HIDDENFACE: STREAMING CONVERSATIONAL FACIAL MOTION FROM HIDDEN STATES OF A LARGE AUDIO-LANGUAGE MODEL

Anonymous authors
Paper under double-blind review

ABSTRACT

Full-duplex large audio-language models (LALMs) enable voice agents to take part in conversations, but embodying these agents as avatars also requires facial motion that stays aligned with their speech and continues through listening and turn transitions. However, audio-driven methods must reprocess the generated speech in a cascade, incurring processing and synchronization overhead. Methods that drive facial motion from LALM internal states mitigate this, but they train the LALM jointly, rely on additional generated speech units, or delay the output to refine motion with later hidden states. We propose HiddenFace, which generates conversational facial motion using only the hidden states already computed by a frozen LALM (Voila) during speech generation. HiddenFace combines Voila's audio and text hidden states with its own motion-token history to autoregressively predict the next facial motion token, finalizing each 160 ms block as soon as its hidden states are available, without waiting for future speech. Under matched 160 ms block-causal conditions, HiddenFace matches or exceeds the lip accuracy of audio-driven baselines on held-out real dyadic conversations and adds only a small, constant per-step cost to speech generation. At the dialogue level, it is closer to real human facial behavior than any baseline, even those using the full audio, and receives the highest mean ratings in a human study.

1 INTRODUCTION

Recent advances in large audio-language models (LALMs) are changing spoken dialogue systems. Traditional systems connect automatic speech recognition, a language model, and text-to-speech synthesis in a cascade, whereas LALMs model audio directly within a language modeling framework to generate both speech and text (Zhang et al., 2023; Nguyen et al., 2025). Models such as Moshi, Voila, and PersonaPlex extend this paradigm to full-duplex interaction, continuously processing incoming user speech while generating responses and supporting backchanneling, interruptions, and turn-taking (Défossez et al., 2024; Shi et al., 2025; Roy et al., 2026). These models allow voice agents to join conversations as they unfold, instead of replying only after each user turn.

Extending these voice interactions to avatars requires more than speech generation. Facial motion is an output modality that shapes an agent's conversational behavior together with speech, rather than a visual effect added after speech has been generated (Cassell et al., 2000). Conversational facial motion generation therefore extends beyond utterance-level co-speech animation: facial motion must align with the agent's own speech while speaking and remain continuous while listening and through turn transitions (Chu et al., 2026).

Audio-driven facial motion methods condition on available waveforms or acoustic features, providing direct acoustic cues for speech-motion alignment (Fan et al., 2022; Xing et al., 2023). When integrated with an LALM, however, they form a cascade that reprocesses generated speech through an audio encoder and a motion model. Full-utterance methods must wait until the complete utterance is available, while streaming methods (Zhen et al., 2025) still incur additional processing and synchronization overhead for each audio chunk.

To reduce this overhead, recent methods connect facial motion generation directly to LALM representations, but jointly train the LALM and motion branch (Hu et al., 2026; Jiang et al., 2026),

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

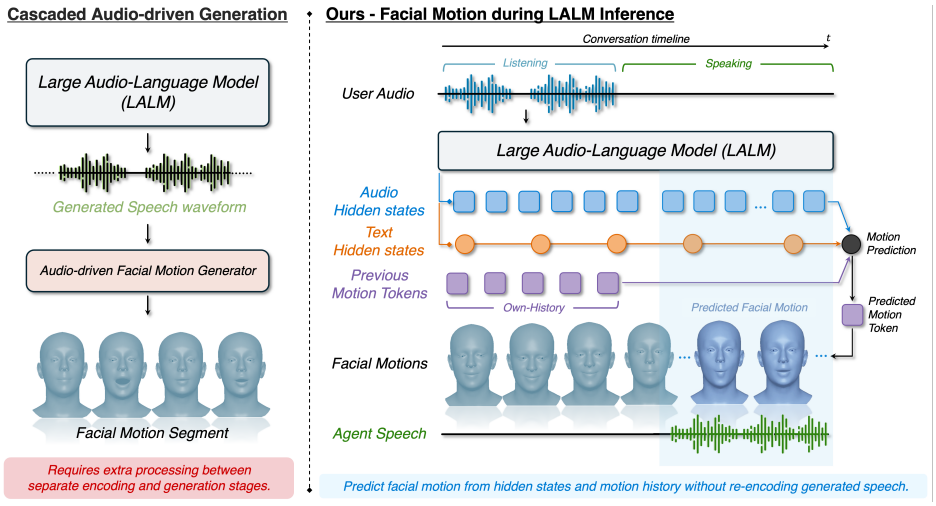

Figure 1: Cascade vs. HiddenFace. HiddenFace generates facial motion from LALM hidden states and past motion tokens without reprocessing generated speech.

additionally condition on generated speech units (Zhang et al., 2026), or delay output to refine motion using subsequent hidden states (Lim et al., 2026). Can the hidden states already computed by a frozen LALM support speech-aligned conversational facial motion without reprocessing generated speech or waiting for context beyond the current 160 ms block? To address this question, we propose HiddenFace (Figure 1), which combines the audio and text hidden states computed when Voila predicts audio rows and text tokens, respectively, with its own previously generated motion tokens to autoregressively predict the next facial motion token.

We evaluate HiddenFace against four audio-driven baselines on held-out real conversations (Chu et al., 2026; 2025; Sun et al., 2024; Peng et al., 2025). In matched 160 ms block-causal comparisons with three of these baselines, HiddenFace outperforms the strongest baseline, UniLS, on most lip-accuracy metrics and the other two baselines on all reported facial-quality metrics. It also outperforms full-audio baselines on dialogue-level metrics and receives the highest mean human ratings for lip synchronization, listening behavior, and overall naturalness. These results come with a low per-step computational cost that is independent of conversation length. Our contributions are as follows.

- We introduce a facial motion interface that uses hidden states already computed by a frozen LALM to generate conversational facial motion in 160 ms blocks, without reprocessing generated speech or accessing context beyond the current block.
- We design an autoregressive predictor that aligns and combines asynchronously updated audio and text hidden states with motion memory from the previous segment. Auxiliary objectives for motion position, velocity, and upper-face amplitude complement token-level cross-entropy.
- We evaluate facial motion quality and latency under released inference and matched block-causal settings, complemented by a human study and ablations of input representations, motion memory, and training objectives.

2 Related Work

Audio-driven facial motion generation. Following VOCA (Cudeiro et al., 2019), subsequent methods have combined contextual speech representations with autoregressive decoding (Fan et al., 2022) or used learned latent motion spaces to encourage realistic facial motion (Richard et al., 2021; Xing et al., 2023; Chu et al., 2025). Diffusion models conditioned on speaking styles extracted from reference motion further improve expressive diversity (Sun et al., 2024). Work on listening and turn-taking has progressed from listener motion generation (Ng et al., 2022) to modeling alternating speaking and listening behaviors using both speakers' audio and partner motion (Peng et al., 2025). UniLS (Chu et al., 2026) combines a motion prior trained without audio with dual-track audio conditioning to maintain facial dynamics even when the avatar is listening without speaking. These

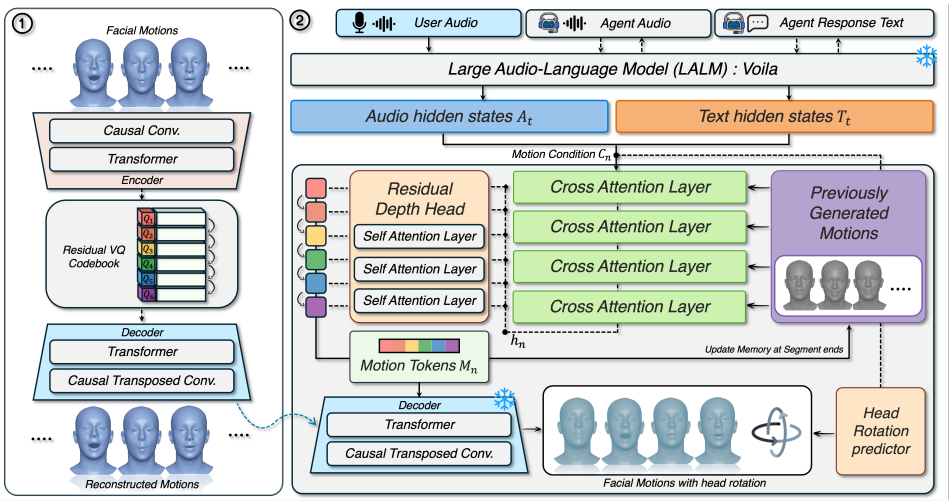

Figure 2: HiddenFace architecture. ① A pretrained, frozen motion tokenizer encodes each 160 ms block into six residual codes; its decoder is used in ② (Section 3.2). ② The predictor forms motion conditions only from a frozen LALM's audio and text hidden states and predicts motion tokens using its own motion from the previous segment (Section 3.3).

methods, however, require speech waveforms or features as input, so deploying them in LALM-based avatars entails reprocessing generated speech.

Motion generation from LALM representations. Full-duplex LALMs continuously process user and agent speech streams within a shared causal context (Défossez et al., 2024; Veluri et al., 2024; Zhang et al., 2025; Shi et al., 2025; Roy et al., 2026). Their generation process exposes hidden states computed before token prediction and the resulting tokens. Recent facial motion methods differ in backbone adaptation, conditioning inputs, and temporal context. A^2-LLM (Hu et al., 2026) generates facial tokens every 200 ms from audio-generation hidden states and previous motion while fine-tuning the backbone. Ex-Omni (Zhang et al., 2026) combines generated speech units as a temporal scaffold with hidden states to predict the full motion sequence. Moshi-Face (Jiang et al., 2026) combines hidden states with generated text and audio token embeddings to predict face tokens, which a non-causal face codec decodes. Both methods also fine-tune the backbone. FacePlex (Lim et al., 2026) delays speech and motion by 240 ms to refine motion using subsequent hidden states.

Related approaches also generate full-body motion. DyaPlex (Nagano et al., 2026) combines per-layer hidden states from a frozen PersonaPlex with partner motion, while MIBURI (Mughal et al., 2026) uses embeddings of speech and text tokens generated by Moshi. HiddenFace keeps Voila frozen and conditions only on audio and text hidden states and its own previous motion tokens, without partner motion or generated speech/text tokens as additional inputs. It commits motion in 160 ms blocks without waiting for hidden states beyond the current block.

3 Method

We describe how HiddenFace generates facial motion from the hidden states computed by a frozen Voila model during speech generation (Figure 2).

