CONFABMUSE: STRUCTURED INTERACTION CONTROL FOR MULTI-PERSON TALKING AVATARS

Anonymous authors
Paper under double-blind review

ABSTRACT

Generating realistic multi-avatar conversational videos requires more than synchronizing multiple audio streams with visible avatars. A multi-avatar conversational generation framework must determine *when* each stream should influence motion, *who* should be animated by each stream, and *how* avatar behavior should evolve as the interaction unfolds. Existing methods largely treat multi-avatar generation as audio routing. They associate audio streams with avatars, but provide limited modeling of cross-stream turn dynamics and little structured control over how actions and relations evolve over time. To address this, we introduce ConfabMuse, a diffusion Transformer framework for controllable two-person conversational video generation from a reference image, two driving audio streams, and optional text guidance. ConfabMuse introduces a *Turn-Context Audio Adapter* for within and cross-stream speech modeling, *Audio-Vision Binding* for learned stream-to-avatar correspondence, and a *Scene-Grounded Interaction Planner* for structured control of evolving avatar actions and relations. Experiments across single and multi-person talking-avatar benchmarks show that ConfabMuse achieves state-of-the-art conversational interactivity, improves structured interaction control, and maintains strong audio-visual synchronization and generation quality. These results demonstrate the benefit of jointly modeling conversational timing, avatar correspondence, and interaction dynamics for controllable multi-person talking-avatar generation.

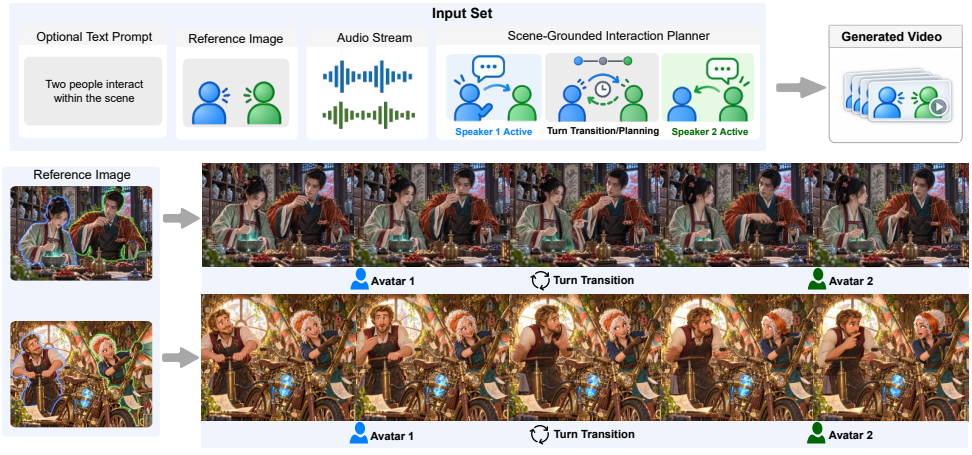

Figure 1: **Controllable two-person conversational generation with ConfabMuse.** Given a reference image and two driving audio streams, ConfabMuse coordinates speaker turns while preserving ongoing actions and adapting avatar-specific behavior across the interaction.

1 INTRODUCTION

Audio-driven avatar generation aims to synthesize realistic human videos from a reference image and driving speech, with applications in content creation, film production, and virtual presenters Nazarieh et al. (2024). Early methods mainly focused on talking-head generation, where audio drives lip motion Cui et al. (2025); Chen et al. (2025b); Wang et al. (2024a). More recent

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.

video-diffusion models extend this setting to richer head and upper-body motion Meng et al. (2025); Nazarieh et al. (2026); Lin et al. (2025). However, most approaches still assume a single visible speaker and a one-to-one correspondence between the driving audio and the animated avatar. A multi-avatar conversation is fundamentally different. Avatars alternate between speaking and listening, respond to one another, and maintain coordinated behavior over time.

Recent methods have begun to support multi-person generation through multiple audio streams and audio-to-avatar correspondence mechanisms Zhong et al. (2025); Chen et al. (2025a); Ma et al. (2025a); Kong et al. (2025); Huang et al. (2025). However, these methods largely focus on assigning speech to visible avatars. A coherent conversation requires three coupled questions to be addressed: *when* should each audio stream drive speech-related motion, *who* should be animated by each stream, and *how* should avatar behavior evolve across conversational turns.

First, conversational timing cannot be inferred reliably from each audio stream in isolation. Local acoustic cues, such as rhythm and intensity, describe short-term speech behavior within each stream, while activity in the other stream provides context for interpreting turn transitions. For example, a short silence may indicate either a within-turn pause or a handover to another speaker. Distinguishing these cases motivates modeling both *within-stream speech dynamics* and *cross-stream turn context*. Second, knowing *when* speech occurs does not determine *who* should be animated by each stream. Existing methods use identity labels or spatial routing mechanisms such as masks Kong et al. (2025); Huang et al. (2025), but do not explicitly model compatibility between each audio stream and the visual representation of each avatar.

Finally, correct timing and audio-avatar correspondence are not sufficient for natural interaction. An avatar that is not speaking should still listen, shift gaze, gesture, or continue an ongoing action while remaining consistent with the evolving interaction. A global text prompt can describe an interaction, but controlling multiple avatars through a single description remains difficult. It does not explicitly specify which avatar behaviors and relations should persist or change across conversational turns. This becomes especially important when avatars perform concurrent actions. For example, one avatar may speak while chopping vegetables, while the other continues stirring and responds through gaze or gesture. Explicit pose or landmark sequences could provide such avatar-level control, but require additional frame-level motion inputs and predefined trajectories. We instead represent the interaction explicitly, allowing it to evolve across the conversation while preserving the generative flexibility of the video model.

To address these challenges, we introduce *ConfabMuse*, a diffusion Transformer framework for multi-person conversational avatar generation. We focus on the two-person setting with two visible avatars, two driving audio streams, and an optional text prompt (Fig. 1). Our framework models conversation through three complementary components corresponding to *when*, *who*, and *how*.

The *Turn-Context Audio Adapter* models *when* and how strongly each audio stream should influence speech-related motion. It combines *within-stream speech dynamics* with *cross-stream turn context* to produce *turn-aware audio features* that retain expressive speech cues while accounting for turn transitions (§ 3.1). The *Audio-Vision Binding* module determines *who* should be animated by each stream by learning pairwise compatibility between audio and visual avatar representations and using these scores as avatar-specific biases in a binding-aware audio cross-attention operation (§ 3.2). This provides soft, content-dependent binding without hard audio-to-avatar routing at inference.

To model *how* the avatars interact as the conversation unfolds, we introduce a *Scene-Grounded Interaction Planner* that represents the interaction as a structured graph (§ 3.3). Avatar nodes encode states and actions, while edges capture pairwise relations. Rather than predicting a new interaction state at every frame, the planner updates the graph only at pause-delimited conversational segments, preserving ongoing behaviors and modifying only those that change. The planner is trained with GRPO Shao et al. (2024) to encourage visual grounding, turn consistency, temporal persistence, and relation consistency, while a reflector identifies unsupported or inconsistent updates and guides their revision. The resulting graph states are serialized into compact interaction instructions for the video generator. Our main contributions are:

- We introduce a *Turn-Context Audio Adapter* that jointly models *within-stream speech dynamics* and *cross-stream turn context*, producing turn-aware audio features for conversational motion conditioning.

