EMBODIED INTERACTION POLICY: A UNIFIED MODEL FOR REAL-TIME HUMAN–ROBOT INTERACTION

Anonymous authors
Paper under double-blind review

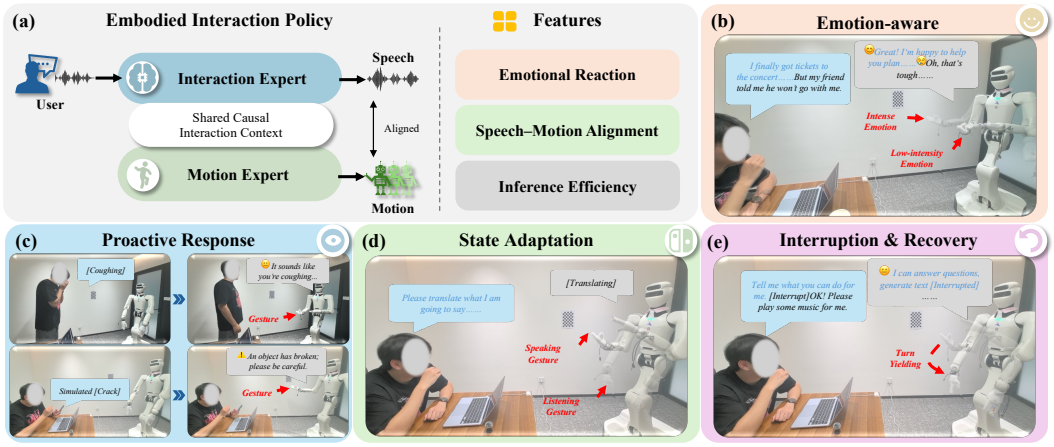

Figure 1: (a) We propose Embodied Interaction Policy (EIP), which is a unified streaming architecture that couples interaction and motion experts for speech–motion alignment. EIP enables diverse interaction scenarios, including (b) emotion-aware gesture generation, (c) proactive response, (d) seamless state adaptation, and (e) fast reaction to user interruption and task recovery.

ABSTRACT

Real-time human-robot social interaction requires continuous coordination between speech and physical behavior. Existing systems often separate interaction agents from motion generation, complicating coordination as conversational states change. We propose **Embodied Interaction Policy (EIP)**, a unified streaming framework that couples an Audio-Interaction expert with a motion expert through layer-wise joint attention. This allows motion generation to directly access the evolving interaction context. A chunk-wise causal mask prevents access to future context, while a vocal token encodes speaking state and motion intensity derived from the response emotion. We construct **Embodied Interaction Dataset (EI-Data)** from co-speech and dyadic interaction data and train our model in two stages. EIP supports proactive responses to salient acoustic events, state adaptation between listening and speaking, and interruption-aware turn-taking. Quantitative evaluations demonstrate improved motion quality over representative gesture-generation baselines. Compared with RoboGesture, EIP achieves higher interaction success rates and reduces warm-start latency from speech input to the first generated motion chunk from 3.78 s to 1.35 s. Physical deployment and user studies further demonstrate improved motion naturalness and interaction comfort.

1 INTRODUCTION

Natural human–robot social interaction requires a robot to continuously determine what to say, when to speak, and how to move during interaction (Zhang et al., 2026; Wang et al., 2026). Beyond generating appropriate verbal responses, the robot must exhibit natural and smoothly transitioning physical behaviors that remain consistent with the evolving conversational context. In particular, the robot should maintain smooth and expressive motion as it switches between interaction states,

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.

such as listening, speaking, and responding to user interruptions (Skantze, 2021). Such nonverbal behavior is closely coupled with conversational timing: decisions about when to respond, remain silent, or yield to an interruption directly affect how physical movements should initiate, sustain, and transition. Achieving this tight temporal coordination between interaction and motion remains a key challenge in embodied social robotics.

Existing co-speech gesture generation methods have shown great success in generating realistic and diverse body, hand, and facial motions aligned with speech audio (Yi et al., 2023; Liu et al., 2024; Mughal et al., 2026). However, these methods typically generate motion in a human-centric representation and in digital space (*e.g.*, SMPL-X (Pavlakos et al., 2019)). Using retargeting post-processing method Araujo et al. (2026) can be a solution for adapting to robot. However, such retargeting may introduce kinematic discrepancies that compromise the expressiveness and temporal fidelity of the original motion. To address this embodiment gap, recent approaches propose to generate robot motion for physical deployment. ProAct (Zhang et al., 2026) combines proactive reasoning with continuous robot motion generation, while retaining turn-based verbal responses. RoboGesture (Wang et al., 2026) integrates an Automatic Speech Recognition (ASR)–Large Language Model (LLM)–Text-to-Speech (TTS) pipeline with a streaming speech-to-motion generator for humanoid deployment.

Although these systems enable embodied interaction to some extent, they largely follow a turn-based interaction paradigm, where the robot waits for a user turn to complete before generating a verbal response and the corresponding motion. This design couples interaction responsiveness with the latency of the upstream ASR–LLM–TTS pipeline, limiting rapid responses to user utterances and interruptions (Fig. 2, left). More recent duplex interaction models, such as Audio Interaction Model (AIM) (Xie et al., 2026), remove this constraint by continuously processing the ongoing audio stream and supporting response generation without waiting for a complete turn to finish. However, timely interaction responses alone do not guarantee natural physical behavior. As the interaction state changes continuously, the robot must also smoothly initiate, sustain, and transition its physical behavior accordingly, rather than switching abruptly between discrete motion states. How to translate such evolving interaction context into continuous, expressive, and temporally coherent robot motion remains unexplored.

Our key observation is that the internal representations of a pretrained interaction model already encode rich, continuously evolving cues about the ongoing interaction, and can therefore serve as a natural context for robot motion generation. Building on this insight, we propose **Embodied Interaction Policy (EIP)**, a unified streaming framework that grounds such interaction representations into continuous physical motion. EIP introduces a dedicated motion expert that directly consumes the evolving interaction context through layer-wise interaction–motion coupling, allowing robot motion to be generated in lockstep with conversational processing. To preserve causality during streaming generation, we introduce a chunk-wise causal attention mask that restricts motion generation to interaction information available up to the current time step. Finally, we introduce a vocal token that explicitly encodes the robot's speaking state and response emotion, enabling more state-appropriate

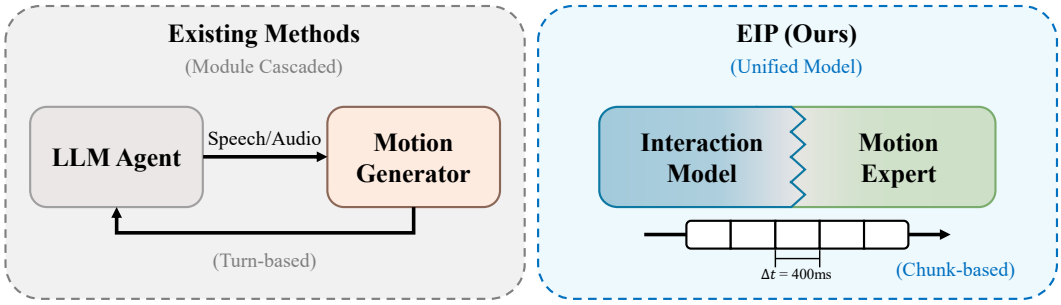

Figure 2: **Comparison of embodied interaction modeling paradigms.** Existing embodied interaction methods typically follow a cascaded, turn-based paradigm, in which the interaction model and motion expert are connected through an intermediate representation. In contrast, our approach unifies the interaction model and motion expert within a single architecture and performs continuous chunk-based processing at a 400-ms temporal granularity.

and expressive motion. Furthermore, to learn such policies and address the lack of systematic evaluation for continuous embodied interaction, we construct **Embodied Interaction Dataset (EI-Data)** by carefully curating data from existing co-speech and dyadic interaction datasets.

Our contributions are threefold:

- We reveal that the hidden states of interaction experts can effectively serve as interaction context for robot motion generation. Building on this insight, we develop EIP, a unified streaming framework that couples interaction and motion experts through layer-wise attention to coordinate streaming speech and continuous robot motion.
- We introduce a chunk-wise causal mask to preserve causality in streaming motion generation, together with a vocal token for state- and emotion-aware motion control.
- We construct EI-Data for training and evaluating embodied interaction, and deploy EIP on the Astribot S1. Quantitative experiments and user studies validate motion quality, interaction success, response latency, and user experience.

2 Embodied Interaction Policy

2.1 Overview

As illustrated in Fig. 3, EIP is a unified streaming policy that couples a frozen interaction expert with a compact motion expert. The interaction model continuously processes human audio, provides rich context, and determines whether the robot should respond and how the robot responds, while the motion expert generates the corresponding robot behavior. Following (Xie et al., 2026), both experts operate on a shared interaction clock of $\Delta = 400$ ms. At interaction step t, EIP takes the latest human-audio chunk $\mathbf{x}_t$, the robot state $\mathbf{q}_t$, and the accumulated context $\mathbf{h}_{t-1}$, and produces synchronized speech and motion chunks for the next interval:

$$(\hat{\mathbf{s}}_t, \hat{\mathbf{a}}_t) = \pi_{\text{EIP}}(\mathbf{x}_t, \mathbf{q}_t, \mathbf{h}_{t-1}). \tag{1}$$

Here, $\hat{\mathbf{s}}_t \in \mathbb{R}^{\Delta \cdot r_s}$ denotes a 400-ms speech chunk, and $\hat{\mathbf{a}}_t \in \mathbb{R}^{\Delta \cdot r_a \times N}$ denotes the robot motion chunk. Here r_s, r_a, N are the sample rate of audio, the motion frequency, and the dimension of action space, respectively.

EIP is a unified model through layer-wise joint attention between the interaction and motion experts, allowing motion generation to access the evolving interaction context directly. To preserve causality, a chunk-wise attention mask blocks future interaction context and motion information from other chunks. We further introduce a vocal token that explicitly encodes the robot's speaking state and emotion-derived motion intensity, thereby aligning physical behavior with verbal output. Next, we elaborate on the design and implementation of our method.

