Motion-example-controlled Co-speech Gesture Generation Leveraging Large Language Models

BOHONG CHEN, State Key Lab of CAD&CG, Zhejiang University, China
YUMENG LI, State Key Lab of CAD&CG, Zhejiang University, China
YOUYI ZHENG, State Key Lab of CAD&CG, Zhejiang University, China
YAO-XIANG DING, State Key Lab of CAD&CG, Zhejiang University, China
KUN ZHOU*, State Key Lab of CAD&CG, Zhejiang University, China

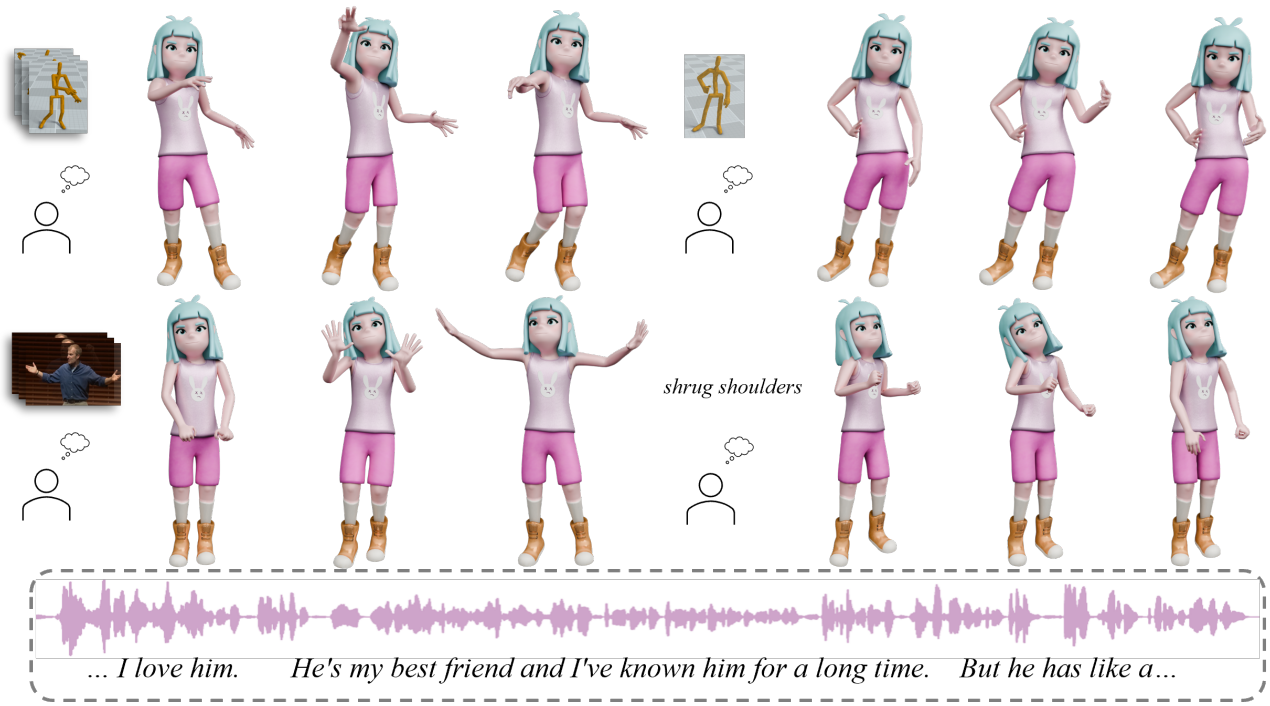

Fig. 1. Given a motion example and a speech audio clip, our method generates vivid co-speech gestures. Motion examples can be a motion clip, a single pose, a human video, or even a text prompt. The four gestures above are generated by the same speech and four different motion examples. The character model is from Adobe Mixamo.

*Corresponding author

Authors' addresses: Bohong Chen, bohongchen@zju.edu.cn, State Key Lab of CAD&CG, Zhejiang University, Hangzhou, China; Yumeng Li, yumeng.li@zju.edu.cn, State Key Lab of CAD&CG, Zhejiang University, Hangzhou, China; Youyi Zheng, youyizheng@zju.edu.cn, State Key Lab of CAD&CG, Zhejiang University, Hangzhou, China; Yao-Xiang Ding, dingyx.gm@gmail.com, State Key Lab of CAD&CG, Zhejiang University, Hangzhou, China; Kun Zhou, kunzhou@acm.org, State Key Lab of CAD&CG, Zhejiang University, Hangzhou, China.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
SIGGRAPH Conference Papers '25, August 10–14, 2025, Vancouver, BC, Canada
© 2025 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 979-8-4007-1540-2/2025/08
https://doi.org/10.1145/3721238.3730611

The automatic generation of controllable co-speech gestures has recently gained growing attention. While existing systems typically achieve gesture control through predefined categorical labels or implicit pseudo-labels derived from motion examples, these approaches often compromise the rich details present in the original motion examples. We present MECo, a framework for motion-example-controlled co-speech gesture generation by leveraging large language models (LLMs). Our method capitalizes on LLMs' comprehension capabilities through fine-tuning to simultaneously interpret speech audio and motion examples, enabling the synthesis of gestures that preserve example-specific characteristics while maintaining speech congruence. Departing from conventional pseudo-labeling paradigms, we position motion examples as explicit query contexts within the prompt structure to guide gesture generation. Experimental results demonstrate state-of-the-art performance across three metrics: Fréchet Gesture Distance (FGD), motion diversity, and example-gesture similarity. Furthermore, our framework enables granular control of individual body parts and accommodates diverse

input modalities including motion clips, static poses, human video sequences, and textual descriptions.

CCS Concepts: • **Computing methodologies** → **Motion processing**; **Computer graphics**.

Additional Key Words and Phrases: co-speech motion generation, motion tokens, text-to-motion, multimodal control

ACM Reference Format:
Bohong Chen, Yumeng Li, Youyi Zheng, Yao-Xiang Ding, and Kun Zhou. 2025. Motion-example-controlled Co-speech Gesture Generation Leveraging Large Language Models. In *Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers (SIGGRAPH Conference Papers '25), August 10–14, 2025, Vancouver, BC, Canada.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3721238.3730611

1 INTRODUCTION

Gestures are the spontaneous and stylized movements of arms, hands and feet that occur while people talk. Just as people have habitual verbal expressions that unconsciously appear in their speech, everyone has their own set of characteristic gestures that they consistently use. These co-speech gestures constitute an essential component of human communication, making the generation of natural and style-appropriate gestures crucial for virtual avatars and digital humans in computer graphics and animation.

Deep learning has become the dominant approach for co-speech gesture generation, yet existing systems often lack fine-grained control mechanisms to effectively translate user intent into precise outputs [Ao et al. 2023; Ghorbani et al. 2023]. Current controllable methods fall into two categories: label-based and example-based. Label-based methods rely on predefined style labels, such as speaker identities [Liu et al. 2022d], emotions [Yang et al. 2023], or hand attributes (e.g., height, radius, and velocity) [Alexanderson et al. 2020; Habibie et al. 2022], which are learned from annotated motion data. While effective, their performance is inherently limited by label availability and granularity, with annotation costs posing practical constraints. Example-based methods [Aberman et al. 2020b; Ao et al. 2023; Ghorbani et al. 2023; Raab et al. 2024] address this limitation by mimicking motion examples as implicit pseudo-labels [Chen et al. 2024a]. Although these methods achieve comprehensive control, they tend to prioritize temporally independent features and often compromise rich details present in the original motion examples.

Recent advances in large language models (LLMs) demonstrate remarkable generalization capabilities in text-related tasks [Chung et al. 2024]. Through fine-tuning with structured input-output pairs, these models outperform traditional methods even in cross-modal tasks. Their universal competence has been validated in audio synthesis [Liao et al. 2024], robotic control [Brohan et al. 2023], and motion generation [Jiang et al. 2024].

In this paper, we present MECo, a framework that leverages LLMs for motion-example-controlled co-speech gesture generation. Our method capitalizes on LLMs' comprehension capabilities through a three-stage fine-tuning mechanism to simultaneously interpret speech audio and motion examples. Departing from conventional pseudo-labeling paradigms, we position motion examples as explicit query contexts within the prompt structure to guide gesture generation, enabling the synthesis of gestures that preserve example-specific characteristics while maintaining speech congruence.

Compared with existing speech-to-gesture methods, our method achieves the state-of-the-art performance evaluated under the most used human-preference-aligned metric for gesture generation and the motion diversity metric. A user study also demonstrates that our method outperforms other example-based methods in terms of the similarity between the generated motions and the input motion examples. Furthermore, our method provides granular control of individual body parts and accommodates diverse input modalities including motion clips, static poses, human video sequences, and textual descriptions (see Figure 1).

Our main contributions include:

- We propose a method to directly use motion examples to control co-speech gesture generation, producing gesture motions that closely resemble the input examples.
- We introduce a three-stage fine-tuning mechanism that effectively integrates audio and motion modalities into the LLM, achieving state-of-the-art performance on the speech-to-gesture task. Interestingly, this approach has a small impact on the LLM's original text comprehension capabilities.
- We build a comprehensive framework for co-speech gesture generation upon our example-based method, which supports multi-modal controls including motion clips, static poses, video sequences and text prompts.

2 RELATED WORK

2.1 Co-Speech Gesture Generation

Gestures enhance the realism of artificial agents by conveying critical social cues like personality and emotional states [Clough and Duff 2020]. Early gesture generation systems used rule-based methods [Cassell et al. 1994, 2001; Kopp et al. 2006; Lee and Marsella 2006; Lhommet et al. 2015], translating speech into predefined gestures via linguistic rules. However, these approaches proved labor-intensive, requiring significant manual effort for rule creation and motion segmentation. Recent advances have shifted to data-driven methods. While traditional deterministic models often produce overly smooth motions [Habibie et al. 2022; Kucherenko et al. 2020; Liu et al. 2022d; Yoon et al. 2020; Zhou et al. 2022] due to their inability to handle many-to-many mappings, modern generative models address this limitation through various architectures: normalizing flows [Alexanderson et al. 2020; Ye et al. 2022], VAEs [Ghorbani et al. 2023; Li et al. 2021; Shi et al. 2024], VQVAEs [Ao et al. 2022; Liu et al. 2022b,c; Lu et al. 2023; Yazdian et al. 2022; Yi et al. 2023], GANs [Wu et al. 2021], and diffusion models [Alexanderson et al. 2023; Ao et al. 2023; Cheng et al. 2024; Yang et al. 2023; Zhang et al. 2024b].

2.2 Controllable Human Motion Generation

Speech-to-gesture mapping constitutes a many-to-many problem, where single speech signals prove inadequate for meeting users' precision demands. This necessitates integrating supplementary control signals with speech for controllable co-speech motion generation. Existing research has explored various control signals to guide gesture synthesis: motion examples [Aberman et al. 2020b; Li et al. 2023; Liu et al. 2024a], text [Goel et al. 2024; Hong et al. 2022; Tevet et al. 2023; Zhang et al. 2022], video [Liu et al. 2022c],

images [Tevet et al. 2022], poses [Ng et al. 2024], trajectories [Karunratanakul et al. 2023; Shafir et al. 2024; Wan et al. 2023; Xie et al. 2023], emotions [Yang et al. 2023], identities [Liu et al. 2022d], hand height and radius [Alexanderson et al. 2020]. However, such signals (e.g., emotion/identity) are typically dataset-specific and resource-intensive to acquire, limiting flexible user control.