- We propose *Audio-Vision Binding*, which learns *soft pairwise compatibility* between audio streams and visual avatar representations and injects it as an avatar-specific bias into audio cross-attention, enabling learned stream-to-avatar binding through soft compatibility-guided audio cross-attention.
- We introduce a *Scene-Grounded Interaction Planner* that represents multi-avatar evolving interactions as a structured graph and updates it over conversational segments, preserving ongoing actions and relations while modifying only the avatar states that change.

Extensive experiments on InteractiveEyes Zhong et al. (2025), the Multi-human Interactive Talking dataset Zhu et al. (2025), and single-person talking-avatar benchmarks Zhang et al. (2021); Ephrat et al. (2018); Meng et al. (2025) show that *ConfabMuse* achieves state-of-the-art performance in conversational interactivity and interaction control, while maintaining strong audio-visual synchronization, visual quality, and identity preservation.

2 Related Work

2.1 Audio-Driven Talking Avatar Generation

Early audio-driven talking-face methods relied on GAN-based models Zhou et al. (2020); Prajwal et al. (2020) or explicit 3D representations Richard et al. (2021); He et al. (2023) to synchronize speech with facial motion, but primarily focused on facial synchronization and offered limited control over richer expressive motion. Recent diffusion-based methods Wang et al. (2024a); Chen et al. (2025b); Cui et al. (2025); Wang et al. (2025a); Gan et al. (2025); Chen et al. (2025a) improve visual quality, identity preservation, and audio-visual synchronization. Several systems further extend audio-driven generation beyond the face to upper-body motion Meng et al. (2025); Lin et al. (2025); Nazarieh et al. (2026), enabling richer head and body dynamics. Despite this progress, most methods are designed for a single visible speaker and therefore do not address the additional challenges of multi-person conversation, including turn transitions, audio-to-avatar correspondence, and coordinated behavior across avatars.

2.2 Multi-Person Conversational Talking Video Generation

Recent work extends audio-driven avatar generation to multi-person settings, with particular emphasis on associating multiple audio streams with the corresponding visible avatars. MultiTalk Kong et al. (2025) introduces Label Rotary Position Embeddings for audio-to-avatar binding, while Bind-Your-Avatar Huang et al. (2025) uses a dynamic mask-based embedding router for fine-grained audio-character correspondence. HunyuanVideo-Avatar Chen et al. (2025a) uses face-aware conditioning, whereas Playmate2 Ma et al. (2025b) employs mask-based classifier-free guidance for character-specific audio routing. AnyTalker Zhong et al. (2025) further incorporates multi-stream audio modeling and interactivity refinement. Recent multi-person methods improve multi-stream audio conditioning, avatar localization, and stream-to-avatar correspondence, while related dyadic speaker-listener methods model responsive listener behavior Zhang et al. (2026); Ki et al. (2026); Shan et al. (2026). However, these aspects are typically addressed separately. In contrast, ConfabMuse jointly models cross-stream turn context, learned stream-to-avatar compatibility, and explicit segment-level evolution of avatar actions and relations.

3 Methodology

Given a reference image containing two visible avatars, two driving audio streams, and an optional text prompt, ConfabMuse generates two-person conversational videos with synchronized speech, preserved identity, and coordinated avatar behavior. As shown in Fig. 2, the framework consists of three components that model *when*, *who*, and *how* in a conversation: a *Turn-Context Audio Adapter* that models within-stream speech dynamics and cross-stream turn context, an *Audio-Vision Binding* module that learns stream-to-avatar correspondence, and a *Scene-Grounded Interaction Planner* that uses a structured interaction graph to control how avatar-specific actions and relations evolve across conversational turns.

3.1 Turn-Context Audio Adapter

We introduce a *Turn-Context Audio Adapter* to model local speech dynamics within each audio stream together with conversational context across streams. The adapter contains two complementary branches: *Within-Stream Speech Dynamics*, which modulates audio features according to local speech dynamics such as rhythm and vocal intensity, and *Cross-Stream Turn Context*, which uses activity across streams to distinguish within-turn pauses from speaker handovers.

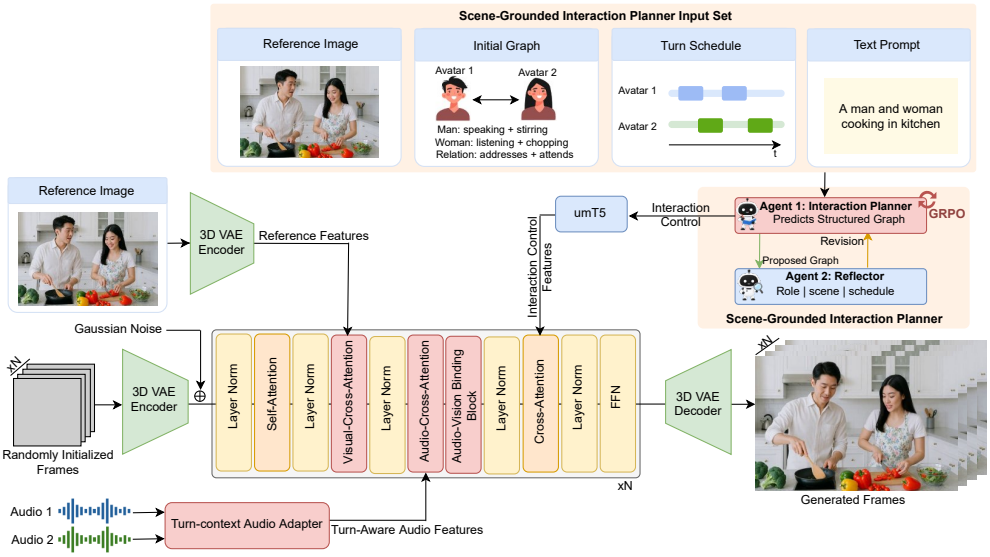

Figure 2: **Overview of ConfabMuse.** Given a reference image, two driving audio streams, and an optional text prompt, ConfabMuse models *when* each audio stream should drive motion with the Turn-Context Audio Adapter, *who* it should affect with Audio-Vision Binding, and *how* the avatars should interact with the Scene-Grounded Interaction Planner. These conditions guide the diffusion Transformer to generate coherent two-person conversational video.

Together, they produce turn-aware audio features whose strength reflects both local speech dynamics and cross-stream turn context.

For the *within-stream branch*, let $\{A^{(k)} \in \mathbb{R}^{T\times d}\}_{k=1}^{K}$ denote the Wav2Vec2 Baevski et al. (2020) features of the driving audio streams, where $K = 2$ in our setting, T is the number of temporal steps, and d is the feature dimension. At time step t, a temporal MLP f_{temp} processes the local window $A^{(k)}_{t-r:t+r}$, where $r = 3$ is the local window radius, and predicts a scalar weight $w_t^{(k)}$. The feature is reweighted as $\tilde{A}_t^{(k)} = w_t^{(k)} A_t^{(k)}$, preserving temporal alignment while allowing its contribution to vary with local speech dynamics.