3.1 Problem Formulation

At 25 Hz, the agent's facial motion $y_t \in \mathbb{R}^{105}$ comprises FLAME (Li et al., 2017) expression (100) parameters and reduced jaw (1) and eye (4) pose parameters. A separate predictor generates head rotation (Section 3.3). Voila tokenizes and embeds the user and agent audio streams, averages the embeddings for backbone input, and interleaves response text tokens with audio to generate speech (Shi et al., 2025). We call each 40 ms group of audio tokens an *audio row*; audio rows and motion frames share the temporal index t.

We group four consecutive audio rows into a 160 ms *audio-state block*. For each block n, a single motion token M_n represents facial motion at frames $4n$ through $4n + 3$ (Section 3.2). HiddenFace receives only audio hidden states A_t and text hidden states T_t from Voila; waveforms, audio tokens, and transcripts are not inputs to the motion predictor. The predictor partitions motion tokens into fixed-length segments and uses only its own tokens from the previous segment, $\mathcal{M}_{b-1}$, as memory (Section 3.3). It models

$$M_n \sim p_\theta(M_n \mid A_{4n:4n+3},\, T_{4n:4n+3},\, \mathcal{M}_{b-1}) \tag{1}$$

Here, b indexes the segment containing token n. Both A_t and T_t are causal Voila states based on preceding audio from both speakers and the agent's text available at that step. Thus, Equation (1) uses no information beyond the current audio-state block. Once all four rows' states are available, the predictor commits a motion token representing all four frames, without subsequent revision. HiddenFace is therefore block-causal. During predictor training, Voila and the motion tokenizer remain frozen. Generated motion is not fed back and does not alter Voila's speech predictions.

3.2 Motion Tokenization

Representing facial motion as discrete tokens allows the predictor to use next-token prediction, as in the backbone. To avoid accessing frames beyond the current block, we discretize motion using a residual VQ-VAE (RVQ-VAE) with a causal encoder and decoder (Zeghidour et al., 2022; Mughal et al., 2026). The encoder uses causal convolution and a causal transformer to produce one latent z_n per four frames. Residual vector quantization with $Q = 6$ stages and 256 codes per stage yields a motion token $M_n = (m_{n,1}, \ldots, m_{n,Q})$ and a quantized latent $\hat{z}_n = \sum_q E_q[m_{n,q}]$, where E_q is the qth codebook. The causal decoder reconstructs $\hat{y}_{4n:4n+3}$ from $\hat{z}_{\le n}$, so the reconstructed motion does not depend on subsequent tokens. The tokenizer is trained with the following loss:

$$\mathcal{L}_{\text{tok}} = \mathcal{L}_{\text{param}} + \mathcal{L}_{\text{geo}} + \lambda_{\text{commit}}\mathcal{L}_{\text{commit}} \tag{2}$$

$\mathcal{L}_{\text{param}}$ measures reconstruction, temporal-difference, and motion-amplitude errors in FLAME parameter space, while $\mathcal{L}_{\text{geo}}$ measures mesh vertex position and temporal-difference errors. $\mathcal{L}_{\text{commit}}$ encourages encoder outputs to remain close to their codebook representations. We first train the tokenizer on the motion-only subset of UniLS-Talk (Chu et al., 2026), then continue training on the paired conversational training split used for the predictor, and finally freeze it (Appendix A).

3.3 Facial Motion Predictor

Hidden-state conditioning. Before generating audio row t, Voila computes the final-layer backbone state A_t, which we use to condition facial motion without reprocessing generated speech. Updated at every row, A_t provides the temporal reference for facial prediction and encodes both speakers' preceding audio and the agent's preceding text. Partner speech is therefore not separately provided to the motion predictor. The text hidden state T_t is the state used to predict the agent's most recent response text token and is held until the next text token is predicted. It conveys the current response context while speaking and retains the previous response state while listening. We use no explicit speaking/listening indicator or separate processing branches. To align the different update rates, we pair each audio row with the latest available text state, sum their linear projections, and combine the four rows within each block using a 1D convolution to obtain the motion condition C_n (Figure 2).

Previous-segment memory. Because Voila's context excludes motion tokens, C_n captures conversational context but not the predictor's generated motion history. To promote temporal continuity, the predictor uses its own previous motion as memory. All tokens in a segment share the memory Z_{b-1}, the sequence of quantized latents corresponding to the tokens $\mathcal{M}_{b-1}$ generated in the previous segment. The token representation h_n is computed as follows:

$$x_t = W_A A_t + W_T T_t, \qquad C_n = \text{Conv}(x_{4n:4n+3})\,, \qquad h_n = \text{CrossAttn}(C_n,\, Z_{b-1}) \tag{3}$$

CrossAttn comprises four cross-attention blocks using C_n as the query and projected Z_{b-1} as keys and values (Figure 2). Without self-attention across queries, token n depends only on its audio-state block and previous-segment memory, without directly accessing earlier tokens in the same segment. Memory is updated at segment boundaries (Appendix B), and its fixed size keeps the predictor's per-token computational cost constant regardless of conversation length. This memory is distinct

from the latent history used by the decoder for frame reconstruction. We construct memory from model-generated tokens during both training and inference, stopping gradients through the training history. Using predicted rather than ground-truth (GT) tokens for memory narrows the training–inference mismatch and substantially reduces motion freezing (Section 4.4). When no motion history is available, such as at the start of a conversation, we use a single learned no-history token as memory.

Residual-depth head. Each code in a motion token quantizes the residual left by preceding stages, so we predict the codes sequentially conditioned on h_n and the previously predicted codes.

$$p_\theta(M_n \mid A_{4n:4n+3},\, T_{4n:4n+3},\, \mathcal{M}_{b-1}) = \prod_{q=1}^{Q} p_\theta(m_{n,q} \mid h_n,\, m_{n,<q}) \tag{4}$$

A three-layer causal transformer predicts each code from h_n and the codebook embedding of the previous stage's code. Its attention operates only across residual stages within the same token. Thus, cross-attention handles temporal context, while the depth head models residual dependencies.

Streaming inference. Inference is synchronized with Voila's generation loop. Once both hidden-state streams for block n are available, we form C_n, compute h_n using the current memory, and greedily select codes in residual-depth order, yielding deterministic motion for a fixed hidden-state sequence. The frozen decoder reconstructs four frames at 25 Hz from these codes. Each block's motion is finalized after its last hidden state becomes available and the facial branch completes its computation. In parallel, a lightweight head-rotation predictor uses the same C_n to predict codes for a dedicated RVQ-VAE (Appendix B).

3.4 TRAINING

Data. We train the predictor on real dyadic and synthetic self-play conversations. For real data, we feed both participants' recorded audio into Voila's two streams at 40 ms intervals and insert the target participant's transcript words as response text no earlier than their onset rows to extract hidden states. No future words or partner text are supplied. Target codes are obtained by encoding the aligned, tracked facial motion with the tokenizer. For self-play between two Voila instances taking turns, we generate pseudo motion using UniLS trained on real conversations and encode it in the same way (Chu et al., 2026). Real data provides human motion supervision, while self-play exposes the predictor to hidden states from Voila's native speech generation. We use half of the synthetic training conversations that pass integrity, dialogue-quality, and motion-quality checks. The resulting training set contains 598 hours of real and 222 hours of synthetic data, counted per role stream (Appendix C). Both sources share the same input interface and training objective. The tokenizer, fixed matrices, affine maps, and normalization statistics all come from training data only. All valid tokens receive supervision across speaking, listening, quiet, and overlapping speech.

Objective. The primary loss is cross-entropy over target codes produced by the tokenizer.

$$\mathcal{L}_{\mathrm{CE}} = -\sum_{n}\sum_{q} \log p_\theta\big(m^{\mathrm{gt}}_{n,q} \mid h_n,\, m^{\mathrm{gt}}_{n,<q}\big) \tag{5}$$

$\mathcal{L}_{\mathrm{CE}}$ supervises code probabilities without directly constraining decoded motion, resulting in underestimated upper-face motion amplitude (Section 4.4). We therefore add motion-level losses.

Decoded-motion terms. Full-batch decoding and backpropagation through FLAME increase training cost severalfold. We instead linearize the decoder and FLAME mapping around the targets, approximating squared vertex position and velocity errors as quadratic forms $\mathcal{L}_{\mathrm{G}}$ and $\mathcal{L}_{\mathrm{V}}$ of latent prediction errors. Their symmetric positive semidefinite matrices Q_{pos} and Q_{vel} are precomputed from vector-Jacobian product samples on training data and kept fixed, measuring vertex-space discrepancies in latent space. To match upper-face motion amplitude, $\mathcal{L}_{\mathrm{S}}$ maps predicted latents to upper-face vertex motion through an affine transformation fitted on training data, then compares predicted and target standard deviations separately for speaking and listening. The three terms are:

$$\mathcal{L}_{\mathrm{G}} = \sum_{n} \Delta_{\le n}^{\top} Q_{\mathrm{pos}} \Delta_{\le n}, \quad \mathcal{L}_{\mathrm{V}} = \sum_{n} \Delta_{\le n}^{\top} Q_{\mathrm{vel}} \Delta_{\le n}, \quad \mathcal{L}_{\mathrm{S}} = \sum_{s} \|\sigma_s(\mathrm{pred}) - \sigma_s(\mathrm{target})\|^2 \tag{6}$$

Here, $\Delta_{\le n}$ denotes latent prediction errors over the decoder context for token n, and s indexes speaking and listening intervals.

Table 1: Comparison with audio-driven baselines. Step reports latency at 64/128 s. For HiddenFace, it includes the predictor, motion decoder, and parallel head-rotation predictor, excluding Voila and the audio codec. **Bold** indicates the best and underline the second best. † Identical results in both settings. ‡ By 128 s, 300 of 800 steps exceed 160 ms.