2.2 Unified Interaction–Motion Architecture

Audio interaction expert. We initialize the interaction expert with the pre-trained Audio-Interaction model (Xie et al., 2026), which follows the perceive–decide–respond paradigm. Given the current human-audio chunk $\mathbf{x}_t$, it updates the interaction context and predicts whether to respond and how to respond:

$$\big(d_t\big) = F_{\text{dec}}(\mathbf{h}_t), \qquad \text{here } \mathbf{h}_t = F_{\text{per}}\big(\mathbf{h}_{t-1}, E_{\text{aud}}(\mathbf{x}_t)\big), \tag{2}$$

where $d_t \in \{\langle\text{silent}\rangle, \langle\text{response}\rangle\}$ is the silence–response decision, E_{aud} denotes the audio encoder, F_{per} updates the interaction context using the current audio input, and F_{dec} maps the updated context h_t to a silence–response decision. When $d_t = \langle\text{silent}\rangle$, the output is silent, *i.e.*, $\hat{\mathbf{s}}_t = \varnothing$, otherwise, the response module F_{resp} generates the response audio.

$$\hat{\mathbf{s}}_t = \begin{cases} \varnothing, & d_t = \langle\text{silent}\rangle, \\ F_{\text{resp}}(\mathbf{h}_t), & d_t = \langle\text{response}\rangle. \end{cases} \tag{3}$$

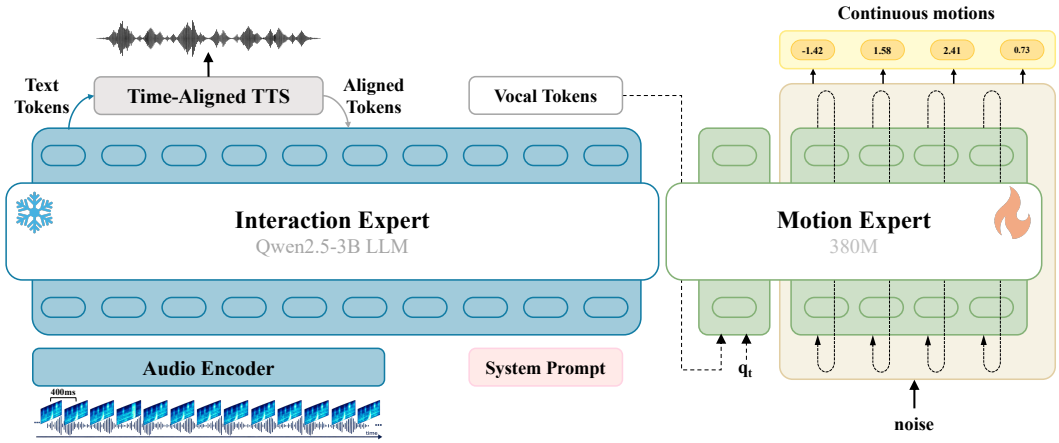

Figure 3: **Overview of the EIP framework.** EIP consists of a pretrained interaction expert that processes streaming audio and models the evolving interaction context, together with a motion expert that generates continuous and synchronized robot motions.

Motion expert. Inspired by the design of π_0(Black et al., 2024), we design a flow-matching based expert for motion generation. For each interaction step t, the motion expert receives the current robot state $\mathbf{q}_t$, a vocal condition $\mathbf{v}_t$ (which will be introduced in Eq. (8)), and a noisy motion trajectory $\mathbf{a}_t^\tau$. Firstly, the robot state and noisy motion are projected into the motion-expert hidden space as

$$\mathbf{Q}_t = \mathcal{E}_{\text{state}}(\mathbf{q}_t),\ \mathbf{A}_t^\tau = \mathcal{E}_{\text{act}}\left(\mathbf{a}_t^\tau, \mathcal{E}_\tau(\tau)\right). \tag{4}$$

Here, $\mathbf{q}_t, \mathbf{a}_t^\tau \in \mathbb{R}^{\Delta \times r_a \times N}$ are the robot-state sequence and noisy motion trajectory for interaction step t, respectively. $\mathcal{E}_{\text{state}}$ is a state encoder, and position encoder $\mathcal{E}_\tau$ encodes the flow time τ, which is combined with the noisy motion by a fusion block $\mathcal{E}_{\text{act}}$. The resulting state and motion token block $\mathbf{Q}_t, \mathbf{A}_t^\tau \in \mathbb{R}^{\Delta \times r_a \times d_m}$, where d_m is the token dimension, form the motion chunk together with the vocal token $\mathbf{v}_t$:

$$\mathbf{Z}_t^{(0)} = [\mathbf{Q}_t; \mathbf{v}_t; \mathbf{A}_t^\tau], \tag{5}$$

where $[\cdots]$ denotes the concatenate operation.

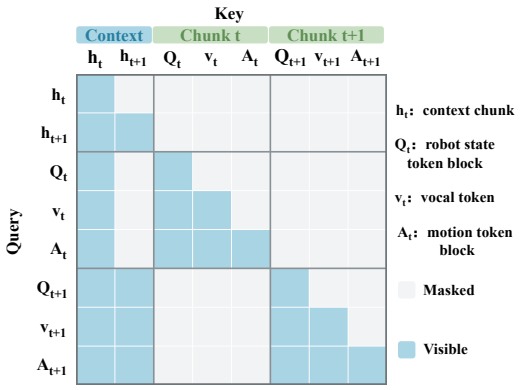

Figure 4: **Chunk-wise causal interaction–motion attention.** Blue entries indicate visible attention, whereas gray entries are masked. Interaction context follows causal attention. Each motion chunk accesses interaction context only up to its current time step and follows the within-chunk dependency $\mathbf{Q}_t \to \mathbf{v}_t \to \mathbf{A}_t$, with tokens within each block remaining mutually visible. Cross-chunk motion attention is masked.

Based on $\mathbf{Z}_t^{(0)}$ and context $\mathbf{h}_t$ from the audio interaction expert, we generate motion by integrating the learned motion expert net $\mathbf{v}_\theta$ from $\tau = 0$ to $\tau = 1$, starting with random noise $\mathbf{a}_t^0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. We use the forward Euler integration rule:

$$\mathbf{a}_t^{\tau+\delta} = \mathbf{a}_t^\tau + \delta \mathbf{v}_\theta(\mathbf{Z}_t^{(0)}, \mathbf{h}_t). \tag{6}$$

Layer-wise interaction–motion coupling. Following the Mixture-of-Transformers (MoT) architecture (Zhou et al., 2025; Black et al., 2024), EIP couples a frozen interaction expert with a trainable motion expert through layer-wise joint attention. To preserve chunk-wise causality, we employ the structured attention mask illustrated in Fig. 4. Interaction context tokens follow standard causal attention, while the state tokens $\mathbf{Q}_t$, vocal token $\mathbf{v}_t$, and motion tokens $\mathbf{A}_t$ can only access the interaction context available up to chunk t. Within each chunk, attention follows $\mathbf{Q}_t \to \mathbf{v}_t \to \mathbf{A}_t$, with tokens within the same block remaining mutually visible. Cross-chunk motion attention and access

to future interaction context are masked. The rationale for the token ordering and attention visibility is detailed in Appendix D.

Vocal token conditioning. The vocal token combines a speaking-state token with an emotion token. When speaking, we obtain a semantic emotion tag e_t from the interaction expert, covering six categories: `happy`, `sad`, `angry`, `surprise`, `normal`, and `urgent`. We map these tags to a compact motion-intensity condition m_t:

$$m_t = \begin{cases} \texttt{none}, & d_t = \langle \text{silent} \rangle, \\ \texttt{low}, & e_t \in \{\texttt{sad}, \texttt{normal}\}, \\ \texttt{high}, & e_t \in \{\texttt{happy}, \texttt{angry}, \texttt{surprise}, \texttt{urgent}\}. \end{cases} \tag{7}$$

The resulting vocal token is

$$\mathbf{v}_t = \mathcal{E}_{\text{spk}}(d_t) + \mathcal{E}_{\text{emo}}(m_t), \tag{8}$$

where $\mathcal{E}_{\text{spk}}$ and $\mathcal{E}_{\text{emo}}$ denote the speaking-state and emotion-condition embeddings, respectively. The motion-intensity condition m_t additionally modulates the motion-token features through FiLM.

2.3 Two-Stage Training

Jointly learning state adaptation and emotion-aware motion can lead to mode collapse. Therefore, we train EIP in two stages. Stage I first learns motion patterns across speaking, listening, and idle states, using only the speaking-state token of the vocal condition. Stage II then introduces emotion-aware motion intensity by conditioning the motion expert on the full vocal token ($\mathbf{v}_t$) defined in Eq. (8).

Stage I: State adaptation motion learning. We train the motion expert on the training split of EI-Data to learn continuous robot motion across speaking, listening and idle states. The interaction expert remains frozen throughout training.