The *cross-stream branch* models turn context from speech activity across streams. For each stream, we estimate the frame-level speech probability $p_t^{(k)} = \sigma(f_{\text{act}}(A_t^{(k)}))$ and supervise it using binary voice-activity Team (2024) labels $y_t^{(k)}$:

$$\mathcal{L}_{\text{act}} = \frac{1}{KT}\sum_{k=1}^{K}\sum_{t=1}^{T}\text{BCE}\left(p_t^{(k)}, y_t^{(k)}\right). \tag{1}$$

While $p_t^{(k)}$ indicates whether stream k is active at time t, it does not reveal whether inactivity corresponds to a brief within-turn pause or a speaker handover. We therefore aggregate activity over a short temporal window,

$$s_t^{(k)} = \frac{1}{2r_s+1}\sum_{\tau=t-r_s}^{t+r_s} p_\tau^{(k)},$$

where $r_s = 3$ is the smoothing radius, and the competing-stream activity is defined as $o_t^{(k)} = \max_{j\neq k} p_t^{(j)}$. The quantity $s_t^{(k)}(1-o_t^{(k)})$ serves as a continuation cue: it remains high when stream k exhibits sustained local activity while competing streams remain inactive.

We form the turn-context vector $\mathbf{z}_t^{(k)} = [p_t^{(k)}, s_t^{(k)}, o_t^{(k)}, s_t^{(k)}(1-o_t^{(k)})]^\top$ and compute a scalar gate $m_t^{(k)} = \sigma(\mathbf{W}_g\mathbf{z}_t^{(k)} + b_g)$. The adapted feature is then $\hat{A}_t^{(k)} = m_t^{(k)}\tilde{A}_t^{(k)}$, where $\mathbf{W}_g \in \mathbb{R}^{1\times 4}$ and b_g are learned parameters. Thus, the adapter preserves the original audio feature dimensionality while modulating each feature according to cross-stream turn context. Stream-to-avatar correspondence is handled separately by the *Audio-Vision Binding* module in § 3.2.

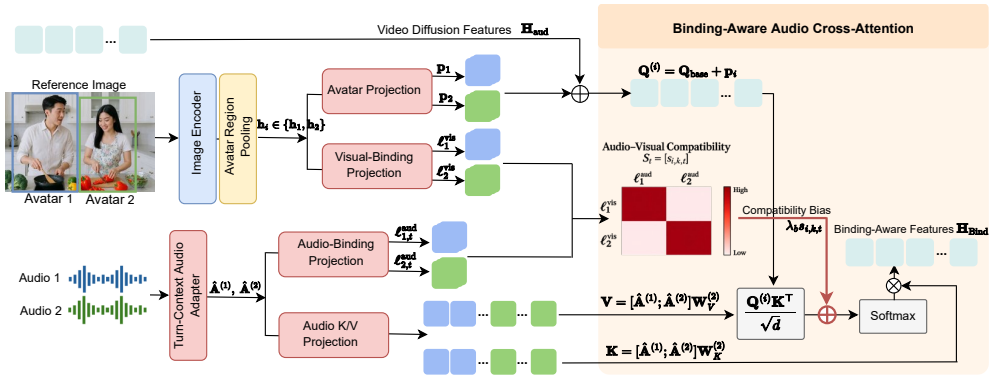

Figure 3: **Audio-Vision Binding.** We learn pairwise compatibility between avatar visual embeddings and adapted audio streams, and use the resulting scores to bias cross-attention toward the matched avatar-audio pairs.

3.2 Audio-Vision Binding

The Turn-Context Audio Adapter determines *when* and *how strongly* each audio stream should influence speech-related motion, but not *who* should be animated by each stream. Existing approaches establish stream-to-avatar correspondence using mechanisms such as identity-aware conditioning or spatial routing masks Kong et al. (2025); Zhong et al. (2025); Ma et al. (2025b); Huang et al. (2025). We instead learn pairwise compatibility between each audio stream and the visual representation of each avatar, and use the resulting score as a soft bias in audio cross-attention.

We first obtain a visual descriptor for each avatar from the reference image. For avatar i, let $\mathbf{b}_i$ denote its image region and $\mathbf{x}_n$ the n-th visual token with patch center $\mathbf{u}_n$. We collect the tokens within this region, $\mathcal{P}_i = \{n : \mathbf{u}_n \in \mathbf{b}_i\}$, and average-pool them as $\mathbf{h}_i = \frac{1}{|\mathcal{P}_i|}\sum_{n\in\mathcal{P}_i}\mathbf{x}_n$. Separate MLPs project the avatar descriptor and each adapted audio feature from § 3.1 into a shared d_ℓ-dimensional binding space: $\boldsymbol{\ell}_i^{\text{vis}} = f_{\text{vis}}(\mathbf{h}_i)$ and $\boldsymbol{\ell}_{k,t}^{\text{aud}} = f_{\text{aud}}(\hat{A}_t^{(k)})$. After ℓ_2 normalization, their compatibility is $s_{i,k,t} = (\boldsymbol{\ell}_i^{\text{vis}})^\top \boldsymbol{\ell}_{k,t}^{\text{aud}}$.

The learned compatibility is incorporated into the binding-aware audio cross-attention. For a video query $\mathbf{q}$ located within avatar region $\mathbf{b}_i$, the attention logit for stream k at time t is

$$\alpha_{\mathbf{q},k,t} = \frac{(\mathbf{W}_Q\mathbf{q})^\top(\mathbf{W}_K\hat{A}_t^{(k)})}{\sqrt{d_{\text{attn}}}} + \lambda_{\text{b}} s_{i,k,t}, \qquad \text{loc}(\mathbf{q}) \in \mathbf{b}_i. \tag{2}$$

We set $\lambda_{\text{b}} = 0.5$ in all experiments. The compatibility term therefore increases the relative attention assigned to audio streams that better match the visual representation of avatar i, while retaining soft attention over all streams.

During training, paired samples provide the target audio stream $k^*(i)$ for each avatar. To reduce the contribution of silent frames, we pool the compatibility scores using the speech activity $p_t^{(k)}$ from § 3.1: $\bar{s}_{i,k} = \frac{\sum_t p_t^{(k)} s_{i,k,t}}{\sum_t p_t^{(k)} + \epsilon}$. We then encourage each avatar to match its paired stream using

$$\mathcal{L}_{\text{bind}} = -\frac{1}{N_p}\sum_{i=1}^{N_p}\log\frac{\exp(\bar{s}_{i,k^*(i)}/\tau)}{\sum_{k=1}^{K}\exp(\bar{s}_{i,k}/\tau)}, \tag{3}$$

where N_p is the number of avatars and $\tau = 0.07$ is the temperature parameter. At inference, the learned compatibility guides each avatar region to attend more strongly to its corresponding audio stream without hard avatar-specific audio routing. With the *who* correspondence established, the *Scene-Grounded Interaction Planner* next models *how* avatar actions and relations evolve across the conversation.