GestureDiffuCLIP [Ao et al. 2023] aligns motion sequences with CLIP embeddings [Radford et al. 2021] for multimodal control, yet remains constrained by CLIP's inherent motion representation limitations. To mitigate this, SynTalker [Chen et al. 2024a] construct a dedicated text-motion alignment space, yet still require multimodal weight balancing during inference to reconcile textual and auditory constraints. The core challenge stems from motion's inherent semantic ambiguity - while gestures can be semantically described, precise motion specification remains elusive. ZeroEGGS [Ghorbani et al. 2023] circumvents this by directly conditioning on motion examples, but collapses arbitrary-length sequences into single-style vectors, preserving only coarse semantic attributes (e.g., emotion) while losing kinematic details. Inspired by voice cloning techniques [Liao et al. 2024], our approach eliminates feature extraction networks and instead directly prepends motion examples as generation prefixes, establishing explicit kinematic references for subsequent sequence synthesis.

2.3 Multimodal LLMs

Recent advances in large language models (LLMs) have sparked widespread multimodal extensions, with speech integration achieved by [Zhang et al. 2023a], speech-image-text unification by [Zhan et al. 2024], and motion incorporation through [Jiang et al. 2024]. Unlike existing approaches focused on cross-modal alignment with LLM text representations [Chen et al. 2024c,b; Jiang et al. 2024; Pang et al. 2024; Zhang et al. 2023a], our framework harnesses LLMs' native capacity to decode structured inputs and approximate novel distributions. Crucially, our pipeline elimintes textual supervision beyond basic instruction formatting, revealing an emergent property: the model maintains 99% performance parity on MMLU, GSM8K, and PIQA benchmarks compared to its original version, preserving foundational language understanding capacities.

Our work is also related to recent methods utilizing language models to synthesize motions. T2M-GPT [Zhang et al. 2023b] employs the GPT architecture to perform text-to-motion tasks. MotionGPT [Jiang et al. 2024] finetunes T5 [Raffel et al. 2020] to tackle various text-motion tasks. M^{3}GPT [Luo et al. 2024] further extends it by incorporating text-music-dance related tasks. None of these methods is designed for the co-speech gesture generation task, and it is difficult to conduct direct comparisons with these methods. We discuss them in more details in Section 3 of the supplementary material.

3 METHOD

Given a speech audio and a reference motion sequence, our goal is to synthesize co-speech gestures with stylistic consistency to the reference motions, as depicted in Figure 2. By harnessing LLMs' dual capabilities in instruction following and conditional generation, we develop a multimodal fusion framework that processes both auditory and kinematic inputs to produce contextually appropriate co-speech gesture motions.

Fig. 2. Our model takes motion examples and speech audio as inputs. Both inputs are converted into token sequences by tokenizers and fed into an LLM for autoregressive generation. The generated motion tokens are then processed through a motion decoder to produce the target gesture motion.

To enable LLMs to comprehend speech audio and motion data, these multimodal inputs must first be mapped to tokens within the LLM's embedding space. However, directly training models with randomly initialized tokens presents significant challenges, as their initial distribution diverges from the pre-existing token embedding distribution of the base LLM. This misalignment causes early-stage training instability and hinders effective utilization of the LLM's inherently well-structured parameter space, which risks degrading the model's original capabilities. To address this issue, we propose a novel token initialization method. As shown in Figure 3, during initialization, only the parameters associated with the newly introduced tokens are made trainable. This strategy yields more optimal initial values for the additional tokens, ensuring enhanced compatibility with the established embedding space while preserving the integrity of the pre-trained model.

Building upon this initialization, we employ two training stages to enable example-controlled co-speech gesture generation. The first stage exclusively trains the model's speech-to-gesture mapping capability, establishing core correlations between these two modalities. The subsequent stage introduces motion-example-conditioned training objectives, where the model learns to adapt gestures to both speech content and reference motion examples. This progressive training strategy significantly enhances generation robustness, particularly in data-scarce scenarios where available motion examples are limited and insufficient for gesture generation.

To enhance practical applicability, we design a parameterized sampling mechanism during inference that provides users with granular controllability over motion-example adherence levels. This continuous spectrum ranges from strict compliance with reference motions to partial integration or complete ignorance, enabling context-aware adaptation of gesture generation fidelity. We further generalize the framework to incorporate various input modalities, including poses, video sequences, and textual descriptions, for more flexiable control of gesture generation.

3.1 Motion Representation

A motion $\mathbf{m}_{1:N} \in \mathbb{R}^{N\times(4+6J)}$ is a sequence of poses, where N denotes the motion length. Each pose $m \in \mathbb{R}^{4+6J}$ consists of root angular velocity along Y-axis, root linear velocities on XZ-plane, root height and the rotations of its J joints, where the rotations are parameterized as 6D vectors [Zhou et al. 2019].

For simplicity, the motion sequence is represented as $\mathbf{m}_{1:N} \in \mathbb{R}^{N\times(4+6J)}$. It is firstly encoded into a latent vector sequence $\mathbf{z}_{1:n} \in$

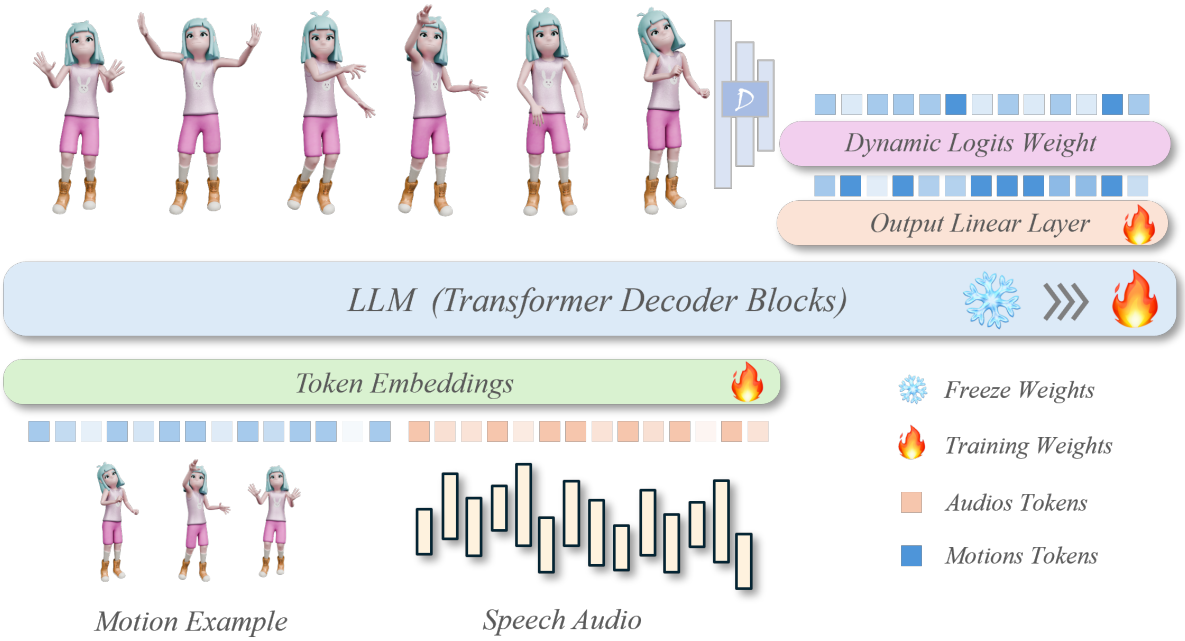

Fig. 3. The structure of our example-guided co-speech generation model. Both motion and audio are tokenized and fed into a large language model (LLM) to generate co-speech motion tokens. Initially, we fine-tune the embedding layer and output linear layer (unembedding space) to adapt the new tokens to the token distribution of the LLM. Subsequently, we perform full parameter fine-tuning to enable the LLM to generate motion tokens.

$\mathbb{R}^{n\times f}$ with a downsampling ratio of n/N and latent feature dimension f, using 1D convolutional encoder $\mathcal{E}$. The $\mathbf{z}_{1:n} \in \mathbb{R}^{n\times f}$ obtained through the encoder then enters the base quantization layer $\mathbf{Q}_0$. Each vector subsequently finds its nearest code entry in the layer's codebook $\mathbf{C}_0 = \{\mathbf{c}_k^0\}_{k=1}^K \subset \mathbb{R}^f$ to get the quantization vector $\hat{\mathbf{z}}_{1:n}^0$. We calculate the quantization residual $\mathbf{r}_{1:n} = \hat{\mathbf{z}}_{1:n}^0 - \mathbf{z}_{1:n}$, which then enters the first residual quantization layer $\mathbf{Q}_1$ and finds its nearest code entry in the layer's codebook $\mathbf{C}_1 = \{\mathbf{c}_k^1\}_{k=1}^K \subset \mathbb{R}^f$ to get the first residual quantization vector $\hat{\mathbf{z}}_{1:n}^1$. Accordingly, $\hat{\mathbf{z}}_{1:n}^2$,$\hat{\mathbf{z}}_{1:n}^3$... can be calculated in this manner. As the final step of motion encoding, we sum all quantization vectors together to get the final code $\hat{\mathbf{z}} = \sum_{q=0}^{Q} \hat{\mathbf{z}}_{1:n}^q$, where $q = 0$ corresponds to the base quantization layer and $q \in \{1, 2, ..., Q\}$ represent the residual quantization layers. Then $\hat{\mathbf{z}}$ is fed into decoder $\mathcal{D}$ for decoding it to motion $\hat{\mathbf{m}}_{1:N}$.

To train the encoder/decoder and all codebooks, we execute the reconstruction task using the following loss function

$$\mathcal{L}_{rec} = \|\hat{\mathbf{m}}_{1:N} - \mathbf{m}_{1:N}\|_1 + \eta \sum_{q=0}^{Q} \|\mathbf{z}_{1:n}^q - \text{sg}[\hat{\mathbf{z}}_{1:n}^q]\|_2^2, \tag{1}$$

where $\text{sg}[\cdot]$ denotes the stop-gradient operation, and η is a weighting factor for the embedding constraint.

To maximize the information captured in the first quantization layer, we randomly drop subsequent residual quantization layers after the base quantization layer during model training [Guo et al. 2023]. This ensures that the base quantization layer learns to encode as much information as possible. We only utilize the base quantization layer without performing VQ completion operations like [Guo et al. 2023; Zhang et al. 2024a]. Although our approach shares the same architecture with vanilla VQVAE during inference, it achieves superior performance in both reconstruction quality and downstream tasks (see detailed validations in Section 2 of the supplementary material).