	Geometric accuracy (mm)				Upper face (10^{-5} m)		Lip sync		Dialogue level				Latency	
Method	S-LVE↓	S-MHD↓	MOD↓	Lip-W↓	S-FDD↓	L-FDD↓	AVC↑	MTM (ms)↓	MSE↓	FD↓	P-FD↓	rPCC↓	First (s)↓	Step (ms)↓
Released inference on the full audio (non-causal)														
DiffPoseTalk	11.592	2.990	4.170	4.693	32.43	28.58	0.212	75.7	0.806	72.80	73.85	0.293	24.17	–
ARTalk	9.123	2.401	3.966	3.879	39.19	36.66	0.206	74.6	0.234	18.86	19.60	0.173	4.04	–
DualTalk	13.274	2.850	5.498	4.255	35.94	31.81	0.134	84.2	0.765	71.44	72.21	0.256	16.05	–
UniLS	**7.763**	2.314	**3.281**	3.657	23.96	**23.18**	**0.485**	**50.2**	0.221	14.81	15.79	0.126	4.09	–
HiddenFace (Ours)†	8.229	**2.247**	3.730	**3.452**	**23.33**	26.88	0.374	63.9	**0.178**	**11.00**	**11.96**	**0.113**	**0.18**	5.8/5.8
160 ms causal streaming (same condition for all)														
ARTalk	9.993	2.561	4.253	4.047	36.44	33.21	0.115	92.7	0.242	17.89	18.92	0.178	0.20	45/45
DualTalk	13.148	2.859	5.427	4.293	33.87	30.12	0.162	82.4	0.767	70.74	71.60	0.260	0.19	126/282‡
UniLS	8.604	2.484	**3.725**	3.940	24.58	**22.99**	0.305	**58.4**	0.241	15.32	16.44	0.135	0.25	86/87
HiddenFace (Ours)	**8.229**	**2.247**	3.730	**3.452**	**23.33**	26.88	**0.374**	63.9	**0.178**	**11.00**	**11.96**	**0.113**	**0.18**	**5.8/5.8**

Amplitude decoupling. The quadratic error decomposes into magnitude and direction terms. At low prediction–target correlation, the magnitude component of its gradient favors amplitude shrinkage, conflicting with $\mathcal{L}_{\mathrm{S}}$. We therefore modify this component in the velocity loss to match target amplitude independently of correlation, while preserving the direction and timing gradients and the forward loss value. The position term retains its squared error, as applying the same modification produces excessive lip motion.

Schedule. The overall loss is:

$$\mathcal{L} = \mathcal{L}_{\mathrm{CE}} + \lambda_{\mathrm{G}}\mathcal{L}_{\mathrm{G}} + \lambda_{\mathrm{V}}\mathcal{L}_{\mathrm{V}} + \lambda_{\mathrm{S}}\mathcal{L}_{\mathrm{S}} \tag{7}$$

We initially optimize only $\mathcal{L}_{\mathrm{CE}}$ and introduce all three auxiliary terms when upper-face motion amplitude begins to decline. For nondifferentiable code selection, we use straight-through Gumbel-Softmax (Jang et al., 2017) at each residual stage to construct latents. We select λ to balance gradient contributions and limit dominance or cancellation among the auxiliary terms (Appendix D).

4 Experiments

4.1 Experimental Setup

Baselines. We evaluate on UniLS's real dyadic conversation test split, constructed from Seamless Interaction (Agrawal et al., 2025) and partitioned by interaction group. This held-out split contains 563 conversations, 112 participants, 3,108 role streams, and 30.0 hours. Baselines include single-speaker methods DiffPoseTalk (Sun et al., 2024) and ARTalk (Chu et al., 2025), and dyadic methods DualTalk (Peng et al., 2025) and UniLS (Chu et al., 2026), which use both speakers' audio to model listening responses. Existing methods using LALM representations require retraining on a common backbone for matched comparison; we instead compare audio-code and hidden-state conditioning from the same backbone (Section 4.4). We use released implementations and checkpoints without additional training. DualTalk's partner facial motion input is set to zero because the agent does not observe the partner's face.

Evaluation conditions. Since the baselines assume offline access to future audio, we evaluate two settings. Offline evaluation follows each baseline's released inference procedure using full audio. In 160 ms block-causal streaming, each method generates motion after each block using only audio available through that block, without revising previous outputs. HiddenFace operates this way natively, yielding identical results in both settings. Quality metrics use completed outputs regardless of computation time; latency separately assesses real-time feasibility. We exclude DiffPoseTalk from streaming evaluation because it regenerates the entire available sequence at each step, taking 8.2 s per generation and making full-test-set evaluation impractical.

Evaluation metrics (Appendix E). All facial metrics use a fixed head pose. S-/L- denote speaking/listening intervals based on dataset speech masks; unprefixed metrics use all frames of a role stream. Geometric accuracy is measured by lip vertex error (S-LVE), mean head distance (S-MHD),

Table 2: Human evaluation (24 raters; 1–5 scale). Brackets show rater-level bootstrap 95% confidence intervals. Best/Worst report the percentage of selections of each method as the best/worst panel.

Method	Lip sync↑	Listening↑	Overall↑	Best↑	Worst↓
DiffPoseTalk	2.75 [2.36, 3.17]	1.37 [1.15, 1.69]	2.01 [1.73, 2.32]	3 %	55 %
ARTalk	2.36 [2.10, 2.65]	2.81 [2.29, 3.36]	2.49 [2.18, 2.81]	4 %	29 %
DualTalk	3.83 [3.60, 4.07]	3.85 [3.58, 4.12]	3.71 [3.47, 3.94]	34 %	4 %
UniLS	3.13 [2.83, 3.43]	3.44 [3.18, 3.70]	3.22 [2.96, 3.48]	12 %	10 %
HiddenFace (Ours)	**4.06** [3.81, 4.30]	**4.11** [3.85, 4.35]	**4.01** [3.76, 4.25]	**47** %	**1** %

mouth-opening error (MOD), and lip-width MAE (Lip-W). We measure upper-face dynamics using upper-face dynamic deviation (S-FDD, L-FDD), and lip synchronization using mouth-opening velocity correlation (AVC) and opening/closing timing error (MTM). Following DualTalk (Peng et al., 2025), dialogue-level metrics include expression coefficient MSE, Fréchet distance from real expressions (FD), distribution distance after pairing with the partner's GT expressions (P-FD), and partner-motion correlation deviation (rPCC). Latency measures time to first motion (First) and processing time per 160 ms step at 64/128 s into a conversation (Step).

Implementation details. We train the predictor on two NVIDIA RTX PRO 6000 GPUs (Appendix D). Latency is measured for all methods on the same single GPU with batch size 1, excluding rendering.

4.2 Comparison with Audio-Driven Baselines

Quantitative results. Under the same block-causal condition, existing LALM hidden states match or exceed the lip accuracy of audio-driven methods. HiddenFace achieves the lowest streaming S-LVE, S-MHD, Lip-W, and MSE (S-LVE: 8.229 vs. UniLS's 8.604), with comparable MOD (3.730 vs. 3.725). Its S-MHD, Lip-W, and MSE also outperform all offline baselines. Offline UniLS has lower S-LVE and MOD, but both worsen under streaming, suggesting a benefit from look-ahead within its 4 s chunks. HiddenFace has the highest streaming AVC (0.374 vs. 0.305), whereas UniLS has lower MTM in both settings (streaming: 58.4 vs. 63.9 ms). HiddenFace also achieves the lowest S-FDD, FD, P-FD, and rPCC across both settings. UniLS, however, better matches listening dynamics (L-FDD: 22.99/23.18 for streaming/offline vs. 26.88). Despite trade-offs in mouth timing and listening dynamics, HiddenFace most closely matches human expression distributions and partner-motion correlations among the compared methods.

Latency. Streaming methods produce the first motion in roughly 0.2 s, with larger differences in per-step cost. HiddenFace adds only 5.8 ms per 160 ms step at both 64 and 128 s, independent of conversation length. In contrast, DualTalk regenerates the entire history at each step, slowing as conversations grow and eventually exceeding the 160 ms budget. The released offline baseline pipelines require 4.0–24.2 s before producing the first motion. End-to-end real-time performance depends on the backbone and hardware. In a 160 s live run, emitting audio and motion together in 160 ms blocks kept every frame in sync with its audio while Voila operated in real time (Appendix F).

Qualitative results. Figure 3 compares faces generated by each method across four phonetic categories and shows HiddenFace during a brief turn transition. Generated videos, including turn transitions, are provided in the supplementary material.

4.3 Human Study

We assess listening responses and overall naturalness, which metrics alone cannot fully capture, through a human study. We select 30 clips from Voila self-play conversations, one per conversation, based on content and audio (10.2–22.7 s; mean 15.5 s). Each clip presents five systems' outputs for the same dialogue in anonymous, randomly ordered panels. Baselines follow the released offline protocols in Table 1. Each method retains its generated head motion, so the study evaluates overall avatar motion. Twenty-four raters score each panel on lip synchronization (Lip sync), listening responses (Listening), and overall naturalness (Overall) from 1 to 5, then select the best and worst panels. Questions and procedures are provided in Appendix G.

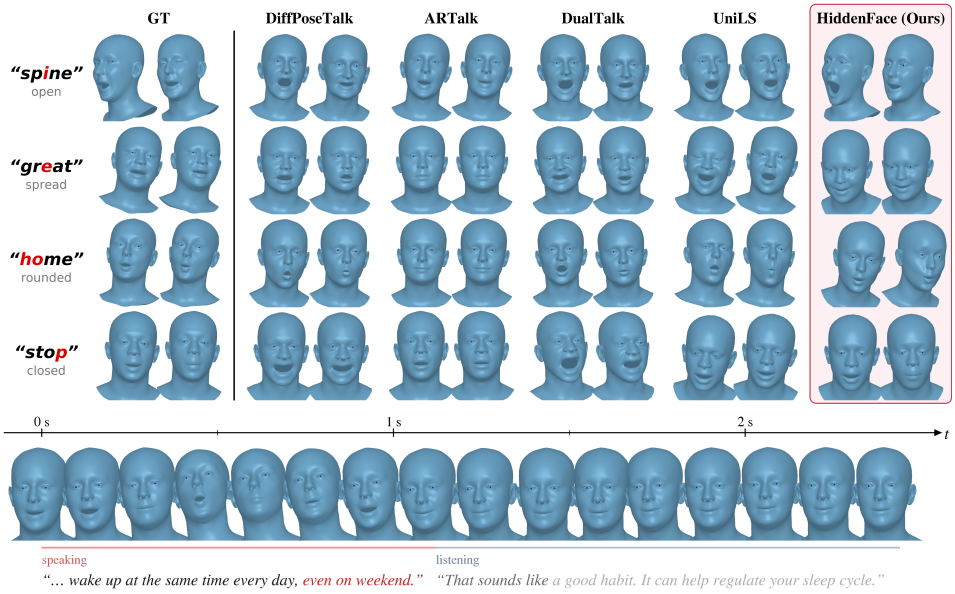

Figure 3: Qualitative results. Top: Four mouth-shape examples from validation conversations, showing frames 0.16 s before (left) and at the selected moment (right). Baselines use released full-audio inference; HiddenFace uses 160 ms block-causal generation. Bottom: HiddenFace outputs across a turn transition in Voila self-play, with speaking/listening intervals marked.

HiddenFace achieves the highest mean scores on all three criteria and is selected best most often and worst least often (Table 2). Rater-level bootstrap 95% confidence intervals for paired differences indicate that its 0.23–0.31-point gains over runner-up DualTalk are not statistically significant, whereas gains over UniLS, ARTalk, and DiffPoseTalk are significant on all three criteria. DualTalk ranks second and UniLS third, reversing their relative standing in Table 1. This may reflect differences in evaluation settings: quantitative metrics measure agreement with GT in recorded conversations, whereas raters judge perceptual plausibility as a conversational partner in generated conversations without GT. HiddenFace's lead on most quantitative metrics and its highest mean human ratings show that frozen LALM states yield high-quality facial motion, including when the LALM generates its own conversations.

