

EmbodiChat: Towards Zero-Latency Text-driven Talking Face Generation for Embodied Dialogue

TIECHENG ZHANG^{1,*}, Huan Yu^{1,*}, Zhu Runjie¹, Zequan Xu²,
Wei Wang², Kaixiang Huang¹, Jin Wang^{1,†}

¹Zhejiang University, ²Zhejiang Sci-Tech University

Abstract

Seamless translation of natural language into embodied behaviors remains critical for open domain embodied dialogue. We formalize this task as mapping large language model text to synchronized speech and facial animations. Prevailing cascaded Text to Speech (TTS) and Talking Face Generation (TFG) pipelines suffer from prohibitive latency and poor cross platform generalizability. We present EmbodiChat, an end-to-end real-time dialogue system overcoming these bottlenecks. The key insight of EmbodiChat is that TTS acoustic representations capture pure “articulatory intent,” making them far more robust than audio-derived semantic features. Aligning facial generation directly onto this representation without additional adapters enables decoupled audio visual synthesis, fundamentally eliminating the serial latency of conventional audio first architectures. Concurrently, our framework decodes these representations into hardware agnostic action sequences. EmbodiChat achieves a stable Time-to-First-Audio (TTFA) of about 150 ms on a consumer GPU, significantly outperforming mainstream methods. Furthermore, it delivers a 3x speedup relative to its ablated audio driven baseline while maintaining comparable animation quality. Deployment across two distinct embodied agent platforms configurations demonstrates a robust, practically deployable text driven solution for embodied dialogue. The data, code and model will be released at https://github.com/JiangMingZi/Embodi_Chat

1 Introduction

The objective of embodied spoken dialogue systems is to facilitate natural and expressive interaction. For humanoid facial robots (Li et al., 2024; Hu et al., 2024, 2026; Liu et al., 2024b; Xu et al., 2026b,a), this requires translating language inputs into synchronized speech and facial

animation in real-time (Yoshida et al., 2025; Troussas et al., 2025; Sheng et al., 2025b; Dritsas et al., 2025; Sheng et al., 2025a). While Large Language Models can generate contextually rich responses, executing these outputs with open-domain fluidity under strict low-latency constraints remains a formidable challenge in multimodal generation. Consequently, the research field has diverged into two primary paradigms: audio-driven and text-driven Talking Face Generation (TFG).

Audio-driven methods (Prajwal et al., 2020; Zhou et al., 2020; Zhang et al., 2023) typically follow a sequential processing pipeline where audio synthesis precedes animation generation. These approaches map facial movements directly from features extracted by an audio encoder. This hard alignment ensures intrinsic temporal consistency between speech and lip motion, but the sequential nature inherently introduces significant end-to-end latency. More critically, it makes the system highly vulnerable to cascade errors. Streaming concatenation artifacts or acoustic noise from the preceding Text-to-Speech (TTS) stage can severely interfere with downstream facial generation. Furthermore, the final motion mapping is often tied to specific hardware, hindering generalization across diverse robotic platforms. Conversely, text-driven methods (Wang et al., 2025b; Jang et al., 2024; Yang et al., 2026a) decouple facial generation from the audio signal to improve inference speed. However, this decoupling often sacrifices precise audio-visual synchronization. Existing works attempt to bridge this gap using soft alignment via complex cross-modal adapters or indirect supervision signals. These strategies require massive aligned datasets that are scarce in real-world scenarios and still struggle to match the synchronization precision of their audio-driven counterparts.

This divergence highlights a fundamental challenge in cross-modal representation learning: the long-held assumption that rigorous cross-modal

*Equal contribution.

†Corresponding author.

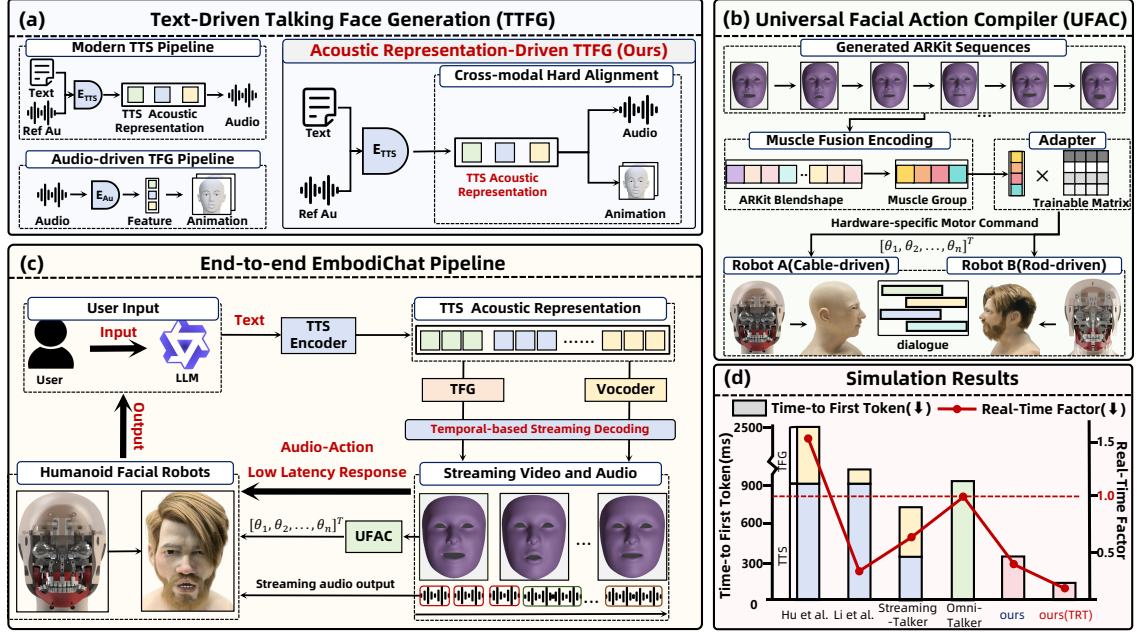


Figure 1: **Overview of EmbodiChat.** (a) Our adapter-free TTFG aligns TTS acoustic representations with facial animation. (b) UFAC maps ARKit blendshapes to robot-specific motor commands. (c) End-to-end pipeline for low-latency text-to-audio-action generation. (d) Simulation results showing improved latency and real-time performance.

alignment and modularly decoupled generation are mutually exclusive (Gao et al., 2024; Xiao et al., 2023; Ma et al., 2025; Pan et al., 2023; Shen et al., 2023; Xie et al., 2024; Wehrmann et al., 2020). In this work, we demonstrate that this dichotomy can be resolved by shifting the condition signal from terminal modalities (raw text or final audio) to the intermediate acoustic representation space of modern TTS models. We hypothesize that this latent space acts as an **idealized information bottleneck**. Raw text lacks critical prosodic timing, and terminal audio is overburdened with speaker-specific timbre and acoustic artifacts. In contrast, intermediate TTS latents perfectly isolate the purely linguistic and rhythmic motor plan of speech. By intercepting these latents via hard alignment, we achieve the precise temporal synchronization of audio-driven methods while bypassing the latency and acoustic interference inherent in sequential waveform synthesis. Because these latents are explicitly optimized to capture the underlying phonetic mechanics of speech production rather than surface-level acoustics, they are structurally isomorphic to the physical manifold of facial articulation.

By extracting and quantizing large-scale data within this purified subspace, we transform the challenging cross-modal alignment task into a stable feature-mapping problem. We systematize these insights into EmbodiChat, a unified text-driven TFG

framework. As a novel disentangled architecture, it replaces semantic representations from audio encoders with acoustic representations from the TTS module. To improve cross-platform transferability, we propose a Universal Facial Action Compiler (UFAC). Raw ARKit blendshape sequences are high-dimensional and highly correlated, which causes adaptation difficulties and temporal jitter when directly applied to robot control. UFAC exploits human facial muscle coordination patterns to perform amplitude inversion and linear fusion on blendshapes belonging to the same muscle group. This yields a set of linearly independent and low-dimensional fused muscle group codes. This structured latent space reduces control dimensions while preserving global facial geometry, enabling a lightweight learnable adapter to accurately align dynamic semantics with target robot actuators.

Combined with UFAC, EmbodiChat realizes full end-to-end generation from text input to synchronized robot speech and action. We evaluate our method on two structurally distinct humanoid robots equipped with LLMs. Experiments show that EmbodiChat achieves state-of-the-art performance in both dialogue responsiveness and expression naturalness. With an acoustic token chunk size of 20, our system achieves a Time-to-First-Audio (TTFA) of approximately 150 ms and a Real-Time Factor (RTF) of 0.17 on consumer-grade hardware

such as an NVIDIA RTX 4060 Ti GPU. Notably, compared to an ablated audio-driven baseline of our own architecture, this framework delivers a three-fold speedup. In summary, our main contributions are as follows:

1) We demonstrate that TTS acoustic representations capture pure articulatory intent, providing a more robust grounding signal for lip synchronization than audio-derived semantic features. This latent space acts as an information bottleneck that filters out non-articulatory noise.

2) We propose a streamlined text-driven talking face generation (TTFG) framework that eliminates cross-modal adapters, achieving natural audio-visual decoupling and efficient streaming inference.

3) We design the Universal Facial Action Compiler (UFAC), a hardware-agnostic latent space interface that maps high-dimensional blendshapes to low-dimensional muscle-group embeddings for deployment across heterogeneous robotic platforms.

4) We present EmbodiChat, an end-to-end text-to-face dialogue framework. By integrating TTFG and UFAC, EmbodiChat achieves state-of-the-art (SOTA) responsiveness (about 150 ms TTFA) on two distinct robots, delivering a 3x speedup over its audio-driven baseline and maintaining robustness under TTS artifacts at 0.17 RTF on a commodity GPU.

2 Related Works

Language-driven Talking Face Generation.

Language-driven talking face generation falls into two primary paradigms based on the conditioning signal, namely audio-driven and text-driven generation. Audio-driven methods (Fan et al., 2022; Chu et al., 2025; Pan et al., 2025) achieve high lip-sync accuracy by leveraging TTS-generated speech and self-supervised audio tokenizers. However, they suffer from cascaded coupling because animation generation must await full audio synthesis, which inherently limits real-time interaction (Stan et al., 2023). While recent state-of-the-art approaches (Chung et al., 2025; Liu et al., 2024a; Yang et al., 2026b) use sliding-window or streaming inference to alleviate this latency, they often introduce temporal boundary artifacts that degrade output quality and robustness. In contrast, text-driven methods (Wang et al., 2025b; Jang et al., 2024; Yang et al., 2026a; Mitsui et al., 2023) jointly model speech and facial animation in a unified framework to bypass the audio-first bottleneck. However, they typi-

cally rely on complex cross-modal supervision or intricate adapter modules. These methods also require scarce, high-quality aligned multimodal data, which restricts their practical deployment. We propose *EmbodiChat*, a novel text-driven 3D talking face framework trained without adapters or cross-modal supervision. Coupled with a Latent Streaming Decoder, it generates smooth facial animation incrementally from streaming TTS acoustic tokens, effectively decoupling audio dependency and enabling millisecond-level latency.

Embodied Dialogue and Facial Actuation.

High-fidelity facial expression is a cornerstone for grounding natural language processing in embodied dialogue systems. Early work focused on static emotional expressions via image-based imitation learning (Hu et al., 2024; Liu et al., 2024b; Xu et al., 2026a) to reproduce discrete affective states. Though expressive, these methods lack generalization for open-domain, real-world conversation involving dynamic viseme articulation. Recent efforts (Hu et al., 2026; Li et al., 2024; Xu et al., 2026b) map TFG outputs directly to physical action sequences, improving lip-sync generalization. Yet, they remain tightly coupled to specific hardware structures, limiting cross-platform transferability. We introduce UFAC, which encodes high-dimensional ARKit blendshapes into a disentangled, musculature-aware low-dimensional control space. A lightweight, learnable adapter then maps this universal behavioral representation to platform-specific actuation commands. To our knowledge, EmbodiChat integrated with UFAC constitutes the first end-to-end, real-time embodied conversational framework for facial robots that achieves seamless cross-hardware adaptability.

