context and producing coherent, expressive, rhythm-aligned motion in real time.