4.4 Ablations

The input-condition and memory ablations use only cross-entropy to isolate their effects from auxiliary objectives. All ablations use the same test split, which was not used for design decisions.

Input condition. When only the predictor input changes (Table 3(a)), generated audio codes yield the best AVC and MTM but worse lip-position accuracy and dialogue-level metrics than combined hidden states (S-LVE: 9.295 vs. 8.212; FD: 13.64 vs. 11.66). This condition receives only the agent's audio codes, without partner audio, accumulated LALM context, or response text. The comparison therefore reflects differences in available context as well as representation. Text states alone lack fine-grained speech-timing cues, resulting in poor timing (MTM: 119.0 ms). Audio states alone outperform codes in lip-position accuracy and dialogue-level metrics; adding text states yields small but consistent gains in speaking lip accuracy and motion amplitude (S-LVE: 8.364 vs. 8.212; S-FDD: 26.82 vs. 25.83). Audio states thus provide the temporal basis, while text states complement them with response context.

Temporal memory. The strongest effect in Table 4 comes from how memory is populated during training. Training with GT codes instead of model predictions creates a mismatch with self-generated memory at deployment, causing repeated code tuples and motion freezing (same-tuple: $0.074 \rightarrow 0.668$; freeze gap: $0.0053 \rightarrow 0.5062$). Training with self-generated memory is therefore crucial for stable deployment of our predictor. Compared with no memory, results are mixed: removing memory slightly improves F-MAE, L-FDD, and MTM, whereas using memory better matches human speaking

Table 3: Ablations on (a) predictor conditioning and (b) auxiliary objectives. Models in (a) use cross-entropy only and are not directly comparable to Table 1. In (b), Full objective is HiddenFace in Table 1, while w/o all is the same model as Audio and text hidden in (a) and Own history in Table 4. **Bold** indicates the best.

	Geometric accuracy (mm)				Upper face (10^{-5} m)		Lip sync		Dialogue level			
Setting	S-LVE↓	S-MHD↓	MOD↓	Lip-W↓	S-FDD↓	L-FDD↓	AVC↑	MTM (ms)↓	MSE↓	FD↓	P-FD↓	rPCC↓
(a) Input condition *(cross-entropy only)*												
Audio codes only	9.295	2.454	4.080	4.408	26.02	30.54	**0.349**	**68.1**	0.199	13.64	14.52	0.155
Audio hidden only	8.364	2.273	3.799	3.600	26.82	**28.30**	0.324	70.1	0.183	11.76	12.72	0.119
Text hidden only	9.337	2.461	4.252	3.931	28.90	30.85	0.133	119.0	0.207	13.52	14.62	0.121
Audio and text hidden (Ours)	**8.212**	**2.252**	**3.728**	**3.537**	**25.83**	28.34	0.330	69.0	**0.182**	**11.66**	**12.62**	**0.116**
(b) Training objective *(same initialization checkpoint, data order and schedule)*												
Full objective (Ours)	8.229	**2.247**	3.730	3.452	**23.33**	26.88	0.374	63.9	0.178	**11.00**	**11.96**	**0.113**
w/o $\mathcal{L}_{\mathrm{G}}$	8.314	2.260	3.769	3.542	23.73	27.08	0.376	63.4	0.181	11.25	12.21	**0.113**
w/o $\mathcal{L}_{\mathrm{V}}$	8.348	2.287	3.748	3.567	23.91	**26.50**	0.331	70.0	0.190	11.66	12.66	**0.113**
w/o $\mathcal{L}_{\mathrm{S}}$	8.366	2.248	3.816	**3.435**	27.58	30.67	**0.386**	**62.5**	**0.175**	11.32	12.24	0.114
w/o all (CE only)	**8.212**	2.252	**3.728**	3.537	25.83	28.34	0.330	69.0	0.182	11.66	12.62	0.116

Table 4: Ablation on predictor memory. F-MAE averages errors equally across expression, jaw, and eye parameter groups; freeze gap measures the difference from the human motion-freeze rate during speech; deployed CE is token cross-entropy with model-generated memory; and same-tuple is the fraction of consecutive tokens with identical code tuples. Other metric units follow Table 1.

Setting	F-MAE↓	Freeze gap↓	S-FDD↓	L-FDD↓	MTM (ms)↓	Lip-W↓	Deployed CE↓	Same-tuple
Own history (Ours)	0.1501	**0.0053**	**25.83**	28.34	69.0	**3.537**	**5.287**	0.074
No history	**0.1459**	0.0185	26.10	**27.68**	**68.4**	3.616	5.344	0.076
GT history	0.1981	0.5062	35.15	36.24	87.1	4.519	6.379	0.668

freeze rates (freeze gap: 0.0053 vs. 0.0185) and improves speaking motion amplitude, lip-width accuracy, and deployed cross-entropy.

Training objective. Removing auxiliary terms individually (Table 3(b)) reveals distinct effects. Removing $\mathcal{L}_{\mathrm{G}}$ worsens lip accuracy and S-FDD, while removing $\mathcal{L}_{\mathrm{V}}$ degrades lip accuracy and mouth-opening velocity correlation (AVC). Removing $\mathcal{L}_{\mathrm{S}}$ slightly improves AVC, MTM, Lip-W, and MSE but substantially increases amplitude errors during both speaking and listening (S-FDD: $23.33 \rightarrow 27.58$; L-FDD: $26.88 \rightarrow 30.67$). This term thus preserves motion amplitude at a small cost to timing, lip width, and expression accuracy. Removing all three lowers S-LVE and MOD by at most 0.02 mm (8.212 vs. 8.229; 3.728 vs. 3.730) but worsens lip width, motion amplitude, lip synchronization, and dialogue-level metrics. Overall, the auxiliary objectives bring motion amplitude closer to human levels while keeping framewise lip accuracy essentially unchanged.

5 Conclusion

We presented HiddenFace, which commits conversational facial motion every 160 ms using only a frozen LALM's audio/text hidden states and its own motion tokens, without reprocessing speech or waiting for context beyond the current block. On real conversations under matched causal streaming conditions, it achieves competitive or better lip accuracy than audio-driven methods. It also better matches human expression distributions than offline baselines and obtains the highest mean human ratings for lip synchronization, listening responses, and overall naturalness. Its additional facial computation is small and constant per step, independent of conversation length. With the same predictor, replacing generated audio codes with hidden states improves lip-position accuracy and dialogue-level metrics. These results support using existing LALM representations as a facial-motion interface for LALM-based avatars, without speech reprocessing or backbone retraining. However, we validate this interface only on Voila. Extending it to LALMs with different audio/text arrangements and comparing methods that use LALM representations on a shared backbone remain future work. We also do not directly assess whether listening responses are semantically appropriate; whether conversational context in hidden states translates into such responses remains to be investigated.

AI USE STATEMENT

We used generative AI tools as assistants for code implementation and execution, data processing, translation, and writing. The authors conceived the research, designed the method and experiments, made all methodological and experimental decisions, and interpreted the results. The authors reviewed all AI-assisted work and take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

All human-study participants were volunteers who gave informed consent, and only anonymous ratings, timings, and brief demographic answers were stored. The study was not reviewed by an institutional review board. Real conversations come from Seamless Interaction (Agrawal et al., 2025) under its non-commercial license, and the self-play conversations condition Voila on reference voices of its participants. We render only identity-free FLAME meshes. Avatars driven by LALM speech could be misused to impersonate people, so deployed systems should disclose that the agent is artificial.

REPRODUCIBILITY STATEMENT

Sections 3 and 4 and Appendices A–G describe the model, training data, training procedure, hyperparameters, evaluation metrics, latency and streaming measurements, and human-study protocol. Evaluation uses UniLS's test split and the released baseline implementations. We will release the code, trained models, and the scripts that generate the self-play conversations upon acceptance.

REFERENCES

Vasu Agrawal et al. Seamless Interaction: Dyadic audiovisual motion modeling and large-scale dataset, 2025. URL https://arxiv.org/abs/2506.22554.

Justine Cassell, Joseph Sullivan, Scott Prevost, and Elizabeth F. Churchill (eds.). *Embodied Conversational Agents*. MIT Press, 2000. ISBN 9780262032780. URL https://mitpress.mit.edu/9780262032780/embodied-conversational-agents/.

Xuangeng Chu, Nabarun Goswami, Ziteng Cui, Hanqin Wang, and Tatsuya Harada. ARTalk: Speech-driven 3D head animation via autoregressive model. In *Proceedings of the SIGGRAPH Asia 2025 Conference Papers*, 2025. doi: 10.1145/3757377.3763955. URL https://arxiv.org/abs/2502.20323.

Xuangeng Chu, Ruicong Liu, Yifei Huang, Yun Liu, Yichen Peng, and Bo Zheng. UniLS: End-to-end audio-driven avatars for unified listening and speaking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 25142–25152, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Chu_UniLS_End-to-End_Audio-Driven_Avatars_for_Unified_Listening_and_Speaking_CVPR_2026_paper.html.

Daniel Cudeiro, Timo Bolkart, Cassidy Laidlaw, Anurag Ranjan, and Michael J. Black. Capture, learning, and synthesis of 3D speaking styles. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10101–10111, 2019. URL https://openaccess.thecvf.com/content_CVPR_2019/html/Cudeiro_Capture_Learning_and_Synthesis_of_3D_Speaking_Styles_CVPR_2019_paper.html.

Alexandre Défossez, Laurent Mazaré, Manu Orsini, Amélie Royer, Patrick Pérez, Hervé Jégou, Edouard Grave, and Neil Zeghidour. Moshi: A speech-text foundation model for real-time dialogue, 2024. URL https://arxiv.org/abs/2410.00037.

Yingruo Fan, Zhaojiang Lin, Jun Saito, Wenping Wang, and Taku Komura. FaceFormer: Speech-driven 3D facial animation with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18770–18780, 2022. URL https://openaccess.thecvf.com/content/CVPR2022/html/Fan_FaceFormer_Speech-Driven_3D_Facial_Animation_With_Transformers_CVPR_2022_paper.html.

Xiaolin Hu, Hang Yuan, Xinzhu Sang, Binbin Yan, Zhou Yu, Cong Huang, and Kai Chen. A^2-LLM: An end-to-end conversational audio avatar large language model, 2026. URL https://arxiv.org/abs/2602.04913.

Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with Gumbel-Softmax. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=rkE3y85ee.