3 Method

3.1 Preliminaries and Task Definition

Two-Stage Modern TTS Architecture. Modern TTS systems (Jiang et al., 2023; Deng et al., 2025; Wang et al., 2025a; Team, 2026) adopt a two-stage pipeline that decouples high-level acoustic modeling from low-level waveform synthesis. This architecture is crucial to our work, as it exposes a structured, discrete acoustic representation space that can be leveraged independently of audio generation. The system relies on a pretrained Vector Quantized Variational Autoencoder (VQ-VAE) (Razavi et al., 2019) with a codebook $\mathcal{C} = \{e_k\}_{k=1}^K$ of K embedding vectors. Given a reference audio x_{ref} , it is first

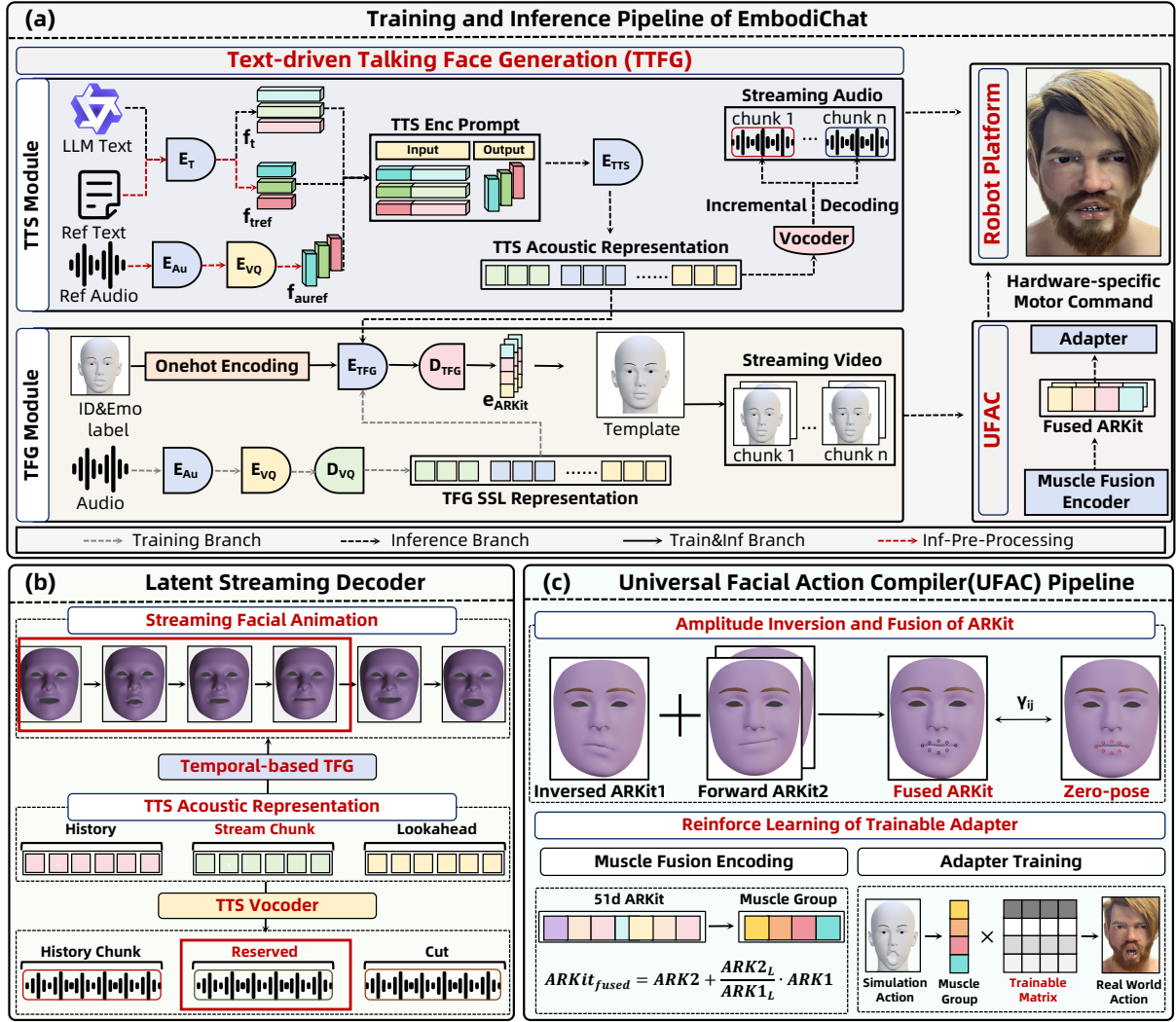


Figure 2: **Overview of the EmbodiChat Framework.** (a) Training and Inference Pipeline. (b) Latent Streaming Decoder. (c) Universal Facial Action Compiler (UFAC)..

encoded and quantized into a sequence of discrete codebook indices:

$$\mathbf{z}_{ref} = Q(E_{audio}(x_{ref})) \in \{1, \dots, K\}^{T_{ref}}, \quad (1)$$

where E_{audio} is a pretrained self-supervised audio encoder (Hsu et al., 2021; Baevski et al., 2020, 2022), and $Q(\cdot)$ is the VQ-VAE quantizer mapping each continuous vector to the nearest codebook entry index. During inference, given an input text sequence x and the reference audio x_{ref} , the system converts the text into a phoneme sequence embedded as a contextual representation h_x . A discrete acoustic autoregressive Transformer-based model $M_{acoustic}$ generates the target discrete token sequence $\mathbf{z} = \{z_t\}_{t=1}^T \in \{1, \dots, K\}^T$. This discrete sequence $\mathbf{z}$ is subsequently mapped to a continuous acoustic representation $\hat{h}$ by looking up the corresponding codebook embeddings. Finally,

a neural vocoder $G_{vocoder}$ synthesizes the output speech waveform as $y = G_{vocoder}(\hat{h})$.

Representation Alignment. In conventional TFG models, animations are driven by semantic features $H_{audio} = E_{audio}(x_{audio})$. We ground the equivalence between TTS and TFG representations in the VQ-VAE optimization objective. During pre-training, the VQ-VAE minimizes the reconstruction loss, ensuring the residual ϵ between the raw feature H_{audio} and its quantized continuous proxy $\hat{h}$ is strictly bounded by a convergence threshold δ :

$$\|\epsilon\| = \|H_{audio} - \hat{h}\| \leq \delta. \quad (2)$$

When driving our LSTM-based TFG generator Φ , the temporal error propagation is mathematically constrained by the contractive properties of the recurrent gates. Consequently, the final animation discrepancy Δ_{anim} is strictly bounded by a constant

multiple of the input residual:

$$\Delta_{\text{anim}} = \|\Phi(H_{\text{audio}}) - \Phi(\hat{h})\| \leq C \cdot \delta, \quad (3)$$

where C is an architectural constant (A rigorous formal proof detailing this bounded error propagation is provided in Appendix A.6). This theoretical guarantee confirms that $\hat{h}$ acts as an invariant proxy for H_{audio} , permitting direct, adapter-free substitution of TTS tokens into the TFG manifold and effectively eliminating the dependency on intermediate audio synthesis.

ARKit Blendshapes. ARKit blendshapes parameterize facial expressions through 52 interpretable action units (Inc., 2026). Formally, a facial expression at time t is represented as a vector $\mathbf{b}_t \in [0, 1]^{52}$. Following common practice (Aldenhoven et al., 2026), we exclude the TongueOut coefficient and adopt the remaining 51-dimensional ARKit representation as the target output space.

Problem Formulation. Building on the two-stage TTS architecture and the information equivalence between acoustic representation and raw audio features, we unify existing audio-driven and text-driven paradigms and formulate the following. Given input text x and reference audio x_{ref} , simultaneously generate synchronized streaming audio y and a 3D facial animation sequence $\mathbf{B} \in [0, 1]^{51 \times T}$ (ARKit blendshapes excluding TongueOut). For embodied deployment on humanoid facial robots, $\mathbf{B}$ must be compiled into a low-dimensional control sequence $\Delta \mathbf{m}_{\text{target}}$ tailored to specific kinematics. This two-stage objective eliminates cascaded delays and enables cross-platform generalization.

3.2 Text-Driven Talking Face Generation

Adapter-free Cross-modal Alignment. The core of our TTFG framework is an adapter-free training paradigm that aligns facial animation directly with the continuous acoustic representation space of a pretrained TTS system, as illustrated in Fig. 2(a). In the training branch, raw audio is first processed by the frozen TTS audio encoder and VQ-VAE to produce the semantic speech representation $\hat{h}$, which preserves rich prosodic structure. Leveraging natively aligned pairs $(\hat{h}, \mathbf{B})$ from the BEAT (Liu et al., 2022) dataset, where $\mathbf{B} \in [0, 1]^{51 \times T}$ represents ARKit blendshapes, the TFG module learns to predict facial motions from this shared latent space. We adopt a hybrid reconstruction loss on even-numbered frames to capture positional ac-

curacy:

$$\mathcal{L}_{\text{recon}} = \text{MSE}(\mathbf{B}_{\text{even}}, \hat{\mathbf{B}}_{\text{even}}) + \text{MAE}(\mathbf{B}_{\text{even}}, \hat{\mathbf{B}}_{\text{even}}), \quad (4)$$

and a smoothing loss on odd frames to enforce temporal coherence and suppress high-frequency jitter:

$$\mathcal{L}_{\text{smooth}} = \text{MSE}(\mathbf{B}_{\text{odd}}, \bar{\mathbf{B}}) + \text{MAE}(\mathbf{B}_{\text{odd}}, \bar{\mathbf{B}}), \quad (5)$$

where $\bar{\mathbf{B}} = \frac{1}{2}(\hat{\mathbf{B}}_{\text{even}, t-1} + \hat{\mathbf{B}}_{\text{even}, t+1})$. The total loss $\mathcal{L}_{\text{total}} = \lambda_{\text{recon}} \mathcal{L}_{\text{recon}} + \lambda_{\text{smooth}} \mathcal{L}_{\text{smooth}}$ ensures stable animation generation, allowing the TFG module to act as a plug-and-play component during text-driven inference.

Latent Streaming Decoder. The primary latency bottleneck in conventional pipelines stems from the sequential generation of full audio waveforms. We overcome this using a Latent Streaming Decoder, visualized in Fig. 2(b). During the inference branch, the autoregressively generated discrete acoustic tokens act as a shared clock. These chunks of tokens synchronously drive the temporal-based streaming decoding for both the TTS vocoder and the TFG module.

For facial animation, our TFG satisfies the streaming consistency property, meaning the output at any incremental step t matches exactly the prefix of a full-sequence inference. Letting $\mathbf{h}_{1:T} = [h_1, \dots, h_T]$, we guarantee drift-free incremental animation via:

$$\text{TFG}_{\text{stream}}(\mathbf{h}_{1:t}) = [\text{TFG}_{\text{full}}(\mathbf{h}_{1:T})]_{1:t}. \quad (6)$$

Simultaneously, the TTS vocoder processes each token chunk $\mathcal{I}_k = \{(k-1)C + 1, \dots, kC\}$ using H historical and L future tokens:

$$\mathbf{h}_{\text{ctx}, k} = \mathbf{h}_{[\max(1, (k-1)C-H+1): \min(T, kC+L)]}. \quad (7)$$

Only the waveform segment strictly corresponding to $\mathcal{I}_k$ is retained. This parallel decoupled design maintains rigorous audiovisual synchronization while eliminating the need to wait for audio synthesis completion. Operating directly on TTS acoustic tokens enables fine-grained facial animation down to small chunks, drastically reducing TTFA and ensuring high responsiveness for interactive scenarios. Details of our framework versus traditional methods were shown in Appendix A.8

3.3 Universal Facial Action Compiler (UFAC)

To bridge the gap between digital animations and physical hardware, UFAC maps the 51-dimensional

ARKit blendshapes to low-dimensional robot actuation signals, as detailed in Fig. 2(c).

Amplitude Inversion and Muscle Fusion. Directly applying high-dimensional ARKit sequences to robot control causes adaptation difficulties due to structural correlations and motor linkage conflicts. We resolve this by partitioning the blendshapes $\{b_k\}_{k=1}^{51}$ into n anatomically coherent muscle groups. Within each group, for any antagonistic pair (b_i, b_j) , we compute a fixed fusion ratio γ_{ij} by minimizing the 3D keypoint MSE between the neutral zero-pose and a counterbalanced physical activation:

$$\gamma_{ij} = \arg \min_{\gamma} \|\mathcal{K}(f_0 - \Delta f_i + \gamma \Delta f_j) - \mathcal{K}(f_0)\|_2^2, \quad (8)$$

where f_0 is the neutral facial mesh, Δf_i represents the deformation induced by setting $b_i = 1$, and $\mathcal{K}(\cdot)$ extracts 3D keypoints. This geometrical compression eliminates invalid deformations, allowing all blendshapes in a group to be fused into a canonical scalar c_i .

Motor Calibration and Adapter Reinforcement Learning. Each fused channel is first actuated physically to record the resulting motor angle deviation vector $\Delta \mathbf{m}_i \in \mathbb{R}^{n \times 1}$, forming the hardware-specific motion matrix $\mathbf{M} = [\Delta \mathbf{m}_1, \dots, \Delta \mathbf{m}_n]$. Subsequently, a trainable adapter row vector $\mathbf{A} = [\alpha_1, \dots, \alpha_n]$ is introduced to scale each channel’s contribution. The final linear superposition is

$$\Delta \mathbf{m}_{\text{final}} = \sum_{i=1}^n \alpha_i c_i \Delta \mathbf{m}_i = \mathbf{M}(\mathbf{A}\mathbf{C})^T, \quad (9)$$

and the target motor pose is $\mathbf{m}_{\text{target}} = \mathbf{m}_0 + \Delta \mathbf{m}_{\text{final}}$, where $\mathbf{C} = \text{diag}(c_1, \dots, c_n)$ and $\mathbf{m}_0$ is the neutral motor pose. The parameters α are fine-tuned via reinforcement learning to minimize the discrepancy between the physical robot face and the digital simulation action. This effectively yields a universal and hardware-agnostic compiler capable of seamlessly adapting text-driven expressions to diverse robotic platforms, as shown in Appendix B.2.

4 Experiments

4.1 Simulation Experiment

Benchmark. We construct a simulation benchmark as follows. The Gemini model generates 20 diverse text sentences of varying lengths covering everyday conversation topics, serving as system

Method	TFG FPS $\uparrow$	TTEA (s) $\downarrow$	RTF $\downarrow$
Hu et al.	1.06	4.213 $\pm$ 1.773	1.51
Li et al.	1250	1.251 $\pm$ 0.411	0.464
StreamingTalker	40	0.796 $\pm$ 0.021	0.524
OmniTalker	25	1.213 $\pm$ 0.324	1.000
StreamChar	24.00	1.340	0.975
Hallo-Live	20.38	0.940	1.180
TokTalk	53.50	0.400	1.000
Ours (PyTorch)	7500	0.378$\pm$0.021	0.483
Ours (TensorRT)	13000	0.148$\pm$0.004	0.173

Table 1: Comparison of EmbodiChat and baselines.

inputs. The reference timbre for the TTS module is randomly sampled from the BEAT (Liu et al., 2022) dataset. Given the limited prior work on text-to-talking-face generation for bionic facial robots, our benchmark focuses on end-to-end system performance from text input to final facial animation output. For audio-driven cascaded methods, we fix the upstream TTS module as GPT-SoVITS (RVC-Boss, 2026) to form a complete pipeline and compute end-to-end metrics by summing individual module latencies. This TTS choice is consistent with the pre-trained TTS model adopted in our proposed approach, ensuring a fair alignment between the cascaded baselines and our method. For streaming audio-driven methods, we adopt the same audio chunk size as our proposed approach for a fair comparison. In contrast, text-driven methods are evaluated directly on their end-to-end response performance.