Given our objective to enable motion-example-guided gesture synthesis, we confront practical constraints in obtaining comprehensive full-body motion examples. This is particularly relevant as many current motion-related works primarily focus on either upper body movements or full-body motions excluding fingers. To bridge this gap, we implement anatomically partitioned tokenization through three functional regions: upper body, lower body, and hands. Each partition is encoded and trained separately, using the same training process. During synthesis, all body partitions are generated simultaneously (see implementation details in Section 1 of the supplementary material). For description simplicity, we use full-body motion modeling as the baseline configuration in the following.

3.2 Finetune LLM

3.2.1 Token embedding initialization. We use the encoder $\mathcal{E}$ to tokenize motion sequence $\mathbf{m}_{1:N}$ into a sequence of discrete units $\mathbf{c}_{1:T_c}$ and a Hidden-unit BERT (HuBERT) [Hsu et al. 2021] to encode speech audio $\mathcal{A}$ into a sequence of discrete units $\mathbf{a}_{1:T_a}$. Since the origin LLM does not have corresponding audio and motion tokens, we need to first extend the vocabulary of the LLM. Token embedding is crucial in LLMs, especially for tokens representing new modalities. To further find better initialization values for these new

tokens that leverage the LLM's existing capabilities while minimizing disruption to its core functions, we initially freeze the main LLM parameters and only train the token embedding layer and the final output projection layer using the LLM's original pretraining task

$$\mathcal{L}(\theta_0) = -\sum_{t=1}^{T_a} \log p(\mathbf{a}_t \mid \mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_{t-1}) - \sum_{t=1}^{T_c} \log p(\mathbf{c}_t \mid \mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_{t-1}), \quad (2)$$

where $\theta_0 = (\theta_{\text{embedding}}, \theta_{\text{output projection}})$.

3.2.2 Speech to gesture. Building upon the previous step, we enable training of all LLM parameters while fine-tuning the model by executing Supervised Fine-Tuning (SFT) [Ouyang et al. 2022] tasks

$$\mathcal{L}(\theta) = -\sum_{t=1}^{T_c} \log p(m_t \mid \mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_{T_a}, \mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_{t-1}). \quad (3)$$

In this training setup, speech audio serves as the user's query in the conversation, while gesture motion acts as the assistant's response. This structure naturally frames our task within the typical conversation format used in LLM training. Note the main purpose of this training step is to establish a mapping between the two modalities in the LLM (audio and motion).

3.2.3 Example-controlled co-speech gesture generation. Finally, we further finetune the LLM augmented with motion examples. The target generated gesture token sequence naturally serves as a motion example input. However, using it directly as a condition which would lead to the generated motions directly copying the motion example while ignoring the audio. Therefore, we process the token sequence through deduplication and shuffling operations to create our motion example. Deduplication removes repeated tokens from the token sequence, while shuffling randomly reorders the tokens in it.

Additionally, since in practice, the motion examples provided by users may be insufficient to generate reasonable co-speech gestures, we also want the model to automatically fill in missing movements during the process. Therefore, we incorporated a random dropout operation on the motion example elements during training. This process can be described as

$$\mathbf{E}_c = \text{Drop\&Shuffle\&Dedup}(\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_{T_c}). \quad (4)$$

In this stage, we train using the same data as in the previous stage while using the following loss function

$$\mathcal{L}(\theta) = -\sum_{t=1}^{T_c} \log p(\mathbf{c}_t \mid \mathbf{E}_c, \mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_{T_a}, \mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_{t-1}) + \lambda \sum_{i=1}^{T_c} p(\mathbf{c}_i \notin \{\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_{T_C}\}). \quad (5)$$

We further utilize the motion example $\mathbf{E}_c$ as a system prompt to assist in the generation process. To ensure the model effectively learns from the given motion examples, we introduce an additional penalty term to the loss function, which discourages the model from generating outputs that deviate from the motion examples. In details, We check the probability of tokens not appearing in motion example and punish their probability. It is important to note that the motion examples used in this penalty term are the original data without dropout operations.

3.3 Inference

Similar to traditional text-based LLMs, during model inference, we use audio as the user query and motion examples as system prompts. To accommodate the requirement of specific initial character states, we set the character's initial pose as the starting point of the model's answer sequence. For arbitrary long audio sequences, we adopt a segmented generation approach, with each segment having the same length as the audio used during training. To ensure temporal consistency across each generated motion segment, when generating the next segment, we use the last three codes of the currently generated motion as the answer sequence, with the corresponding audio aligned accordingly.

To control the frequency of example motions in the generated sequence, we propose a logit-based sampling strategy. Instead of merely adjusting the sampling temperature, which could lead to motion discontinuities, we introduce a manually selected hyperparameter, denoted as β, to the logits of tokens corresponding to motion examples. Additionally, to avoid the repetitive sampling of specific tokens and promote diversity in the motion examples, we apply a decay factor, γ, to the logits of each subsequent token after one is sampled. The adjusted logits are given by

$$logits_i' = (logits_i + \beta) \cdot \gamma^t, \quad (6)$$

where i denotes the index of the i-th token, and t represents the frequency of occurrence of the i-th token in the previously sampled sequence. The computed $logits_i'$ replaces the original logit in the subsequent softmax computation to derive the final sampling probability.

4 EXPERIMENTS

4.1 System Setup

4.1.1 Dataset. We train and test our model on two high-quality mocap co-speech gesture datasets: (1) BEAT2 [Liu et al. 2024b] provides 60 hours of SMPL-X [Pavlakos et al. 2019] formatted full-body motion from 25 speakers. Following the benchmark protocol, we only use the second speaker's data for training and testing; (2) ZeroEGGS [Ghorbani et al. 2023] features two hours of English monologue data from a female performer across 19 different styles, including synchronized motion and audio. We use the same datasets split as in their work.

4.1.2 Settings. Our system generates motions at 30 frames per second. The motion RQVAEs, described in Section 3.1, are trained with a downsampling ratio of n/N = 4, K = 512, d = 512, Q = 6, batch size = 256, η = 0.1, a learning rate of 4e-4, and a step learning rate scheduler. For fine-tuning the LLM, we use Qwen2.5-0.5b-instruction [Yang et al. 2024] as our base model and detach its tied embeddings. In Sections 3.2.1, 3.2.2, and 3.2.3, the batch sizes per GPU are 32, 20, and 12, respectively. The gradient accumulation steps are set to 4, 6, and 10, and the learning rates are 2e-4, 5e-5, and 5e-5 for each

Table 1. Comparison with the state-of-the art methods on BEAT2 [Liu et al. 2024b] test set. Quantitative evaluation on BEAT2. We report FGD $\times 10^{-1}$, BC $\times 10^{-1}$, and diversity. **Bold** face indicates the best result.

Method	FGD ↓	BC ↑	Diversity ↑
S2G[Ginosar et al. 2019]	28.15	4.683	5.971
Trimodal[Yoon et al. 2020]	12.41	5.933	7.724
HA2G[Liu et al. 2022c]	12.32	6.779	8.626
DisCo[Liu et al. 2022a]	9.417	6.439	9.912
CaMN[Liu et al. 2022d]	6.644	6.769	10.86
DiffStyleGesture[Yang et al. 2023]	8.811	7.241	11.49
Habibie *et al.*[Habibie et al. 2021]	9.040	7.716	8.213
TalkShow[Yi et al. 2023]	6.209	6.947	13.47
SynTalker[Chen et al. 2024a]	6.413	**7.971**	12.72
EMAGE [Liu et al. 2024b]	5.512	7.724	13.06
MECo	3.401	7.346	15.30
MECo (w/ examples)	**2.999**	7.472	15.01
MECo (7b llm)	3.456	7.470	**15.64**
MECo (7b llm w/ examples)	3.195	7.554	15.46
MECo (w/o freeze)	8.512	4.551	13.46
MECo (w/o freeze&pretrain)	4.575	6.936	15.45
MECo (w/o s2g)	4.845	6.910	15.09
MECo (w/o s2g ; w/ examples)	4.413	7.138	14.76
MECo (w/o llm)	10.32	5.813	13.47
MECo (w/o Instruct llm)	4.133	6.962	15.13

stage. We use the cosine schedule with warmup as our learning rate scheduler. In Sections 3.3, β is set to 5 by default, and $\gamma = 0.9$.

During RVQVAEs training, we randomly sample 64-frame motion sequences. For LLM fine-tuning, given Hubert's audio encoding rate of 50Hz and motion encoding rate of 7.5Hz, we use 4-second audio and motions, corresponding to 200 audio tokens and 90 motion tokens, with 30 tokens for each of the three body parts. We train all these models using four NVIDIA RTX 4090 GPUs in 22 hours. During inference, using an NVIDIA RTX 4090 GPU and Intel i9-13900KF CPU, our model achieves an inference speed of 147 tokens per second with the default Hugging Face inference pipeline [Wolf et al. 2020]. When deploying the same model using the vLLM [Kwon et al. 2023] inference framework, the generation throughput increases to 270 tokens per second. It means that we can generate 36 seconds of motion in just 1 second.

4.2 Comparisons

4.2.1 Speech-to-gesture benchmark. In the traditional speech-to-gesture task, in order to avoid leaking information from the test dataset, the motion examples in the model's input conditions are set to be empty during sampling. To ensure the reproducibility of results, only greedy sampling is used when calculating quantitative metrics. We have achieved SOTA performance as shown in Table 1, particularly on FGD [Yoon et al. 2020], which is currently the most used human-preference-aligned metric for gesture generation evaluation [Kucherenko et al. 2024]. We further test the inference process with the inclusion of motion examples (w/ examples), and find that our results are further improved. These results not only demonstrate our model's superiority in the pure speech-to-gesture task but also validate the effectiveness of motion examples in controlling co-speech gesture generation.

4.2.2 Speech-to-gesture with motion examples. 3 For the co-speech gesture generation task, we select two works that similarly support motion examples as input for comparison: ZeroEGGS [Ghorbani et al. 2023] and SynTalker [Chen et al. 2024a]. While ZeroEGGS is originally trained on the ZeroEGGS dataset, we follow its data split to retrain our model on it. For SynTalker, we directly use their published code and pre-trained checkpoints.

To validate whether the generated motions follow the examples, we still use FGD as the metric, as it is the most effective in determining whether two sequences are similar. Since the ZeroEGGS dataset does not provide a feature extractor, following [Liu et al. 2022d; Yoon et al. 2020], we use an autoencoder as our feature extractor to compute the Fréchet distance, which we refer to as FGD1. Since this autoencoder is trained by us, for fairness, like [Ng et al. 2023], we directly calculate the Fréchet distance on the representations of the motions themselves, which we refer to as FGD2. For the BEAT2 dataset, we compute FGD1 using its default feature extractor [Aberman et al. 2020a]. To compare the performance of different methods on both training and test sets, we sampled 20 motion sequences with durations ranging from 3 to 6 seconds as motion examples. For input audio, we consistently used a neutral-style audio clip from the test set. For each generated result, we computed its FGD1 and FGD2 with the corresponding motion example. The final metrics are calculated by averaging all results. The experimental results in Table 2 show that our method achieves superior performance compared to existing approaches on both training and test sets. Figure 5 demonstrates that we can use different motion examples for different body parts to achieve granular control. Additionally, the visualization comparison results in Figure 6 and 7 clearly show that our generated results are more closely aligned with those of the motion example.