Jingjing Jiang, Atsumoto Ohashi, and Ryuichiro Higashinaka. Integrating facial generation into full-duplex spoken dialogue systems. In *Interspeech 2026*, pp. 3432–3436, 2026. URL https://www.isca-archive.org/interspeech_2026/jiang26g_interspeech.html.

Chae-Yeon Lee, Hyun-Bin Oh, EunGi Han, Sung-Bin Kim, Suekyeong Nam, and Tae-Hyun Oh. Perceptually accurate 3D talking head generation: New definitions, speech-mesh representation, and evaluation metrics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21065–21074, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Chae-Yeon_Perceptually_Accurate_3D_Talking_Head_Generation_New_Definitions_Speech-Mesh_Representation_CVPR_2025_paper.html.

Tianye Li, Timo Bolkart, Michael J. Black, Hao Li, and Javier Romero. Learning a model of facial shape and expression from 4D scans. *ACM Transactions on Graphics*, 36(6):194:1–194:17, 2017. doi: 10.1145/3130800.3130813.

Habin Lim, Jae-Ho Lee, Hah Min Lew, Ji-Su Kang, and Gyeong-Moon Park. FacePlex: Full-duplex joint speech-facial motion generation for conversational avatars, 2026. URL https://arxiv.org/abs/2606.30145.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

M. Hamza Mughal, Rishabh Dabral, Vera Demberg, and Christian Theobalt. MIBURI: Towards expressive interactive gesture synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 40031–40041, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/papers/Mughal_MIBURI_Towards_Expressive_Interactive_Gesture_Synthesis_CVPR_2026_paper.pdf.

Koki Nagano, Hongyu Liu, Seonwook Park, Tianye Li, Amrita Mazumdar, Christian Jacobsen, Shengze Wang, Michael Stengel, Rajarshi Roy, Ka Chun Cheung, Simon See, and Shalini De Mello. DyaPlex: Full-duplex speech-motion model for dyadic interaction, 2026. URL https://arxiv.org/abs/2606.03874.

Evonne Ng, Hanbyul Joo, Liwen Hu, Hao Li, Trevor Darrell, Angjoo Kanazawa, and Shiry Ginosar. Learning to listen: Modeling non-deterministic dyadic facial motion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20395–20405, 2022. URL https://openaccess.thecvf.com/content/CVPR2022/html/Ng_Learning_To_Listen_Modeling_Non-Deterministic_Dyadic_Facial_Motion_CVPR_2022_paper.html.

Tu Anh Nguyen, Benjamin Muller, Bokai Yu, Marta R. Costa-jussà, Maha Elbayad, Sravya Popuri, Christophe Ropers, Paul-Ambroise Duquenne, Robin Algayres, Ruslan Mavlyutov, Itai Gat, Mary Williamson, Gabriel Synnaeve, Juan Pino, Benoît Sagot, and Emmanuel Dupoux. SpiRit-LM: Interleaved spoken and written language model. *Transactions of the Association for Computational Linguistics*, 13:30–52, 2025. URL https://aclanthology.org/2025.tacl-1.2/.

Ziqiao Peng, Yanbo Fan, Haoyu Wu, Xuan Wang, Hongyan Liu, Jun He, and Zhaoxin Fan. DualTalk: Dual-speaker interaction for 3D talking head conversations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21055–21064, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/papers/Peng_DualTalk_Dual-Speaker_Interaction_for_3D_Talking_Head_Conversations_CVPR_2025_paper.pdf.

Alexander Richard, Michael Zollhöfer, Yandong Wen, Fernando de la Torre, and Yaser Sheikh. MeshTalk: 3D face animation from speech using cross-modality disentanglement. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1173–1182, 2021. URL https://openaccess.thecvf.com/content/ICCV2021/papers/Richard_MeshTalk_3D_Face_Animation_From_Speech_Using_Cross-Modality_Disentanglement_ICCV_2021_paper.pdf.

Rajarshi Roy, Jonathan Raiman, Sang-gil Lee, Teodor-Dumitru Ene, Robert Kirby, Sungwon Kim, Jaehyeon Kim, and Bryan Catanzaro. PersonaPlex: Voice and role control for full duplex conversational speech models. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 16137–16141, 2026. doi: 10.1109/ICASSP55912.2026.11463413. URL https://arxiv.org/abs/2602.06053.

Yemin Shi, Yu Shu, Siwei Dong, Guangyi Liu, Jaward Sesay, Jingwen Li, and Zhiting Hu. Voila: Voice-language foundation models for real-time autonomous interaction and voice role-play, 2025. URL https://arxiv.org/abs/2505.02707.

Zhiyao Sun, Tian Lv, Sheng Ye, Matthieu Lin, Jenny Sheng, Yu-Hui Wen, Minjing Yu, and Yong-Jin Liu. DiffPoseTalk: Speech-driven stylistic 3D facial animation and head pose generation via diffusion models. *ACM Transactions on Graphics*, 43(4):46:1–46:9, 2024. doi: 10.1145/3658221. URL https://diffposetalk.github.io/.

Bandhav Veluri, Benjamin N. Peloquin, Bokai Yu, Hongyu Gong, and Shyamnath Gollakota. Beyond turn-based interfaces: Synchronous LLMs as full-duplex dialogue agents. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 21390–21402, 2024. URL https://aclanthology.org/2024.emnlp-main.1192/.

Jinbo Xing, Menghan Xia, Yuechen Zhang, Xiaodong Cun, Jue Wang, and Tien-Tsin Wong. CodeTalker: Speech-driven 3D facial animation with discrete motion prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12780–12790, 2023. URL https://openaccess.thecvf.com/content/CVPR2023/html/Xing_CodeTalker_Speech-Driven_3D_Facial_Animation_With_Discrete_Motion_Prior_CVPR_2023_paper.html.

Neil Zeghidour, Alejandro Luebs, Ahmed Omran, Jan Skoglund, and Marco Tagliasacchi. SoundStream: An end-to-end neural audio codec. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30:495–507, 2022. doi: 10.1109/TASLP.2021.3129994.

Dong Zhang, Shimin Li, Xin Zhang, Jun Zhan, Pengyu Wang, Yaqian Zhou, and Xipeng Qiu. SpeechGPT: Empowering large language models with intrinsic cross-modal conversational abilities. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 15757–15773. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.findings-emnlp.1055. URL https://aclanthology.org/2023.findings-emnlp.1055/.

Haoyu Zhang, Zhipeng Li, Yiwen Guo, and Tianshu Yu. Ex-Omni: Enabling 3D facial animation generation for omni-modal large language models, 2026. URL https://arxiv.org/abs/2602.07106.

Qinglin Zhang, Luyao Cheng, Chong Deng, Qian Chen, Wen Wang, Siqi Zheng, Jiaqing Liu, Hai Yu, Chao-Hong Tan, Zhihao Du, and ShiLiang Zhang. OmniFlatten: An end-to-end GPT model for seamless voice conversation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 14570–14580, 2025. URL https://aclanthology.org/2025.acl-long.709/.

Dingcheng Zhen, Shunshun Yin, Shiyang Qin, Hou Yi, Ziwei Zhang, Siyuan Liu, Gan Qi, and Ming Tao. Teller: Real-time streaming audio-driven portrait animation with autoregressive motion generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21075–21085, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Zhen_Teller_Real-Time_Streaming_Audio-Driven_Portrait_Animation_with_Autoregressive_Motion_Generation_CVPR_2025_paper.html.

Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5745–5753, 2019. URL https://openaccess.thecvf.com/content_CVPR_2019/html/Zhou_On_the_Continuity_of_Rotation_Representations_in_Neural_Networks_CVPR_2019_paper.html.

A Motion Tokenizer Details

Architecture. In the tokenizer of Section 3.2 (Table 5), the encoder embeds the expression, jaw, and eye parameters by group and combines them, extracts per-frame features with a causal convolution layer and three dilated residual blocks, and reduces every four frames to one token with a stride-4 causal convolution before a six-layer causal transformer (width 512, 8 heads). Each transformer layer attends only to the seven most recent tokens, including the current one. The decoder mirrors this stack and is made causal by discarding the last four frames of each transposed-convolution output. The codebooks are initialized with k-means and updated with EMA (decay 0.99), and rarely used codes are periodically re-initialized (Zeghidour et al., 2022).

Loss. Table 6 lists the terms and weights of $\mathcal{L}_{\text{param}}$ and $\mathcal{L}_{\text{geo}}$ in Equation (2). $\mathcal{L}_{\text{param}}$ is computed on de-standardized FLAME parameters, and its amplitude term compares, with a smooth L1 loss, the jaw range (maximum minus minimum) and the standard deviation of each expression and eye dimension within a training window. $\mathcal{L}_{\text{geo}}$ is computed on FLAME2020 meshes (mm) with zero head rotation and shape, using the 1,787 face-region vertices and the 250 lip-region vertices among them. The weights of these terms are calibrated by gradient magnitude so that errors in mm^2 are on the same scale as the parameter terms. $\mathcal{L}_{\text{commit}}$ sums, over the Q stages, the MSE between each stage's input residual and the selected code, with $\lambda_{\text{commit}} = 0.25$.

Training. Stage 1 trains from random initialization for 100 epochs on 307,883 training clips of the motion-only subset. Stage 2 trains for 200 epochs on 144,944 training clips of the paired conversational subset, starting from the last Stage 1 checkpoint with a newly initialized optimizer and learning-rate schedule. Both stages use batches of 1,638 four-second windows, AdamW (Loshchilov & Hutter, 2019) with learning rate 10^{-4} and weight decay 0.002, gradient-norm clipping at 1.0, and a learning rate halved every 20 epochs. λ_{commit} increases linearly from 0 over the first 25 epochs of Stage 1. We use the Stage 2 checkpoint with the lowest validation loss; on the Stage 2 validation set, it uses all 256 codes at each of the six stages (per-stage perplexity 208–229). Normalization statistics for both stages are computed on the Stage 2 training split, and no interaction group in the test split of Section 4.1 is used for training or checkpoint selection in either stage.

Head-rotation tokenizer. Head rotation is discretized by a separate RVQ-VAE with the same architecture; Table 5 lists its differences from the facial motion tokenizer. Only the input and output layers are adapted to 6D rotations, and the input is the 6D representation (Zhou et al., 2019) of the rotation relative to the first frame of each clip. The geodesic and angular-velocity errors in the loss are divided by the RMS rotation angle and RMS angular velocity of that stage's training split, respectively, before squaring. The training data and settings are the same as for the facial motion tokenizer. We use the last Stage 2 epoch, which also has the lowest validation rotation and angular-velocity errors among the evaluated checkpoints. On the Stage 2 validation set, it uses all 256 codes at each of the four stages (per-stage perplexity 148–177).