For a ground-truth motion chunk $\mathbf{a}_t$, we sample Gaussian noise $\boldsymbol{\epsilon}_t$ and a flow timestep $\tau \in [0, 1]$, and construct the interpolated motion:

$$\mathbf{a}_t^\tau = (1-\tau)\boldsymbol{\epsilon}_t + \tau \mathbf{a}_t, \qquad \tau \sim \mathcal{U}(0,1), \quad \boldsymbol{\epsilon}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \tag{9}$$

where $\mathbf{a}_t$ is the ground-truth motion chunk and $\mathbf{a}_t^\tau$ is its noisy interpolation at flow timestep τ. For each behavioral phase $r \in \{\text{spk}, \text{lis}, \text{idle}\}$, we optimize

$$\mathcal{L}_r = \mathbb{E}\left[\frac{1}{|\mathcal{M}_r|} \sum_{t \in \mathcal{M}_r} \left\| \mathbf{v}_\theta\left(\mathbf{a}_t^\tau, \mathbf{q}_t, \tau \mid \mathbf{h}_t, \mathbf{v}_t\right) - (\mathbf{a}_t - \boldsymbol{\epsilon}_t) \right\|_2^2 \right]. \tag{10}$$

where $\mathcal{M}_r$ selects the motion chunks in phase r. We combine three losses: the co-speech speaking loss $\mathcal{L}_{\text{spk}}$ learns speech-correlated gestures; the Seamless listening loss $\mathcal{L}_{\text{lis}}$ exposes the model to diverse non-speaking reactions while the user is speaking; and the co-speech idle loss $\mathcal{L}_{\text{idle}}$ encourages stable motion when neither participant is speaking. The Stage-I objective is:

$$\mathcal{L}_{\text{I}} = \lambda_1 \mathcal{L}_{\text{spk}} + \lambda_2 \mathcal{L}_{\text{lis}} + \lambda_3 \mathcal{L}_{\text{idle}}. \tag{11}$$

Stage II: Emotion-aware motion learning. Starting from the Stage-I checkpoint, FiLM conditioning is introduced into the final four motion-expert blocks, while the final eight motion-expert blocks and the motion head are jointly fine-tuned. We curate and balance naturally occurring low- and high-intensity motion examples from the co-speech datasets to supervise this adaptation. We optimize the flow-matching objective over the speaking motion chunks:

$$\mathcal{L}_{\text{II}} = \mathbb{E}_{m_t \sim \mathcal{U}(\{\text{low}, \text{high}\})}\left[\mathcal{L}_{\text{spk}}(\theta_{\text{II}}; m_t)\right]. \tag{12}$$

where θ_{II} denotes the FiLM modules and the selected motion-expert parameters adapted in Stage II. During inference, the semantic emotion tag is mapped to the corresponding motion-intensity condition according to Eq. (7), enabling emotion-aware gesture intensity.

2.4 PLAYBACK-SYNCHRONIZED STREAMING INFERENCE

During training, response tokens are teacher-forced using ground-truth speech timestamps. During inference, we obtain this alignment through incremental speech synthesis. The interaction expert decodes a response into punctuation-delimited phrases, each of which is submitted to TTS upon completion. Speech and motion output can begin once the first playable audio chunk is available, while subsequent phrases are decoded and synthesized in the background.

To preserve causality, we maintain separate decoding and committed KV caches that share the frozen interaction-expert parameters. The decoding cache supports response generation ahead of playback, whereas the committed cache receives response tokens only as their corresponding audio intervals are played. The motion expert accesses only the committed cache, preventing future response text from entering its interaction context. At step t, the committed context, robot state, and vocal condition are used to generate the next motion chunk $\hat{a}_t$, which is dispatched together with the corresponding speech chunk $\hat{s}_t$ on the shared 400-ms timeline. Algorithm 1 in the Appendix C details the incremental generation, temporal alignment, and interruption handling.

3 EXPERIMENTS

We conduct empirical studies to justify three central claims: (i) EIP supports natural and expressive human–robot interaction by coordinating speech with continuous, state- and emotion-aware motion; (ii) its streaming motion generation improves motion realism, speech alignment, and diversity over representative gesture-generation baselines; and (iii) EIP achieves higher interaction success rates and lower response latency than a cascaded interaction pipeline across proactive response, interruption-aware turn-taking, and emotion-aware motion scenarios.

3.1 EXPERIMENTAL SETUP

3.1.1 DATASETS PROCESSING

We construct **EI-Data** by combining co-speech data from **BEAT2** (Liu et al., 2024), **GUAVA** (Zhang et al., 2025), and **SHOW** (Yi et al., 2023) with dyadic interaction data from **Seamless Interaction** (Agrawal et al., 2025). Co-speech recordings are augmented with synthesized user turns and fixed listening segments, while dyadic recordings preserve natural turn-taking and listening behavior. We retarget motions to Astribot S1, filter motion artifacts, and annotate chunk-level interaction states to provide supervision across speaking, listening and idle. The resulting training set contains about 60,000 interaction episodes of 20 seconds each. For testing, we select 20 high-quality episodes based on clear distinctions between speaking and listening behaviors, motion diversity, and consistency between motion and the expressed emotion, while ensuring that the training and test sets are strictly disjoint at the episode level. Detailed data construction, selection criteria, and preprocessing procedures are provided in the Appendix B.

3.1.2 IMPLEMENTATION & DEPLOYMENT DETAILS

The interaction expert is initialized from the pre-trained Audio-Interaction model based on Qwen2.5-3B and remains frozen throughout training. The motion expert consists of 36 Transformer blocks with 16 attention heads, a hidden dimension of $d_m = 768$, and an intermediate dimension of 3072, totaling approximately 380M parameters. We use $\Delta = 0.4$s and $r_a = 30$Hz, giving 12 frames per motion chunk, with $N = 20$ for Astribot S1 and $N = 165$ for SMPL-X skeleton. Input audio is sampled at 16 kHz, and output speech at $r_s = 24\,\text{kHz}$.

Stage I trains the motion expert for 15K optimization steps using AdamW with a global batch size of 32. The loss weights are set to $\lambda_1 = 1$, $\lambda_2 = 2$, and $\lambda_3 = 1$. A sensitivity analysis of these loss weights is provided in Appendix E. The learning rate is warmed up for 500 steps to 1×10^{-4} and then cosine-decayed to 1×10^{-5}, with a weight decay of 0.01 and gradient clipping at 1.0.

Stage II starts from the Stage-I checkpoint and introduces FiLM-based intensity conditioning in the final four motion-expert blocks. The final eight motion-expert blocks and the motion head are fine-tuned for 1K steps with a global batch size of 16. Newly introduced conditioning parameters use a peak learning rate of 1×10^{-4}, while the unfrozen pre-trained parameters use a $0.02\times$ learning-rate

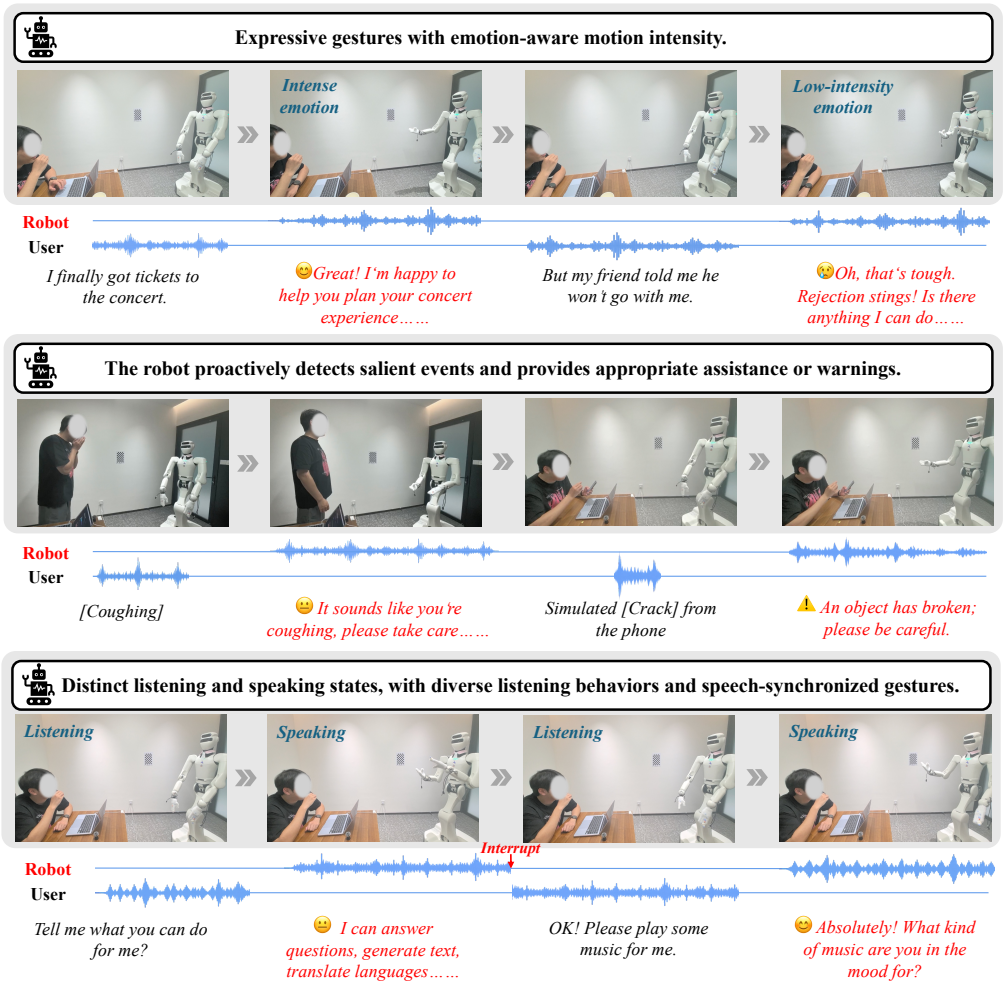

Figure 5: **Qualitative results of EIP on the physical Astribot S1.** Top: emotion-aware gesture intensity. Middle: proactive responses to salient acoustic events. Bottom: interruption-aware turn-taking with distinct listening behaviors and speech-synchronized gestures.

scale. The learning rate is warmed up for 100 steps and then follows a cosine schedule. Both stages are trained on eight NVIDIA RTX A6000 GPUs. During inference, each motion chunk is generated using 10 Euler steps.

We deploy the model on an Astribot S1, with online inference running on an NVIDIA RTX 5090. The generated 30-Hz motions and synchronized speech are transmitted in 400-ms chunks to the safety gate. For collision avoidance, we apply a filter based on model predictive control (MPC) to refine the generated trajectories. A Ruckig-based online trajectory generator (OTG) further enforces velocity, acceleration, and jerk limits, producing smooth commands at 50 Hz.

3.2 Evaluation

3.2.1 Qualitative Results

Fig. 5 shows EIP's behavior during real-world interactions on the Astribot S1 across three representative scenarios. EIP adapts its motion intensity to the emotion of the generated response, producing more expansive gestures under high-intensity conditioning and subtler movements under low-intensity conditioning. It also proactively responds to salient acoustic events, such as coughing and the sound of an object breaking, with relevant verbal responses and accompanying gestures. During user interruptions, it transitions from speaking to listening and subsequently responds to the updated request, exhibiting distinct listening behaviors and speech-synchronized gestures. These

Table 1: Observer-based user study results for different training stages. All scores are reported on a seven-point Likert scale.