4.3 User Study

Following [Alexanderson et al. 2023; Ao et al. 2023; Zhang et al. 2024a], we conduct a similar user study to validate the effectiveness of our method. For each method to be evaluated, eighty 10-second audio segments were employed to generate animations. 29 participants were recruited for this study, and each questionnaire included 24 paired comparisons. They selected their preferred clip and rated their preference intensity on a 0-2 scale (0 indicating no preference). The unselected clip automatically received the opposite/negative score. We evaluate the generated gestures using three subjective metrics: *Human Likeness* for realism and human-like quality, *Appropriateness* for alignment with speech rhythm and semantics, and *Example Consistency* for similarity to the reference motion example. As shown in Table 4, our method achieves the best performance under these metrics, especially in the *Example Consistency*.

4.4 Ablation Study

4.4.1 Initialize token embedding. To validate the importance of this step in our experiment, we conducted two ablation studies. We

Table 2. Comparison of the similarity between the generated results and the motion example. The values in this table represent the mean and standard deviation, where the standard deviation is shown after '±'.

Dataset	System	$FGD1_{train}$ ↓	$FGD1_{test}$ ↓	$FGD2_{train}$ ↓	$FGD2_{test}$ ↓
ZEGGS	ZeroEGGS	3.39 ± 0.179	4.54 ± 0.282	23.14 ± 1.973	26.04 ± 1.939
	MECo	1.83 ± 0.118	1.98 ± 0.593	10.22 ± 2.221	10.20 ± 2.881
	MECo (w/o freeze&pretrain)	2.47 ± 0.239	2.82 ± 0.647	16.63 ± 2.752	18.94 ± 3.326
	MECo (w/o s2g)	2.29 ± 0.137	2.65 ± 0.683	13.47 ± 2.390	14.98 ± 3.217
	MECo (w/o instruct llm)	1.96 ± 0.153	2.31 ± 0.591	11.86 ± 2.528	12.14 ± 2.946
BEAT2	SynTalker	4.19 ± 0.477	8.21 ± 0.771	24.74 ± 1.455	37.72 ± 3.136
	MECo	2.65 ± 0.243	4.12 ± 0.513	17.23 ± 1.338	21.73 ± 2.647
	MECo (w/o freeze&pretrain)	2.95 ± 0.206	4.55 ± 0.483	20.41 ± 1.452	26.17 ± 2.441
	MECo (w/o s2g)	2.83 ± 0.272	4.68 ± 0.656	19.11 ± 1.539	28.53 ± 2.968
	MECo (w/o instruct llm)	2.81 ± 0.254	4.29 ± 0.577	18.62 ± 1.430	22.95 ± 2.703

first directly omit our embedding initialization step described in Section 3.2.1 and use the default token embedding initialization in PyTorch for the newly added vocabulary.

As shown in Tables 1 and Table 3, this method led to a degradation in co-speech gesture generation quality metrics. Moreover, it impairs some of the LLM's inherent capabilities. We also provide more detailed comparisons of models in terms of degradation rates in Section 5 of the supplementary material.

We further conduct an ablation study on the partial freezing strategy described in Section 3.2.1, where we make the main body of the LLM frozen. In this experiment, we made all model parameters trainable during the training process, instead of our original approach where only token embeddings and the output linear layer are trainable. As shown in Tables 1 and 3, this modification severely degrades the model's performance, resulting in the generation of static motions lacking any meaningful variation.

4.4.2 Speech-to-gesture Training. In this experiment, we investigate eliminating the speech-to-gesture training phase prior to example-controlled co-speech gesture training (i.e., using only Sections 3.2.1 and 3.2.3). This decision is motivated by the valid concern that Section 3.2.3 already involves training the speech-to-gesture task, raising the question of whether a dedicated training phase is necessary. As shown in Table 1, after removing this training task, the performance metrics for co-speech gesture generation show a noticeable decline for both MECo (w/o s2g) and (w/o s2g; w/ examples). Here, "w/o s2g" refers to training without the speech-to-gesture task, as described in Section 3.2.2, and "w/ examples" indicates the use of motion examples during the metric evaluation. Including the speech-to-gesture task during training helps improve the quality of gesture generation, especially in cases where no motion examples are given or the provided motion examples are short in length.

4.4.3 LLM backbone. In this experiment, to validate the effectiveness of instruction LLM for our work, we introduced two additional LLM variants for comparison: a base LLM that only underwent pre-training, and an untrained LLM with randomly initialized parameters. It is worth noting that all LLMs share the same network architectures, differing only in their parameter values. As demonstrated in Table 1 and Table 2, the instruction-tuned LLM consistently outperforms the other two variants in gesture generation capabilities in both co-speech gesture generation ability and motion examples followed ability.

Table 3. Comparison with the original LLM in performance metrics.

Model	MMLU↑	GSM8K↑	PIQA↑
Qwen2.5-0.5b-instrcut	46.50	20.47	70.13
MECo	46.27	20.47	69.64
MECo (w/o freeze)	24.63	0.15	52.50
MECo (w/o freeze&pretrain)	39.62	15.24	65.83
Qwen2.5-7b-instrcut	74.20	82.18	80.30
MECo (7b llm)	74.13	81.96	79.54

We also conducted parameter scaling tests using Qwen2.5-Instruct (7B parameters). As shown in Table 1, no scaling benefits were observed - performance actually slightly decreased. This likely indicates that current co-speech gesture data scarcity renders even 0.5B models sufficiently capable, leaving no advantage for larger architectures. For instance, the current BEAT2 benchmark uses the second character's data (containing only about 2 hours of co-speech gestures) as its test set. After tokenization, this yields only 54k motion tokens and 360k audio tokens. Even with data augmentation through mirroring and speed variation, this scale remains minimal compared to text models like the 0.5B-parameter model in Qwen2.5 [Yang et al. 2024], which was trained from scratch on 18T tokens.

4.5 Multimodal Controls

Our example-based method can support diverse input modalities. For video input, we extract SMPL-X parameters via monocular motion capture [Yi et al. 2023]. For image input, we reconstruct static poses using SMPLify-X [Pavlakos et al. 2019]. For text input, we employ two approaches: 1) Annotate our co-speech gesture dataset with text labels to establish gesture-text mappings; 2) Train a motion-text retrieval system (TMR [Petrovich et al. 2023]). Given text queries, ChatGPT first checks for matching annotations in our

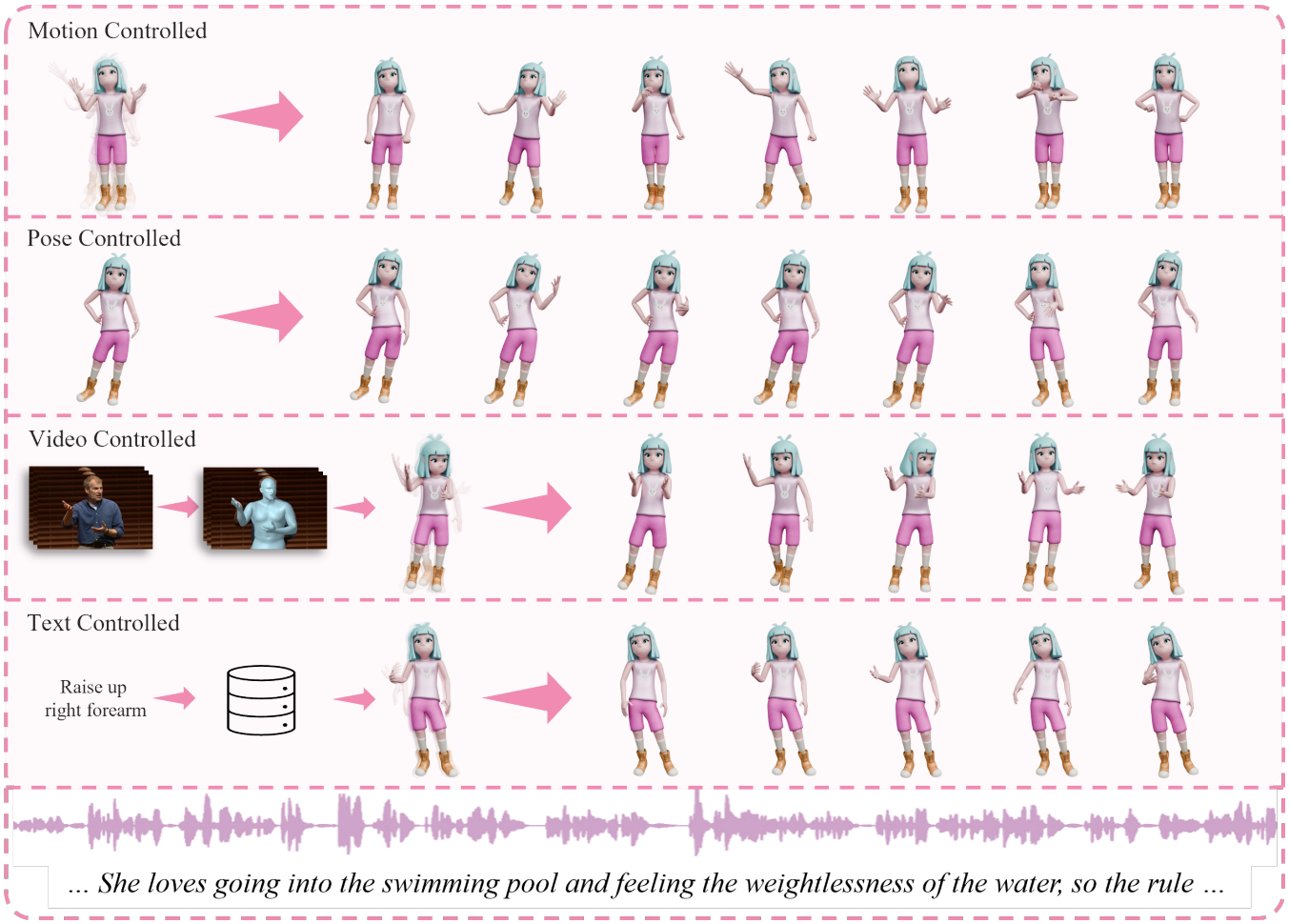

Fig. 4. We demonstrate the versatility of our method across various control modalities, including direct motion control, pose control, video control, and text control. These diverse modalities are unified as motion examples, which serve as prompts for our system. By leveraging these prompts, our method performs co-speech motion generation that is not only aware of the speech audio but also aligned with the provided motion examples to reflect user intent.