Table 5: Differences between the facial motion and head-rotation tokenizers. The remaining architecture and training settings are identical. Parameter counts exclude the EMA-updated codebooks and thus do not depend on the number of stages; the head-rotation predictor itself has 2.67M parameters (Appendix B).

	Facial motion	Head rotation
Input (25 Hz)	expression, jaw, eye (100, 1, 4)	6D rotation (6)
Codebook	6 stages × 256 codes (256-d)	4 stages × 256 codes (256-d)
Loss	Equation (2), Table 6	SO(3) geodesic error^2 + angular-velocity error^2 + 0.25·commitment
Parameters	41.8M	42.1M

Table 6: Terms and weights of $\mathcal{L}_{\text{param}}$ and $\mathcal{L}_{\text{geo}}$ in Equation (2). Differences and Laplacians are taken along the time axis of the 25 Hz sequences.

	Term	Weight
$\mathcal{L}_{\text{param}}$	MSE (all 105 dims / jaw / eye)	1 / 5 / 1
	Smooth L1 (expression / jaw / eye)	0.1 / 1 / 0.1
	First-difference MSE (all / jaw)	1 / 2
	Second-difference MSE (all / jaw)	0.5 / 1
	Five-point Laplacian MSE (all)	0.1
	Amplitude (jaw range; expression and eye s.d.)	0.1
$\mathcal{L}_{\text{geo}}$	Vertex position MSE (face / lips)	6.12×10^{-4} / 2.64×10^{-4}
	Face-vertex first- / second-difference MSE	4.89×10^{-3} / 2.01×10^{-3}

B Predictor and Head-Rotation Predictor

Facial motion predictor. The two 4096-dimensional hidden states are each projected to 1024 dimensions by a bias-free linear layer. The two projections are summed at each row, and a 1D convolution (kernel 4, stride 4) with GELU combines the four rows into C_n. The states of control tokens (e.g., <eos>) are not used as T_t, and $T_t = 0$ before the first response text token, leaving only the audio path. A segment spans six tokens (0.96 s), and memory is updated all at once at the end of each segment; the no-history token therefore serves as memory during the first 0.96 s of a conversation. Memory projects the quantized latents of the previous segment's tokens to 1024 dimensions and is attended by four cross-attention blocks (width 1024, 16 heads, FF 4096). The depth head consists of a three-layer causal transformer (width 256, 4 heads) and six stage-specific 256-way classifiers. Memory, the depth head, and the decoder share the same fixed codebooks of the tokenizer, and the predictor has 67.9M trainable parameters.

Head-rotation predictor. Head rotation is generated by a small predictor (2.67M parameters) that predicts the four-stage codes of the head RVQ-VAE in Appendix A (Table 5). Its input is C_n of the trained facial motion predictor, and no gradient flows back into the facial motion predictor. Because head rotation continues from the previous pose, the predictor, like the facial motion predictor, uses the head codes of the previous segment as memory. After reducing C_n to 256 dimensions, two cross-attention blocks (width 256, 4 heads, FF 1024) and a two-layer depth head (width 128, 4 heads) predict the four stages in order, and the frozen head decoder reconstructs them as rotations relative to the first frame, which serves as the reference (identity) pose.

Head decoding and training. Because selecting every stage greedily left the head nearly still on the validation set, we sample only the first-stage code at temperature 1.0 and select the remaining stages greedily. Unlike the facial motion predictor, which generates greedily, the sampling-based head-rotation predictor uses GT head codes as memory during training (teacher forcing) and its own generated codes at inference. We chose this because, on the validation set, training in this way made the speed and acceleration of the sampled head rotation close to those of real human head motion, with motion continuing naturally across segments. We train it on the same data as the facial motion predictor (Appendix C) with cross-entropy on the four-stage codes for 20 epochs, using AdamW (learning rate 10^{-4}) with a short warm-up followed by cosine decay to 0.1 times the initial rate. We select the final model by comparing its sampled head rotation with real human head motion on the validation set at each epoch: speed and acceleration as ratios to real human motion, and motion amplitude as pose dynamic deviation (PDD) (Chu et al., 2026) during speaking and listening. Because the acceleration ratio was an almost constant multiple of the speed ratio at every epoch, we use the model with the lowest summed PDD among the epochs whose speed ratio lies within 0.9–1.1.

C Training Data

Real conversations. Real conversations are continuous runs of conversations in the training split of the paired conversational subset of UniLS-Talk (Seamless Interaction (Agrawal et al., 2025)). Each of the two participants is treated as the agent, giving two role streams per run (27,323 runs; 54,646 role streams). Voila runs in the same full-duplex mode as at inference, but both audio streams and the target's text are replayed from the dataset. Each run's recorded audio is encoded once with the

released Voila tokenizer; re-encoding prefixes shows that a 40 ms code frame depends on up to four later audio frames (three at the median), apart from rare isolated code changes, and no normalization uses later audio. These codes are the tokens that decode to the recorded speech, so the replayed states are those Voila holds with this speech in its context. The states are causal in these tokens, and the predictor reads them only up to the current block. The live run in Appendix F encodes only past audio. The agent's voice is given by a reference voice obtained from another conversation of the same participant.

Self-play conversations. Synthetic conversations are generated by two frozen Voila instances that take turns speaking. For each conversation, the two instances receive different voices from 811 reference voices of real participants, and each of 13,354 opening utterances is generated up to five times. A conversation has 2–32 turns with a target length of 64 s, and we record the text, audio codes, and speaking states that each instance actually generates. Generated conversations then pass three checks. (i) The integrity check removes conversations with corrupted records or a truncated final utterance. (ii) The dialogue-quality check removes conversations without a single complete exchange (at least 4 s and more than six words per pair) and keeps a conversation only up to the point where it starts echoing the partner or repeating the same content. (iii) The motion-quality check compares the pseudo motion that a frozen UniLS (Chu et al., 2026) generates from both speakers' audio with criteria set on real training motion. If a role's lip–speech agreement (PLRS (Lee et al., 2025)) falls below a fixed threshold or its mouth and jaw motion leaves the state-specific ranges, its pseudo motion is regenerated up to three times, and we keep the first version that passes rather than the highest-scoring one. Of the 35,998 generated conversations, 28,122 pass (i) and (ii), and 25,179 of these (50,358 role streams) pass (iii).

Table 7: Comparison of synthetic data amounts. Every row adds 0, 30, 50, or 100% of the synthetic training conversations that pass the quality checks to the same full set of real conversations and is trained with cross-entropy only, without auxiliary terms, for the same number of epochs; rows with more synthetic data therefore also receive more updates. The 50% row is the reference model of Tables 3 and 4.

	Geometric accuracy (mm)				Upper face (10^{-5} m)		Lip sync		Dialogue level					
Synthetic	S-LVE↓	S-MHD↓	MOD↓	Lip-W↓	S-FDD↓	L-FDD↓	AVC↑	MTM (ms)↓	MSE↓	FD↓	P-FD↓	rPCC↓	F-MAE↓	Freeze gap↓
0 % (real only)	8.973	2.374	4.091	3.787	24.89	**25.89**	0.324	70.7	0.197	12.29	13.34	**0.115**	0.1553	0.0298
30 %	8.523	2.311	4.036	3.688	**24.21**	26.47	0.332	69.5	0.184	**11.31**	**12.31**	0.116	0.1506	0.0183
50 % (Ours)	**8.212**	**2.252**	**3.728**	**3.537**	25.83	28.34	0.330	69.0	**0.182**	11.66	12.62	0.116	**0.1501**	**0.0053**
100 %	8.397	2.282	3.854	3.796	24.64	26.60	**0.347**	**66.3**	0.186	11.74	12.73	0.123	0.1503	0.0249

The self-play conversations are split into training and validation sets by prompt group, a group of related opening prompts (851 groups; 23,793 training and 1,386 validation conversations). The final training set contains all real training role streams and all conversations generated from 3,319 of the 6,616 opening utterances in the synthetic training split (23,970 role streams). In the plane of the first two principal components, most synthetic frames lie within the region occupied by real motion in every state (Figure 4). On the first two principal components, the overlap between the two distributions (Bhattacharyya coefficient) is 0.92 for speaking, 0.90 for listening, and 0.84 for quiet, and the difference between their means is smaller than one standard deviation for speaking and listening. Synthetic conversations thus add the hidden states that Voila produces when it speaks on its own, while their motion stays close to real motion in this projection. We examine their contribution by varying only the amount of synthetic data while keeping all real conversations (Table 7). Adding synthetic conversations, which also adds updates, improves geometric errors (S-LVE, S-MHD, MOD), the speaking freeze gap, and dialogue-level distributions (FD, P-FD) over real data alone. In this cross-entropy-only comparison, however, upper-face motion amplitude (S-FDD, L-FDD) deviates most at 50%, which the final model compensates for with the auxiliary losses (Table 3(b)). We chose the ratio during development by comparing the four ratios, trained with the same settings, on the validation set. At 50%, F-MAE and freeze gap were lowest on the validation set (0.147 and 0.0002, compared with 0.156 and 0.041 without synthetic data), and both are also lowest at 50% on the test set (Table 7). Adding backbone self-play conversations to real conversations and generating their pseudo motion with a teacher model is also used in the concurrent FacePlex (Lim et al., 2026).

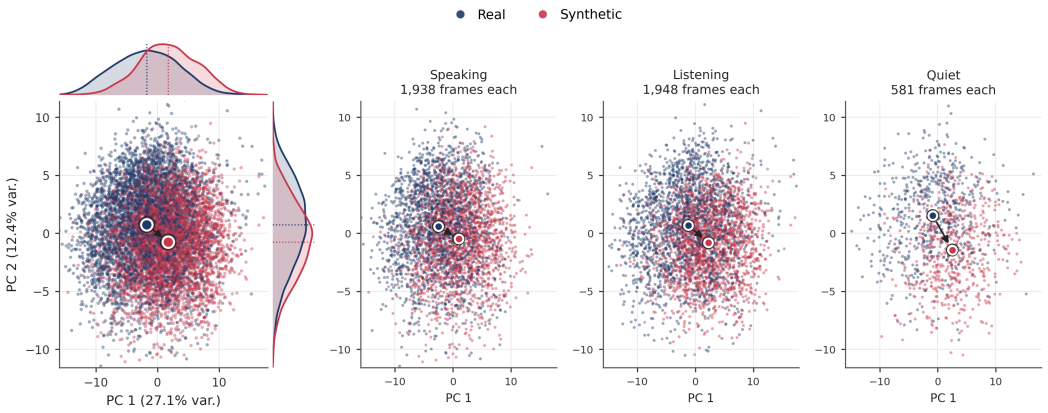

Figure 4: Joint frame-level PCA of real motion and synthetic pseudo motion (training split; 105-dimensional expression, jaw, and eye parameters).