Training Details. The training set is constructed by extracting paired audio and 3D facial animation sequences from the BEAT dataset. Specifically, we exclusively use its facial animation and audio data and restrict all training samples to English utterances. To align the temporal resolution of the kinematics with the acoustic tokens, the raw ARKit blendshape animations are downsampled from 60 fps to 50 fps. Furthermore, the dataset is systematically categorized based on speaker identity and emotion labels. To extract the driving features, we leverage GPT-SoVITS’s zero-shot audio cloning pipeline to instantiate the TTS acoustic representation space: it sequentially passes through the pre-trained audio encoder, the VQ-VAE encoder, and the VQ-VAE quantization decoder. The resulting continuous feature vector represents the precise projection of the audio within the TTS acoustic representation space. These features, paired with their corresponding ARKit sequences, form the training data for the TFG module. The TFG module of Em-

bodiChat takes as input the joint acoustic representation encoded by the GPT-SoVITS audio encoder and its VQ-VAE, and is trained for 600 epochs using the AdamW optimizer on an NVIDIA RTX 4060 Ti GPU. At inference time, the TFG module is seamlessly integrated into the TTS system in a plug-and-play manner. It takes the acoustic representation as input and runs asynchronously in parallel with the vocoder, jointly generating lip-synchronized facial animations alongside the audio output.

Evaluation Details. We compare EmbodiChat against seven SOTA talking-face generation methods, including Hu et al. (Hu et al., 2026), Li et al. (Li et al., 2024), StreamingTalker (Yang et al., 2026b), OmniTalker (Wang et al., 2025b), StreamChar (Tian et al., 2026), Hallo-Live (Li et al., 2026), and TokTalk (Zhao et al., 2026). The first two are audio-driven methods previously validated on humanoid facial robots; StreamingTalker is a streaming audio-driven variant that achieves the best inference performance among audio-based approaches; OmniTalker, in contrast, follows a text-driven paradigm. Since our initial submission, several concurrent streaming text-driven TFG methods have appeared, namely StreamChar, Hallo-Live, and TokTalk, which were not available at the time of our experimental freeze. We include them as additional baselines in the camera-ready version. For baseline models, we use the official pretrained parameters and deploy them in PyTorch. Model warm-up is performed before inference to eliminate kernel launch overhead. During inference, EmbodiChat employs a fixed streaming decoding window of 20 tokens, corresponding to an initial audio chunk of 1.6 seconds to ensure audio quality. We also include a TensorRT-optimized variant as an additional baseline. All methods use the same set of input texts. We report the Frames Per Second (FPS) of the system, the Time-to-First-Audio (TTFA), and the Real-Time Factor (RTF) for each method. Notably, an $RTF < 1$ can be considered indicative of a real-time interactive system, and a $TTFA < 500\text{ms}$ can be considered a low-latency real-time system.

Hardware Platforms. All audio-driven methods and our EmbodiChat are evaluated on a single NVIDIA RTX 4060 Ti GPU. The text-driven baselines are reported to use the official hardware configurations: OmniTalker on an NVIDIA RTX 4090; StreamChar on an NVIDIA H100; Hallo-Live on dual NVIDIA H200 GPUs; and TokTalk

on an NVIDIA L40S. We note that all these text-driven baselines are evaluated on significantly more powerful GPUs than our RTX 4060 Ti, which gives them a substantial hardware advantage. StreamingTalker is driven by EmbodiChat-generated audio chunks with the same audio chunk size as our method for fairness, and its reported latency includes both TTS and TFG inference. The reported results for StreamChar, Hallo-Live, and TokTalk are based on their best officially reported or estimated values.

Quantitative results. As shown in Table 1, EmbodiChat establishes a new SOTA in real-time text-to-talking-face generation for humanoid facial robots. It achieves a consistently low TTFA of 0.378 seconds in PyTorch and just 0.148 seconds with TensorRT optimization—well below the 500-millisecond threshold essential for natural human-robot interaction, outperforming all compared methods, including recently published concurrent works, despite running on a consumer-grade GPU while several baselines rely on datacenter-class hardware. To the best of our knowledge, EmbodiChat is the first system that simultaneously meets all requirements for a truly low-latency, real-time talking-face pipeline on bionic platforms under consumer-grade hardware constraints. It operates in a streaming manner with a chunk size of 20 tokens, attains an RTF as low as 0.173 for audio, and delivers facial animation generation throughput of up to 13,000 frames per second, exceeding existing approaches by over one orders of magnitude. Notably, EmbodiChat-TensorRT achieves the lowest latency and RTF among all compared methods on a single consumer-grade GPU, while the concurrent baselines require high-end datacenter GPUs (H100, H200, L40S) and still yield substantially higher TTFA and RTF values. In contrast, audio-driven methods by Hu et al. and Li et al., along with the text-driven OmniTalker, are non-streaming and exhibit significantly higher Time-to-First-Audio and Real-Time Factor, rendering them unsuitable for interactive robotic deployment. These results confirm that EmbodiChat is the first real-time talking-face system demonstrated to achieve seamless integration with humanoid facial robots.

4.2 Ablation Study

To analyze the contribution of each component to EmbodiChat’s overall performance, we perform ablation experiments at three levels using the same

Table 2: **Ablation study results.** TTS1 denotes GPT-SoVITS (pre-trained on 7,000 hours of audio) and TTS2 denotes IndexTTS (Deng et al., 2025) (12,000 hours). All groups adopt the LSTM-based TFG module; for the same TTS framework, the TFG module is kept identical with only the input source changed.

Configuration	LSE-D ↓	LSE-C ↑	TTFA (s) ↓	RTF ↓
TTS1 + Audio-driven	13.177	1.442	1.231±0.411	0.521
TTS1 + Text-driven(ours)	13.204	1.261	1.163±0.394	0.453
TTS2 + Audio-driven	12.368	3.042	2.991±1.173	1.132
TTS2 + Text-driven	12.174	3.017	2.914±1.133	1.062
TTS1 (Vocoder broken) + Audio-driven	15.543	0.394	/	/
TTS1 (Vocoder broken) + Text-driven	14.942	0.495	/	/
ours (Streaming chunk size=20)	13.307	1.137	0.378±0.021	0.483
ours (Streaming chunk size=40)	13.354	1.294	0.826±0.037	0.456

input settings, as shown in Table 2.

(1) Compatibility with Modern Two-Stage TTS Architectures. We select two representative open-source two-stage TTS systems that perform acoustic modeling followed by vocoder-based waveform synthesis, both supporting reference-audio voice cloning. For each system, audio clips from the BEAT dataset are processed through its native audio encoder and VQ-VAE to extract acoustic representations, which replace conventional hand-crafted audio features as inputs for TFG training. We evaluate two driving paradigms: in the audio-driven pipeline, text is first synthesized into full audio via TTS, and the generated audio is then encoded into acoustic representations to drive facial animation through the TFG module; in contrast, the text-driven paradigm directly predicts TTS acoustic representations from text and feeds them into TFG without intermediate audio generation. Strong performance on both TTS systems demonstrates the generalizability of our TFG training framework. It fully exploits the inherent decoupling between acoustic modeling and waveform synthesis in modern TTS pipelines, enforces strict speech-face alignment through hard constraints in the acoustic latent space, and remains agnostic to the underlying TTS architecture. Moreover, TTS models pretrained for longer durations yield more expressive acoustic latent spaces, leading to improved TFG generation performance. **(2) Comparison Between Audio-Driven and Our Text-Driven Paradigm.** We further compare the robustness of the proposed text-driven scheme with conventional audio-driven facial animation. Beyond standard lip-sync and perceptual quality metrics, we intentionally degrade vocoder performance by adjusting sampling bitrate to test the model’s sensitivity to downstream audio

distortion. Experimental results demonstrate that TTS acoustic representations exhibit stronger robustness than raw audio semantic features. More importantly, our text-driven design fully decouples facial animation generation from the quality of final synthesized audio, greatly improving system resilience. **(3) Impact of Latent Streaming Decoding on System Responsiveness.** We evaluate three decoding strategies under unified PyTorch settings: full-sequence decoding, and incremental streaming decoding with fixed chunk sizes of 20 and 40 tokens. The results show that incremental decoding substantially reduces the Time-to-First-Audio (TTFA) and only marginally increases the Real-Time Factor ($RTF < 1$) in interactive scenarios, yielding superior system responsiveness with further acceleration potential via TensorRT. As TFG is a temporal network, output discrepancies between full-sequence and incremental decoding are limited to negligible floating-point errors. Accordingly, latent streaming decoding introduces only slight degradation to concatenated audio quality and exerts nearly no influence on the temporal consistency of facial animation.

4.3 Real-World Evaluation

Evaluation Baseline. To evaluate EmbodiChat’s real-world performance, we conduct static and dynamic experiments on two robots with distinct actuation: Robot A (cable-driven) and Robot B (rod-driven). In static evaluation, we collect human subjective scores from 32 participants for lip-sync and naturalness, and also employ Qwen3.5 Omni Plus as an automatic evaluator on rendered digital human videos and robot-captured videos, as shown in Fig. 3. In dynamic evaluation, we integrate a cloud-hosted LLM for real-time conversation and

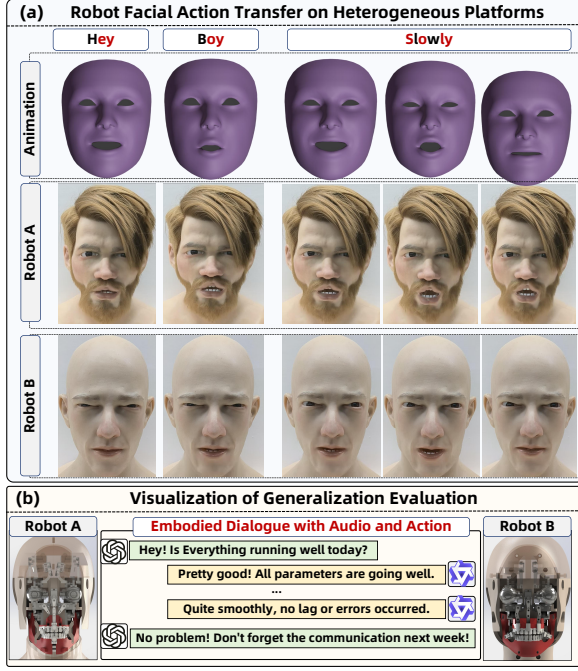


Figure 3: Facial Action Transfer onto Heterogeneous Humanoid Robot Platforms.

measure end-to-end latency. Additional details are in Appendix A.1 and A.2.

Evaluation Details. For static evaluation, participants assessed each video on a 5-point Likert scale with respect to lip-sync accuracy and naturalness. Qwen3.5 Omni Plus processed the videos at 10 fps and generated five scores per utterance; the highest and lowest scores were excluded, and the mean of the remaining three was computed. Human ratings were averaged over 32 participants. For dynamic evaluation, only high-level scenario prompts were provided to the LLMs, enabling unconstrained responses.

Quantitative Analysis. As summarized in Table 3, EmbodiChat demonstrates strong cross-platform adaptability by deploying the same facial animation pipeline onto rendered digital human videos and two physically distinct robots without hardware-specific redesign. The high human and LLM naturalness scores across all platforms confirm that EmbodiChat preserves expressive quality regardless of the underlying actuation mechanisms, achieving hardware-agnostic facial animation transfer. The framework latency, defined as the time introduced by EmbodiChat’s audio-visual synchronization and actuation pipeline, is consistently low at 0.158 ± 0.005 seconds for both robots. This latency is imperceptible to human users and negligible compared to the network de-

lay incurred by cloud-based LLM inference (about 0.6 s). Consequently, the end-to-end response time from text input to synchronized facial motion remains well within the threshold for real-time interaction. These results collectively validate that EmbodiChat enables robust, real-world deployment across heterogeneous robotic platforms while maintaining high perceptual quality and low system latency. Additional cross-lingual generalization results are provided in Appendix A.3.

Platform	Human		LLM	
	Lip-Sync $\uparrow$	Natural $\uparrow$	Lip-Sync $\uparrow$	Natural $\uparrow$
Video	4.70	4.84	4.53	4.83
Robot A	4.20	4.28	4.14	4.36
Robot B	4.06	4.42	4.16	4.30

Table 3: Performance comparison across different platforms with human and LLM evaluation. Framework latency: Robot A/B = 0.158 ± 0.005 s, Video = N/A.

5 Conclusion

We present EmbodiChat, the first real-time text-driven talking face generation framework for humanoid facial robots. We quantitatively demonstrate that the acoustic representation space of modern TTS systems with zero-shot voice cloning capability can be directly used for downstream TFG tasks, and the TFG quality scales with TTS pretraining data volume. Based on this insight, we design an adapter-free TFG framework trained under a single audio-driven paradigm and introduce a Latent Streaming Decoder to achieve about 150 ms end-to-end latency on commodity GPUs. We further propose UFAC, which encodes the ARKit parameters output by TFG into a universal low-dimensional intermediate representation via muscle-group fusion encoding, and maps the representation onto structurally heterogeneous humanoid robot platforms through a trainable lightweight adapter, forming a complete end-to-end pipeline from text to synchronized speech and facial motion. Ethical considerations and broader societal impact are discussed in Appendix C.