3.3 SCENE-GROUNDED INTERACTION PLANNER

Audio-Vision Binding determines *who* should be animated by each audio stream, but not *how* the interaction should evolve. Across conversational turns, avatar-specific roles, gaze, actions, and relations may change while ongoing behaviors should persist. A global text prompt does not explicitly track these state changes, making concurrent behavior difficult to control. We therefore introduce a *Scene-Grounded Interaction Planner* that represents the interaction as a *structured interaction graph* grounded in the reference image and updates it over audio-derived conversational segments, optionally guided by the text prompt. The graph is updated only at audio-derived conversational segments, preserving states and actions that remain unchanged.

Table 1: **Quantitative comparison on HDTF.**

Method	FID↓	FVD↓	Sync-C↑	Sync-D↓	IQA↑
V-Express	62.8	972.3	4.13	10.55	3.14
EchoMimic	59.8	581.6	7.15	6.91	3.41
Hallo3	56.4	579.1	7.19	6.73	3.47
FantasyTalk	62.7	594.6	5.48	9.21	3.21
HunyuanVideo-Avatar	54.9	528.3	7.34	6.48	3.56
AnyTalker	54.6	521.9	7.53	6.41	3.63
OmniAvatar	53.7	514.7	7.92	6.36	3.68
MultiTalk	53.1	511.6	7.95	6.24	3.69
SyncDreamer	52.8	508.1	8.04	6.15	3.72
ConfabMuse	**51.6**	**493.7**	**8.21**	**6.03**	**3.84**

Let $\mathcal{S} = \{\rho_j\}_{j=1}^{T_s}$ denote the turn schedule derived from the speech-activity estimates of the Turn-Context Audio Adapter (§ 3.1), where ρ_j indicates the active audio stream within segment j. These pause-aligned segments define the planning points at which the interaction state may change. Before segment-wise planning, the planning agent $\mathcal{M}_P$, instantiated with Qwen3-VL-8B Bai et al. (2025), constructs a scene-grounded reference graph $\mathcal{G}^{(0)}$ from the reference image I. Each avatar is represented by a node containing its image region, appearance, pose, gaze, visible action, and nearby objects, while edges represent pairwise relations such as *attends-to*, *coordinates-with*, and *physically-supports*. The reference graph provides a fixed anchor for the avatars and their initial scene configuration throughout planning.

At each segment j, the planner predicts only the graph changes δ_j required by the current conversational context:

$$\delta_j = \mathcal{M}_P(I, \mathcal{G}^{(0)}, \mathcal{G}^{(j-1)}, \rho_j, \mathcal{T}_{\text{text}}), \qquad \mathcal{G}^{(j)} = \mathcal{G}^{(j-1)} \oplus \delta_j. \tag{4}$$

Here, $\mathcal{T}_{\text{text}}$ is the optional text prompt and $\oplus$ applies the predicted edit to the previous graph state. Only attributes that change, such as speaker role, gaze, action, or relation, are updated. All others are inherited from $\mathcal{G}^{(j-1)}$. For example, a speaker may shift gaze toward the listener while an ongoing cooking action is preserved. This sparse update allows the interaction to evolve without unnecessarily rewriting stable states.

Before applying each edit, a frozen reflector $\mathcal{M}_R$, sharing the vision-language backbone of $\mathcal{M}_P$, checks the proposal against the reference image, previous graph state, and turn schedule for visual grounding, turn consistency, relation consistency, and schema validity. If needed, it returns revision feedback to the trainable planner $\mathcal{M}_P$. We fine-tune $\mathcal{M}_P$ with GRPO Shao et al. (2024) using rewards for schema validity, turn consistency, temporal persistence, and relation consistency. *Further training details are provided in the Appendix (§ H).*

4 IMPLEMENTATION DETAILS

Our framework builds on a diffusion Transformer Wan et al. (2025) and is trained in three stages: **Stage 1** learns general human motion, scene composition, and graph-conditioned generation on HumanVID. **Stage 2** introduces single-person audio conditioning for synchronized facial and body motion, and **Stage 3** trains on multi-person conversational data. During video training, audio, image, and text conditions are dropped with probability 0.1 for classifier-free guidance. The 42 block video generator is trained on two NVIDIA A100 GPUs and uses 100 denoising steps at inference. For evaluation, we use both real and synthetically generated multi-person reference images to broaden the diversity of evaluated scenes and assess generalization beyond the real benchmark images. Synthetic references are generated using GPT-4o OpenAI et al. (2024) and FLUX Labs et al. (2025). *Additional training and optimization details are provided in the Appendix.*

4.1 DATASETS AND EVALUATION METRICS

Training uses **HumanVID** Wang et al. (2024b) for full-body motion, **Hallo3** Cui et al. (2025) for upper-body motion, **HDTF** Zhang et al. (2021) and **AVSpeech** Ephrat et al. (2018) for face-centric speech, and the **MIT** Zhu et al. (2025) and **InteractiveEyes** Zhong et al. (2025) datasets for two-person conversation. SyncNet-based Chung & Zisserman (2016) audio-visual alignment and

Table 3: **Quantitative comparison on multi-person benchmarks.** Results on InteractiveEyes and MIT. Best results are shown in **bold**.

Method	InteractiveEyes			MIT		
	Inter. ↑	Sync-C ↑	FVD ↓	Inter. ↑	Sync-C ↑	FVD ↓
Bind-Your-Avatar	0.38	2.54	716.3	0.36	2.39	671.8
AnyTalker	0.52	4.81	624.8	0.48	2.97	645.2
MultiTalk	0.74	6.27	571.3	0.62	4.15	**576.9**
ConfabMuse	**0.86**	**6.42**	**539.2**	**0.75**	**4.63**	582.4

Table 4: **Quantitative comparison on EMTD.** Comparison on upper-body avatar generation. Best results are shown in **bold**.

Method	FID ↓	FVD ↓	PSNR ↑	SSIM ↑	CSIM ↑
AnimateAnyone	56.31	871.36	20.63	0.713	0.269
MimicMotion	51.82	614.98	19.71	0.725	0.371
EchoMimicV2	42.69	528.10	22.09	0.741	0.393
SyncDreamer	41.77	483.59	22.38	0.746	0.598
ConfabMuse	**40.26**	**465.49**	**23.85**	**0.762**	**0.635**

InsightFace-based Deng et al. (2019) identity filtering yield approximately 700K training clips. For each dataset, we first split the original source videos into 80% training and 20% evaluation subsets, and then extract clips within each split, ensuring that clips from the same source video do not appear in both sets.