D Training Details

Optimization and schedule. The predictor is trained for 996 updates (about 100k motion tokens per update) with AdamW (learning rate 10^{-4}, weight decay 0, gradient-norm clipping at 1.0), using a linear warm-up over the first 166 updates and cosine decay toward 10^{-5} over a planned 1,660 updates. The cross-entropy-only models in Section 4.4 are also trained for the same 996 updates. The final model with auxiliary terms starts training with $\mathcal{L}_{\mathrm{CE}}$ alone and, from update 665, at which validation S-FDD and L-FDD stop changing and begin to worsen, linearly increases the weights of the three auxiliary terms from 0 to their full values over 32 updates. We set the coefficients to $\lambda_{\mathrm{G}} = 0.05$, $\lambda_{\mathrm{V}} = 0.010$, and $\lambda_{\mathrm{S}} = 0.5$ by comparing models with different coefficient combinations on the validation set, while checking gradient magnitudes so that no auxiliary term overwhelms $\mathcal{L}_{\mathrm{CE}}$ or cancels another. $\mathcal{L}_{\mathrm{CE}}$ is averaged over the supervised tokens and residual stages of an update; $\mathcal{L}_{\mathrm{G}}$ and $\mathcal{L}_{\mathrm{V}}$ are summed over vertices and normalized by the numbers of frames and frame-to-frame differences; and $\mathcal{L}_{\mathrm{S}}$ is averaged over vertices and over speaking and listening, with role streams weighted by their length. Table 3(b) reports the contribution of each term. The auxiliary terms are computed from codes drawn at each depth-head stage with straight-through Gumbel-Softmax (Jang et al., 2017) at temperature 0.1, and each drawn code is the input to the next stage. Algorithm 1 summarizes one update. For $\mathcal{L}_{\mathrm{CE}}$, each depth-head stage receives the GT code prefix (teacher forcing), and memory receives, without gradient and without a separate greedy rollout, the self-predicted codes obtained by taking the stage-wise argmax of these teacher-forced logits. At inference, the depth head uses its own greedy code prefix. This leaves a residual training–inference mismatch in depth conditioning that can also propagate to temporal memory.

Auxiliary terms. Q_{pos} and Q_{vel} in Equation (6) are obtained from the mapping through the frozen decoder and FLAME2020 (zero head rotation and shape), linearized around the target latents, using all 5,023 FLAME vertices and their 25 Hz differences. With J denoting the Jacobian of this mapping at the target latents, the squared error of the reconstructed vertices is, to first order, $\|J\Delta\|^2 = \Delta^\top J^\top J \Delta$, so Q is the average of $J^\top J$ over training contexts. Separate matrices are used for each of the four frame positions reconstructed from a token, and the input $\Delta_{\le n}$ is the latent difference over the span that the decoder uses to reconstruct that frame (about 45 tokens), zero-padded before the start of a role stream. Each matrix is estimated by computing vector-Jacobian products $J^\top v$ with 16 random ± 1 vectors v for each of 1,024 contexts sampled from the training data and averaging their outer products $(J^\top v)(J^\top v)^\top$, and it is kept fixed during training. Let p and r denote the mean-subtracted predicted and target latent histories of a role stream. In the inner product defined by Q, the squared error then splits into magnitude and direction terms:

$$\|p - r\|_Q^2 = (a - c)^2 + 2ac\,(1 - \rho), \qquad a^2 = p^\top Q p, \quad c^2 = r^\top Q r, \quad \rho = \frac{p^\top Q r}{ac} \tag{8}$$

Decomposing the gradient of the squared error, $g = 2Q(p - r)$, as $g = 2\kappa Qp + g_\perp$ with $p^\top g_\perp = 0$ gives $\kappa = 1 - \rho c/a$. In the velocity term, we use $\tilde{g} = 2(1 - c/a)Qp + g_\perp$ instead of g. This replaces the magnitude derivative by $2(a - c)$, removing correlation-dependent shrinkage from this component. The loss value itself is unchanged. When the predicted amplitude is negligible ($a^2 \leq 10^{-12}(a^2 + c^2)$), the velocity term falls back to the squared error. The affine readout in $\mathcal{L}_\mathrm{S}$ is a fixed map from a 45-token history of predicted latents to the motion (mm) of 827 upper-face vertices, fitted on 256 training role streams to reproduce the output of the frozen decoder and FLAME for the same latents. For each role stream, $\mathcal{L}_\mathrm{S}$ computes per-vertex standard deviations separately over speaking and listening intervals and averages their squared differences from the same statistics of the target motion.

Algorithm 1 One training update of the facial motion predictor. At inference, the depth head uses its own greedy prefix, and no Gumbel sampling is performed.

Require: Hidden states A_t, T_t of the role streams, target codes m^{gt}, fixed codebooks E_q

for each role stream in the update **do**
  $Z \leftarrow$ no-history token
  for each segment b **do**
    for each token n in segment b **do** ▷ tokens in a segment share Z
      $h_n \leftarrow \mathrm{CrossAttn}(C_n, Z)$ ▷ Eq. (3)
      $\ell_{n,q} \leftarrow \mathrm{DepthHead}_q(h_n, m^{\mathrm{gt}}_{n,<q}),\ q = 1, \ldots, Q$ ▷ teacher-forced pass
      $\mathcal{L}_{\mathrm{CE}} \leftarrow \mathcal{L}_{\mathrm{CE}} - \sum_q \log \mathrm{softmax}(\ell_{n,q})[m^{\mathrm{gt}}_{n,q}]$
      $\hat{m}_{n,q} \leftarrow \arg\max \ell_{n,q}$ ▷ self-prediction for memory
      for $q = 1, \ldots, Q$ **do** ▷ Gumbel pass for auxiliary terms
        $\tilde{\ell}_{n,q} \leftarrow \mathrm{DepthHead}_q(h_n, \tilde{m}_{n,<q})$ ▷ drawn prefix
        $\tilde{m}_{n,q} \leftarrow$ straight-through Gumbel-Softmax of $\tilde{\ell}_{n,q}$ (temperature 0.1)
      end for
    end for
    $Z \leftarrow \mathrm{stopgrad}\big(\{\sum_q E_q[\hat{m}_{n,q}]\}_{n \in b}\big)$
  end for
  $\Delta \leftarrow \sum_q E_q[\tilde{m}_{\cdot,q}] - \sum_q E_q[m^{\mathrm{gt}}_{\cdot,q}]$
  Compute $\mathcal{L}_\mathrm{G}, \mathcal{L}_\mathrm{V}, \mathcal{L}_\mathrm{S}$ from Δ and $\sum_q E_q[\tilde{m}_{\cdot,q}]$ ▷ Eqs. (6) and (8)
end for
$\mathcal{L} \leftarrow \mathcal{L}_{\mathrm{CE}} + w(u)\,(\lambda_\mathrm{G}\mathcal{L}_\mathrm{G} + \lambda_\mathrm{V}\mathcal{L}_\mathrm{V} + \lambda_\mathrm{S}\mathcal{L}_\mathrm{S})$ ▷ $w(u)$: 0 before the auxiliary terms are enabled, then linear up to 1
Update θ with AdamW

E Evaluation Metrics

The metrics of Table 1 are computed in the same way in the other tables. Facial metrics compare FLAME2020 meshes with zero head rotation and shape, together with their 68 landmarks, at 25 Hz, and distances are in mm. As in the UniLS reference implementation (Chu et al., 2026), listening intervals comprise all frames in which the target speaker is not speaking according to the speech mask. Except for MTM, the metrics of Table 1 are computed per role stream and averaged over role streams, and S- and L- metrics are averaged only over role streams that contain the corresponding intervals.

With predicted and GT vertex positions $\hat{\mathbf{v}}_{t,k}$ and $\mathbf{v}_{t,k}$, and the sets of speaking and listening frames $\mathcal{F}_\mathrm{S}$ and $\mathcal{F}_\mathrm{L}$, the vertex metrics are:

$$\text{S-LVE} = \Big(\frac{1}{|\mathcal{F}_\mathrm{S}|}\sum_{t \in \mathcal{F}_\mathrm{S}} \max_{k \in \mathcal{V}_{\mathrm{lip}}} \|\hat{\mathbf{v}}_{t,k} - \mathbf{v}_{t,k}\|^2\Big)^{1/2} \tag{9}$$

$$\text{S-MHD} = \Big(\frac{1}{|\mathcal{F}_\mathrm{S}|\,|\mathcal{V}|}\sum_{t \in \mathcal{F}_\mathrm{S}}\sum_{k \in \mathcal{V}} \|\hat{\mathbf{v}}_{t,k} - \mathbf{v}_{t,k}\|^2\Big)^{1/2} \tag{10}$$

$$\mathrm{FDD}(\mathcal{F}) = \frac{1}{|\mathcal{V}_{\mathrm{up}}|}\sum_{k \in \mathcal{V}_{\mathrm{up}}} \Big|\,\mathrm{std}_{t \in \mathcal{F}}\left(\mathbf{1}^\top \hat{\mathbf{v}}_{t,k}\right) - \mathrm{std}_{t \in \mathcal{F}}\left(\mathbf{1}^\top \mathbf{v}_{t,k}\right)\Big| \tag{11}$$

Here, $\mathcal{V}$ is the set of all FLAME vertices, $\mathcal{V}_{\text{lip}}$ and $\mathcal{V}_{\text{up}}$ are the lip and upper-face vertices defined by UniLS, and $\mathbf{1}^\top \mathbf{v}$ is the sum of the three coordinates. S-FDD and L-FDD are FDD($\mathcal{F}_{\text{S}}$) and FDD($\mathcal{F}_{\text{L}}$), respectively, reported in units of 10^{-5} m. LVE and FDD were proposed in MeshTalk and CodeTalker (Richard et al., 2021; Xing et al., 2023), respectively, and S-MHD is the mean head distance of UniLS; all three are computed with the UniLS reference implementation (Chu et al., 2026).

MOD and AVC follow the mouth opening distance and velocity correlation of A^2-LLM (Hu et al., 2026). With the mouth opening o_t given by the mean distance of three upper and lower inner-lip landmark pairs and the mouth width w_t by the distance between the two mouth corners, MOD and Lip-W are the absolute errors of o_t and w_t averaged over all frames, and AVC is the Pearson correlation between the predicted and GT frame-to-frame changes $o_t - o_{t-1}$. MTM follows the public implementation of Lee et al. (2025). For each speaking turn (a continuous speaking interval in the speech mask), it finds the maxima (openings) and minima (closings) of the distance between the central upper- and lower-lip vertices after Gaussian smoothing ($\sigma = 2$ frames), aligns the predicted and GT curves with derivative dynamic time warping, and averages the time differences between matched extrema of the same type. These per-turn values are averaged over the turns with matched extrema.