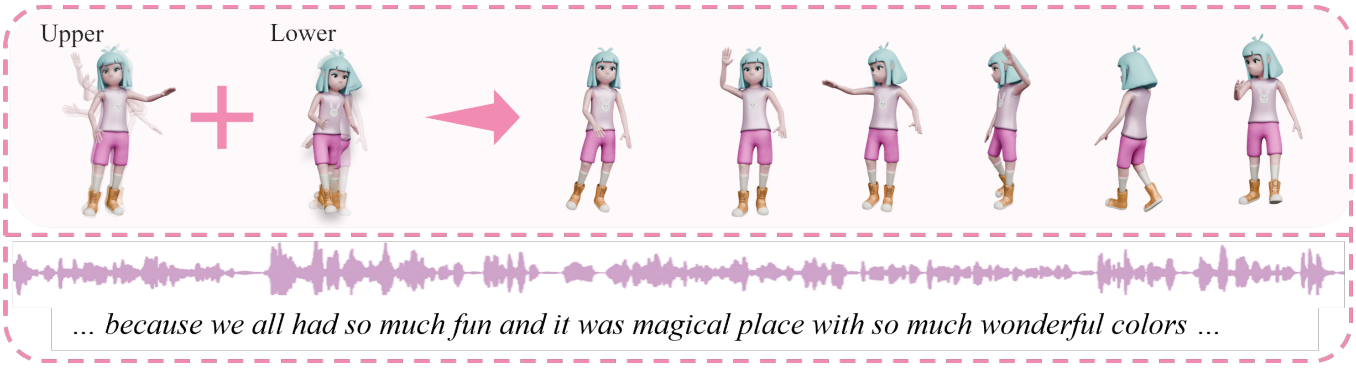

Fig. 5. We can control specific body parts by tokenizing examples and combining their corresponding tokens. For instance, we tokenize two examples, use the upper body token from the first and the lower body token from the second as a prompt. The generated motion effectively reflects these references.

gesture dataset. If available, the corresponding motion is used; otherwise, TMR retrieves the most similar motion from the broader corpus. Figure 4 demonstrates our framework's multimodal operation and outputs.

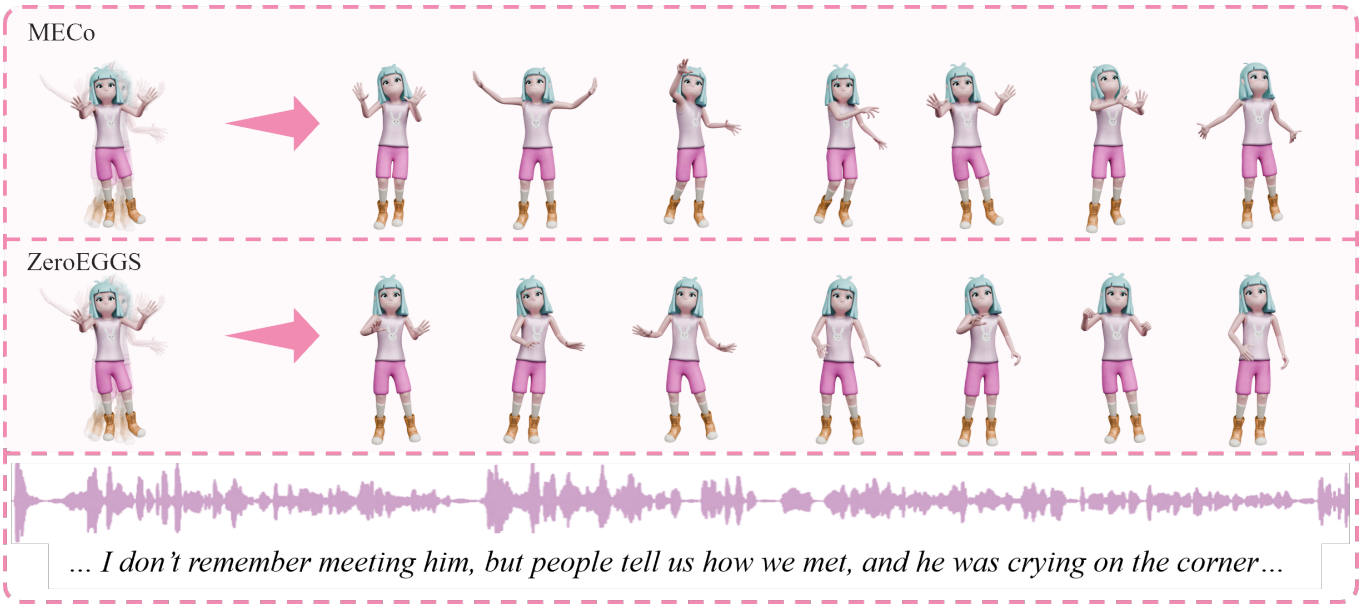

Fig. 6. A qualitative comparison between our method and ZeroEGGS. Both methods use the same input, with the motion example displayed on the left side of the arrows in the figure, and the input audio presented at the bottom of the figure.

Fig. 7. A qualitative comparison between our method and SynTalker. Both methods use the same input, with the motion example displayed on the left side of the arrows in the figure, and the input audio presented at the bottom of the figure.

5 DISCUSSIONS AND FUTURE WORK

Our experiments reveal degraded performance when using video prompts. Analysis shows that for many in-the-wild videos, monocular motion capture derived SMPL-X parameters exhibit significant reconstruction errors after VQ-VAE processing. This highlights our motion VQ-VAE's limited generalization to out-of-distribution data. Addressing this limitation by improving the VQ-VAE's generalization capability constitutes our next research priority.

Our generated motions also exhibit physically implausible artifacts (e.g., foot sliding), a common challenge in kinematic motion generation, which could be addressed via inverse kinematics (IK) or physics-based simulations [Luo et al. 2023; Yao et al. 2024].

Table 4. User study of different systems on BEAT2 and ZeroEGGS datasets. The results are reported as average scores with 95% confidence intervals.

Dataset	System	HumanLikeness ↑	Appropriateness ↑	ExampleConsistency ↑
BEAT2	EMAGE	-0.59 ± 0.17	-0.48 ± 0.19	–
	SynTalker(U)	0.18 ± 0.19	0.12 ± 0.19	–
	SynTalker	-0.28 ± 0.20	-0.42 ± 0.22	-0.64 ± 0.21
	MECo(U)	0.34 ± 0.18	0.30 ± 0.20	–
	MECo	0.28 ± 0.20	0.42 ± 0.22	0.64 ± 0.21
ZeroEGGS	ZeroEGGS	-0.53 ± 0.24	-1.10 ± 0.16	-1.17 ± 0.17
	MECo	0.53 ± 0.24	1.10 ± 0.16	1.17 ± 0.17

ACKNOWLEDGEMENTS

This work is partially supported by NSF China (No. 62421003, 62206245) and the XPLORER PRIZE.

REFERENCES

Kfir Aberman, Peizhuo Li, Dani Lischinski, Olga Sorkine-Hornung, Daniel Cohen-Or, and Baoquan Chen. 2020a. Skeleton-aware networks for deep motion retargeting. ACM Trans. Graph. 39, 4, Article 62 (Aug. 2020), 14 pages.

Kfir Aberman, Yijia Weng, Dani Lischinski, Daniel Cohen-Or, and Baoquan Chen. 2020b. Unpaired Motion Style Transfer from Video to Animation. ACM Transactions on Graphics (TOG) 39, 4 (2020), 64.

Simon Alexanderson, Gustav Eje Henter, Taras Kucherenko, and Jonas Beskow. 2020. Style-controllable speech-driven gesture synthesis using normalising flows. Computer Graphics Forum 39, 2 (2020), 487–496.

Simon Alexanderson, Rajmund Nagy, Jonas Beskow, and Gustav Eje Henter. 2023. Listen, Denoise, Action! Audio-Driven Motion Synthesis with Diffusion Models. ACM Trans. Graph. 42, 4, Article 44 (July 2023), 20 pages.

Tenglong Ao, Qingzhe Gao, Yuke Lou, Baoquan Chen, and Libin Liu. 2022. Rhythmic gesticulator: Rhythm-aware co-speech gesture synthesis with hierarchical neural embeddings. ACM Transactions on Graphics (TOG) 41, 6 (2022), 1–19.

Tenglong Ao, Zeyi Zhang, and Libin Liu. 2023. GestureDiffuCLIP: Gesture Diffusion Model with CLIP Latents. ACM Trans. Graph. (2023), 18 pages.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, Pete Florence, Chuyuan Fu, Montse Gonzalez Arenas, Keerthana Gopalakrishnan, Kehang Han, Karol Hausman, Alex Herzog, Jasmine Hsu, Brian Ichter, Alex Irpan, Nikhil Joshi, Ryan Julian, Dmitry Kalashnikov, Yuheng Kuang, Isabel Leal, Lisa Lee, Tsang-Wei Edward Lee, Sergey Levine, Yao Lu, Henryk Michalewski, Igor Mordatch, Karl Pertsch, Kanishka Rao, Krista Reymann, Michael Ryoo, Grecia Salazar, Pannag Sanketi, Pierre Sermanet, Jaspiar Singh, Anikait Singh, Radu Soricut, Huong Tran, Vincent Vanhoucke, Quan Vuong, Ayzaan Wahid, Stefan Welker, Paul Wohlhart, Jialin Wu, Fei Xia, Ted Xiao, Peng Xu, Sichun Xu, Tianhe Yu, and Brianna Zitkovich. 2023. RT-2: Vision-Language-Action Models Transfer Web Knowledge to Robotic Control. In arXiv preprint arXiv:2307.15818.

Justine Cassell, Catherine Pelachaud, Norman Badler, Mark Steedman, Brett Achorn, Tripp Becket, Brett Douville, Scott Prevost, and Matthew Stone. 1994. Animated Conversation: Rule-Based Generation of Facial Expression, Gesture & Spoken Intonation for Multiple Conversational Agents. In Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '94). Association for Computing Machinery, New York, NY, USA, 413–420.

Justine Cassell, Hannes Högni Vilhjálmsson, and Timothy Bickmore. 2001. BEAT: The Behavior Expression Animation Toolkit. In Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '01). Association for Computing Machinery, New York, NY, USA, 477–486.

Bohong Chen, Yumeng Li, Yao-Xiang Ding, Tianjia Shao, and Kun Zhou. 2024a. Enabling Synergistic Full-Body Control in Prompt-Based Co-Speech Motion Generation. In Proceedings of the 32nd ACM International Conference on Multimedia. ACM, New York, NY, USA, 10.

Changan Chen, Juze Zhang, Shrinidhi Kowshika Lakshmikanth, Yusu Fang, Ruizhi Shao, Gordon Wetzstein, Li Fei-Fei, and Ehsan Adeli. 2024c. The Language of Motion: Unifying Verbal and Non-verbal Language of 3D Human Motion. In arXiv.

Ling-Hao Chen, Shunlin Lu, Ailing Zeng, Hao Zhang, Benyou Wang, Ruimao Zhang, and Lei Zhang. 2024b. MotionLLM: Understanding Human Behaviors from Human Motions and Videos. arXiv preprint arXiv:2405.20340 (2024).