We report FID Heusel et al. (2017), FVD Unterthiner et al. (2019), Sync-C/D Chung & Zisserman (2016), PSNR Horé & Ziou (2010), SSIM Wang et al. (2004), CSIM, IQA Wu et al. (2023), and the eye-focused Interactivity metric Zhong et al. (2025). These metrics do not assess whether a requested multi-person interaction is correctly followed across conversational turns. We therefore introduce four MLLM-based interaction-control metrics: Interaction State Compliance Score (ISCS), Interaction Transition Success (ITS), Gaze-to-Target Accuracy (GTA), and Temporal Interaction Fidelity (TIF), using *GPT-5.6* OpenAI (2026) as the visual evaluator. To assess evaluator dependence, we additionally repeat the interaction-control evaluation with *InternVL3.5* Wang et al. (2025b), obtaining the same method ranking across all four metrics(§ I.6). *Metric definitions and evaluation prompts are provided in the Appendix (§ I).*

5 Results

5.1 Quantitative Results on Single-Person Benchmarks

We first verify that ConfabMuse retains standard single-speaker generation capability. On HDTF, we compare against recent single-speaker audio-driven avatar generation methods, including V-Express Wang et al. (2024a), EchoMimic Chen et al. (2025b), Hallo3 Cui et al. (2025), FantasyTalk Wang et al. (2025a), HunyuanVideo-Avatar Chen et al. (2025a), OmniAvatar Gan et al. (2025), and SyncDreamer Nazarieh et al. (2026), as well as the multi-speaker methods AnyTalker Zhong et al. (2025) and MultiTalk Kong et al. (2025), evaluated under their single-speaker settings. As shown in Table 1, ConfabMuse maintains strong visual quality, temporal consistency, identity preservation, and audio-visual synchronization. *Similar results on AVSpeech are provided in the Appendix (§ B).*

Table 2: **Multi-person motion coordination.** FD, P-FD, and rPCC are reported for expression and head-pose trajectories.

Method	FD ↓		P-FD ↓		rPCC ↓	
	Exp.	Pose $\times 10^2$	Exp.	Pose $\times 10^2$	Exp. $\times 10^2$	Pose $\times 10^1$
AnyTalker	20.31	10.57	26.97	12.69	13.87	9.41
Playmate2	17.84	7.28	23.58	9.37	11.65	7.29
MultiTalk	14.35	4.94	14.61	5.11	7.58	4.83
ConfabMuse	**12.79**	**4.18**	**13.82**	**4.35**	**6.21**	**3.16**

5.2 Quantitative Results on Multi-Person Benchmarks

We evaluate two-person conversational generation on InteractiveEyes and MIT against Bind-Your-Avatar Huang et al. (2025), MultiTalk Kong et al. (2025), and AnyTalker Zhong et al. (2025). As shown in Table 3, ConfabMuse achieves the highest Interactivity score on both benchmarks while maintaining strong audio-visual synchronization and temporal quality. Importantly, Interactivity (Inter.) measures listener-side motion during non-speaking periods, making these gains directly relevant to our goal of modeling responsive behavior beyond active-speaker animation. *Additional qualitative examples of two-person conversation are provided in the Appendix (Figs. 7 and 8).*

5.3 Multi-Person Motion Coordination

We further evaluate ConfabMuse on SpeakerVid (Zhang et al., 2026) against AnyTalker Zhong et al. (2025), Playmate2 Ma et al. (2025b), and MultiTalk Kong et al. (2025), using the same reference image and audio streams. Following Peng et al. (2025), we report FD, P-FD, and rPCC over facial-expression and head-pose trajectories. Table 2 shows that ConfabMuse achieves the lowest scores across all six measurements, indicating stronger cross-avatar motion coordination.

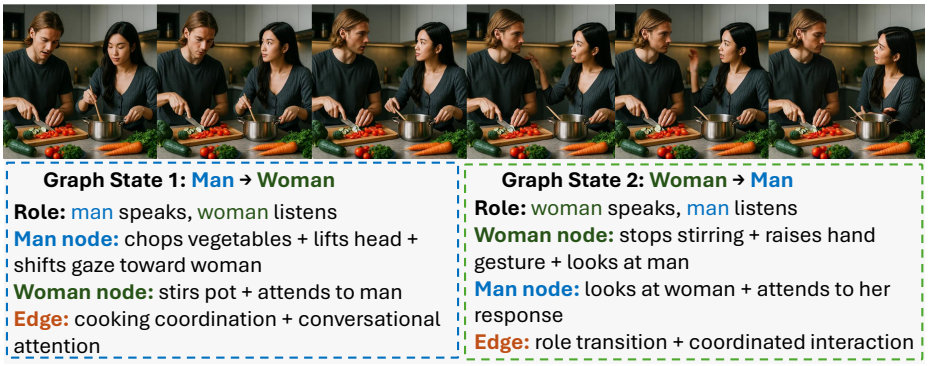

Figure 4: **Controllable multi-person motion with structured interaction guidance.** ConfabMuse follows specified speaker-listener roles and coordinated actions across turns while preserving ongoing behavior.



Figure 5: **Effect of Audio-Vision Binding.** Without binding, audio can leak across avatars, causing unintended listener activation.

5.4 Generation Beyond Face-Centric Portraits

We further evaluate whether ConfabMuse retains strong upper and full-body generation in the single-person setting despite its multi-person design. On EMTD Meng et al. (2025), we compare with recent methods designed for upper and full-body human animation: AnimateAnyone Hu et al. (2024), MimicMotion Zhang et al. (2025), EchoMimicV2 Meng et al. (2025), and SyncDreamer Nazarieh et al. (2026). As shown in Table 4, ConfabMuse achieves the best performance across the reported visual quality, temporal consistency, reconstruction, and identity metrics. These results show that the proposed multi-person framework retains strong single-person upper and full-body generation capability.

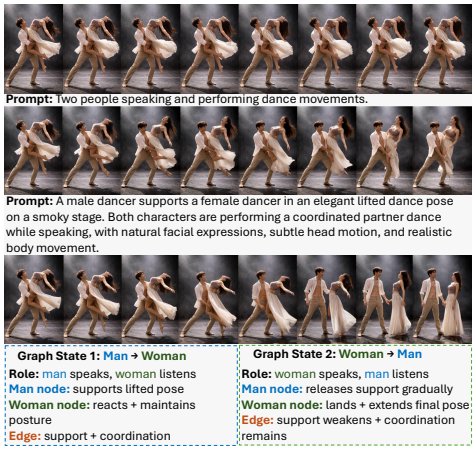

Figure 6: **Effect of interaction guidance.** ConfabMuse better follows the requested interaction between two talking avatars.

5.5 Controllability of Multi-Person Interaction

We next examine whether generated multi-person videos follow the requested interaction and its evolution across conversational turns. We evaluate this on the two-person subset of MIT Zhu et al. (2025). We first assess whether the requested avatar motions and spatial relations are correctly represented, using Motion Binding and Spatial Relationships from T2V-CompBench Sun et al. (2025). We then assess whether interaction states, transitions, attention targets, and timing follow the intended turn sequence, using our proposed MLLM-based metrics ISCS, ITS, GTA, and TIF. As shown in Table 5, ConfabMuse achieves the best performance across all six metrics, demonstrating more accurate interaction content and temporal control. *Additional examples are provided in the Appendix (§ F).*

Table 5: **Quantitative evaluation of interaction controllability on MIT.** Motion Binding and Spatial Relationships follow T2V-CompBench Sun et al. (2025). ISCS, ITS, GTA, and TIF evaluate interaction state, transitions, attention, and temporal control. Higher is better for all metrics.