Dialogue-level metrics are computed per role stream on the 100 FLAME expression coefficients by adapting the DualTalk implementation (Peng et al., 2025). With generated, GT, and partner GT coefficients $\hat{\boldsymbol{\psi}}_t$, $\boldsymbol{\psi}_t$, and $\boldsymbol{\psi}_t^{\text{P}}$, MSE is the mean element-wise squared error between $\hat{\boldsymbol{\psi}}_t$ and $\boldsymbol{\psi}_t$, and FD is the Fréchet distance between Gaussian approximations of the two frame distributions. P-FD applies the same computation to the concatenations $[\boldsymbol{\psi}_t^{\text{P}}; \hat{\boldsymbol{\psi}}_t]$ and $[\boldsymbol{\psi}_t^{\text{P}}; \boldsymbol{\psi}_t]$, and rPCC is $|\text{corr}(\boldsymbol{\psi}, \boldsymbol{\psi}^{\text{P}}) - \text{corr}(\hat{\boldsymbol{\psi}}, \boldsymbol{\psi}^{\text{P}})|$, with correlations computed after flattening all frames and coefficients into one vector.

In the streaming setting, UniLS and ARTalk regenerate their current 4 s chunk at every step with the audio not yet received set to zero, keep that step's four frames, and start the next chunk from the chunk generated once its audio was complete; ARTalk's released smoothing filter, which needs future frames, is not applied. DualTalk, whose layers are bidirectional, is rerun at every step on the conversation up to that step, with its waveform normalized using statistics of that prefix only.

First is the time from the start of the audio to the first motion frame. In the offline setting, it covers waiting for the input required by the released inference (a 4 s chunk for ARTalk and UniLS; the whole clip for DiffPoseTalk and DualTalk, measured on 16 s clips) and completing the computation; in the streaming setting, it covers receiving the first 160 ms and completing the first step. Step is the median computation time of the last 25 steps when each method's streaming inference runs step by step without pauses until the conversation reaches 64 or 128 s. HiddenFace's Step is measured with its kernel launches grouped into CUDA graphs, with outputs identical to eager execution. Without CUDA graphs, it is 23.0 and 22.9 ms at 64 and 128 s, respectively, still shorter than that of every baseline in the streaming setting, all of which run their released implementations as is.

F-MAE, freeze gap, deployed CE, and same-tuple in Table 4 (F-MAE and freeze gap also appear in Table 7) are computed by pooling frames or tokens over the whole test set rather than averaging per role stream. F-MAE computes the mean absolute error separately for the expression, jaw, and eye groups of the unnormalized FLAME parameters and averages the three. A freeze frame is a frame in which only the target speaker is speaking and the mean absolute frame-to-frame change of the 105 parameters falls below the 20th percentile of the same quantity in real motion; the freeze gap is the absolute difference between the predicted freeze rate and the real rate (0.2). Deployed CE is the cross-entropy of the GT codes averaged over the six stages, and same-tuple is the fraction of adjacent tokens in the generated sequence whose six codes are all identical.

F Streaming Timeline and Audio–Visual Synchronization

We run HiddenFace together with Voila in real time and measure when the hidden states, audio, and face become ready within each block. Because the released Voila audio tokenizer is not a streaming codec, we emulate streaming by re-encoding and decoding the most recent 1.28 s at every row. Voila's LLM and audio tokenizer run on separate GPUs, and HiddenFace runs on the same GPU as the audio

tokenizer. We feed the partner's speech in real time and measure a 160 s conversation that Voila generates in response.

When Voila starts a response, it generates several text tokens in succession, pausing audio generation for up to 1.6 s before processing the queued input. Because the audio itself is delayed during these intervals regardless of the facial branch, we define blocks in which Voila operates in real time as those in which the partner's audio input is not delayed by more than one row and Voila does not generate such a run of text tokens (849 of 1,000 blocks). We report the 796 of these blocks after the motion decoder's input window has filled, about 8.6 s into the conversation, as the steady state.

Table 8: Timeline of the 796 steady-state blocks (ms from the start of the block's first row interval; medians). The four row intervals end at 160 ms, and both the fourth row's audio and the block's face become ready after that. Offset is the face-ready time minus the time at which the fourth row's audio becomes playable.

Row in block	1	2	3	4
Row interval start	0	40	80	120
Hidden state computed	47.0	77.9	122.1	157.3
Audio playable	–	–	–	169.4
Face ready	–	–	–	165.0
Offset	–	–	–	−4.4

Table 8 gives the timeline of the 796 steady-state blocks. The four facial frames of a block become ready together right after the hidden state of its fourth row is computed. Audio and motion are therefore emitted per block: the audio of the first three rows is held until the face is ready, and each frame is played with its row's audio, so frames stay aligned with their audio by construction. If the face is ready after the fourth row's audio, the block waits for the face, delaying audio and motion equally. In the steady state, the face is ready a median of 4.4 ms before the fourth row's audio (95th percentile: 3.6 ms before) and later in only one of the 796 blocks, by 2.1 ms.

G Human Study Details

We create one clip from each of 30 self-play validation conversations not used for training. After reading the transcripts of all 376 candidate conversations, we select conversations in which the two speakers respond to each other on a single topic, and cut 2–10 complete turns in which both speakers talk (at least 4 s of agent speech and 3 s of user speech in total) at turn boundaries. No animation system's output is used in this selection. Because self-play conversations are split into training and validation sets by prompt group, no conversation from the same prompt group is used for training. Each clip shows the outputs of the five systems for the same conversation side by side in five numbered panels, whose order is randomized per clip (Figure 5). All systems are rendered with the same FLAME mean shape and camera, and the first panel indicates whether the agent or the user is speaking. Baselines follow their released inference procedures, as in the offline setting of Table 1: ARTalk and DiffPoseTalk, which cannot take both speakers' audio, receive only the agent's audio, and DualTalk receives zeros in place of partner motion. HiddenFace uses its 160 ms causal output as is.

After reading the consent form and instructions, raters view one practice clip, which is not analyzed, and then rate 10 assigned clips. The 30 clips are assigned evenly across raters, and the presentation order is shuffled for each rater. Questions can be answered only after a clip has been watched to the end, and clips can be replayed while answering. Raters answer the three questions on a 1–5 scale for each panel and select the best and worst panels (Table 9). At the end, one previously rated clip is shown again to check rating consistency, and this repeated rating is excluded from aggregation. Twenty-four raters provide ratings, 20 of whom rate all 10 clips, giving 208 ratings per system and question. Each mean in Table 2 pools these 208 ratings, and each interval resamples raters with replacement and recomputes the pooled mean (10,000 resamples). Table 10 reports the paired differences between HiddenFace and each baseline with the same bootstrap. Only the differences from DualTalk have confidence intervals that include 0; for the other three baselines, the intervals lie above 0 for all three questions (Section 4.3).

Table 9: Questions shown to raters (verbatim). The intermediate scale points are 2 Poor, 3 Fair, and 4 Good for all questions.

Item	Question	Scale
Lip sync (while the agent speaks)	While the agent is speaking (green label), how well do the mouth movements match the speech you hear?	1 Bad: mouth barely moves or is clearly out of sync; 5 Excellent: perfectly in sync
Listening behaviour (while the user speaks)	While the user is speaking (blue label), does the agent look like it is listening: attentive, with reactions that are well timed to the user's speech, neither frozen nor moving randomly?	1 Bad: frozen or random; 5 Excellent: attentive and well timed
Overall naturalness	Overall, how natural and convincing is this face as a conversational partner (expressions, head motion, timing)?	1 Bad; 5 Excellent
Best / worst	Which panel is the best overall? / Which panel is the worst overall?	One panel

Table 10: Mean score differences between HiddenFace and each baseline (HiddenFace − baseline) with rater-level bootstrap 95% confidence intervals.

Baseline	Lip sync	Listening	Overall
DiffPoseTalk	+1.30 [+0.88, +1.75]	+2.74 [+2.33, +3.09]	+2.00 [+1.61, +2.38]
ARTalk	+1.70 [+1.38, +2.03]	+1.30 [+0.83, +1.79]	+1.53 [+1.16, +1.93]
DualTalk	+0.23 [−0.12, +0.60]	+0.26 [−0.01, +0.56]	+0.31 [−0.05, +0.67]
UniLS	+0.93 [+0.64, +1.22]	+0.67 [+0.42, +0.94]	+0.80 [+0.53, +1.07]

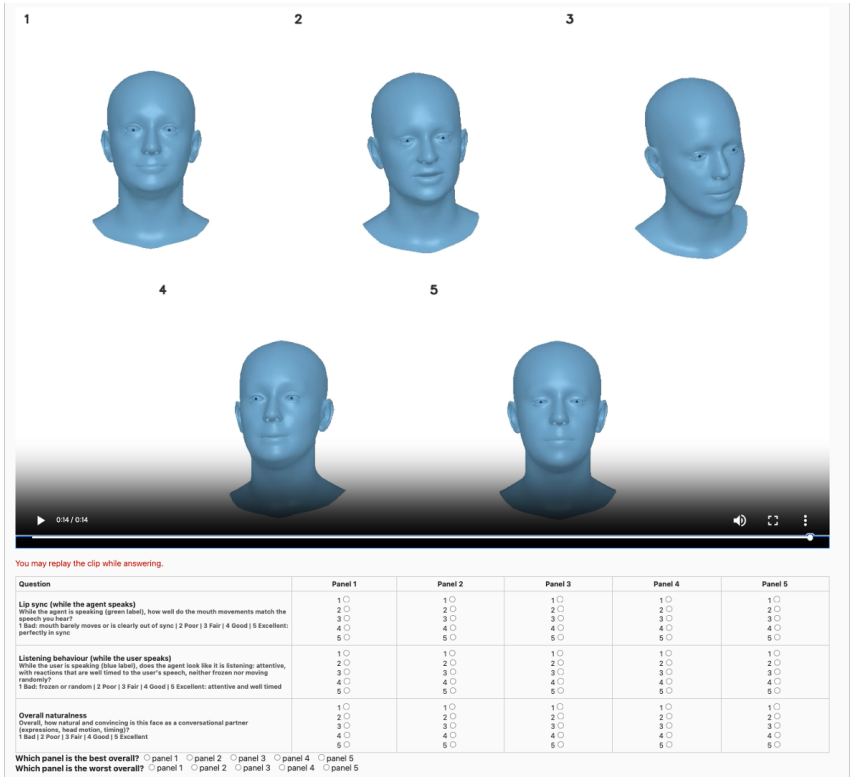

Figure 5: Rating screen. Below the clip, raters answer the three questions for each panel and select the best and worst panels.