Acknowledgement. This research was supported by the Zhejiang Provincial Natural Science Foundation of China (Grant No. LD24E050010), the "Pioneer" and "Leading Goose" R&D Program of Zhejiang (Grant No. 2024C01020), and the Ningbo Science and Technology Project (Grant No. 2024Z295).

Limitations

While EmbodiChat achieves state-of-the-art performance in low-latency embodied dialogue, several limitations remain that offer opportunities for future research.

Trade-off Between Visual Fidelity and System Agility. To achieve ultra-low latency and hardware-agnostic deployment on physical robots, EmbodiChat relies on a compact 51-dimensional ARKit blendshape representation. Consequently, compared to state-of-the-art digital human generation methods that directly regress thousands of 3D vertices or high-resolution 2D pixels, our approach involves an inherent trade-off in visual fidelity for micro-expressions. This ceiling on generation fineness is also constrained by the abstraction level of the blendshape dataset itself. However, we argue that in the context of real-time physical embodied interaction, the agility of system response and highly synchronized lip movements are vastly more critical to a natural user experience than extreme static rendering quality, making this a necessary and practical compromise.

Dependence on Cloud-based LLMs. As analyzed in our real-world evaluations (Table 3), while our framework latency is consistently low (≈ 158 ms), the total system response time is still dominated by the cloud-hosted Large Language Model (LLM) inference. In environments with unstable network connectivity, the end-to-end user experience may degrade due to external communication overhead. Integrating high-performance locally-deployed LLMs is a logical next step to ensure complete system autonomy and robustness.

Physical Actuation Bottleneck. EmbodiChat provides highly responsive time-to-first-audio and high-fidelity ARKit facial animation, which the UFAC module then compiles into low-dimensional facial action representations. Although these low-dimensional action representations are approximately decoupled from one another, on hardware platforms with limited degrees of freedom, certain low-dimensional action representations cannot be directly realized. EmbodiChat can still operate on such platforms; however, this inability to fully execute the compiled action set can disrupt the continuity of facial expression delivery. Therefore, while UFAC substantially reduces dependence on specific low-level hardware structures, the underlying hardware must still support the low-dimensional

facial action representations to achieve the most faithful rendering possible.

Servo-Induced Acoustic Noise. Cost constraints led us to use standard hobby servo motors instead of silent actuators; these servos produce audible mechanical noise that contaminates audio recordings with broadband transients and low-frequency tones, mildly masking speech and confounding fine-grained audio analyses such as voice activity detection, emotion recognition, and turn-taking latency measurement. Although dialogue remains intelligible and user study ratings unaffected, future hardware iterations can mitigate this issue through mechanical isolation, sound dampening, or speech enhancement leveraging the known servo noise profile.

Social Perception and Uncanny Valley Considerations. While our user study focused on task-oriented naturalness and response agility, the long-term social acceptance of EmbodiChat raises potential uncanny valley effects not captured by single-session metrics. This concern stems particularly from the combination of high-fidelity lip synchronization, audible servo noise, and the absence of full-body gestures. A dedicated discussion of these social perception factors along with preliminary mitigation strategies is provided in Appendix C. Nevertheless, systematic longitudinal user studies across different cultural and age demographics remain necessary to fully assess the boundary conditions of comfortable human-robot interaction with our system.

References

- Céline Madeleine Aldenhoven, Leon Nissen, Marie Heinemann, Cem Doğdu, Alexander Hanke, Stephan Jonas, and Lara Marie Reimer. 2026. Real-time emotion recognition performance of mobile devices: A detailed analysis of camera and truedepth sensors using apple’s arkit. *Sensors*, 26(3):1060.
- Alexei Baevski, Wei-Ning Hsu, Qiantong Xu, Arun Babu, Jiatao Gu, and Michael Auli. 2022. Data2vec: A general framework for self-supervised learning in speech, vision and language. In *International conference on machine learning*, pages 1298–1312. PMLR.
- Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in neural information processing systems*, 33:12449–12460.

- Xuangeng Chu, Nabarun Goswami, Ziteng Cui, Hanqin Wang, and Tatsuya Harada. 2025. Artalk: Speech-driven 3d head animation via autoregressive model. In *Proceedings of the SIGGRAPH Asia 2025 Conference Papers*, pages 1–9.
- Chaeyeon Chung, Ilya Fedorov, Michael Huang, Aleksey Karmanov, Dmitry Korobchenko, Roger Blanco i Ribera, and Yeongho Seol. 2025. [Audio2face-3d: Audio-driven realistic facial animation for digital avatars](#). *CoRR*, abs/2508.16401.
- Wei Deng, Siyi Zhou, Jingchen Shu, Jinchao Wang, and Lu Wang. 2025. Indextts: An industrial-level controllable and efficient zero-shot text-to-speech system. *arXiv preprint arXiv:2502.05512*.
- Elias Dritsas, Maria Trigka, Gerasimos Vonitsanos, and Phivos Mylonas. 2025. Human-computer interaction in the big data era: Systems, cognition, and interactive intelligence. In *Proceedings of the 3rd International Conference of the ACM Greek SIGCHI Chapter*, pages 61–70.
- Yingruo Fan, Zhaojiang Lin, Jun Saito, Wenping Wang, and Taku Komura. 2022. Faceformer: Speech-driven 3d facial animation with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18770–18780.
- Yuting Gao, Jinfeng Liu, Zihan Xu, Tong Wu, Enwei Zhang, Ke Li, Jie Yang, Wei Liu, and Xing Sun. 2024. Softclip: Softer cross-modal alignment makes clip stronger. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 1860–1868.
- 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*, 29:3451–3460.
- Yuhang Hu, Boyuan Chen, Jiong Lin, Yunzhe Wang, Yingke Wang, Cameron Mehlman, and Hod Lipson. 2024. Human-robot facial coexpression. *Science Robotics*, 9(88):eadi4724.
- Yuhang Hu, Jiong Lin, Judah A. Goldfeder, Philippe Martin Wyder, Yifeng Cao, Steven Tian, Yunzhe Wang, Jingran Wang, Mengmeng Wang, Jie Zeng, Cameron Mehlman, Yingke Wang, Delin Zeng, Boyuan Chen, and Hod Lipson. 2026. [Learning realistic lip motions for humanoid face robots](#). *Sci. Robotics*, 11(110).
- Apple Inc. 2026. ARKit — apple developer documentation. <https://developer.apple.com/documentation/arkit>. Accessed: 2026-04-24.
- Youngjoon Jang, Ji-Hoon Kim, Junseok Ahn, Doyeop Kwak, Hongsun Yang, Yooncheol Ju, Ilhwan Kim, Byeong-Yeol Kim, and Joon Son Chung. 2024. [Faces that speak: Jointly synthesising talking face and speech from text](#). *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8818–8828.
- Ziyue Jiang, Yi Ren, Zhenhui Ye, Jinglin Liu, Chen Zhang, Qian Yang, Shengpeng Ji, Rongjie Huang, Chunfeng Wang, Xiang Yin, Zejun Ma, and Zhou Zhao. 2023. [Mega-tts: Zero-shot text-to-speech at scale with intrinsic inductive bias](#). *CoRR*, abs/2306.03509.
- Boren Li, Hang Li, and Hangxin Liu. 2024. Driving animatronic robot facial expression from speech. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 7012–7019. IEEE.
- Chunyu Li, Jiaye Li, Ruiqiao Mei, Haoyuan Xia, Hao Zhu, Jingdong Wang, and Siyu Zhu. 2026. Hallo-live: Real-time streaming joint audio-video avatar generation with asynchronous dual-stream and human-centric preference distillation. *arXiv preprint arXiv:2604.23632*.
- Haiyang Liu, Zihao Zhu, Giorgio Becherini, Yichen Peng, Mingyang Su, You Zhou, Xuefei Zhe, Naoya Iwamoto, Bo Zheng, and Michael J Black. 2024a. Emage: Towards unified holistic co-speech gesture generation via expressive masked audio gesture modeling. In *Proc. 2024 IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 1144–1154.
- Haiyang Liu, Zihao Zhu, Naoya Iwamoto, Yichen Peng, Zhengqing Li, You Zhou, Elif Bozkurt, and Bo Zheng. 2022. Beat: A large-scale semantic and emotional multi-modal dataset for conversational gestures synthesis. In *European conference on computer vision*, pages 612–630. Springer.
- Xiaofeng Liu, Rongrong Ni, Biao Yang, Siyang Song, and Angelo Cangelosi. 2024b. Unlocking human-like facial expressions in humanoid robots: A novel approach for action unit driven facial expression disentangled synthesis. *IEEE Transactions on Robotics*, 40:3850–3865.
- Wei Ma, Yucheng Huang, Shengjun Tang, Xianwei Zheng, Zhen Dong, Liang Ge, Jianping Pan, Qingquan Li, and Bing Wang. 2025. [Cross-modal 2d-3d feature matching: simultaneous local feature description and detection across images and point clouds](#). *ISPRS Journal of Photogrammetry and Remote Sensing*, 229:155–169.
- Kentaro Mitsui, Yukiya Hono, and Kei Sawada. 2023. Uniflg: Unified facial landmark generator from text or speech. *arXiv preprint arXiv:2302.14337*.
- Yifang Pan, Karan Singh, and Luiz Gustavo Hafemann. 2025. Model see model do: Speech-driven facial animation with style control. In *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers*, pages 1–10.
- Zhengxin Pan, Fangyu Wu, and Bailing Zhang. 2023. Fine-grained image-text matching by cross-modal hard aligning network. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 19275–19284.

- KR Prajwal, Rudrabha Mukhopadhyay, Vinay P Nambodiri, and CV Jawahar. 2020. A lip sync expert is all you need for speech to lip generation in the wild. In *Proceedings of the 28th ACM international conference on multimedia*, pages 484–492.
- Ali Razavi, Aaron Van den Oord, and Oriol Vinyals. 2019. Generating diverse high-fidelity images with vq-vae-2. *Advances in neural information processing systems*, 32.
- RVC-Boss. 2026. GPT-SoVITS: An efficient llm-based text-to-speech model. <https://github.com/RVC-Boss/GPT-SoVITS>. Accessed: 2026-04-24.
- Junhong Shen, Liam Li, Lucio M Dery, Corey Staten, Mikhail Khodak, Graham Neubig, and Ameet Talwalkar. 2023. Cross-modal fine-tuning: Align then refine. In *International Conference on Machine Learning*, pages 31030–31056. PMLR.
- Qincheng Sheng, Wei Tang, Hao Qin, Yujie Kong, Haokai Dai, Yiding Zhong, Yonghao Wang, Jun Zou, and Huayong Yang. 2025a. A review on an ai-driven face robot for human-robot expression interaction. *Science China Technological Sciences*, 68(10):2010301.
- Qincheng Sheng, Zhongxiang Zhou, Jinhao Li, Xiangyu Mi, Pingyu Xiang, Zhenghan Chen, Haocheng Xu, Shenhan Jia, Xiyang Wu, Yuxiang Cui, Shuhao Ye, Jiyu Yu, Yuhan Du, Shichao Zhai, Kechun Xu, Yifei Yang, Zhichen Lou, Zherui Song, Zikang Yin, and 4 others. 2025b. [A comprehensive review of humanoid robots](#). *SmartBot*, 1(1):e12008.
- Stefan Stan, Kazi Injamamul Haque, and Zerrin Yumak. 2023. Facediffuser: Speech-driven 3d facial animation synthesis using diffusion. In *Proceedings of the 16th ACM SIGGRAPH Conference on Motion, Interaction and Games*, pages 1–11.
- Qwen Team. 2026. [Qwen3-tts technical report](#). *CoRR*, abs/2601.15621.
- Linrui Tian, Qi Wang, and Bang Zhang. 2026. Stream-char: Long-horizon streaming character audio-video generation with decoupled orchestration. *arXiv preprint arXiv:2605.25659*.
- Christos Troussas, Akrivi Krouska, and Cleo Sgouropoulou. 2025. *Human-Computer Interaction and Augmented Intelligence*. Springer.
- Xinsheng Wang, Mingqi Jiang, Ziyang Ma, Ziyu Zhang, Songxiang Liu, Linqin Li, Zheng Liang, Qixi Zheng, Rui Wang, Xiaoqin Feng, Weizhen Bian, Zhen Ye, Sitong Cheng, Ruibin Yuan, Zhixian Zhao, Xinfu Zhu, Jiahao Pan, Liumeng Xue, Pengcheng Zhu, and 6 others. 2025a. [Spark-tts: An efficient llm-based text-to-speech model with single-stream decoupled speech tokens](#). *CoRR*, abs/2503.01710.
- Zhongjian Wang, Peng Zhang, Jinwei Qi, Yuan Wang, Sheng Xu, and Bang Zhang. 2025b. [Omnitalker: One-shot real-time text-driven talking audio-video generation with multimodal style mimicking](#). In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*.
- Jonatas Wehrmann, Camila Kolling, and Rodrigo C Barros. 2020. Adaptive cross-modal embeddings for image-text alignment. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 12313–12320.
- Luwei Xiao, Xingjiao Wu, Shuwen Yang, Junjie Xu, Jie Zhou, and Liang He. 2023. Cross-modal fine-grained alignment and fusion network for multimodal aspect-based sentiment analysis. *Information Processing & Management*, 60(6):103508.
- Chuyun Xie, Wei Gao, and Ren Guo. 2024. Cross-modal learning for event-based semantic segmentation via attention soft alignment. *IEEE Robotics and Automation Letters*, 9(3):2359–2366.
- Zequan Xu, Wei Wang, Qinchuan Li, Jin Wang, and Gang Chen. 2026a. A novel method for composite facial expressions generation based on multimodal reinforcement learning. *Science China Technological Sciences*, 69(2):1220301.
- Zhuoxiong Xu, Xuanchen Li, Yuhao Cheng, Fei Xu, Yichao Yan, and Xiaokang Yang. 2026b. Singing-bot: An avatar-driven system for robotic face singing performance. *arXiv preprint arXiv:2601.02125*.
- Hao Yang, Yanyan Zhao, Tian Zheng, Hongbo Zhang, Bichen Wang, Di Wu, Xing Fu, Xuda Zhi, Yongbo Huang, and Hao He. 2026a. Text-driven emotionally continuous talking face generation. *arXiv preprint arXiv:2603.06071*.
- Yifan Yang, Zhi Cen, Sida Peng, Xiangwei Chen, Yifu Deng, Xinyu Zhu, Fan Jia, Xiaowei Zhou, and Hujun Bao. 2026b. Streamingtalker: Audio-driven 3d facial animation with autoregressive diffusion model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 11766–11774.
- Takahide Yoshida, Atsushi Masumori, and Takashi Ikegami. 2025. From text to motion: grounding gpt-4 in a humanoid robot “alter3”. *Frontiers in Robotics and AI*, 12:1581110.
- Wenxuan Zhang, Xiaodong Cun, Xuan Wang, Yong Zhang, Xi Shen, Yu Guo, Ying Shan, and Fei Wang. 2023. Sadtalker: Learning realistic 3d motion coefficients for stylized audio-driven single image talking face animation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8652–8661.
- Qingcheng Zhao, Yifang Pan, and Karan Singh. 2026. Toktalk: Expressive real-time facial animation from audio-llm tokens. *arXiv preprint arXiv:2605.31294*.