	Compositional Consistency		Interaction Control			
Method	**Motion Bind. ↑**	**Spatial Rel. ↑**	**ISCS (%) ↑**	**ITS (%) ↑**	**GTA (%) ↑**	**TIF (%) ↑**
AnyTalker	0.28	0.26	56.3	28.0	80.0	22.7
Playmate2	0.41	0.47	60.4	32.0	82.7	38.1
MultiTalk	0.76	0.81	90.6	62.0	93.2	87.9
ConfabMuse	**0.83**	**0.85**	**93.8**	**81.0**	**94.6**	**92.8**

Table 6: **Ablation of the Scene-Grounded Interaction Planner.**

Guidance	**TSA ↑**	**TTE ↓**	**FSC ↑**
Simple Prompt	17.6	N/R	45.0
Expanded Prompt	76.5	0.968 s	82.0
Scene-Grounded Planner	**94.1**	**0.028 s**	**98.0**

Table 7: **Ablation of the Turn-Context Audio Adapter.**

Method	**Sync-C ↑**	**Active-Speaker AP (%) ↑**	**Listener FPR (%) ↓**
Wav2Vec2 only	3.86	89.71	45.52
+ Within-Stream Branch	4.19	90.46	45.27
+ Cross-Stream Branch (Full Adapter)	**5.76**	**94.38**	**13.63**

5.6 Ablation Study

Effect of the Turn-Context Audio Adapter. We progressively ablate the Turn-Context Audio Adapter, starting from direct Wav2Vec2 conditioning and adding within-stream weighting and cross-stream context. As shown in Table 7, direct Wav2Vec2 conditioning performs the weakest, while within-stream weighting provides modest gains. Adding cross-stream context improves active-speaker detection and substantially reduces unintended listener activation, demonstrating its importance for determining when each audio stream should influence speech-related motion. *Metric definitions and qualitative results are provided in the Appendix (§ D.2).*

Effect of Audio-Vision Binding. We disable the learned compatibility bias while keeping all other components unchanged. Without binding, audio can leak across avatars, causing unintended listener activation (Fig. 5). Quantitative results in Appendix E show the same effect, with substantially higher Listener FPR and lower Active-Speaker AP when the compatibility bias is removed.

Effect of the Scene-Grounded Interaction Planner. We isolate the contribution of structured interaction representation by comparing our graph guidance with two text-only alternatives: a simple prompt and a Qwen-expanded prompt Bai et al. (2025), under the same generation setting. As shown in Table 6 and Fig. 6, prompt expansion improves state alignment, but the interaction graph achieves substantially better transition timing and final-state completion. This indicates that increasing prompt detail alone does not reliably specify which avatar behaviors should change, which should persist, or when the transition should occur.

6 Limitations and Applications

Our framework is trained and evaluated for two-person conversations. Extending to larger groups requires suitable multi-party data. Our MLLM-based interaction metrics are additionally evaluated with the open-source InternVL3.5 model, which preserves the method ranking across all four metrics, while human validation remains future work. Potential applications include filmmaking, virtual production, and interactive digital content. *Further discussion is provided in Appendix G.*

7 Conclusion

We presented ConfabMuse, a framework for controllable two-person conversational avatar generation that jointly models *when* each audio stream should influence motion, *who* it should animate, and *how* avatar behavior should evolve across turns. By combining the Turn-Context Audio Adapter, Audio-Vision Binding, and Scene-Grounded Interaction Planner, ConfabMuse coordinates speaker turns, stream-to-avatar correspondence, and evolving avatar actions and relations while preserving ongoing behaviors. Across single and multi-person benchmarks, ConfabMuse improves conversational interactivity and interaction control while maintaining strong audio-visual synchronization and generation quality.

8 AI USE STATEMENT

Generative AI tools were used to assist with manuscript editing. GPT-4o OpenAI et al. (2024) and FLUX Labs et al. (2025) were used to generate a subset of synthetic multi-person reference images for evaluation beyond the benchmark datasets. GPT-5.6 Sol was used as the fixed visual evaluator for the MLLM-based interaction-control metrics described in the paper. Gemini 2.5 Pro was used to generate structured interaction-graph annotations used during training.

REFERENCES

Alexei Baevski, Henry Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations, 2020.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, et al. Qwen3-vl technical report, 2025. URL https://arxiv.org/abs/2511.21631.

Yi Chen, Sen Liang, Zixiang Zhou, et al. Hunyuanvideo-avatar: High-fidelity audio-driven human animation for multiple characters, 2025a. URL https://arxiv.org/abs/2505.20156.

Zhiyuan Chen, Jiajiong Cao, Zhiquan Chen, et al. Echomimic: lifelike audio-driven portrait animations through editable landmark conditions. AAAI, 2025b. doi: 10.1609/aaai.v39i3.32241.

J. S. Chung and A. Zisserman. Out of time: automated lip sync in the wild. In *Workshop on Multi-view Lip-reading, ACCV*, 2016.

Jiahao Cui, Hui Li, Yun Zhan, et al. Hallo3: Highly dynamic and realistic portrait image animation with video diffusion transformer. In *CVPR*, pp. 21086–21095, June 2025.

Jiankang Deng, Jia Guo, Xue Niannan, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *CVPR*, 2019.

Ariel Ephrat, Inbar Mosseri, Oran Lang, et al. Looking to listen at the cocktail party: a speaker-independent audio-visual model for speech separation. *ACM Trans. Graph.*, 2018. doi: 10.1145/3197517.3201357.

Qijun Gan, Ruizi Yang, Jianke Zhu, et al. Omniavatar: Efficient audio-driven avatar video generation with adaptive body animation, 2025. URL https://arxiv.org/abs/2506.18866.

Shan He, Haonan He, Shuo Yang, et al. Speech4mesh: Speech-assisted monocular 3d facial reconstruction for speech-driven 3d facial animation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 14192–14202, 2023.

Martin Heusel, Hubert Ramsauer, et al. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Alain Horé and Djemel Ziou. Image quality metrics: Psnr vs. ssim. In *ICPR*, pp. 2366–2369, 2010.

Li Hu, Xin Gao, Peng Zhang, et al. Animate anyone: Consistent and controllable image-to-video synthesis for character animation, 2024. URL https://arxiv.org/abs/2311.17117.

Yubo Huang, Weiqiang Wang, Sirui Zhao, Tong Xu, Lin Liu, and Enhong Chen. Bind-your-avatar: Multi-talking-character video generation with dynamic 3d-mask-based embedding router, 2025. URL https://arxiv.org/abs/2506.19833.

Taekyung Ki, Sangwon Jang, Jaehyeong Jo, Jaehong Yoon, and Sung Ju Hwang. Avatar forcing: Real-time interactive head avatar generation for natural conversation, 2026. URL https://arxiv.org/abs/2601.00664.

Zhe Kong, Feng Gao, Yong Zhang, Zhuoliang Kang, Xiaoming Wei, Xunliang Cai, Guanying Chen, and Wenhan Luo. Let them talk: Audio-driven multi-person conversational video generation, 2025. URL https://arxiv.org/abs/2505.22647.

Black Forest Labs, Stephen Batifol, Andreas Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, Jack English, Zion English, Patrick Esser, Sumith Kulal, Kyle Lacey, Yam Levi, Cheng Li, Dominik Lorenz, Jonas Müller, Dustin Podell, Robin Rombach, Harry Saini, Axel Sauer, and Luke Smith. Flux.1 kontext: Flow matching for in-context image generation and editing in latent space, 2025. URL https://arxiv.org/abs/2506.15742.