Method	Naturalness ↑	State Adaptation ↑	Listening Diversity ↑	Emotional Expressiveness ↑	Average ↑
a) Vanilla	2.75	2.75	3.56	3.38	3.11
b) +Stage I	5.50	6.19	**4.94**	4.25	5.22
c) +Stage II (Final)	**6.56**	**6.44**	4.88	**6.06**	**5.99**

examples illustrate coordinated verbal and non-verbal behavior throughout an interaction. Further visualizations of EIP's results in simulation are provided in Appendix F (Fig. 8).

3.2.2 Quantitative User Study

We evaluate EIP from two complementary perspectives: observer-based evaluation of offline-generated videos and participant-based evaluation with the physical robot. In the observer-based study, 15 participants rated three configurations on a seven-point Likert scale: **Vanilla**, which uses the same motion backbone and training data, is conditioned only on the interaction context h_t and trained with the standard flow-matching objective, without vocal-token conditioning, phase-specific supervision; **+Stage I**, which adds phase-specific supervision for speaking, listening, and idle states; and **+Stage II (Final)**, which further introduces FiLM-based emotion-intensity conditioning. We evaluate Motion Naturalness, State Adaptation, Listening Diversity, and Emotional Expressiveness, measuring motion realism, state-appropriate behavior, listening-gesture variety, and emotion-motion alignment, respectively. In the physical-robot study, 12 participants rated Motion Naturalness, Interaction Comfort, and Willingness to Re-engage.

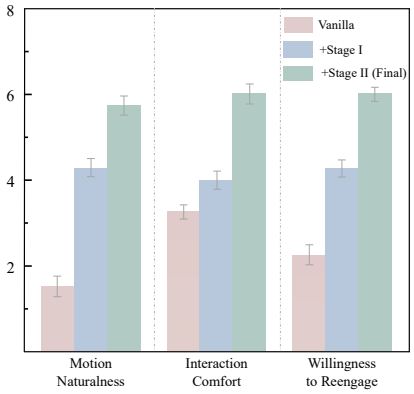

Figure 6: Participant-based user study results on the physical robot. Twelve participants rated the Vanilla, +Stage I, and +Stage II (Final) configurations on a seven-point Likert scale using Motion Naturalness, Interaction Comfort, and Willingness to Re-engage. Bars show mean ratings, and error bars indicate standard deviations.

As shown in Table 1, Stage I substantially improves the average score from 3.11 to 5.22, particularly for Motion Naturalness and State Adaptation. Stage II further increases the average to 5.99 and improves Emotional Expressiveness from 4.25 to 6.06, while largely preserving the interaction behaviors learned in Stage I. Figure 6 shows consistent improvements in the physical setting, where the final model achieves the highest scores on all three metrics. These results indicate that the two-stage training strategy improves both motion quality and the overall comfort and engagement of human–robot interaction.

3.2.3 Quantitative Streaming Motion Evaluation

We compare EIP with MIBURI (Mughal et al., 2026), EMAGE (Liu et al., 2024), Photoreal (Ng et al., 2024), and RoboGesture (Wang et al., 2026) on the EI-Data test set using Fréchet Gesture Distance (FGD), Beat Consistency (BC), and Diversity (Div). These baselines generate motion conditioned on reference target-speaker speech. To isolate motion generation from response-generation errors, we condition EIP on user audio and teacher-forced target text for each predicted motion chunk. As shown in Table 2, EIP achieves the best results across all metrics on both skeletons. Compared with the strongest baseline on each skeleton, EIP reduces FGD from 74.51 to 43.72 on S1 and from 25.18 to 5.97 on SMPL-X, indicating a closer match to the reference motion distribution. The S1 ablations further show consistent improvements across all three metrics from Vanilla to +Stage I and Final, supporting our two-stage training strategy. **Ours (w/o $\mathbf{h}_t$)** blocks $\mathbf{Q}_t, \mathbf{v}_t$ and $\mathbf{A}_t^\tau$ from attending to $\mathbf{h}_t$; its poorer performance relative to **Ours (Final)** supports the effectiveness of interaction-expert hidden states as context for robot motion generation.

Table 2: Quantitative results on the S1 and SMPL-X skeletons. For baseline methods, * denotes training on our co-speech data in the target skeleton; otherwise, generated motions are directly retargeted to the target skeleton for evaluation.

	FGD↓		BC→		Div↑	
System	S1	SMPL-X	S1	SMPL-X	S1	SMPL-X
GT	–	–	0.780	0.685	1.118	0.499
MIBURI	137.74	25.25	0.666	0.487	0.910	0.397
EMAGE	174.39	32.21	0.689	0.485	0.912	0.468
Photoreal	138.03	30.76	0.665	0.463	0.927	0.469
RoboGesture	198.71	39.67	0.513	0.356	0.936	0.447
MIBURI*	74.51	26.22	0.697	0.529	1.007	0.471
RoboGesture*	92.99	25.18	0.654	0.454	0.967	0.400
Vanilla	126.84	-	0.601	-	0.948	-
Ours (w/o $\mathbf{h}_t$)	90.27	-	0.677	-	0.931	-
+Stage I	69.33	-	0.703	-	0.981	-
Ours (Final)	**43.72**	**5.97**	**0.744**	**0.547**	**1.014**	**0.483**

Table 3: Success rates across three interaction scenarios with 20 test cases per scenario. Latency is measured under warm-start conditions from speech input to the generation of the first motion chunk. RTF is calculated as the end-to-end processing time divided by the generated motion duration; lower values indicate faster generation. All metrics are evaluated on an NVIDIA RTX A6000 GPU.

Method	**Proactive Event Response**	**Interruption-Aware Turn Taking**	**Emotion-Aware Motion**	**Average Success (%) ↑**	**Latency (s) ↓**	**RTF ↓**
RoboGesture	0	15	40	18.3	3.78	0.821
Ours	**100**	**100**	**100**	**100**	**1.35**	**0.804**

3.2.4 Quantitative Comparison of Interaction Success Rate and Latency

We compare EIP with RoboGesture across three interaction scenarios: proactive response, interruption-aware turn-taking, and emotion-aware motion, using 20 AI-generated speech samples per scenario. A trial is considered successful if the robot responds to the input event in proactive response or stops its ongoing response upon a user interruption in interruption-aware turn-taking. For emotion-aware motion, success is judged subjectively based on whether the generated motion is consistent with the emotion conveyed by the speech. As shown in Table 3, EIP achieves a 100% success rate in all three scenarios, compared with 18.3% on average for RoboGesture. Under warm-start conditions, EIP also reduces the latency from speech input to the first generated motion chunk from 3.78 s to 1.35 s. Moreover, EIP achieves a lower RTF of 0.804 compared with 0.821 for RoboGesture, indicating slightly faster end-to-end generation relative to the generated motion duration.

4 Conclusion

We presented Embodied Interaction Policy (EIP), a unified streaming framework that couples interaction modeling and motion generation on a shared 400-ms timeline. With a two-stage training strategy, EIP enables proactive responses and interruption-aware interaction, with its physical behavior adapting to the robot's conversational state and response emotion. We also introduced EI-Data to support policy learning and evaluate motion quality. Quantitative evaluations, user studies, and deployment on the Astribot S1 demonstrate improved motion quality, higher interaction success rates, reduced response latency, and greater interaction comfort and engagement. Despite these promising results, EIP currently relies on audio to perceive interaction partners, limiting its ability to interpret visual cues such as gaze, facial expressions, and body posture. Future work will incorporate visual perception to capture these non-verbal cues and support more context-aware embodied interaction.

AI USE STATEMENT

Generative AI tools were used to assist with language editing and coding-related tasks. Specifically, they were used to improve the clarity, readability, and grammatical quality of the manuscript, as well as to assist with code writing, editing, debugging, and minor implementation tasks. Generative AI was not used to formulate research hypotheses, develop the proposed methodology, generate or interpret experimental results, or derive scientific conclusions. All AI-assisted text and code were carefully reviewed, revised, and verified by the authors. The authors take full responsibility for all claims, results, text, code, and artifacts presented in this work.

ETHICS STATEMENT

This work develops a real-time embodied interaction system that continuously processes human speech and generates both verbal responses and robot motion. As such systems become more natural and responsive, they may inherit harmful, biased, or inappropriate behaviors from pretrained models and training data, while physical deployment introduces additional safety risks associated with generated motion. We therefore emphasize the importance of appropriate content safeguards, motion-safety constraints, and human oversight in real-world deployments. Continuous audio interaction may also raise privacy concerns, and users should be clearly informed when they are interacting with an AI-controlled robot and when their audio is being processed. Our work is intended to support research toward more natural and responsible human–robot interaction.

REPRODUCIBILITY STATEMENT

The dataset construction and preprocessing pipeline, including source datasets, motion retargeting, interaction-state annotation, and train/test data construction, are described in Section 3.1.1 and Appendix B. Implementation and deployment details, including model dimensions, optimization hyperparameters, training schedules, hardware, motion chunk size, sampling rates, and inference settings, are provided in Section 3.1.2. The evaluation protocols for user studies, streaming motion generation, interaction success rate, and latency are described in Section 3.2, with additional implementation details and analyses provided in the appendix. We will release the training and evaluation code, configuration files, and data-processing scripts to facilitate reproduction of the reported results.

REFERENCES

Vasu Agrawal, Akinniyi Akinyemi, Kathryn Alvero, Morteza Behrooz, Julia Buffalini, Fabio Maria Carlucci, Joy Chen, Junming Chen, Zhang Chen, Shiyang Cheng, et al. Seamless interaction: Dyadic audiovisual motion modeling and large-scale dataset. *arXiv preprint arXiv:2506.22554*, 2025.

Joao Pedro Araujo, Yanjie Ze, Pei Xu, Jiajun Wu, and C. Karen Liu. Retargeting matters: General motion retargeting for humanoid motion tracking. In *2026 IEEE International Conference on Robotics and Automation (ICRA)*, 2026.

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, et al. π_0: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024.