Yang Zhou, Xintong Han, Eli Shechtman, Jose Echevarria, Evangelos Kalogerakis, and Dingzeyu Li. 2020. [Makelttalk: speaker-aware talking-head animation](#). *ACM Trans. Graph.*, 39(6):221:1–221:15.

The supplementary material is organized as follows:

- **Appendix A:** Implementation Details of TTFG
- **Appendix B:** Details of our Humanoid Facial Robots
- **Appendix C:** Ethics Statement and Societal Impact

A Implementation Details of TTFG

A.1 User–LLM Experimental Setup

Experimental Protocol. To validate the subjective quality of the generated facial animations, we conducted a static evaluation involving both human participants and an automated LLM evaluator (Qwen3.5 Omni Plus).

Human Evaluation. 32 participants (16 male, 16 female, aged 22–35) with normal or corrected-to-normal vision were recruited. Each participant independently watched 24 video clips (8 for digital human videos, 8 for Robot A, and 8 for Robot B) and rated two dimensions on a 5-point Likert scale (0.5 increments allowed):

- **Lip-Sync Accuracy:** how well the mouth movements match the spoken audio.
- **Naturalness:** smoothness and realism of facial expressions, absence of jitter or deformation.

The same 5-point scale definition (as detailed below) was provided to both human raters and the LLM.

LLM Evaluation. The LLM (Qwen3.5 Omni Plus) was given the same video clips and the identical scoring rubric. For each utterance, the model generated five independent scores; the highest and lowest were discarded, and the average of the remaining three was used as the final LLM score.

Scoring Rubric (Prompt for LLM and Human Raters). The following prompt was used for both the LLM and human participants:

You are a professional lip-sync evaluator. First, listen to the speech content in the video and output the corresponding text (if unclear, reply “unrecognizable”). Then, based on your expertise, objectively evaluate the digital human talking-head video according to the criteria below, which tightly integrate audio and visual modalities.

Lip-sync Accuracy (0–5, 1 decimal):

- 4.2–5 (Excellent): Mouth shapes precisely match phonemes; consonants/vowels (e.g., /a/, /i/, /u/, /b/, /p/, /s/, /l/) are clearly distinguished; no perceptible delay or misalignment.
- 3.5–4.2 (Good): Generally synchronized; minor deviations on complex sounds (e.g., “th”, “v”) do not impair understanding.
- 2.5–3.5 (Fair): Speech is discernible but noticeable lag or lead in some phrases.
- 1.5–2.5 (Poor): Frequent desynchronization between mouth and audio.
- 0–1.5 (Very Poor): Almost no correspondence, like a mismatched dubbing.

Key aspects: vowel/consonant shaping, smooth transitions in multi-syllable words, rhythm consistency.

Facial Animation Naturalness (0–5, 1 decimal):

- 4.2–5 (Excellent): Extremely smooth facial motion, no stuttering, twitching, or model distortion; natural transitions.
- 3.5–4.2 (Good): Mostly smooth, occasional minor edge jitter or frame rate fluctuations, but not distracting.
- 2.5–3.5 (Fair): Noticeable jerkiness, popping, or geometric distortion, especially during fast movements.
- 1.5–2.5 (Poor): Frequent tearing, model twisting, or broken motion.
- 0–1.5 (Very Poor): Severe errors, unwatchable.

Key aspects: smooth jaw opening/closing, stability of mask edges, nostrils, eyebrows under motion, consistency across speaking rates.

Output format:

Recognized text: ...
Lip-sync Accuracy: X.Y
Naturalness: X.Y
Brief justification:
– Lip-sync: [pros/cons]
– Naturalness: [pros/cons]

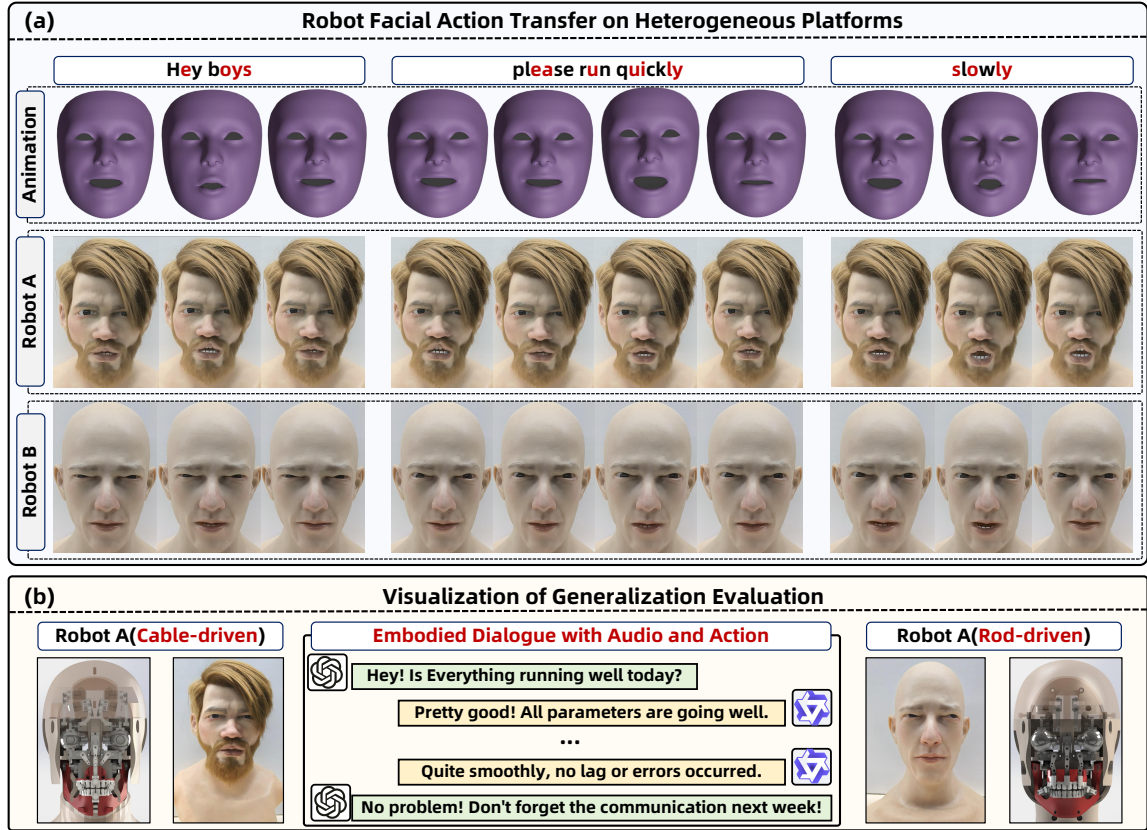


Figure 4: Visualization of (a)Facial Action Transfer onto Heterogeneous Humanoid Robot Platforms and (b)Generalization Evaluation.

A.2 Real-world Experiment Visualization

Static Experiment. The static experiment tests the EmbodiChat system using fixed sentences, and records the embodied dialogue results via animation rendering or video capture, as shown in Fig. 4(a).

Dynamic Experiment. The dynamic experiment involves two LLM-driven robots, Robot A and Robot B, engaging in embodied dialogue directly, as shown in Fig. 4(b).

A.3 Cross-lingual generalization of TTFG

Verification baseline. We select three representative TTS systems as baselines: GPT-Sovits, IndexTTS, and Qwen3-TTS. For each system, we evaluate the lip-sync accuracy of its officially supported synthesis languages using the TTFG framework. The evaluation metrics are LSE-D (Lip Sync Error Distance, lower is better) and LSE-C (Lip Sync Error Confidence, higher is better).

Verification details. The experiment is conducted strictly on the languages officially claimed by each TTS model. GPT-Sovits adopts distinct synthesis pipelines: the Chinese path relies on an additional BERT module, while the non-Chinese path does not need it. This architectural discrepancy makes a direct cross-lingual generalization from Chinese to other languages infeasible. Consequently, for GPT-Sovits we only report the non-Chinese languages it supports (English, Japanese, Korean) and mark Chinese as unavailable, because the two paths are not comparable under the same cross-lingual condition. IndexTTS provides official support exclusively for Chinese and English; the remaining languages are therefore marked as unsupported. Qwen3-TTS, in contrast, natively covers all ten languages listed. All results are summarized in Table 4.

Quantitative findings. Qwen3-TTS consistently delivers the best overall performance, achieving the lowest LSE-D and highest LSE-C across all tested languages, which indicates a strong cross-lingual generalization capability. For example, on Chinese, English, and Korean, Qwen3-TTS yields LSE-D scores of 12.221, 12.175, and 12.295, respectively, all accompanied by high LSE-C values. IndexTTS

Table 4: **Cross-lingual generalization of TTFG.** The officially supported languages of TTS are used for generalization experiments. GPT-Sovits uses different synthesis paths for Chinese and non-Chinese (Chinese path requires extra BERT), thus direct generalization is not feasible.

Language	GPT-Sovits		IndexTTS		Qwen3-TTS	
	LSE-D↓	LSE-C↑	LSE-D↓	LSE-C↑	LSE-D↓	LSE-C↑
Chinese	/	/	12.561	2.674	12.221	3.178
English	13.204	1.261	12.174	3.017	12.175	2.979
Japanese	13.673	1.122	/	/	12.416	2.937
Korean	13.872	0.523	/	/	12.295	3.333
German	/	/	/	/	12.466	3.354
French	/	/	/	/	12.514	3.341
Russian	/	/	/	/	12.476	3.017
Portuguese	/	/	/	/	12.381	2.914
Spanish	/	/	/	/	12.651	2.813
Italian	/	/	/	/	12.436	2.978

attains reasonable lip-sync quality on Chinese and English (LSE-D 12.561 and 12.174), yet it cannot serve the other eight languages. GPT-Sovits, although capable of generating non-Chinese speech, produces markedly higher LSE-D (e.g., 13.872 on Korean) and considerably lower LSE-C, revealing inferior lip-sync alignment for cross-lingual voices. These results demonstrate that the unified multilingual design of Qwen3-TTS effectively eliminates the generalization gap that exists in systems relying on language-specific branches or external linguistic features.

A.4 Physical Motion Accuracy and Global Stability Analysis

To further verify that our TTFG module maintains precise control and global stability, we conducted a quantitative geometric analysis on the BEAT validation set. This evaluation focuses on the fidelity of the mapping from TTS acoustic tokens to the 51-dimensional ARKit manifold.

Evaluation Protocol. During inference, EmbodiChat generates facial animation sequences at 25 FPS to ensure real-time responsiveness. Since the ground-truth motion capture data in the BEAT validation set is recorded at 50 FPS, we applied a factor-of-two downsampling to the validation sequences to ensure precise temporal alignment for frame-by-frame comparison.

To quantify the spatial deviation, we utilized a standardized 3D facial template consisting of **5,023 vertices**. Both the predicted and downsampled ground-truth ARKit coefficients were mapped onto this identical template via a fixed linear blendshape basis. We then calculated the Landmark Distance (LMD) as the average Euclidean distance between corresponding vertices in 3D space. Additionally, we directly computed the Mean Absolute Error (MAE) within the 51-dimensional ARKit coefficient space.

Results and Conclusion. As summarized in Table 5, EmbodiChat achieves an ARKit MAE of 0.018 and a sub-millimeter global LMD of 0.176 mm. Notably, the mouth-region LMD (0.677 mm) remains highly accurate despite the compact representation. These results demonstrate that our framework **maintains exceptional stability and precision across the entire face**. The extremely low global error confirms that our model does not introduce unwanted jitter or drift in non-speech facial regions, ensuring a coherent and realistic humanoid presence during open-domain dialogue.

Table 5: Quantitative geometric fidelity and stability results on the BEAT validation set. LMD is reported in millimeters (mm).