Qingrong Cheng, Xu Li, and Xinghui Fu. 2024. SIGGesture: Generalized Co-Speech Gesture Synthesis via Semantic Injection with Large-Scale Pre-Training Diffusion Models. In SIGGRAPH Asia 2024 Conference Papers (SA '24). Association for Computing Machinery, New York, NY, USA, Article 133, 11 pages.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2024. Scaling instruction-finetuned language models. Journal of Machine Learning Research 25, 70 (2024), 1–53.

Sharice Clough and Melissa C. Duff. 2020. The Role of Gesture in Communication and Cognition: Implications for Understanding and Treating Neurogenic Communication Disorders. Frontiers in Human Neuroscience 14 (2020).

Saeed Ghorbani, Ylva Ferstl, Daniel Holden, Nikolaus F. Troje, and Marc-André Carbonneau. 2023. ZeroEGGS: Zero-shot Example-based Gesture Generation from Speech. Computer Graphics Forum 42, 1 (2023), 206–216. arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.14734

Shiry Ginosar, Amir Bar, Gefen Kohavi, Caroline Chan, Andrew Owens, and Jitendra Malik. 2019. Learning individual styles of conversational gesture. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 3497–3506.

Purvi Goel, Kuan-Chieh Wang, C. Karen Liu, and Kayvon Fatahalian. 2024. Iterative Motion Editing with Natural Language. In ACM SIGGRAPH 2024 Conference Papers (Denver, CO, USA) (SIGGRAPH '24). Association for Computing Machinery, New York, NY, USA, Article 71, 9 pages.

Chuan Guo, Yuxuan Mu, Muhammad Gohar Javed, Sen Wang, and Li Cheng. 2023. MoMask: Generative Masked Modeling of 3D Human Motions. (2023). arXiv:2312.00063 [cs.CV]

Ikhsanul Habibie, Mohamed Elgharib, Kripasindhu Sarkar, Ahsan Abdullah, Simbarashe Nyatsanga, Michael Neff, and Christian Theobalt. 2022. A Motion Matching-based Framework for Controllable Gesture Synthesis from Speech. In ACM SIGGRAPH 2022 Conference Proceedings (Vancouver, BC, Canada) (SIGGRAPH '22). Association for Computing Machinery, New York, NY, USA, Article 46, 9 pages.

Ikhsanul Habibie, Weipeng Xu, Dushyant Mehta, Lingjie Liu, Hans-Peter Seidel, Gerard Pons-Moll, Mohamed Elgharib, and Christian Theobalt. 2021. Learning Speech-driven 3D Conversational Gestures from Video. arXiv preprint arXiv:2102.06837 (2021).

Fangzhou Hong, Mingyuan Zhang, Liang Pan, Zhongang Cai, Lei Yang, and Ziwei Liu. 2022. AvatarCLIP: Zero-Shot Text-Driven Generation and Animation of 3D Avatars. ACM Transactions on Graphics (TOG) 41, 4 (2022), 1–19.

Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. 2021. Hubert: Self-supervised speech representation learning by masked prediction of hidden units. IEEE/ACM transactions on audio, speech, and language processing 29 (2021), 3451–3460.

Biao Jiang, Xin Chen, Wen Liu, Jingyi Yu, Gang Yu, and Tao Chen. 2024. Motiongpt: Human motion as a foreign language. Advances in Neural Information Processing Systems 36 (2024).

Korrawe Karunratanakul, Konpat Preechakul, Supasorn Suwajanakorn, and Siyu Tang. 2023. Guided Motion Diffusion for Controllable Human Motion Synthesis. In Proceedings of the IEEE/CVF International Conference on Computer Vision. 2151–2162.

Stefan Kopp, Brigitte Krenn, Stacy Marsella, Andrew N. Marshall, Catherine Pelachaud, Hannes Pirker, Kristinn R. Thórisson, and Hannes Vilhjálmsson. 2006. Towards a Common Framework for Multimodal Generation: The Behavior Markup Language. In Proceedings of the 6th International Conference on Intelligent Virtual Agents (Marina Del Rey, CA) (IVA'06). Springer-Verlag, Berlin, Heidelberg, 205–217.

Taras Kucherenko, Patrik Jonell, Sanne van Waveren, Gustav Eje Henter, Simon Alexandersson, Iolanda Leite, and Hedvig Kjellström. 2020. Gesticulator: A framework for semantically-aware speech-driven gesture generation. In Proceedings of the 2020 International Conference on Multimodal Interaction (Virtual Event, Netherlands) (ICMI '20). Association for Computing Machinery, New York, NY, USA, 242–250.

Taras Kucherenko, Pieter Wolfert, Youngwoo Yoon, Carla Viegas, Teodor Nikolov, Mihail Tsakov, and Gustav Eje Henter. 2024. Evaluating Gesture Generation in a

Large-scale Open Challenge: The GENEA Challenge 2022. 43, 3, Article 32 (jun 2024).

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient Memory Management for Large Language Model Serving with PagedAttention. In Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles.

Jina Lee and Stacy Marsella. 2006. Nonverbal Behavior Generator for Embodied Conversational Agents (IVA '06). Springer, 243–255.

Margot Lhommet, Yuyu Xu, and Stacy Marsella. 2015. Cerebella: Automatic Generation of Nonverbal Behavior for Virtual Humans (AAAI '15, 1).

Jing Li, Di Kang, Wenjie Pei, Xuefei Zhe, Ying Zhang, Zhenyu He, and Linchao Bao. 2021. Audio2Gestures: Generating Diverse Gestures from Speech Audio with Conditional Variational Autoencoders. In Proceedings of the IEEE/CVF International Conference on Computer Vision. 11293–11302.

Weiyu Li, Xuelin Chen, Peizhuo Li, Olga Sorkine-Hornung, and Baoquan Chen. 2023. Example-Based Motion Synthesis via Generative Motion Matching. ACM Transactions on Graphics (TOG) 42, 4, Article 94 (2023).

Shijia Liao, Yuxuan Wang, Tianyu Li, Yifan Cheng, Ruoyi Zhang, Rongzhi Zhou, and Yijin Xing. 2024. Fish-Speech: Leveraging Large Language Models for Advanced Multilingual Text-to-Speech Synthesis. arXiv:2411.01156 [cs.SD]

Haiyang Liu, Naoya Iwamoto, Zihao Zhu, Zhengqing Li, You Zhou, Elif Bozkurt, and Bo Zheng. 2022a. DisCo: Disentangled Implicit Content and Rhythm Learning for Diverse Co-Speech Gestures Synthesis. In Proceedings of the 30th ACM International Conference on Multimedia (Lisboa, Portugal) (MM '22). Association for Computing Machinery, New York, NY, USA, 3764–3773.

Haiyang Liu, Xingchao Yang, Tomoya Akiyama, Yuantian Huang, Qiaoge Li, Shigeru Kuriyama, and Takafumi Taketomi. 2024a. TANGO: Co-Speech Gesture Video Reenactment with Hierarchical Audio Motion Embedding and Diffusion Interpolation. arXiv:2410.04221 [cs.CV]

Haiyang Liu, Zihao Zhu, Giorgio Becherini, Yichen Peng, Mingyang Su, You Zhou, Naoya Iwamoto, Bo Zheng, and Michael J. Black. 2024b. EMAGE: Towards Unified Holistic Co-Speech Gesture Generation via Masked Audio Gesture Modeling. arXiv:2401.00374 [cs.CV]

Haiyang Liu, Zihao Zhu, Naoya Iwamoto, Yichen Peng, Zhengqing Li, You Zhou, Elif Bozkurt, and Bo Zheng. 2022d. BEAT: A Large-Scale Semantic and Emotional Multi-Modal Dataset for Conversational Gestures Synthesis. arXiv preprint arXiv:2203.05297 (2022).

Xian Liu, Qianyi Wu, Hang Zhou, Yuanqi Du, Wayne Wu, Dahua Lin, and Ziwei Liu. 2022b. Audio-Driven Co-Speech Gesture Video Generation. In Advances in Neural Information Processing Systems, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (Eds.), Vol. 35. Curran Associates, Inc., 21386–21399.

Xian Liu, Qianyi Wu, Hang Zhou, Yinghao Xu, Rui Qian, Xinyi Lin, Xiaowei Zhou, Wayne Wu, Bo Dai, and Bolei Zhou. 2022c. Learning Hierarchical Cross-Modal Association for Co-Speech Gesture Generation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 10462–10472.

Shuhong Lu, Youngwoo Yoon, and Andrew W. Feng. 2023. Co-Speech Gesture Synthesis using Discrete Gesture Token Learning. 2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (2023), 9808–9815.

Mingshuang Luo, Ruibing Hou, Zhuo Li, Hong Chang, Zimo Liu, Yaowei Wang, and Shiguang Shan. 2024. M^{3}GPT: An Advanced Multimodal, Multitask Framework for Motion Comprehension and Generation. Advances in Neural Information Processing Systems (2024).

Zhengyi Luo, Jinkun Cao, Alexander W. Winkler, Kris Kitani, and Weipeng Xu. 2023. Perpetual Humanoid Control for Real-time Simulated Avatars. In International Conference on Computer Vision (ICCV).

Evonne Ng, Javier Romero, Timur Bagautdinov, Shaojie Bai, Trevor Darrell, Angjoo Kanazawa, and Alexander Richard. 2024. From Audio to Photoreal Embodiment: Synthesizing Humans in Conversations. In ArXiv.

Evonne Ng, Sanjay Subramanian, Dan Klein, Angjoo Kanazawa, Trevor Darrell, and Shiry Ginosar. 2023. Can Language Models Learn to Listen?. In Proceedings of the International Conference on Computer Vision (ICCV).

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. Advances in neural information processing systems 35 (2022), 27730–27744.

Haozhou Pang, Tianwei Ding, Lanshan He, Ming Tao, Lu Zhang, and Qi Gan. 2024. LLM Gesticulator: Leveraging Large Language Models for Scalable and Controllable Co-Speech Gesture Synthesis. arXiv:2410.10851 [cs.GR]

Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed A. A. Osman, Dimitrios Tzionas, and Michael J. Black. 2019. Expressive Body Capture: 3D Hands, Face, and Body from a Single Image. In Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR).

Mathis Petrovich, Michael J. Black, and Gül Varol. 2023. TMR: Text-to-Motion Retrieval Using Contrastive 3D Human Motion Synthesis. In International Conference on Computer Vision (ICCV).

Sigal Raab, Inbar Gat, Nathan Sala, Guy Tevet, Rotem Shalev-Arkushin, Ohad Fried, Amit H Bermano, and Daniel Cohen-Or. 2024. Monkey See, Monkey Do: Harnessing Self-attention in Motion Diffusion for Zero-shot Motion Transfer. In SIGGRAPH Asia 2024 Conference Papers. 1–13.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning Transferable Visual Models from Natural Language Supervision. In International Conference on Machine Learning (ICML).

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. Journal of Machine Learning Research 21, 140 (2020), 1–67.

Yoni Shafir, Guy Tevet, Roy Kapon, and Amit Haim Bermano. 2024. Human Motion Diffusion as a Generative Prior. In The Twelfth International Conference on Learning Representations.