Zhongang Cai, Jianping Jiang, Zhongfei Qing, Xinying Guo, Mingyuan Zhang, Zhengyu Lin, Haiyi Mei, Chen Wei, Ruisi Wang, Wanqi Yin, Liang Pan, Xiangyu Fan, Han Du, Peng Gao, Zhitao Yang, Yang Gao, Jiaqi Li, Tianxiang Ren, Yukun Wei, Xiaogang Wang, Chen Change Loy, Lei Yang, and Ziwei Liu. Digital life project: Autonomous 3d characters with social intelligence. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 582–592, 2024.

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. *arXiv preprint arXiv:2410.00037*, 2024.

Hangrui Hu, Xinfa Zhu, Ting He, Dake Guo, Bin Zhang, Xiong Wang, Zhifang Guo, Ziyue Jiang, Hongkun Hao, Zishan Guo, et al. Qwen3-tts technical report. *arXiv preprint arXiv:2601.15621*, 2026.

Jianping Jiang, Weiye Xiao, Zhengyu Lin, Huaizhong Zhang, Tianxiang Ren, Yang Gao, Zhiqian Lin, Zhongang Cai, Lei Yang, and Ziwei Liu. SOLAMI: Social vision-language-action modeling for immersive interaction with 3d autonomous characters. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 26887–26898, 2025.

Haiyang Liu, Zihao Zhu, Giorgio Becherini, Yichen Peng, Mingyang Su, You Zhou, Xuefei Zhe, Naoya Iwamoto, Bo Zheng, and Michael J Black. Emage: Towards unified holistic co-speech gesture generation via expressive masked audio gesture modeling. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1144–1154. IEEE, 2024.

Ziyang Ma, Yakun Song, Chenpeng Du, Jian Cong, Zhuo Chen, Yuping Wang, Yuxuan Wang, and Xie Chen. Language model can listen while speaking. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 24831–24839, 2025. doi: 10.1609/aaai.v39i23.34665.

Miles Macklin. Warp: A high-performance python framework for gpu simulation and graphics. https://github.com/NVIDIA/warp, March 2022. NVIDIA GPU Technology Conference (GTC).

M Hamza Mughal, Rishabh Dabral, Vera Demberg, and Christian Theobalt. Miburi: Towards expressive interactive gesture synthesis. *arXiv preprint arXiv:2603.03282*, 2026.

Evonne Ng, Javier Romero, Timur Bagautdinov, Shaojie Bai, Trevor Darrell, Angjoo Kanazawa, and Alexander Richard. From audio to photoreal embodiment: Synthesizing humans in conversations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1001–1010, 2024.

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

Qwen Team. Qwen3.6-27B: Flagship-level coding in a 27b dense model, April 2026. URL https://qwen.ai/blog?id=qwen3.6-27b.

Jun Saito, Jiefeng Li, Michael de Ruyter, Miguel Guerrero, Edy Lim, Ehsan Hassani, Roger Blanco Ribera, Hyejin Moon, Magdalena Dadela, Marco Di Lucca, et al. Soma: Unifying parametric human body models. *arXiv preprint arXiv:2603.16858*, 2026.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. In *Advances in Neural Information Processing Systems*, volume 36, pp. 68539–68551, 2023.

Gabriel Skantze. Turn-taking in conversational systems and human-robot interaction: A review. *Computer Speech & Language*, 67:101178, 2021. doi: 10.1016/j.csl.2020.101178.

Thinking Machines Lab. Interaction models: A scalable approach to human-ai collaboration. *Thinking Machines Lab: Connectionism*, May 2026. doi: 10.64434/tml.20260511.

Zifan Wang, Ziang Ren, Pengyang Shi, Zirui Wang, Chenghuai Lin, Tianze Wang, Zekun Qi, Liangliang Zhao, He Wang, and Li Yi. RoboGesture: Real-time semantic-aligned co-speech gestures generation for humanoid interaction, 2026.

Zhifei Xie, Zihang Liu, Ze An, Xiaobin Hu, Yue Liao, Ziyang Ma, Dongchao Yang, Mingbao Lin, Deheng Ye, Shuicheng Yan, et al. Audio interaction model. *arXiv preprint arXiv:2606.05121*, 2026.

Jin Xu, Zhifang Guo, Jinzheng He, Hangrui Hu, Ting He, Shuai Bai, Keqin Chen, Jialin Wang, Yang Fan, Kai Dang, Bin Zhang, Xiong Wang, Yunfei Chu, and Junyang Lin. Qwen2.5-omni technical report. *arXiv preprint arXiv:2503.20215*, 2025.

Lujie Yang, Xiaoyu Huang, Zhen Wu, Angjoo Kanazawa, Pieter Abbeel, Carmelo Sferrazza, C. Karen Liu, Rocky Duan, and Guanya Shi. OmniRetarget: Interaction-preserving data generation for humanoid whole-body loco-manipulation and scene interaction. In *2026 IEEE International Conference on Robotics and Automation (ICRA)*, 2026.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R. Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations*, 2023.

Hongwei Yi, Hualin Liang, Yifei Liu, Qiong Cao, Yandong Wen, Timo Bolkart, Dacheng Tao, and Michael J Black. Generating holistic 3d human motion from speech. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 469–480. IEEE, 2023.

Youngwoo Yoon, Woo-Ri Ko, Minsu Jang, Jaeyeon Lee, Jaehong Kim, and Geehyuk Lee. Robots learn social skills: End-to-end learning of co-speech gesture generation for humanoid robots. In *2019 International Conference on Robotics and Automation (ICRA)*, pp. 4303–4309, 2019. doi: 10.1109/ICRA.2019.8793720.

Dongbin Zhang, Yunfei Liu, Lijian Lin, Ye Zhu, Yang Li, Minghan Qin, Yu Li, and Haoqian Wang. Guava: Generalizable upper body 3d gaussian avatar. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 14205–14217. IEEE, 2025.

Zeyi Zhang, Zixi Kang, Ruijie Zhao, Yusen Feng, Biao Jiang, Hanyu Ji, and Libin Liu. ProAct: Harnessing streaming motion generation and agentic reasoning for real-time embodied social interaction. *ACM Transactions on Graphics*, 45(6), 2026. doi: 10.1145/3842576.

Chunting Zhou, Lili Yu, Arun Babu, Kushal Tirumala, Michihiro Yasunaga, Leonid Shamis, Jacob Kahn, Xuezhe Ma, Luke Zettlemoyer, and Omer Levy. Transfusion: Predict the next token and diffuse images with one multi-modal model. In *International Conference on Learning Representations*, volume 2025, pp. 6446–6469, 2025.

SUPPLEMENTARY MATERIALS

This appendix provides additional material that complements the main paper. Section A reviews prior studies on human–robot interaction, streaming interaction models, and co-speech gesture generation, and situates EIP relative to cascaded verbal–motion pipelines. Section B describes the construction of EI-Data, covering motion normalization and robot retargeting (Section B.1), real-interaction episodes from Seamless Interaction (Section B.2), synthetic-context co-speech episodes (Section B.3), and emotion annotation for second-stage training (Section B.4). Section C specifies playback-synchronized token commitment (Algorithm 1), which aligns response-token KV-cache updates with speech playback. Section D explains the rationale behind the token ordering and attention visibility in the chunk-wise causal mask. Section E provides a sensitivity analysis of the Stage-I loss weights. Section F presents additional qualitative results of EIP in simulation (Fig. 8).

A RELATED WORK

A.1 HUMAN–ROBOT INTERACTION

Social interaction has been increasingly explored through virtual humans and autonomous characters, where multimodal social behaviors can be modeled without the physical constraints of robotic embodiment (Cai et al., 2024; Jiang et al., 2025). Conventional approaches often learn motion in human representation and subsequently retarget it to the robot (Yoon et al., 2019). Recent retargeting methods improve the fidelity of this transfer (Araujo et al., 2026; Yang et al., 2026). However, transferring such behaviors to physical robots introduces substantial embodiment gaps, including morphological differences, kinematic feasibility, safety, expressiveness degradation, and real-time execution.

Recent robot-centric work has therefore shifted toward learning directly in the robot's representation, moving cross-embodiment alignment from inference-time post-processing to data preprocessing (Zhang et al., 2026; Wang et al., 2026). Building on this direction, ProAct (Zhang et al., 2026) combines a streaming multimodal interaction model with a streaming motion generator for real-time embodied social interaction. RoboGesture (Wang et al., 2026) learns streaming co-speech motion directly in robot space and deploys it on a physical humanoid robot. Despite these advances, verbal interaction and motion generation retain cascaded, introducing additional inter-module handoff and synchronization overhead that can increase end-to-end latency. In contrast, EIP unifies interaction and motion generation within a unified policy, enabling low-latency and synchronized embodied interaction.

A.2 INTERACTION MODELS

Interactive agents have evolved from text-based systems that interact through language and tools (Yao et al., 2023; Schick et al., 2023), to speech-based full-duplex models (Défossez et al., 2024; Ma et al., 2025), and multimodal models that continuously process audio and visual inputs (Xu et al., 2025). Recent work further treats interaction itself as a native modeling problem. Thinking Machines' Interaction Model (Thinking Machines Lab, 2026) models audio, video, and text as time-aligned continuous streams, enabling interruption, overlap, and responsive interaction. Audio Interaction Model (Xie et al., 2026) formulates a similar paradigm for continuous audio through an always-on perceive–decide–respond loop.

These works primarily model interaction through digital modalities such as speech, text, and vision. We extend this interaction paradigm to physical embodiment, incorporating robot motion into the continuous interaction loop.

A.3 CO-SPEECH GESTURE GENERATION

Another related topic is co-speech gesture generation, which generates expressive motion in sync with speech. Recent methods have advanced from holistic speech-driven motion generation (Yi et al., 2023; Liu et al., 2024) to photorealistic conversational motion (Ng et al., 2024) and causal online gesture generation (Mughal et al., 2026). While these methods primarily generate digital

motion conditioned on the speaker's speech, EIP targets continuous embodied dialogue interaction, jointly modeling robot behaviors across speaking, listening, and idle states.