Metric	ARKit MAE ↓	Lower-face LMD (mm) ↓	Full-face LMD (mm) ↓
EmbodiChat (Ours)	0.018	0.677	0.176

A.5 Choice of Temporal Models for Text-Driven Talking Face Generation (TTFG)

Table 6: Comparison of Causal Transformer and LSTM on IndexTTS. The input is selected from datasets.

Model	LSE-D ↓	LSE-C ↑	FPS ↑
Video in Datasets	10.349	3.41	/
Causal Transformer	14.75	0.11	2650
LSTM	10.527	3.21	7500

We trained both LSTM-based and causal Transformer-based models to evaluate their suitability for text-driven talking face generation. While Transformers excel at capturing long-range dependencies through self-attention, our analysis revealed that the temporal dynamics in talking face generation exhibit relatively weak long-term dependencies—facial movements are primarily governed by local phonetic and prosodic cues within short time windows.

More critically, we observed that the global receptive field of the causal Transformer, although theoretically capable of modeling arbitrary-length sequences, introduces unintended interference during autoregressive generation. Specifically, as the sequence progresses, the attention mechanism tends to accumulate noise or spurious correlations from earlier timesteps, which progressively degrades the quality of facial animation in later frames. This manifests as unnatural mouth shapes, jittery lip motions, or even complete collapse of articulation toward the end of long utterances—a phenomenon rarely observed with LSTM-based models.

In contrast, LSTM-based models offer several practical advantages: they inherently enforce strict causal temporal modeling (i.e., predictions depend only on past and present inputs), avoid reliance on distant context that may be irrelevant or misleading, and maintain consistent generation quality throughout the entire sequence. Moreover, their sequential processing incurs minimal memory overhead and ensures low inference latency without the need for buffering global context, making them highly suitable for real-time or streaming deployment scenarios.

As evidenced in Table 6, the LSTM not only achieves significantly better lip-sync consistency (LSE-C: 3.21 vs. 0.11) and competitive lip-audio alignment (LSE-D: 10.527 vs. 14.75), but also delivers nearly three times higher inference throughput (7500 vs. 2650 FPS). Given these empirical results, architectural stability, and efficiency considerations, we selected the LSTM architecture as the backbone of our TTFG system for all experiments in this work.

A possible explanation for the Transformer’s collapse. We further conjecture that the extreme failure of the causal Transformer may stem not only from the accumulation of attention noise discussed above, but also from a specific interaction between the training dynamics and the nature of the BEAT dataset. ARKit blendshape sequences, unlike 3D vertex coordinates, provide no explicit geometric constraints, with only coefficient values range from 0 to 1 regressed under MSE loss. In such weakly regularized regression tasks, models are prone to converge to a local optimum where they simply output the dataset’s average facial expression, which already achieves a deceptively low loss for a large fraction of frames. We observed that the LSTM-based model, after initially exhibiting such “mean-face” behavior during training, gradually escaped from this local optimum and recovered responsiveness to the acoustic input. We speculate that the LSTM’s recurrent gating dynamics inherently inject small state perturbations that prevent the model from settling into a static output pattern. In contrast, the causal Transformer showed no such self-recovery: once its autoregressive prediction trajectory deviated toward a static average face, the exposure bias in teacher-forcing-free generation compounded this error, locking the model irreversibly into the local optimum. This hypothesis is consistent with the near-zero LSE-C (0.11) and high LSE-D (14.75) we observed for the Transformer, but warrants more targeted investigation in future work. Meanwhile, a side-by-side visual comparison of the generated talking-face animations is provided in the supplementary material (LSTM.mp4 and Transformer.mp4), where the progressive articulation collapse of the Transformer on long utterances is clearly observable.

A.6 Proof of the Boundedness of TFG

In this section, we provide the formal proof that the temporal error propagation within our lightweight LSTM-based Text-to-Face Generation (TFG) model is constrained, thereby bounding the final animation discrepancy Δ_{anim} .

Theorem 1. *Let Φ be an LSTM-based TFG generator with recurrent weight matrix W_h and input weight matrix W_x . Assume the activation functions (e.g., σ , $\tanh$) are 1-Lipschitz continuous. If the input residual between the raw semantic feature H_{audio} and its continuous proxy $\hat{h}$ is bounded by $\|H_{\text{audio}} - \hat{h}\| \leq \delta$, then the discrepancy between the resulting generated animations is strictly bounded by $\Delta_{\text{anim}} \leq C \cdot \delta$, where $C = \|W_x\| \sum_{i=0}^{T-1} \|W_h\|^i$.*

Proof. Let H_t and $\hat{h}_t$ denote the input feature at time step t for the continuous audio representation and its quantized proxy, respectively. According to the convergence threshold of the VQ-VAE optimization objective, the input residual at any time step $t \in \{1, 2, \dots, T\}$ satisfies:

$$\|H_t - \hat{h}_t\| \leq \delta. \quad (10)$$

During the generation process, the hidden state of the LSTM driven by the original feature H_t updates as $s_t = \sigma(W_h s_{t-1} + W_x H_t + b)$. Symmetrically, the hidden state driven by the proxy $\hat{h}_t$ updates as $\hat{s}_t = \sigma(W_h \hat{s}_{t-1} + W_x \hat{h}_t + b)$.

We denote the hidden state error at time step t as $e_t = \|s_t - \hat{s}_t\|$. Assuming the generation begins from the same initial state (i.e., $s_0 = \hat{s}_0 \implies e_0 = 0$), we calculate the distance e_t . Leveraging the 1-Lipschitz continuity of the activation function $\sigma(\cdot)$, which ensures $\|\sigma(x) - \sigma(y)\| \leq \|x - y\|$, we have:

$$\begin{aligned} e_t &= \|\sigma(W_h s_{t-1} + W_x H_t + b) - \sigma(W_h \hat{s}_{t-1} + W_x \hat{h}_t + b)\| \\ &\leq \|(W_h s_{t-1} + W_x H_t + b) - (W_h \hat{s}_{t-1} + W_x \hat{h}_t + b)\| \\ &= \|W_h(s_{t-1} - \hat{s}_{t-1}) + W_x(H_t - \hat{h}_t)\|. \end{aligned} \quad (11)$$

By applying the triangle inequality and the sub-multiplicative property of matrix norms, the error propagation at step t is bounded by:

$$e_t \leq \|W_h\| \|s_{t-1} - \hat{s}_{t-1}\| + \|W_x\| \|H_t - \hat{h}_t\| \leq \|W_h\| e_{t-1} + \|W_x\| \delta. \quad (12)$$

To find the accumulated error over the sequence, we recursively unroll this inequality from $t = 1$ up to the finite sequence length T :

$$\begin{aligned} e_1 &\leq \|W_x\| \delta, \\ e_2 &\leq \|W_h\| e_1 + \|W_x\| \delta \leq (\|W_h\| + 1) \|W_x\| \delta, \\ &\dots \\ e_T &\leq \left(\sum_{i=0}^{T-1} \|W_h\|^i \right) \|W_x\| \delta. \end{aligned} \quad (13)$$

In our TFG formulation, the animation output discrepancy Δ_{anim} is characterized by the maximum hidden state deviation over the sequence length T . Therefore, the total animation discrepancy is bounded as follows:

$$\Delta_{\text{anim}} = \|\Phi(H_{\text{audio}}) - \Phi(\hat{h})\| \leq e_T \leq \left(\|W_x\| \sum_{i=0}^{T-1} \|W_h\|^i \right) \delta. \quad (14)$$

By defining the architectural constant $C = \|W_x\| \sum_{i=0}^{T-1} \|W_h\|^i$, we obtain $\Delta_{\text{anim}} \leq C \cdot \delta$. Because the spectral norms $\|W_h\|$ and $\|W_x\|$ are structurally regularized and finite, C acts as a definitive upper bound multiplier. This establishes that the semantic alignment between $\hat{h}$ and H_{audio} is theoretically maintained throughout the recurrent manifold generation, completing the proof. ■

A.7 Theoretical Equivalence of Incremental and Full-Sequence Inference

We formally establish that our TTFG framework, which employs a unidirectional LSTM architecture, produces mathematically identical hidden and cell states under both incremental (autoregressive) and full-sequence (batched) inference modes. This equivalence is essential for ensuring that the real-time performance during deployment matches the offline evaluation results.

Formalism. Consider a unidirectional LSTM layer with parameters Θ . Let the input sequence be $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_T]$. For each timestep t , the recursive update is defined as:

$$\mathbf{i}_t = \sigma(W_{xi}\mathbf{x}_t + W_{hi}\mathbf{h}_{t-1} + b_i), \quad (15)$$

$$\mathbf{f}_t = \sigma(W_{xf}\mathbf{x}_t + W_{hf}\mathbf{h}_{t-1} + b_f), \quad (16)$$

$$\tilde{\mathbf{c}}_t = \tanh(W_{xc}\mathbf{x}_t + W_{hc}\mathbf{h}_{t-1} + b_c), \quad (17)$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \tilde{\mathbf{c}}_t, \quad (18)$$

$$\mathbf{o}_t = \sigma(W_{xo}\mathbf{x}_t + W_{ho}\mathbf{h}_{t-1} + b_o), \quad (19)$$

$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t), \quad (20)$$

where $\mathbf{h}_0 = \mathbf{0}, \mathbf{c}_0 = \mathbf{0}$. In *incremental inference*, the model processes $\mathbf{x}_t$ sequentially using cached states $(\mathbf{h}_{t-1}, \mathbf{c}_{t-1})$. In *full-sequence inference*, the recurrence is unrolled over the entire $\mathbf{X}$ within a single execution call.

Algorithmic Identity. Assuming the model is in a deterministic evaluation mode (i.e., stochastic components like Dropout are disabled) and that no temporal-dependent normalization (e.g., BatchNorm across the time dimension) is applied, the two modes implement the exact same recurrence relation. By induction on t , it follows that $\mathbf{h}_t^{(\text{inc})} = \mathbf{h}_t^{(\text{full})}$ and $\mathbf{c}_t^{(\text{inc})} = \mathbf{c}_t^{(\text{full})}$ for all $t \in \{1, \dots, T\}$.

Numerical Implementation. In practical deployment, minor discrepancies may arise due to the finite precision of IEEE 754 floating-point arithmetic. Specifically, full-sequence inference often utilizes highly optimized, fused CUDA kernels (e.g., cuDNN), while incremental inference typically relies on iterative calls to individual cell operators. Although mathematically equivalent, the differing order of floating-point operations can lead to numerical drift. Empirically, we observe that the L_2 norm of the difference between states remains within $[10^{-7}, 10^{-6}]$ on our RTX 4060 Ti platform, confirming that the two modes are numerically equivalent for all practical TTFG applications.

A.8 Our framework versus Traditional Audio-driven and Text-driven Paradigms

Conventional talking-face generation pipelines typically fall into two paradigms: the **audio-driven cascade paradigm** and the **text-driven unified paradigm**. In the audio-driven cascade (Fig. 5(a)), a text-to-speech (TTS) engine first synthesizes the full audio waveform, which is then fed into a subsequent talking-face generator (TFG). This cascaded design inevitably suffers from cumulative inference latency and pipeline bubbles, as the TFG module must wait for the complete upstream audio. Furthermore, any error in the TTS output directly propagates into the facial animation stage, creating a brittle cascade of errors. The text-driven paradigm (Fig. 5(b)) eliminates the cascade dependency by directly mapping text to animation, thus removing the need for a pre-generated audio waveform. However, this comes at the cost of significantly increased training complexity: to synchronize the two modalities, it typically requires dedicated cross-modal temporal supervision (e.g., explicit frame-level alignment losses) and carefully designed adapters to bridge the text and visual domains, making the architecture and training pipeline substantially more complicated.

Essentially, the shift from an audio-driven to a text-driven approach transforms the problem of modality coupling from the inference stage into the training stage. Historically, this has led to the belief that the two paradigms are incompatible and that one must trade off simplicity for end-to-end generation. Our **TTFG framework** (Fig. 5(c)) challenges this view by acting as a fundamental *unification* of both paradigms. It inherits the best of both worlds: on the training side, it retains the simplicity of the audio-driven scheme, obviating the need for hard cross-modal alignment supervision or extra adapters; on the inference side, it

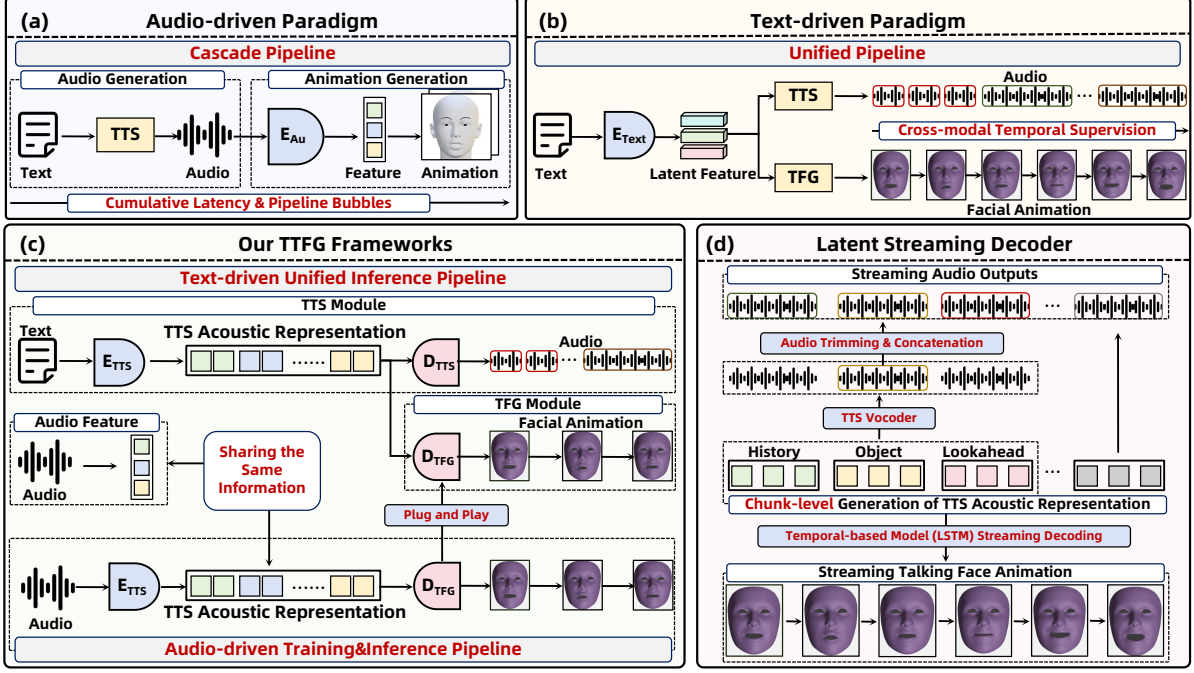


Figure 5: **Our framework versus traditional audio-driven and text-driven paradigms.**(a) Audio-driven cascade pipeline. (b) Text-driven unified pipeline. (c) Our TTFG frameworks with text-driven unified inference and audio-driven training/inference pipelines. (d) Latent streaming decoder for chunk-level audio and facial animation generation.

enjoys the text-driven advantage of directly generating audio and animation without relying on a complete pre-synthesized audio input.