Min Shi, Wenke Feng, Lin Gao, and Dengming Gao. 2024. Generating diverse clothed 3D human animations via a generative model. Computational Visual Media 10, 2 (2024), 261–277.

Guy Tevet, Brian Gordon, Amir Hertz, Amit H Bermano, and Daniel Cohen-Or. 2022. Motionclip: Exposing human motion generation to clip space. In Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXII. Springer, 358–374.

Guy Tevet, Sigal Raab, Brian Gordon, Yoni Shafir, Daniel Cohen-or, and Amit Haim Bermano. 2023. Human Motion Diffusion Model. In The Eleventh International Conference on Learning Representations.

Weilin Wan, Zhiyang Dou, Taku Komura, Wenping Wang, Dinesh Jayaraman, and Lingjie Liu. 2023. TLControl: Trajectory and Language Control for Human Motion Synthesis. arXiv preprint arXiv:2311.17135 (2023).

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-Art Natural Language Processing. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. Association for Computational Linguistics, Online, 38–45.

Bowen Wu, Chaoran Liu, Carlos Toshinori Ishi, and Hiroshi Ishiguro. 2021. Modeling the conditional distribution of co-speech upper body gesture jointly using conditional-GAN and unrolled-GAN. Electronics 10, 3 (2021), 228.

Yiming Xie, Varun Jampani, Lei Zhong, Deqing Sun, and Huaizu Jiang. 2023. OmniControl: Control Any Joint at Any Time for Human Motion Generation. arXiv:2310.08580

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. 2024. Qwen2.5 Technical Report. arXiv preprint arXiv:2412.15115 (2024).

Sicheng Yang, Zhiyong Wu, Minglei Li, Zhensong Zhang, Lei Hao, Weihong Bao, Ming Cheng, and Long Xiao. 2023. DiffuseStyleGesture: Stylized Audio-Driven Co-Speech Gesture Generation with Diffusion Models. In Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI-23. International Joint Conferences on Artificial Intelligence Organization, 5860–5868.

Heyuan Yao, Zhenhua Song, Yuyang Zhou, Tenglong Ao, Baoquan Chen, and Libin Liu. 2024. MoConVQ: Unified Physics-Based Motion Control via Scalable Discrete Representations. ACM Trans. Graph. 43, 4, Article 144 (July 2024), 21 pages.

Payam Jome Yazdian, Mo Chen, and Angelica Lim. 2022. Gesture2Vec: Clustering gestures using representation learning methods for co-speech gesture generation. In 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 3100–3107.

Sheng Ye, Yu-Hui Wen, Yanan Sun, Ying He, Ziyang Zhang, Yaoyuan Wang, Weihua He, and Yong-Jin Liu. 2022. Audio-driven stylized gesture generation with flow-based model. In European Conference on Computer Vision. Springer, 712–728.

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

Youngwoo Yoon, Bok Cha, Joo-Haeng Lee, Minsu Jang, Jaeyeon Lee, Jaehong Kim, and Geehyuk Lee. 2020. Speech gesture generation from the trimodal context of text, audio, and speaker identity. ACM Transactions on Graphics (TOG) 39, 6 (2020), 1–16.

Jun Zhan, Junqi Dai, Jiasheng Ye, Yunhua Zhou, Dong Zhang, Zhigeng Liu, Xin Zhang, Ruibin Yuan, Ge Zhang, Linyang Li, et al. 2024. AnyGPT: Unified Multimodal LLM with Discrete Sequence Modeling. arXiv preprint arXiv:2402.12226 (2024).

Dong Zhang, Shimin Li, Xin Zhang, Jun Zhan, Pengyu Wang, Yaqian Zhou, and Xipeng Qiu. 2023a. SpeechGPT: Empowering Large Language Models with Intrinsic Cross-Modal Conversational Abilities. arXiv:2305.11000 [cs.CL]

Jianrong Zhang, Yangsong Zhang, Xiaodong Cun, Shaoli Huang, Yong Zhang, Hongwei Zhao, Hongtao Lu, and Xi Shen. 2023b. T2M-GPT: Generating Human Motion from Textual Descriptions with Discrete Representations. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).

Mingyuan Zhang, Zhongang Cai, Liang Pan, Fangzhou Hong, Xinying Guo, Lei Yang, and Ziwei Liu. 2022. MotionDiffuse: Text-Driven Human Motion Generation with Diffusion Model. arXiv preprint arXiv:2208.15001 (2022).

Mingyuan Zhang, Daisheng Jin, Chenyang Gu, Fangzhou Hong, Zhongang Cai, Jingfang Huang, Chongzhi Zhang, Xinying Guo, Lei Yang, Ying He, and Ziwei Liu. 2024b. Large Motion Model for Unified Multi-modal Motion Generation. In Computer Vision – ECCV 2024: 18th European Conference, Milan, Italy, September 29–October 4, 2024, Proceedings, Part XIII (Milan, Italy). Springer-Verlag, Berlin, Heidelberg, 397–421.

Zeyi Zhang, Tenglong Ao, Yuyao Zhang, Qingzhe Gao, Chuan Lin, Baoquan Chen, and Libin Liu. 2024a. Semantic Gesticulator: Semantics-Aware Co-Speech Gesture Synthesis. ACM Trans. Graph. (2024), 17 pages.

Chi Zhou, Tengyue Bian, and Kang Chen. 2022. GestureMaster: Graph-based Speech-driven Gesture Generation. In Proceedings of the 2022 International Conference on Multimodal Interaction (Bengaluru, India) (ICMI '22). Association for Computing Machinery, New York, NY, USA, 764–770.

Yi Zhou, Connelly Barnes, Lu Jingwan, Yang Jimei, and Li Hao. 2019. On the Continuity of Rotation Representations in Neural Networks. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

Motion-example-controlled Co-speech Gesture Generation Leveraging Large Language Models: Supplementary Materials

BOHONG CHEN, State Key Lab of CAD&CG, Zhejiang University, China
YUMENG LI, State Key Lab of CAD&CG, Zhejiang University, China
YOUYI ZHENG, State Key Lab of CAD&CG, Zhejiang University, China
YAO-XIANG DING, State Key Lab of CAD&CG, Zhejiang University, China
KUN ZHOU*, State Key Lab of CAD&CG, Zhejiang University, China

ACM Reference Format:
Bohong Chen, Yumeng Li, Youyi Zheng, Yao-Xiang Ding, and Kun Zhou. 2025. Motion-example-controlled Co-speech Gesture Generation Leveraging Large Language Models: Supplementary Materials. In *Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers (SIGGRAPH Conference Papers '25), August 10–14, 2025, Vancouver, BC, Canada.* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/3721238.3730611

1 LLM DATA FORMAT

To help readers better understand how our data is organized and fed into the LLM, we visually present the data format in Figure 1.

2 MOTION REPRESENTATION

To validate the effectiveness of introducing residual quantization layers and using only base quantization layer in the subsequent process, we conduct two ablation study in reconstruction task and one in downstream task. As shown in Table 1, recons (w/ res. in train&infer) refers to the setting where the quantized residual layer is used in both the training and inference stages of the reconstruction task. recons (w/ res. in train) indicates that the quantized residual layer is applied only during training, while recons (w/o res. in train&infer) denotes that it is not used in either training or inference. The results demonstrate that introducing residual layers in training significantly improves the model's reconstruction performance. Moreover, using only the base quantization layer during inference yields comparable results to using all residual quantization layers. Therefore, in subsequent processes, we only model the base quantization layer, instead of modeling all residual quantization layers [Guo et al. 2023; Zhang et al. 2024]. It can be also observed that incorporating the quantized residual layer during the training stage of VQ-VAE not

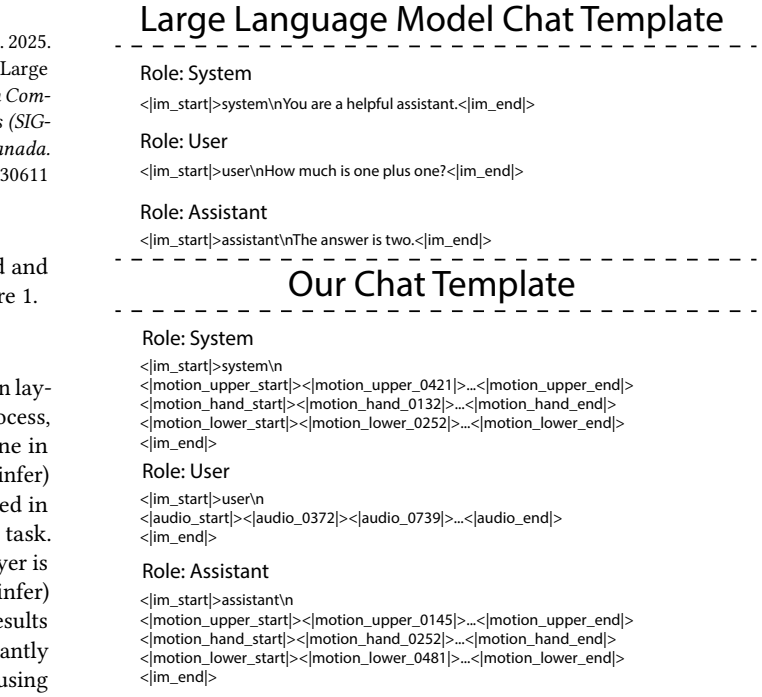

Fig. 1. The prompt format of regular LLM and of our method. We prompt the LLM using tokens from motion and audio modals with different template designs.

Table 1. Abalation study on BEAT2 [Liu et al. 2024]. We report FGD $\times 10^{-1}$, BC $\times 10^{-1}$, and diversity.

Method	FGD ↓	BC ↑	Diversity ↑
GT	0.000	6.897	12.75
recons (w/ res. in train&infer)	2.221	7.544	12.26
recons (w/ res. in train)	2.311	7.779	12.66
recons (w/o res. in train&infer)	2.974	7.940	12.48
MECo	3.401	7.346	15.30
MECo (w/o res. in train&infer)	3.762	7.833	15.19

*Corresponding author

Authors' addresses: Bohong Chen, bohongchen@zju.edu.cn, State Key Lab of CAD&CG, Zhejiang University, Hangzhou, China; Yumeng Li, yumeng.li@zju.edu.cn, State Key Lab of CAD&CG, Zhejiang University, Hangzhou, China; Youyi Zheng, youyizheng@zju.edu.cn, State Key Lab of CAD&CG, Zhejiang University, Hangzhou, China; Yao-Xiang Ding, dingyx.gm@gmail.com, State Key Lab of CAD&CG, Zhejiang University, Hangzhou, China; Kun Zhou, kunzhou@acm.org, State Key Lab of CAD&CG, Zhejiang University, Hangzhou, China.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
SIGGRAPH Conference Papers '25, August 10–14, 2025, Vancouver, BC, Canada
© 2025 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 979-8-4007-1540-2/2025/08
https://doi.org/10.1145/3721238.3730611

only improves reconstruction quality but also enhances the performance on downstream tasks from the comparison between MECo and MECo(w/o res. in train&infer).