B Data Processing Details

We construct two types of training episodes: real two-person interactions from Seamless Interaction (Agrawal et al., 2025), and synthetic-context co-speech episodes derived from BEAT2 (Liu et al., 2024), GUAVA (Zhang et al., 2025), and SHOW (Yi et al., 2023). Throughout the data-processing pipeline, we refer to the person whose motion is retargeted to the robot as the *target*, and to the other participant whose speech is provided as input as the *user* or *partner*.

B.1 Motion Normalization and Robot Retargeting

As illustrated in Figure 7, we process the human motion in the following stages.

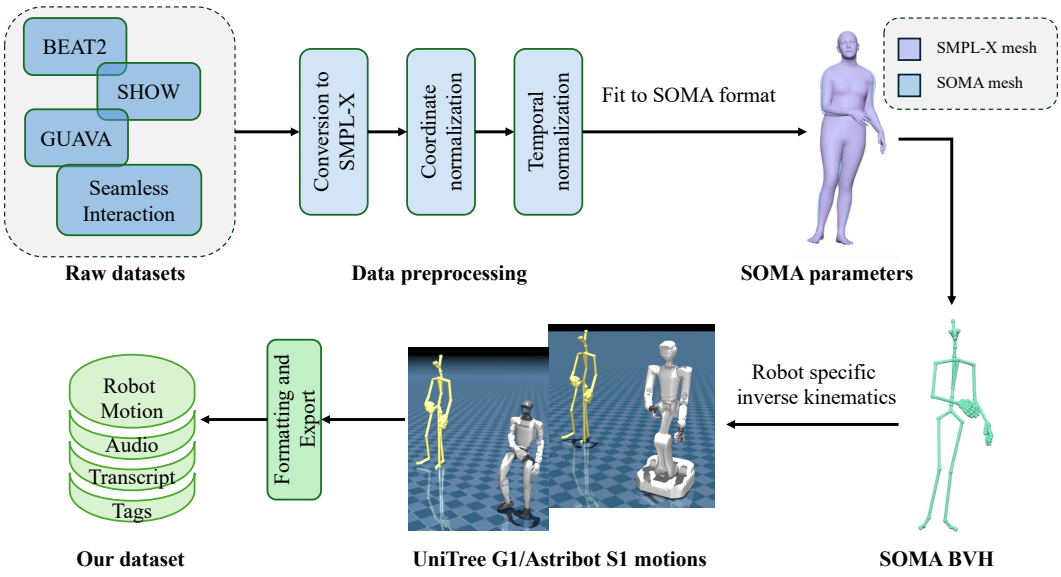

Figure 7: **Motion normalization and robot-retargeting pipeline.** Human motion is converted to a common SMPL-X representation, temporally normalized, fitted to SOMA, and retargeted to the robot via inverse kinematics.

Conversion to a common SMPL-X representation. Dataset-specific loaders first convert all motion sequences into a common SMPL-X representation (Pavlakos et al., 2019) consisting of the global root orientation, body pose, root translation, and body-shape parameters. BEAT2 (Liu et al., 2024) stores each motion frame using a dataset-specific 215-dimensional parameterization containing facial expression, jaw and eye poses, global orientation, body pose, and bilateral hand poses. GUAVA (Zhang et al., 2025), SHOW (Yi et al., 2023), and Seamless Interaction (Agrawal et al., 2025) are loaded from AMASS-style SMPL-X files. We preserve the available hand, facial, and expression parameters, although the robot supervision primarily uses body and arm motion.

Coordinate and temporal normalization. We express the human motion in meters using a canonical Y-up coordinate system. For sequences with a dataset-specific global orientation offset, we remove the sequence-level mean root rotation and select the axis-aligned world rotation that makes the body upright. The same world rotation is applied to both the global orientation and root translation. For GUAVA, we additionally apply a five-frame sliding rotation average on $\mathrm{SO}(3)$ to suppress high-frequency pose jitter.

To reduce the cost of fitting, the input sequence is temporally subsampled by a factor of two while its frame-rate metadata is adjusted accordingly, preserving its physical duration. We then resample all normalized sequences to 50 Hz. Rotations are interpolated with spherical linear interpolation, while translations and other Euclidean quantities are interpolated linearly.

SMPL-X-to-SOMA fitting. We first use the SMPL-X parameters to reconstruct the posed human mesh. SOMA-X is then used to fit the intermediate SOMA skeleton to the mesh vertices (Saito et al., 2026). Shape-dependent identity parameters are prepared once for each body shape and reused across frames. Fitting is performed in batches of 128 frames using two body-fitting iterations, one full-body iteration, and three Lie-algebra refinement iterations. The resulting intermediate representation contains the SOMA joint rotations, root translation, and per-vertex fitting error. We also export the fitted motion as a SOMA BVH sequence, which serves as the common input to the robot-specific retargeters.

Robot-specific inverse kinematics. The SOMA motion is retargeted independently to the Unitree G1 and Astribot S1 embodiments. We first rescale the human skeleton using robot-specific segment-length factors and transform the motion into the robot coordinate system. A Newton-based inverse-kinematics solver, accelerated with NVIDIA Warp (Macklin, 2022), then optimizes the robot configuration at every frame.

For G1, the IK objectives constrain the pelvis, torso, shoulders, elbows, wrists, knees, and feet while respecting joint limits and encouraging temporal smoothness. Foot stabilization is applied as a post-processing step. For S1, the planar chassis pose is obtained from the projected SOMA hip translation and heading, while the upper-body IK objectives match the shoulder, elbow, and wrist trajectories. Head yaw and pitch are recovered from the relative chest–neck orientation and constrained to the physical head-joint range. We use 24 IK iterations for G1 and 40 iterations for S1. Long sequences are processed in contiguous temporal blocks while carrying the final joint state of one block into the next, preventing discontinuities at block boundaries.

Motion representation used for training. The retargeter exports robot joint trajectories at 50 Hz. For S1, the stored trajectory contains the planar chassis, torso, head, two arms, and grippers. Before training, we convert angular quantities from degrees to radians, resample the trajectory to 30 Hz, and retain 20 actuated upper-body degrees of freedom: four torso joints, two head joints, and seven joints for each arm. The planar chassis and gripper joints are not included in the predicted motion vector.

Each 0.4-s interval consequently contains 12 robot frames. At streaming step k, the motion in the current interval is used as the robot state, and the following 12-frame interval is used as the target motion:

$$\mathbf{s}_k = \mathbf{q}_{12k:12(k+1)}, \qquad \mathbf{a}_k = \mathbf{q}_{12(k+1):12(k+2)}, \tag{13}$$

where $\mathbf{q} \in \mathbb{R}^{T\times 20}$ denotes the processed S1 joint trajectory.

B.2 Real-Interaction Episodes from Seamless Interaction

Seamless Interaction (Agrawal et al., 2025) provides synchronized recordings of two interacting participants. We retain the original speech of the partner and the retargeted motion of the target, thereby preserving real turn-taking, interruptions, pauses, and listening behavior. The processing is performed as follows.

Participant pairing. We group recordings by their shared interaction identifier and match each retargeted target sequence with the unique other participant in the same interaction. A valid pair must contain the audio, transcript, and video streams of both participants, together with the target robot motion. Because the construction is directional, either participant can serve as the target when a valid retargeted motion sequence is available.

Common valid interval. We compute the usable duration as the minimum duration among the two audio streams, the two videos, and the target motion. The sequence is further truncated at the last detected speech event so that long, inactive tails do not dominate training.

Chunk-level speech states. Voice-activity intervals are read from the dataset annotations, and neighboring intervals separated by at most 1 s are merged into a single utterance. The synchronized streams are divided into non-overlapping 0.4-s chunks. For each participant, every chunk is assigned one of three states:

$$z_k \in \{\texttt{onset}, \texttt{speaking}, \texttt{non_speech}\}. \tag{14}$$

A chunk is labeled `onset` if it contains the beginning of an utterance, `speaking` if it overlaps an utterance without containing its onset, and `non_speech` otherwise. Response-onset supervision is applied to `onset` and `non_speech` chunks; chunks in which the target is already speaking are excluded from this decision loss because the response has already been initiated. Word-level transcript tokens are assigned to chunks according to their timestamps.

Partner-first episode extraction. Candidate episodes are anchored at a partner-speech onset. Each episode begins at the anchor and spans 20 s:

$$T_{\text{episode}} = 20\ \text{s}, \qquad \Delta t = 0.4\ \text{s}, \qquad N_{\text{chunk}} = \frac{T_{\text{episode}}}{\Delta t} = 50. \tag{15}$$

We retain only complete 50-chunk windows that contain a subsequent target response and at least one valid target transcript token. Accepted windows from the same interaction are made non-overlapping. This construction provides the model with both the user's speech turn and the target's transition from listening to speaking.

Quality filtering and final subset. We discard episodes with missing modalities, incomplete chunk sequences, invalid transcripts, or unavailable robot motion. We additionally measure joint velocity, acceleration, and within-chunk motion range during target-non-speech intervals. Sustained-motion statistics are constrained by a one-sided 90th-percentile envelope, and spike-related statistics by a 95th-percentile envelope, estimated from a manually inspected clean subset. This filtering removes retargeting failures and implausible motion spikes while retaining naturally moving listeners. We deterministically select 30,000 complete episodes, corresponding to 1,500,000 synchronized 0.4-s chunks.

B.3 Synthetic-Context Co-Speech Episodes

BEAT2 (Liu et al., 2024), GUAVA (Zhang et al., 2025), and SHOW (Yi et al., 2023) contain a target speaker's speech, transcript, and co-speech motion, but do not provide the preceding user turn. We therefore construct a synthetic interaction context for each usable target clip.

Preceding-user-utterance generation. Given the complete target transcript, Qwen3.6-27B (Qwen Team, 2026) generates a short utterance that could plausibly occur immediately before the target speech. The prompt includes a dataset-dependent conversational context and asks the model to prefer a natural question without copying the target response. For the English data used in our experiments, the generated utterance is limited to at most 18 words. Invalid, empty, multi-line, overly long, or copied generations are rejected and regenerated.

User-speech synthesis. We synthesize the generated utterance with Qwen3-TTS-12Hz-1.7B-CustomVoice (Hu et al., 2026). One of two English speaker presets is selected deterministically from the source-speaker identity, so clips from the same source speaker use a consistent user voice. The synthesis instruction requests a brief, natural conversational style at a moderate speaking rate. During data loading, the waveform is converted to mono, resampled to 16 kHz, and partitioned into 50 audio chunks of 0.4 s each.