Junhua Liao, Haihan Duan, Kanghui Feng, Wanbing Zhao, Yanbing Yang, and Liangyin Chen. A light weight model for active speaker detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22932–22941, 2023.

Gaojie Lin, Jianwen Jiang, Chao Liang, et al. Cyberhost: Taming audio-driven avatar diffusion model with region codebook attention, 2025. URL https://arxiv.org/abs/2409.01876.

Xingpei Ma, Jiaran Cai, Yuansheng Guan, Shenneng Huang, Qiang Zhang, and Shunsi Zhang. Playmate: Flexible control of portrait animation via 3d-implicit space guided diffusion, 2025a. URL https://arxiv.org/abs/2502.07203.

Xingpei Ma, Shenneng Huang, Jiaran Cai, Yuansheng Guan, Shen Zheng, Hanfeng Zhao, Qiang Zhang, and Shunsi Zhang. Playmate2: Training-free multi-character audio-driven animation via diffusion transformer with reward feedback, 2025b. URL https://arxiv.org/abs/2510.12089.

Rang Meng, Xingyu Zhang, Yuming Li, and Chenguang Ma. Echomimicv2: Towards striking, simplified, and semi-body human animation. In *CVPR*, pp. 5489–5498, 2025.

Fatemeh Nazarieh, Zhenhua Feng, Muhammad Awais, et al. A survey of cross-modal visual content generation. *IEEE TCSVT*, pp. 6814–6832, 2024. doi: 10.1109/TCSVT.2024.3351601.

Fatemeh Nazarieh, Zhenhua Feng, Diptesh Kanojia, Josef Kittler, and Muhammad Awais. Syncdreamer: Controllable and expressive avatar generation beyond the talking head. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 25121–25130, 2026.

OpenAI. GPT-5.6 System Card. https://deploymentsafety.openai.com/gpt-5-6, July 2026. Published July 9, 2026.

OpenAI, :, Aaron Hurst, Adam Lerer, et al. Gpt-4o system card, 2024. URL https://arxiv.org/abs/2410.21276.

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 (CVPR)*, pp. 21055–21064, 2025.

K R Prajwal, Rudrabha Mukhopadhyay, Vinay P. Namboodiri, and C.V. Jawahar. A lip sync expert is all you need for speech to lip generation in the wild. In *ACM MM*, pp. 484–492, 2020.

Alexander Richard, Michael Zollhoefer, Yandong Wen, Fernando De la Torre, and Yaser Sheikh. Meshtalk: 3d face animation from speech using cross-modality disentanglement. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1153–1162, 2021.

Joseph Roth, Sourish Chaudhuri, Ondrej Klejch, Radhika Marvin, Andrew Gallagher, Liat Kaver, Sharadh Ramaswamy, Arkadiusz Stopczynski, Cordelia Schmid, Zhonghua Xi, and Caroline Pantofaru. Ava-activespeaker: An audio-visual dataset for active speaker detection, 2019. URL https://arxiv.org/abs/1901.01342.

Mengyi Shan, Shouchieh Chang, Ziqian Bai, Shichen Liu, Yinda Zhang, Luchuan Song, Rohit Pandey, Sean Fanello, and Zeng Huang. Talking together: Synthesizing co-located 3d conversations from audio, 2026. URL https://arxiv.org/abs/2603.08674.

Zhihong Shao, Peiyi Wang, Qihao Zhu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models, 2024. URL https://arxiv.org/abs/2402.03300.

Kaiyue Sun, Kaiyi Huang, Xian Liu, Yue Wu, Zihan Xu, Zhenguo Li, and Xihui Liu. T2v-compbench: A comprehensive benchmark for compositional text-to-video generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8406–8416, 2025.

Silero Team. Silero vad: pre-trained enterprise-grade voice activity detector (vad), number detector and language classifier. https://github.com/snakers4/silero-vad, 2024.

Thomas Unterthiner, Sjoerd van Steenkiste, et al. Towards accurate generative models of video: A new metric & challenges, 2019. URL https://arxiv.org/abs/1812.01717.

Team Wan, Ang Wang, Baole Ai, et al. Wan: Open and advanced large-scale video generative models, 2025. URL https://arxiv.org/abs/2503.20314.

Cong Wang, Kuan Tian, Jun Zhang, et al. V-express: Conditional dropout for progressive training of portrait video generation. 2024a.

Mengchao Wang, Qiang Wang, Fan Jiang, et al. Fantasytalking: Realistic talking portrait generation via coherent motion synthesis, 2025a. URL https://arxiv.org/abs/2504.04842.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, et al. Internvl3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency, 2025b. URL https://arxiv.org/abs/2508.18265.

Zhenzhi Wang, Yixuan Li, Yanhong Zeng, et al. Humanvid: Demystifying training data for camera-controllable human image animation. In *NeurIPS*, 2024b.

Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE TIP*, pp. 600–612, 2004.

Haoning Wu, Zicheng Zhang, Weixia Zhang, et al. Q-align: Teaching lmms for visual scoring via discrete text-defined levels, 2023. URL https://arxiv.org/abs/2312.17090.

Youliang Zhang, Zhaoyang Li, Duomin Wang, jiahe zhang, Deyu Zhou, Zixin Yin, Xili Dai, Gang Yu, and Xiu Li. Speakervid-5m: A large-scale high-quality dataset for audio-visual dyadic interactive human generation. In *International Conference on Learning Representations*, pp. 117896–117926, 2026.

Yuang Zhang, Jiaxi Gu, Li-Wen Wang, et al. Mimicmotion: High-quality human motion video generation with confidence-aware pose guidance. In *ICML*, 2025.

Zhimeng Zhang, Lincheng Li, Yu Ding, and Changjie Fan. Flow-guided one-shot talking face generation with a high-resolution audio-visual dataset. In *CVPR*, pp. 3661–3670, 2021.

Zhizhou Zhong, Yicheng Ji, Zhe Kong, et al. Anytalker: Scaling multi-person talking video generation with interactivity refinement, 2025. URL https://arxiv.org/abs/2511.23475.

Yang Zhou, Xintong Han, Eli Shechtman, et al. Makelttalk: speaker-aware talking-head animation. 2020.

Zeyu Zhu, Weijia Wu, and Mike Zheng Shou. Multi-human interactive talking dataset, 2025. URL https://arxiv.org/abs/2508.03050.

A Dataset Preparation and Training Details

Training data and splits. Our training corpus combines publicly available audio–visual datasets covering full-body, upper-body, face-centric, and two-person conversational settings. Specifically, we use HumanVID Wang et al. (2024b) for full-body motion, Hallo3 Cui et al. (2025) for expressive upper-body motion, HDTF Zhang et al. (2021) and AVSpeech Ephrat et al. (2018) for face-centric speech, and the Multi-human Interactive Talking (MIT) Zhu et al. (2025) and InteractiveEyes Zhong et al. (2025) datasets for two-person conversational training.

For each dataset, we first split the original source videos into 80% training and 20% evaluation subsets. We then segment the videos within each split into clips of 5–40 s, ensuring that clips from the same source video do not appear in both training and evaluation sets. We apply SyncNet-based Chung & Zisserman (2016) audio–visual alignment filtering and InsightFace-based Deng et al. (2019) identity filtering, resulting in approximately 700K training clips.