Beyond this unification, our framework supports a highly efficient **latent streaming decoder** (Fig. 5(d)). By performing chunk-level generation of TTS acoustic representations and a temporally autoregressive streaming decoding of facial animation, it drastically reduces the system response latency and fully eliminates pipeline bubbles. Crucially, owing to the incremental nature of latent-space streaming and the temporal dependency modeling in our TFG module, the streaming decoder achieves facial animation quality on par with the full-input (non-streaming) setting, while completely removing the overhead of a dedicated audio feature extractor. Moreover, because audio generation and video generation are thoroughly decoupled in our design, streaming audio can exploit a lookahead window and history context to support finer-grained chunk-level decoding without degrading the facial animation quality. This stands in sharp contrast to conventional streaming audio-driven methods, where the chunk size is severely constrained by the audio decoder; overly small chunks often cause extracted audio features to drop frames and severely harm animation coherence.

A.9 Training and Inference Details of TTFG

To ensure rigorous reproducibility, we detail the data processing, training protocol, and streaming inference mechanism of the Text-Driven Talking Face Generation (TTFG) module.

Data Preparation and Acoustic Projection. The training corpus is constructed by extracting paired audio and 3D facial animation sequences from the BEAT dataset (Liu et al., 2022) in **Licensed under the Non-commercial license**. To align the temporal resolution of the kinematics with the acoustic tokens, the raw ARKit blendshape animations are downsampled from 60 fps to 50 fps. Furthermore, the dataset is systematically categorized based on speaker identity and emotion labels. To extract the driving features, we instantiate the TTS acoustic representation space using GPT-SoVITS (RVC-Boss, 2026). The raw audio is processed through its native reference audio encoding pathway: it sequentially passes through the pretrained audio encoder, the VQ-VAE encoder, and the VQ-VAE quantization decoder. The resulting continuous feature vector represents the precise projection of the audio within the TTS acoustic

representation space. These features, paired with their corresponding ARKit sequences, form the dataset for TFG training.

Model Architecture and Conditioning. As demonstrated in Appendix A.1, we employ an LSTM-based architecture as the TTFG backbone due to its strict causal temporal modeling capability and superior efficiency over causal Transformers. To capture personalized and affective dynamics, speaker identity and emotion labels are encoded as one-hot vectors and concatenated with the acoustic representations. Consequently, the model naturally supports explicit control over the speaking style and emotional intensity of the generated ARKit sequences via one-hot manipulation. However, as the core contribution of this work lies in achieving extreme low-latency and real-time embodied interaction rather than advancing affective computing, we do not extensively elaborate on these emotional control mechanisms in the main evaluations.

Training Objective and Hardware Specification. During the training phase, the TTFG module learns to map the acoustic latent space to the facial articulation manifold. To guarantee both spatial precision and temporal coherence, we apply an interleaved supervision strategy. Specifically, as formulated in Eq.(4) and Eq.(5), we enforce a reconstruction loss $\mathcal{L}_{recon}$ exclusively on even-numbered frames to capture positional accuracy, while imposing a smoothing loss $\mathcal{L}_{smooth}$ on odd-numbered frames to suppress high-frequency jitter. The network is optimized using the AdamW optimizer for 600 epochs. **Both the training and subsequent inference processes of our models are conducted on a single consumer-grade NVIDIA RTX 4060 Ti GPU (16GB VRAM).**

Streaming Inference and Unified Latent Flow. During deployment, the TTFG module functions as a zero-shot, plug-and-play extension seamlessly integrated into the GPT-SoVITS pipeline, requiring absolutely no additional fine-tuning or structural modification to the backbone TTS model. By operating directly on the VQ-VAE latent space, the module bypasses the vocoder synthesis stage completely. This structural design decouples facial motion generation from raw audio reconstruction, thereby eliminating vulnerability to cascading errors such as streaming concatenation artifacts or acoustic noise that traditionally interfere with downstream TFG results. In practice, we implement a synchronous latent-to-motion mechanism: as the TTS autoregressive transformer emits acoustic tokens, they are immediately fed into our LSTM-based TTFG in fixed chunks of 20 tokens. This configuration enables the parallel prediction of ARKit blendshape coefficients alongside the neural vocoder’s waveform synthesis. By processing the shared acoustic manifold directly, the module ensures that facial articulation and vocal prosody are derived from the exact same semantic source, maintaining precise temporal synchronization throughout the streaming process without the need for external alignment interventions.

B Details of our Humanoid Facial Robots

B.1 Mechanical Structure of Robot A and Robot B

Robot A and Robot B employ distinct actuation mechanisms to achieve facial expressions, as illustrated in Fig. 6. Robot A, depicted in Fig. 6(a), utilizes a cable-driven system. This design employs tendons or cables routed through the robotic structure to transmit forces from remote actuators to the facial features, enabling movement. In contrast, Robot B, shown in Fig. 6(b), adopts a rod-driven mechanism. This approach uses rigid or semi-rigid rods directly connected to actuators to push or pull specific points on the robot’s face, providing an alternative method for actuation. The mechanical designs lead to differences in their degrees of freedom (DoF). The cable-driven Robot A possesses a total of 21 DoF, distributed as follows: 9 DoF for the mouth, 8 DoF for the eyes, and 4 DoF for the eyebrows. Conversely, the rod-driven Robot B has a total of 16 DoF, with 6 DoF allocated to the mouth, 8 DoF for the eyes, and 2 DoF for the eyebrows. This indicates that while both robots offer significant expressiveness, particularly around the eyes, the cable-driven system of Robot A allows for greater articulation, especially in the mouth and eyebrow regions, compared to the rod-driven system of Robot B.

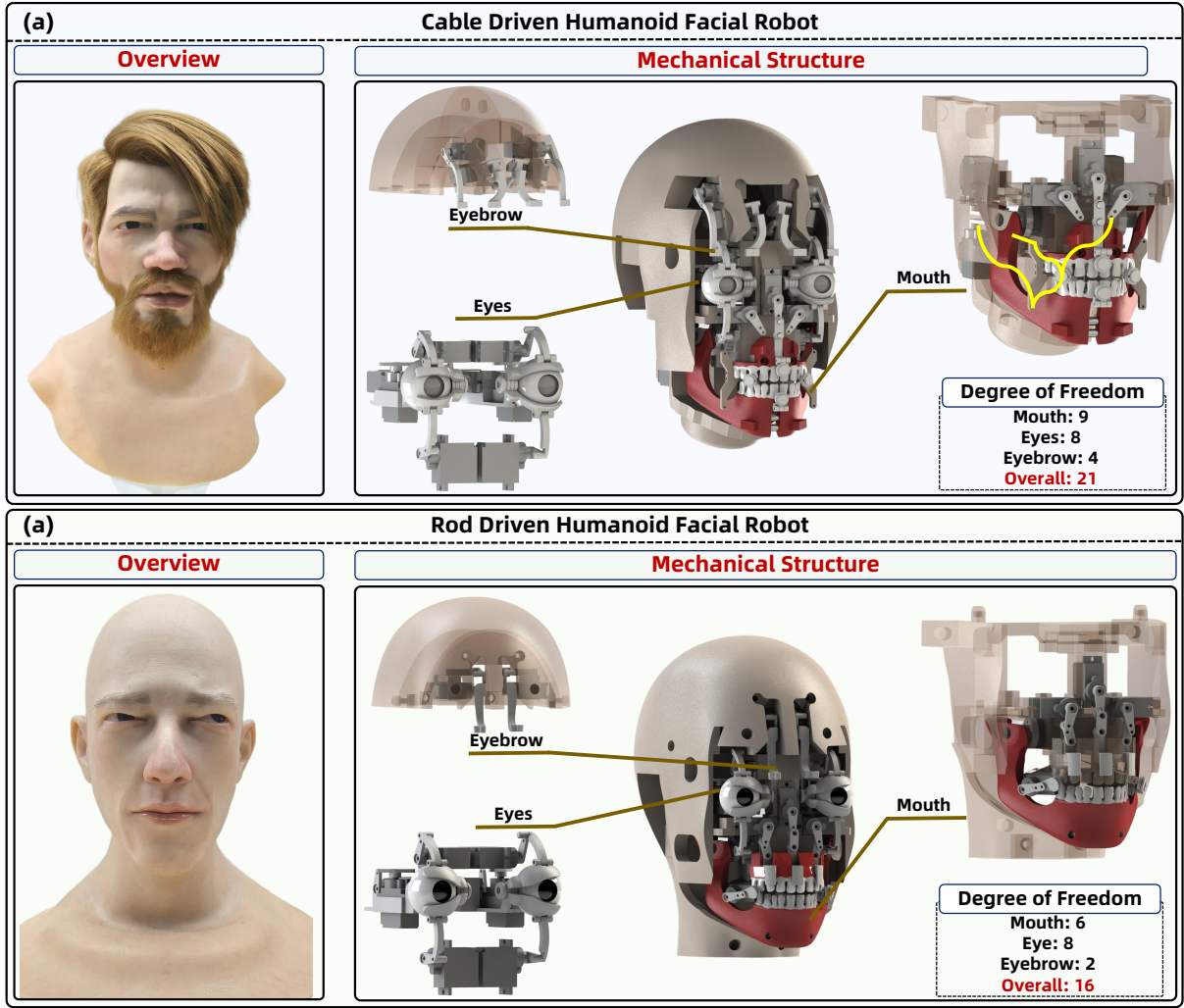


Figure 6: Mechanical structure of (a) Cable-Driven and (b) Rod-Driven Humanoid Facial Robots, along with their respective Degrees of Freedom (DoF) distributions.

B.2 Universal Facial Action Compiler on Humanoid Facial Robot

The Universal Facial Action Compiler (UFAC) translates abstract facial action descriptors into motor commands through two cascaded stages: *Muscle Fusion*, which compresses high-dimensional ARKit blendshapes into a compact set of robot-agnostic muscle groups, and *Motor Calibration with Adapter Reinforcement Learning*, which converts a desired muscle activation into hardware-specific motor trajectories. Figure 7 gives an architectural overview.

Muscle Fusion. Let $\mathbf{b} \in \mathbb{R}^{51}$ be the raw ARKit blendshape vector. We define n muscle groups $\mathbf{f} = [f_1, \dots, f_n]^\top$ that correspond to independent degrees of freedom of the robot mouth. Each muscle group is obtained as a linear combination of the primitive blendshapes (Fig. 7a):

$$f_i = \sum_{j=1}^k w_{ij} b_j, \quad i = 1, \dots, n, \quad (21)$$

with learned fusion weights w_{ij} (red boxes in the figure). The resulting vector $\mathbf{f} \in [0, 1]^n$ constitutes a hardware-agnostic, semantic representation of the facial expression. After fusion, the resulting fused ARKit values are approximately decoupled from one another, since the coupled components have been consolidated into each fused ARKit value itself.