Response-time placement. The generated user utterance is placed at the beginning of the 20-s episode. We deterministically sample a post-utterance thinking delay $\tau \in \{0.4, 0.8, 1.2\}$ s and align the beginning of the target response to the next chunk boundary:

$$t_{\text{target}} = \Delta t \left\lceil \frac{d_{\text{user}} + \tau}{\Delta t} \right\rceil, \qquad \Delta t = 0.4\ \text{s}, \tag{16}$$

where d_{user} is the duration of the synthesized user speech. This guarantees that target speech and motion do not begin before the user has finished speaking. The target word timestamps and the retargeted co-speech motion are shifted by the same t_{target}.

Listening and speaking motion composition. Before t_{target}, the robot follows a fixed listening-motion sequence. We use the first 9 s of a selected S1 listening sequence and loop it when additional duration is required. A 0.4-s linear blend is applied at loop boundaries. At the target-response boundary, we blend from the listening trajectory into the retargeted co-speech motion over 0.4 s. When the target motion ends before the 20-s episode boundary, we similarly blend back to the listening motion. The final listening padding is retained to keep every episode the same length but is excluded from the motion loss and motion-normalization statistics, since it is not part of the observed target motion.

Filtering and statistics. A source clip is retained only if its target audio, word-timestamped transcript, retargeted S1 motion, generated user waveform, and response onset all fit within the 20-s episode. Starting from 29,152 English candidates, 110 clips are rejected because they do not contain valid timed words, and five are rejected because the target speech begins outside the episode. The resulting Co-Speech set contains 29,037 episodes, as summarized in Table 4.

Table 4: Number of synthetic-context co-speech episodes by source dataset.

Source dataset	Number of episodes
BEAT2	9,242
GUAVA	3,858
SHOW	15,937
Total	29,037

B.4 Emotion Annotation for Second-Stage Training

For the second training stage, we annotate the semantic emotion of every target response using Qwen3.6-27B (Qwen Team, 2026). The classifier receives only the complete target transcript; it does not observe the generated user utterance, target audio, or motion. Consequently, the labels describe the semantic and emotional stance of the response rather than its acoustic prosody or motion magnitude.

The model assigns exactly one of six labels:

$$e \in \{\texttt{happy}, \texttt{sad}, \texttt{angry}, \texttt{surprise}, \texttt{normal}, \texttt{urgent}\}. \tag{17}$$

In particular, fear, anxiety, and nervousness are assigned to `sad`, whereas `urgent` is reserved for direct warnings, requests for immediate motion, or emergencies. Neutral descriptions of dangerous events remain `normal`. We use deterministic decoding and store both the predicted label and its confidence score. All 29,037 Co-Speech episodes receive a valid annotation, with the distribution reported in Table 5.

Table 5: Semantic emotion annotations on the Co-Speech set before motion–emotion consistency filtering.

Emotion	Number of episodes
`happy`	2,561
`sad`	1,402
`angry`	1,172
`surprise`	731
`normal`	23,103
`urgent`	68
Total	29,037

Although the raw emotion distribution is highly imbalanced, we do not sample directly from this empirical distribution during second-stage training. We first apply a motion–emotion consistency

filter. For each episode, motion intensity is measured over valid target-speaking motion frames using

$$s_i = \sqrt{\bar{v}_i \, \bar{d}_i}, \tag{18}$$

where $\bar{v}_i$ is the mean ℓ_2 joint displacement between consecutive motion frames and $\bar{d}_i$ is the mean displacement of the predicted motion horizon from the last observed robot state. When possible, the first speaking chunk is excluded because it contains the synthetic transition from the fixed listening motion to the original co-speech motion. Since the three source datasets have different motion statistics, we compute the median score $m_{\mathcal{D}}$ separately for each source dataset $\mathcal{D}$.

We divide the six semantic labels into

$$\mathcal{E}_{\text{high}} = \{\texttt{happy}, \texttt{angry}, \texttt{surprise}, \texttt{urgent}\}, \qquad \mathcal{E}_{\text{low}} = \{\texttt{sad}, \texttt{normal}\}. \tag{19}$$

An episode with label $e_i \in \mathcal{E}_{\text{high}}$ is retained only if $s_i \geq m_{\mathcal{D}}$, whereas an episode with $e_i \in \mathcal{E}_{\text{low}}$ is retained only if $s_i < m_{\mathcal{D}}$. This removes samples for which the semantic emotion and the observed motion intensity strongly disagree. After filtering, the numbers of eligible episodes are 1,454 for `happy`, 561 for `angry`, 359 for `surprise`, 36 for `urgent`, 752 for `sad`, and 11,644 for `normal`.

We then use hierarchical balanced sampling instead of following the remaining class frequencies. In every epoch, we sample the same number M of episodes from $\mathcal{E}_{\text{high}}$ and $\mathcal{E}_{\text{low}}$, producing a 1:1 ratio between the two groups. Samples are also distributed uniformly among the labels within each group:

$$n_e = \begin{cases} M/4, & e \in \mathcal{E}_{\text{high}}, \\ M/2, & e \in \mathcal{E}_{\text{low}}. \end{cases} \tag{20}$$

We use $M = 1{,}536$, resulting in 384 samples per epoch for each of `happy`, `angry`, `surprise`, and `urgent`, and 768 samples per epoch for each of `sad` and `normal`. Classes with more eligible examples than their quota are randomly undersampled without replacement, whereas rare classes are oversampled through independently shuffled passes over their eligible examples. The resulting 3,072 samples are reshuffled at every epoch and then partitioned across distributed training workers. This procedure prevents the dominant `normal` class from overwhelming the training objective while ensuring that rare emotions, particularly `urgent`, contribute consistently to second-stage training.

C Incremental Generation and Playback Synchronization

Algorithm 1 describes response execution after the interaction expert decides to respond. Let t_0 denote the interaction step at which the response begins. During inference, the interaction context h_t is maintained as a committed KV cache synchronized with speech playback. We denote the independent response-decoding cache by h^{dec}. Both caches use the same frozen model parameters. The decoding cache is initialized from the committed context h_{t_0} and seeded with the response-start token.

The buffer $\mathcal{B}$ stores synthesized audio, aligned text units, and a playback cursor. Each synthesized phrase is appended to this buffer, with its local timestamps shifted by the cumulative duration of preceding phrases. Let u_i be an aligned text unit and τ_i its start time on the response-relative audio timeline. The tokens associated with playback interval r are

$$\bar{y}_r = \text{Tokenize}(\text{Concat}\,\{u_i \mid r\Delta \leq \tau_i < (r+1)\Delta\}), \qquad \Delta = 400\,\text{ms}, \tag{21}$$

where text units are concatenated in temporal order. Here, r indexes response-relative playback intervals, whereas t indexes interaction steps. Audio chunks may span phrase boundaries. Only the final incomplete chunk is zero-padded.

The first phrase is decoded and synthesized synchronously; subsequent phrases are processed by a background worker. The response-level emotion tag e is obtained during initial decoding and retained across phrases. The flag done becomes true only after decoding has ended and the final synthesized phrase has been appended.

At each interaction step t, the outer interaction loop supplies h_t updated with the latest human-audio chunk, the robot state q_t, and an interruption flag ι_t. If the response remains active, the text aligned with the previously emitted speech chunk is committed to h_t before generating the current

Algorithm 1 Incremental Response Generation and Playback Synchronization

Require: Frozen interaction model M, TTS model S, committed interaction context h_{t_0} stored as a KV cache
Require: Motion expert and chunk duration $\Delta = 400\,\text{ms}$
Ensure: Synchronized speech–motion output and updated interaction context

$h^{\text{dec}} \leftarrow$ ForkForResponse(M, h_{t_0})
$\mathcal{B} \leftarrow \varnothing$, $e \leftarrow \perp$
done $\leftarrow$ false, cancelled $\leftarrow$ false, sent $\leftarrow$ false
ProducePhrase ▷ Decode and synthesize the first phrase
LaunchAsync(ContinueResponse)
for each interaction step $t \geq t_0$ **do**
  The outer loop supplies updated h_t, q_t, and ι_t.
  if ι_t **then**
    cancelled $\leftarrow$ true
    Clear$(\mathcal{B})$
    $h_t \leftarrow$ CloseResponse(M, h_t)
    return h_t
  end if
  if sent **then**
    $F \leftarrow$ AdvancePlayback$(\mathcal{B}, \Delta)$
    $h_t \leftarrow$ CommitFragment(M, h_t, F, e)
    sent $\leftarrow$ false
  end if
  if done **and** Exhausted$(\mathcal{B})$ **then**
    $h_t \leftarrow$ CloseResponse(M, h_t)
    return h_t
  end if
  $v_t \leftarrow$ VocalCondition(active, e)
  $\hat{a}_t \leftarrow$ GenerateMotion(h_t, q_t, v_t)
  $\hat{s}_t \leftarrow$ PeekAudio$(\mathcal{B}, \Delta, \text{done})$
  if $\hat{s}_t = \perp$ **then**
    $\hat{s}_t \leftarrow \mathbf{0}$ ▷ Keep the response cursor unchanged
  else
    sent $\leftarrow$ true
  end if
  DispatchTogether$(\hat{s}_t, \hat{a}_t)$
end for

procedure ProducePhrase
  $(p, h^{\text{dec}}, e, f) \leftarrow$ DecodePhrase(M, h^{dec}, e)
  if $p \neq \varnothing$ **then**
    $(w, P) \leftarrow$ SynthesizeAndAlign(S, p)
    AppendIfActive$(\mathcal{B}, w, P,$ cancelled$)$
  end if
  if not cancelled **then**
    done $\leftarrow f$
  end if
end procedure

procedure ContinueResponse
  while not done **and not** cancelled **do**
    ProducePhrase
  end while
end procedure

motion chunk $\hat{a}_t$. The speech chunk $\hat{s}_t$ is retrieved from $\mathcal{B}$ without advancing the playback cursor and dispatched together with $\hat{a}_t$. Its aligned text is committed at the next interaction update. If insufficient audio is available, a silent chunk is emitted, and the response playback and text cursors remain unchanged.