It is worth noting that the setting(w/o res. in train&infer) results in relatively high BC scores, which, however, do not reflect better quality. From a visual perspective, this is primarily due to the presence of numerous meaningless jitters compared to our default setting(w/ res. in train).

3 COMPARISONS WITH OTHER LLM-BASED METHODS

We perform comparisons with relevant techniques that synthesize motions with speech or motion examples as input.

3.1 MotionGPT

MotionGPT [Jiang et al. 2024a] focuses on aligning motion modal with text modal. It does not incorporate speech or motion example control.

As MotionGPT is trained on a diverse range (15) of text-motion related tasks, we designed two experiments: (a) whether MotionGPT supports example-based control though not explicitly trained; (b) adding speech modality to MotionGPT.

When motion example and text description are fed into the MotionGPT model, instead of generating example-guided motion, it returned a textual description. This highlights that enabling example-based control requires specifically designed approaches.

We attempted to introduce speech modality to MotionGPT by constructing a speech-to-gesture task using speech-gesture data pairs from BEAT2. We incorporated this task into finetuning process to enable direct comparison with our model. Since the MotionGPT generated results do not include hands, we assign ground truth values for the hands. The result's FGD is 0.8784, which is much weaker than our 0.3401. Though this may be caused by many factors, this result highlights that different motion generation tasks in different modalities require unique design choices when utilizing LLMs, and it is not easy to transfer a model to another task.

3.2 T2M-GPT

T2M-GPT [Zhang et al. 2023] is a model solely trained on text-to-motion model, which does not support speech or example input. In addition, it does not use pre-trained LLM and was trained from scratch using a GPT structure similar to LLMs.

We designed an experiment to explore whether the T2M-GPT architecture and training design can tackle speech-to-gesture generation. To suit its architecture design, we use the speech transcripts, along with speech tokens as input, and train the model to generate the motion tokens. The final model achieved a FGD score of 0.7253, which is slightly worse than TalkSHOW's results and falls behind our 0.3401. Note that their design does not further allow example-based control.

3.3 M^{3}GPT

M^{3}GPT [Luo et al. 2024] adds music-dance tasks based on MotionGPT, and does not support example-based control. Unfortunately, this work is not open-sourced. Their github repository contains a template without training or inference code, and no model checkpoints are available.

4 ALIGNING WITH TEXT

We additionally align the text modality during our training process like MotionGPT and M³GPT, adding speech-to-text and text-to-gesture tasks, where the text corresponds to the speech transcripts. However, experimental results indicate that this negatively impacts speech-to-gesture performance, producing the FGD score of 0.4104, while also degrading the original textual capabilities of the LLM, lowering MMLU to 43.73. As we focus on speech-to-gesture solely, aligning with text modal ultimately compromises our primary objective and harms the fundamental capabilities of the LLM.

5 IMPACT OF FINETUNING ON LLM'S ORIGINAL TEXT CAPABILITIES

Regarding whether finetuning compromises LLM's original text capabilities, we provide several examples. Note that for motion related tasks, we only demonstrate MotionGPT, as T2M-GPT does not use pre-trained LLM and M³GPT is not open-sourced.

As shown in Table 2, when incorporating new modalities, existing works generally experience an inevitable degradation in text capabilities, while our method has small impact on the LLM's original text capabilities.

6 OBJECTIVE METRICS

We follow BEAT2 [Liu et al. 2024] benchmark and use the same way to calculate these metric.

6.1 Fréchet Gesture Distance (FGD)

A lower FGD, as referenced by [Yoon et al. 2020], indicates that the distribution between the ground truth and generated body gestures is closer. It is currently the metric that most closely aligns with human perception in evaluating the quality of gestures [Kucherenko et al. 2024]. Similar to the perceptual loss used in image generation tasks, FGD is calculated based on latent features extracted by a pretrained network:

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

where μ_r and Σ_r represent the first and second moments of the latent features distribution z_r of real human gestures g, and μ_g and Σ_g represent the first and second moments of the latent features distribution z_g of generated gestures $\hat{g}$.

6.2 Beat Constancy (BC)

A higher BC, suggests a closer alignment between the beat of gesture and the speech audio. It can be calculated by:

$$\mathrm{BC} = \frac{1}{g}\sum_{b_g \in g} \exp\left(-\frac{\min_{b_a \in a}\|b_g - b_a\|^2}{2\sigma^2}\right), \quad (2)$$

Table 2. Impact of finetuning on LLMs' original text capabilities, M&S refers to motion and speech.

Modality	Model Name	Base Model	Original MMLU↑	Finetuned MMLU↑	Degradation↓
Motion	MotionGPT	flan-t5-base	33.44	22.95	31.37%
Speech	SpeechGPT	llama-13b	46.90	27.13	42.15%
Vision	QwenVL2.5-7b-instruct	Qwen2.5-7b-instruct	74.20	70.17	5.43%
M&S	MECo	Qwen2.5-0.5b-instruct	46.50	46.27	0.49%
3M&S	MECo(7b llm)	Qwen2.5-7b-instruct	74.20	74.13	0.09%

6.3 L1 Diversity

A higher diversity indicates a larger variance in the given gesture clips. We calculate the average L1 distance from different N motion clips as follows:

$$\text{L1 div.} = \frac{1}{2N(N-1)} \sum_{t=1}^{N} \sum_{j=1}^{N} \left\| p_t^i - \hat{p}_t^j \right\|_1, \tag{3}$$

where p_t represents the position of joints in frame t, note that the character's translation is set to zero.

6.4 User Study

We conducted the user study on webpages to collect data. The user interface is shown in Figure 2. Note that we use two types of avatars during the process: one is the SMPL-X model, and the other is the Amy model from Mixamo. The main reason for using different avatars is that we utilize two datasets during our process: BEAT2 and ZeroEGGS. The BEAT2 dataset is represented in SMPL-X format. Retargeting the hand skeleton from SMPL-X to the Mixamo model is difficult and often produces visually obvious artifacts. Therefore, we directly used the mesh from SMPL-X.

Following GestureDiffuCLIP [Ao et al. 2023], for comparison purposes, we created two webpages to collect results. One evaluates the Human Likeness and Appropriateness of generated motions based solely on audio input. The other evaluates Human Likeness, Appropriateness, and Example Consistency when both audio and motion examples are provided as input. Each webpage is consist of three parts: textual explanation of the evaluation, evaluation videos, and scoring buttons. Each comparison group contains only two cases, which are played simultaneously on the left and right sides of the video for easier comparison.

7 MORE DISCUSSION

Since we first compress the motion into a latent representation using a motion tokenizer, our method struggles to provide joint-level control, such as precisely controlling a character's trajectory. Providing more precise and fine-grained control remains an area worthy of exploration. Additionally, our method can produce up to 36 seconds of motion within 1 second of processing time, but it's important to note that this is offline generation, as our motion tokenizer is not causal—meaning the current motion is influenced by both past and future tokens. A straightforward solution is using a causal motion tokenizer [Jiang et al. 2024b], which can ensure that current motions are only influenced by past tokens, thereby enabling real-time generation.

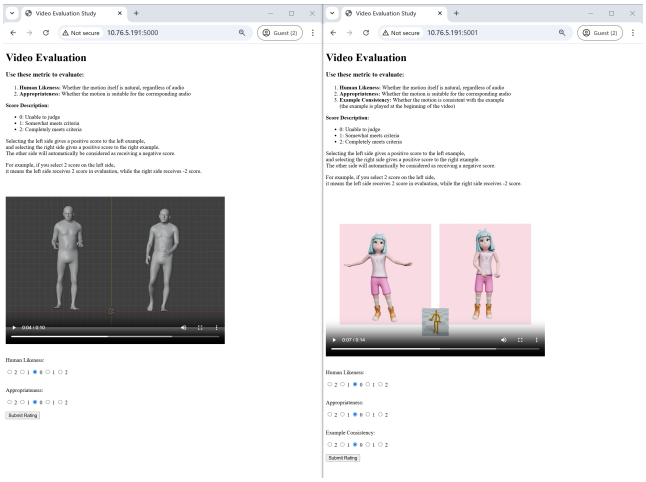

Fig. 2. Screenshot of the user interface used for user study.

REFERENCES

Tenglong Ao, Zeyi Zhang, and Libin Liu. 2023. GestureDiffuCLIP: Gesture Diffusion Model with CLIP Latents. *ACM Trans. Graph.* (2023), 18 pages.

Chuan Guo, Yuxuan Mu, Muhammad Gohar Javed, Sen Wang, and Li Cheng. 2023. MoMask: Generative Masked Modeling of 3D Human Motions. (2023). arXiv:2312.00063 [cs.CV]

Biao Jiang, Xin Chen, Wen Liu, Jingyi Yu, Gang Yu, and Tao Chen. 2024a. Motiongpt: Human motion as a foreign language. *Advances in Neural Information Processing Systems* 36 (2024).

Biao Jiang, Xin Chen, Ailing Zeng, Xinru Sun, Fukun Yin, Xianfang Zeng, Chi Zhang, Xuanyang Zhang, Gang Yu, and Tao Chen. 2024b. Causal Motion Tokenizer for Streaming Motion Generation.

Taras Kucherenko, Pieter Wolfert, Youngwoo Yoon, Carla Viegas, Teodor Nikolov, Mihail Tsakov, and Gustav Eje Henter. 2024. Evaluating Gesture Generation in a Large-scale Open Challenge: The GENEA Challenge 2022. 43, 3, Article 32 (jun 2024).

Haiyang Liu, Zihao Zhu, Giorgio Becherini, Yichen Peng, Mingyang Su, You Zhou, Naoya Iwamoto, Bo Zheng, and Michael J. Black. 2024. EMAGE: Towards Unified Holistic Co-Speech Gesture Generation via Masked Audio Gesture Modeling. arXiv:2401.00374 [cs.CV]

Mingshuang Luo, Ruibing Hou, Zhuo Li, Hong Chang, Zimo Liu, Yaowei Wang, and Shiguang Shan. 2024. M^3GPT: An Advanced Multimodal, Multitask Framework for Motion Comprehension and Generation. *Advances in Neural Information Processing Systems* (2024).

Youngwoo Yoon, Bok Cha, Joo-Haeng Lee, Minsu Jang, Jaeyeon Lee, Jaehong Kim, and Geehyuk Lee. 2020. Speech gesture generation from the trimodal context of text, audio, and speaker identity. *ACM Transactions on Graphics (TOG)* 39, 6 (2020), 1–16.

Jianrong Zhang, Yangsong Zhang, Xiaodong Cun, Shaoli Huang, Yong Zhang, Hongwei Zhao, Hongtao Lu, and Xi Shen. 2023. T2M-GPT: Generating Human Motion from Textual Descriptions with Discrete Representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).*

Zeyi Zhang, Tenglong Ao, Yuyao Zhang, Qingzhe Gao, Chuan Lin, Baoquan Chen, and Libin Liu. 2024. Semantic Gesticulator: Semantics-Aware Co-Speech Gesture Synthesis. *ACM Trans. Graph.* (2024), 17 pages.