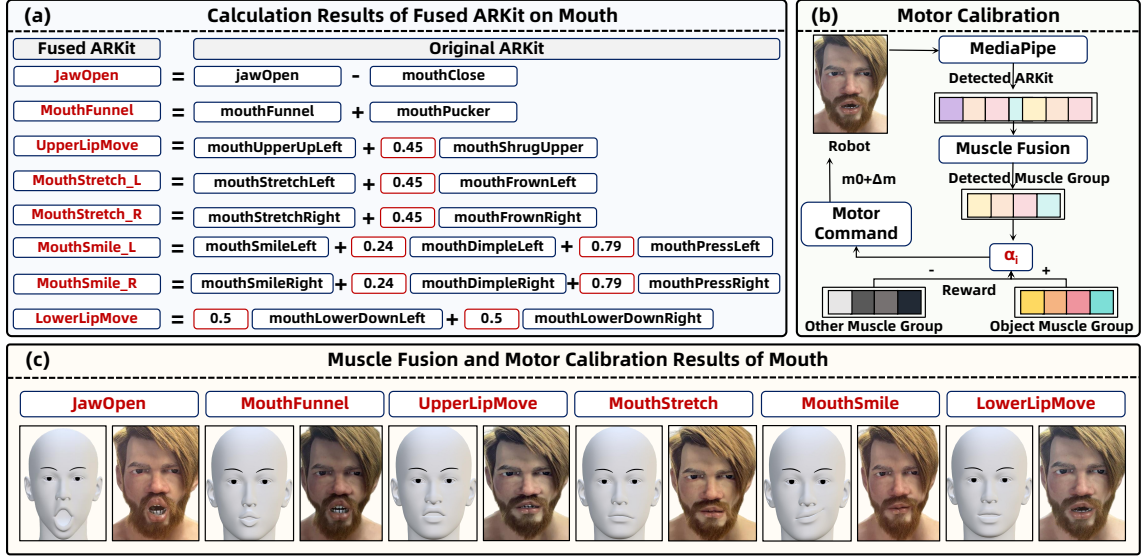


Figure 7: Universal Facial Action Compiler (UFAC) architecture and validation. (a) Fused ARKit calculation: linear combinations of primitive ARKit blendshapes to define semantically meaningful mouth actions; red boxes denote learned fusion weights. (b) Motor calibration loop: robot executes motor command $\mathbf{m}_0 + \Delta \mathbf{m}$; MediaPipe detects facial pose $\rightarrow$ Detected ARKit $\rightarrow$ Muscle Fusion $\rightarrow$ Detected Muscle Group; a trainable scaling vector α modulates the Object Muscle Group, and reward from muscle-space discrepancy drives RL optimization of α . (c) Qualitative results: target expressions (left, white head model) vs. realized expressions on the physical robot (right) for six canonical mouth actions—demonstrating hardware-agnostic fidelity.

Motor Calibration and Adapter Reinforcement Learning. Let $\mathbf{m}_0 \in \mathbb{R}^n$ denote the neutral motor position. For each muscle group i , we wish to find a motor offset vector $\Delta \mathbf{m}_i \in \mathbb{R}^n$ that, when added to $\mathbf{m}_0$, maximally activates the corresponding physical degree of freedom. Because the muscle groups are designed to be mutually independent, the calibration is performed channel by channel.

- **Single-channel exploration with crosstalk suppression.** For a given channel i , the robot executes motor command $\mathbf{m} = \mathbf{m}_0 + \Delta \mathbf{m}$, where $\Delta \mathbf{m}$ is the exploration offset. A consumer camera captures the resulting facial appearance and MediaPipe extracts an ARKit blendshape vector, which is immediately passed through the muscle fusion layer to obtain the detected values $f_k^{\text{det}} \in [0, 1]$ of all fused channels $k = 1, \dots, n$.
- **Reward definition.** The objective is not only to drive f_i^{det} as close to 1 as possible, but also to keep the other channels as close to 0 as possible, thereby decoupling the actuation. We therefore define a scalar reward

$$r = f_i^{\text{det}} - \lambda \sum_{j \neq i} f_j^{\text{det}}, \quad (22)$$

where the hyperparameter $\lambda > 0$ balances expressiveness against crosstalk penalty.

- **Reinforcement learning optimization.** Proximal Policy Optimization (PPO) is employed to search for the optimal offset $\Delta \mathbf{m}_i^*$ in the continuous motor space under the crosstalk-aware reward. The policy outputs a direction and step for updating $\Delta \mathbf{m}$, and the episode terminates when the reward plateaus. Because the action space is low-dimensional and the reward signal remains smooth despite the multi-channel objective, PPO reliably discovers a high-reward solution within a few hundred episodes. For applications where high precision is not critical, a human-in-the-loop manual tuning can be used as a simpler alternative, during which the operator intuitively adjusts the offset while watching both the target expression and unwanted movements on other regions.
- **Calibration time and automation.** The PPO-based calibration runs fully automatically without human intervention. For a robot with n independent muscle groups (typically $n \leq 8$ for the mouth after muscle

fusion), the entire process completes within 1–2 hours on standard hardware. The calibrated parameters $(\Delta \mathbf{m}_i^*, \alpha_i)$ depend only on the robot’s specific hardware; they remain valid across power cycles and require no recalibration as long as the mechanical structure is unchanged. When sub-millimeter precision is not necessary, we provide a manual tuning alternative: a human operator directly adjusts the scaling factor α_i for each of the 8 fused channels while observing the facial expression in real time, reducing the calibration overhead to less than 10 minutes at the cost of slightly lower expression fidelity.

- **Adapter coefficient extraction.** After convergence, we record the optimal motor offset $\Delta \mathbf{m}_i^*$ and the corresponding steady-state detected value $c_i^{\text{real}} \triangleq f_i^{\text{det}}|_{\Delta \mathbf{m}_i^*}$. The scaling factor α_i is then defined as

$$\alpha_i = \frac{1}{c_i^{\text{real}}}, \quad (23)$$

so that, for any desired activation $c_i \in [0, 1]$, the commanded offset $\alpha_i c_i \Delta \mathbf{m}_i^*$ produces a detected value of approximately $\alpha_i c_i \cdot c_i^{\text{real}} = c_i$.

Repeating the procedure for all n channels yields the motion matrix $\mathbf{M} = [\Delta \mathbf{m}_1^*, \dots, \Delta \mathbf{m}_n^*] \in \mathbb{R}^{n \times n}$ and the adapter vector $\boldsymbol{\alpha} = [\alpha_1, \dots, \alpha_n]^\top$. At runtime, given a target muscle vector $\mathbf{c} = [c_1, \dots, c_n]^\top$ (which is the fused ARKit vector $\mathbf{f}_{\text{target}}$ from the simulator), the final motor offset is a weighted linear superposition:

$$\Delta \mathbf{m}_{\text{final}} = \sum_{i=1}^n \alpha_i c_i \Delta \mathbf{m}_i^* = \mathbf{M} \text{diag}(\boldsymbol{\alpha}) \mathbf{c}, \quad (24)$$

and the motor command becomes $\mathbf{m}_{\text{target}} = \mathbf{m}_0 + \Delta \mathbf{m}_{\text{final}}$. This formulation is exactly the one given in the main text with $\mathbf{C} = \text{diag}(c_1, \dots, c_n)$ and $\mathbf{A} = [\alpha_1, \dots, \alpha_n]$. Note that the adapter vector $\boldsymbol{\alpha}$ is neither manually tuned nor learned through a discrepancy-minimizing global RL; it emerges automatically from the per-channel exploration rewards, guaranteeing that the linear superposition respects the physical scaling of each independent actuation.

Qualitative Validation. Figure 7c compares the target expressions rendered on a white head model with the actual expressions produced by a physical humanoid robot for six canonical mouth actions. The high visual fidelity demonstrates that UFAC, with the muscle fusion weights and the scaled motor bases obtained through the calibration procedure above, faithfully transfers text-driven expressions to physical hardware without requiring explicit mechanical modelling.

B.3 Other Details of Our Robots

Beyond the basic expressive movements, our robotic head features precise control over fine facial details. As shown in the top row of Fig. 8, the system supports independent and coordinated control of eye blinking, multi-directional eye movement, and eyebrow motion, allowing the robot to produce natural, human-like facial expressions.

A key component for learning-based control is the intraocular camera embedded within each eye module (bottom-left of Fig. 8). This camera captures the robot’s egocentric visual field in real time. To enable reinforcement learning for self-perception and motor coordination, we employ a mirror-based interaction setup (bottom-right). In this setup, the robot observes its own facial actions through the intraocular camera while facing a mirror, effectively seeing the visual consequences of its blinking, eye movements, and eyebrow gestures. The mirror provides a controlled yet natural environment for the robot to learn a self-model via trial and error, without requiring external motion-capture systems. The visual feedback from the mirror, combined with reward signals based on desired facial configurations, allows the robot to iteratively improve its control policies through RL.

C Ethics Statement and Societal Impact

The development of real-time, high-fidelity human-robot interaction systems presents both significant societal benefits and potential ethical and psychological risks that warrant careful consideration.

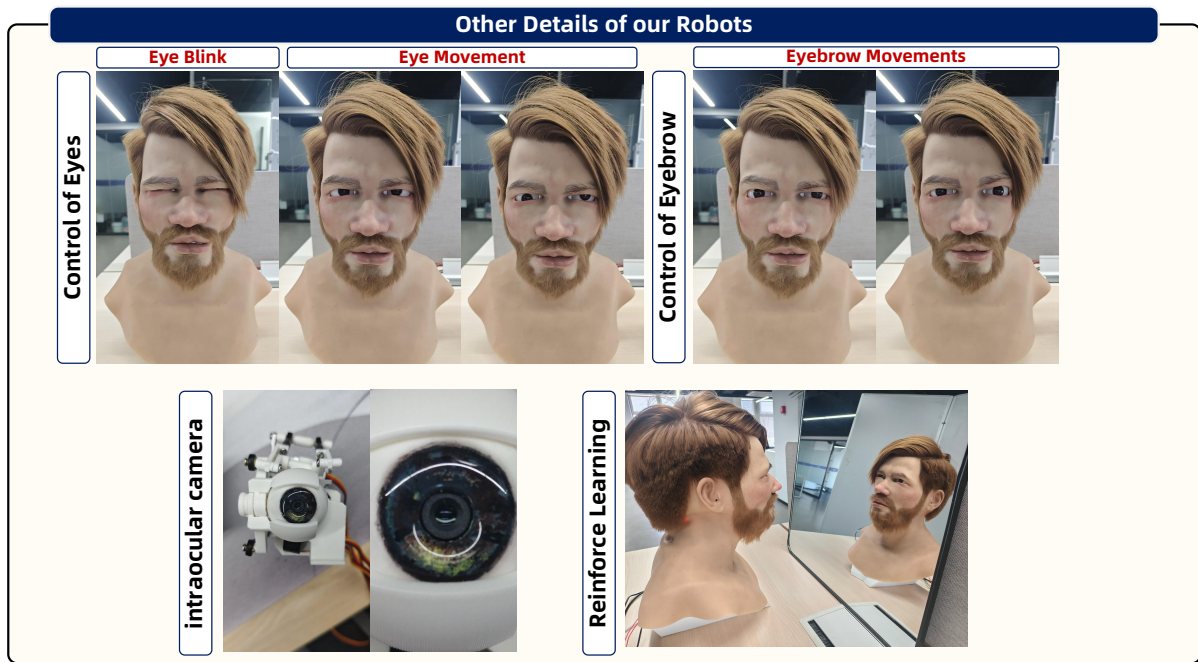


Figure 8: Additional details of the human-like robotic head. The system supports fine-grained facial control, including (top-left) eye blinking, (top-middle) multi-directional eye movement, and (top-right) eyebrow motion, enabling natural and expressive facial behaviors. (Bottom-left) The eye module is equipped with an intraocular camera for visual perception. (Bottom-right) The reinforcement learning setup, where the robot interacts with a mirror to learn self-perception and motion coordination policies.

C.1 Data Privacy and Offensive Content.

In the human-robot interaction experiments conducted to evaluate EmbodiChat, the collected data consist solely of anonymized user ratings and interaction logs. No personally identifying information (PII) — such as names, email addresses, IP addresses, or video recordings — was collected. Each participant was assigned a unique numeric ID. Prior to analysis, all text-based user inputs and rating content were manually reviewed for potentially offensive material (e.g., hate speech, harassment, sexually explicit content). No such content was observed in the collected dataset. All data are stored on a password-protected institutional server, accessible only to the research team, and will be deleted upon project completion.

License Information for Third-Party Artifacts. We use the following third-party artifacts in this work. Their licenses are explicitly acknowledged below:

- **BEAT (PantoMatrix/BEAT):** The BEAT dataset and code are released under a **non-commercial license**. Use is permitted only for academic or research purposes that do not involve commercial activities.
- **GPT-SoVITS:** The GPT-SoVITS repository is released under the **MIT License**, which allows free use, modification, and redistribution, provided that the original copyright notice and license terms are retained.

All usage in this paper complies with the respective license terms.

C.2 Participant Recruitment and Compensation

Volunteers were recruited by the authors through private campus channels (e.g., personal contacts, word of mouth, and campus social networks). All participants took part in the study on a voluntary basis without financial compensation. Before participating, each volunteer was informed of the task objectives, the estimated time required, and their right to withdraw at any time without penalty. Given the voluntary

nature of participation, no payment was provided, and thus a discussion of payment adequacy is not applicable.

C.3 Societal Impact

Positive Impacts. EmbodiChat fundamentally lowers the barrier for natural human-robot dialogue. In healthcare and eldercare, responsive humanoid robots can provide accessible companionship and cognitive assistance. In education, expressive embodied agents can serve as interactive tutors, providing consistent and engaging feedback for learners, particularly in special education scenarios where reliable and predictable facial cues are beneficial.

Negative Impacts and Mitigations: The Uncanny Valley Effect. While pushing the boundaries of realism, highly synchronized and expressive humanoid robots inherently risk falling into the "Uncanny Valley." As EmbodiChat enables near-human conversational fluency and facial articulation, any residual mechanical imperfections—such as slight motor latency, rigid actuator dynamics, or the absence of micro-saccades—can elicit acute psychological discomfort, eeriness, or distrust in human users. This cognitive dissonance could alienate the very demographics (e.g., the elderly or children in special education) these embodied agents are designed to assist.

To mitigate these psychological risks, we advocate for: 1) **Stylized Embodiment:** Deploying the EmbodiChat framework on stylized or appropriately cartoonish robotic shells rather than attempting hyper-realistic human replication, thereby managing user expectations and bypassing the uncanny valley; 2) **Transparent Artificiality:** Incorporating explicit design cues (e.g., visible mechanical joints or distinct synthetic vocal markers) that continuously signal the agent's artificial nature to prevent cognitive deception; and 3) **Human-Centric Evaluation:** Mandating comprehensive psychological user studies alongside technical benchmarks prior to the widespread deployment of bionic interactive systems.