In Algorithm 1, DecodePhrase continues decoding until a valid punctuation boundary or the response-end token. AppendIfActive checks that the response is still active, shifts phrase-local

timestamps onto the shared audio timeline, and appends the synthesized audio and aligned text to $\mathcal{B}$. Cancellation checks and buffer updates are performed atomically.

PEEKAUDIO retrieves the next available audio chunk without advancing the playback cursor. It returns $\perp$ if insufficient audio is available within the allowed wait; only the final incomplete chunk is zero-padded. ADVANCEPLAYBACK advances past the previously emitted chunk and retrieves its aligned text and boundary flags. COMMITFRAGMENT commits this text to h_t, adding the response-start and emotion tokens for the first fragment and the response-end token for the final fragment.

COMMITSILENCE records the silent interval preceding response playback. On interruption, CLOSERESPONSE appends the response-end token if the response has entered the committed context, or a silence-control token otherwise. State is scoped to each response, preventing cancelled generation from appending audio to a subsequent response.

D RATIONALE FOR THE ATTENTION MASK

The attention mask reflects the conditional generation process of EIP: the available interaction context, robot state, and vocal condition guide the generation of the next motion chunk, while the noisy action tokens are jointly updated through flow matching. The ordering $[Q_t; v_t; A_t^\tau]$ separates the conditioning inputs from the variables being denoised.

Interaction-context isolation and temporal visibility. Interaction-context tokens retain autoregressive attention and do not attend to motion-expert tokens. This preserves the frozen interaction expert's original information dependencies and prevents its representations from changing with sampled motion. The interaction context can therefore be computed and cached independently of action denoising. Each motion chunk accesses only the context available at its decision step, preventing attention to later interaction information even when multiple chunks are processed together during training.

Ordering of state, vocal, and action tokens. Within each chunk, Q_t represents the robot's current physical state, v_t specifies its speaking state and emotion-derived motion intensity, and A_t^τ represents the action trajectory being generated. Allowing v_t to attend to Q_t lets the vocal-condition representation incorporate the current physical state. Keeping the reverse connection masked leaves the state representation independent of the vocal-token pathway. Action tokens attend to both blocks, combining the starting state with the desired behavioral condition. Neither state nor vocal tokens attend to action tokens, so their representations remain independent of the changing noisy trajectory throughout flow integration. This ordering is an inductive bias that maintains a separation between fixed conditions and generated actions.

Bidirectional attention within blocks. Tokens within the state block encode observations already available at the decision boundary and can therefore attend to one another. Tokens within the action block represent a trajectory generated jointly through flow matching. Bidirectional attention allows each action token to incorporate information from the entire candidate trajectory and model dependencies among its frames. These tokens are denoising variables available during the current computation, rather than future observations, so their mutual visibility does not violate temporal causality.

Isolation between motion chunks. Direct attention between different motion chunks is masked. Blocking future chunks prevents access to unavailable robot states and future motion information. Blocking past motion chunks is an architectural choice that avoids dependence on their intermediate denoising representations: temporal information is instead carried through the accumulated interaction context and the updated robot state. Given the same conditioning inputs, each chunk therefore has the same local attention dependencies whether processed alongside other chunks during training or generated individually during streaming inference.

These restrictions are applied at every layer, preventing future context or other motion chunks from influencing the current prediction through indirect attention paths. This argument assumes that the input features and conditioning signals themselves are available at the corresponding decision boundary.

Table 6: Observer-based evaluation of Stage-I loss weights. Five participants rate each configuration on a seven-point Likert scale.

λ_{spk}	λ_{lis}	λ_{idle}	Naturalness↑	State Adaptation↑	Listening Diversity↑	Average↑
1	1	1	5.60	5.40	5.00	5.33
1	4	1	5.60	4.00	6.20	5.27
1	2	0.5	5.80	5.00	6.00	5.60
1	2	2	5.80	6.00	4.00	5.27
1	2	1	5.80	6.40	5.80	**6.00**

E Sensitivity to Stage-I Loss Weights

We examine how the Stage-I loss weights affect perceived motion quality. We fix $\lambda_{\text{spk}} = 1$ and vary λ_{lis} and λ_{idle} around the default setting $(1, 2, 1)$. Table 6 summarizes ratings from five participants on a seven-point Likert scale, covering Naturalness, State Adaptation, and Listening Diversity. Average is the unweighted mean of these three dimensions.

The default setting achieves the highest average rating of 6.00 and the highest State Adaptation score of 6.40, while matching the highest Naturalness score of 5.80. Reducing λ_{lis} from 2 to 1 lowers all three ratings, resulting in an average of 5.33. Increasing it to 4 yields the highest Listening Diversity score of 6.20, but State Adaptation decreases to 4.00 and the average falls to 5.27. These ratings suggest that emphasizing listening supervision can favor listening diversity without necessarily improving behavioral adaptation across interaction states.

Changing the idle weight also produces trade-offs. With $\lambda_{\text{idle}} = 0.5$, Listening Diversity increases to 6.00, whereas State Adaptation decreases to 5.00, giving an average of 5.60. With $\lambda_{\text{idle}} = 2$, Listening Diversity drops to 4.00 and the average decreases to 5.27. Naturalness remains at 5.80 for both variants. Among the configurations evaluated in this small observer study, $(1, 2, 1)$ therefore provides the strongest aggregate rating, balancing naturalness, state adaptation, and listening diversity.

F Additional Qualitative Results

Figure 8 shows additional EIP rollouts in simulation under the same three interaction scenarios evaluated in Section 3.2.1. In the top row, the robot produces proactive verbal and gestural responses to salient non-speech acoustic events, such as coughing and object breaking. The middle row illustrates interruption-aware turn-taking: the robot switches from speaking to listening when the user barges in, then resumes with speech-synchronized gestures. The bottom row shows emotion-aware gesture generation, where motion intensity follows the semantic emotion of the response. These examples complement the physical-robot visualizations in the main paper.

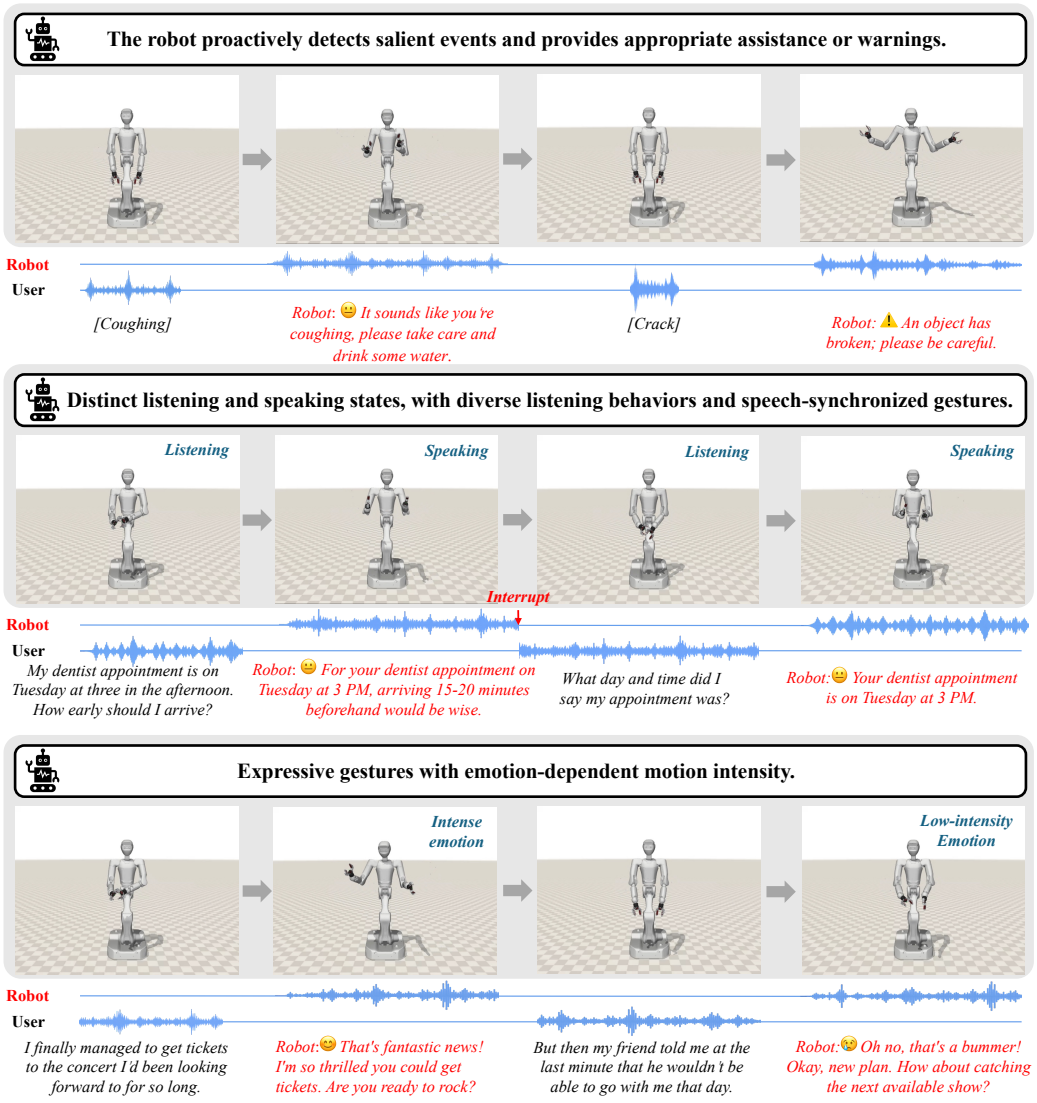

Figure 8: **Qualitative results of EIP in three representative interaction scenarios.** Top: proactive responses to salient acoustic events, such as coughing and object breaking. Middle: continuous turn-taking under user interruption, with distinct listening and speaking behaviors and speech-synchronized gestures. Bottom: emotion-aware gesture generation, where motion intensity adapts to the response emotion.