Interaction-graph annotations. All training datasets are augmented with structured interaction-graph annotations. During the first 100 epochs, we use Gemini 2.5 Pro to generate graph annotations from the training samples. These annotations provide stable structured supervision that allows the video generator to learn the correspondence between graph states and the visual content and motion of the target video.

After this initial stage, we fine-tune the Scene-Grounded Interaction Planner to generate the structured graph conditions used by ConfabMuse. For the remaining 100 epochs, we continue training the video generator using graph conditions produced by the fine-tuned planner. This reduces the discrepancy between the graph conditions observed during training and those produced by the planner at inference. Planner-specific GRPO configuration and reward definitions are provided in Appendix H.

Graph annotation prompt. For each training sample, Gemini 2.5 Pro receives an extracted frame and the following instruction:

Gemini 2.5 Pro prompt for graph annotation

```
You are given a single frame extracted from a human video. Construct a
structured interaction graph describing only the information that is
    directly
visible in the image.

Identify each visible person as P1, P2, etc., ordered from left to
    right.
For each person, describe:
- image region (left, center, or right);
- visible appearance and clothing;
- body pose;
- visible action;
- gaze or attention direction;
- nearby or interacted-with objects;
- role in the visible scene.

Also identify visible relations between people, such as attending to,
coordinating with, supporting, touching, or interacting with the same
    object.

Use concise descriptions. Do not infer speech, intentions, emotions,
    future
actions, hidden objects, or information that is not visually supported
     by the
image. If an attribute cannot be determined reliably, use "unknown".

Return valid JSON only, following this structure:

{
  "graph_id": "G0",
```



Figure 7: **Qualitative comparison on two-person conversational generation.** Our method better follows speaker turns through coherent gaze and head orientation changes while maintaining avatar identity and scene consistency. Speaker icons indicate the active speaker.

```
  "type": "static_reference_graph",
  "participants": [
    {
      "id": "P1",
      "region": "<left|center|right>",
      "appearance": "<visible appearance>",
      "pose": "<visible body pose>",
      "visible_action": "<currently visible action>",
      "gaze": "<visible attention direction>",
      "nearby_objects": ["<object>", "..."],
      "scene_role": "<role in the visible scene>"
    }
  ],
  "edges": [
    {
      "source": "<participant_id>",
      "target": "<participant_id or object>",
      "relation": "<visible relation>"
    }
  ]
}

Do not output explanatory text outside the JSON.
```

Training configuration. The video generator contains 42 Transformer blocks and is trained in three stages. Stage 1 learns general human motion, scene composition, and graph-conditioned generation. Stage 2 introduces single-person audio conditioning, and Stage 3 extends training to multi-person conversational video. We use AdamW with a batch size of 8 for 200 epochs and an initial learning rate of 1×10^{-4}, reduced to 5×10^{-5} when training instability is observed. Audio, image, and text conditions are dropped with probability 0.1 for classifier-free guidance. Training takes approximately 25 days on two NVIDIA A100 GPUs. At inference, we use 100 denoising steps. Generating a 20 s video at 25 FPS takes approximately 35 minutes on a single A100 GPU.

B Quantitative Evaluation on a Single-Speaker Benchmark

To complement the HDTF results reported in the main paper, we further evaluate ConfabMuse on AVSpeech Ephrat et al. (2018). This experiment examines whether the proposed conversational components retain standard single-speaker generation capability on a second face-centric benchmark. We compare against V-Express Wang et al. (2024a), EchoMimic Chen et al. (2025b), Hallo3 Cui et al. (2025), FantasyTalk Wang et al. (2025a), HunyuanVideo-Avatar Chen et al. (2025a), AnyTalker Zhong et al. (2025), OmniAvatar Gan et al. (2025), MultiTalk Kong et al. (2025), and SyncDreamer Nazarieh et al. (2026).

Table 8: **Quantitative comparison on AVSpeech.** Best results are shown in **bold**.

Method	FID↓	FVD↓	Sync-C↑	Sync-D↓	IQA↑
V-Express	89.5	1531	3.74	9.15	2.48
EchoMimic	71.5	83.1	5.33	8.14	2.61
Hallo3	71.8	81.9	5.86	7.28	2.66
FantasyTalk	83.6	91.3	4.07	8.58	2.43
HunyuanVideo-Avatar	69.4	79.5	5.91	7.21	2.71
AnyTalker	69.2	79.1	6.17	6.95	2.74
OmniAvatar	68.4	73.8	6.24	6.73	2.78
MultiTalk	68.6	73.2	6.43	6.51	2.76
SyncDreamer	67.9	72.6	6.31	7.05	2.81
ConfabMuse	**66.3**	**71.5**	**6.48**	**6.23**	**3.52**

As shown in Table 8, ConfabMuse achieves the best performance across the reported visual-quality, temporal-consistency, and audio-visual synchronization metrics. Together with the HDTF results in Table 1, these results show that ConfabMuse retains strong single-speaker generation capability despite its additional multi-person conversational components.

C Additional Multi-Person Results

Figures 7 and 8 provide additional qualitative comparisons on two-person conversational generation. Figure 7 illustrates turn-dependent gaze and head-orientation changes, while Fig. 8 considers a continuous-walking scenario in which the ongoing action must be preserved as speaker roles change. ConfabMuse maintains the walking motion while adapting orientation and responses across turns, whereas the compared baselines show weaker preservation of the requested ongoing action.

D Further Analysis of the Turn-Context Audio Adapter

Figure 9 provides a detailed view of the proposed Turn-Context Audio Adapter. Starting from the Wav2Vec2 features of the two driving audio streams, the adapter models two complementary aspects of conversational speech. The within-stream branch analyzes each stream using local temporal context and estimates a stream-specific weight reflecting short-term speech dynamics. The cross-stream branch reasons over speech activity across streams to distinguish local pauses from speaker handovers and estimates a turn-context gate for each stream. The within-stream weight and cross-stream gate are applied sequentially to produce turn-aware audio features. The adapter therefore determines *when* and *how strongly* each audio stream should influence speech-related motion, while stream-to-avatar correspondence is handled separately by Audio-Vision Binding.

D.1 Evaluation Metrics

We evaluate the Turn-Context Audio Adapter using three complementary measures of speech synchronization, active-speaker identification, and unintended listener activation.

Sync-C. Sync-C measures audio-visual synchronization for the speaking avatar. We compute it over the corresponding speaking segments using SyncNet Chung & Zisserman (2016). Higher Sync-C indicates stronger alignment between the driving speech and generated mouth motion.

Active-Speaker AP. We use a pretrained Light-ASD model Liao et al. (2023) to obtain a frame-level speaking-confidence score for each avatar. The expected active speaker is determined from the audio-derived turn schedule. Following the active-speaker evaluation protocol Roth et al. (2019), we report Average Precision (AP), which measures how well true speaking samples are ranked above non-speaking samples. Higher AP indicates more reliable active-speaker identification.

Listener FPR. Listener false-positive rate (FPR) measures unintended speaker activation in the listening avatar. During intervals in which one avatar is speaking, we measure how often Light-ASD

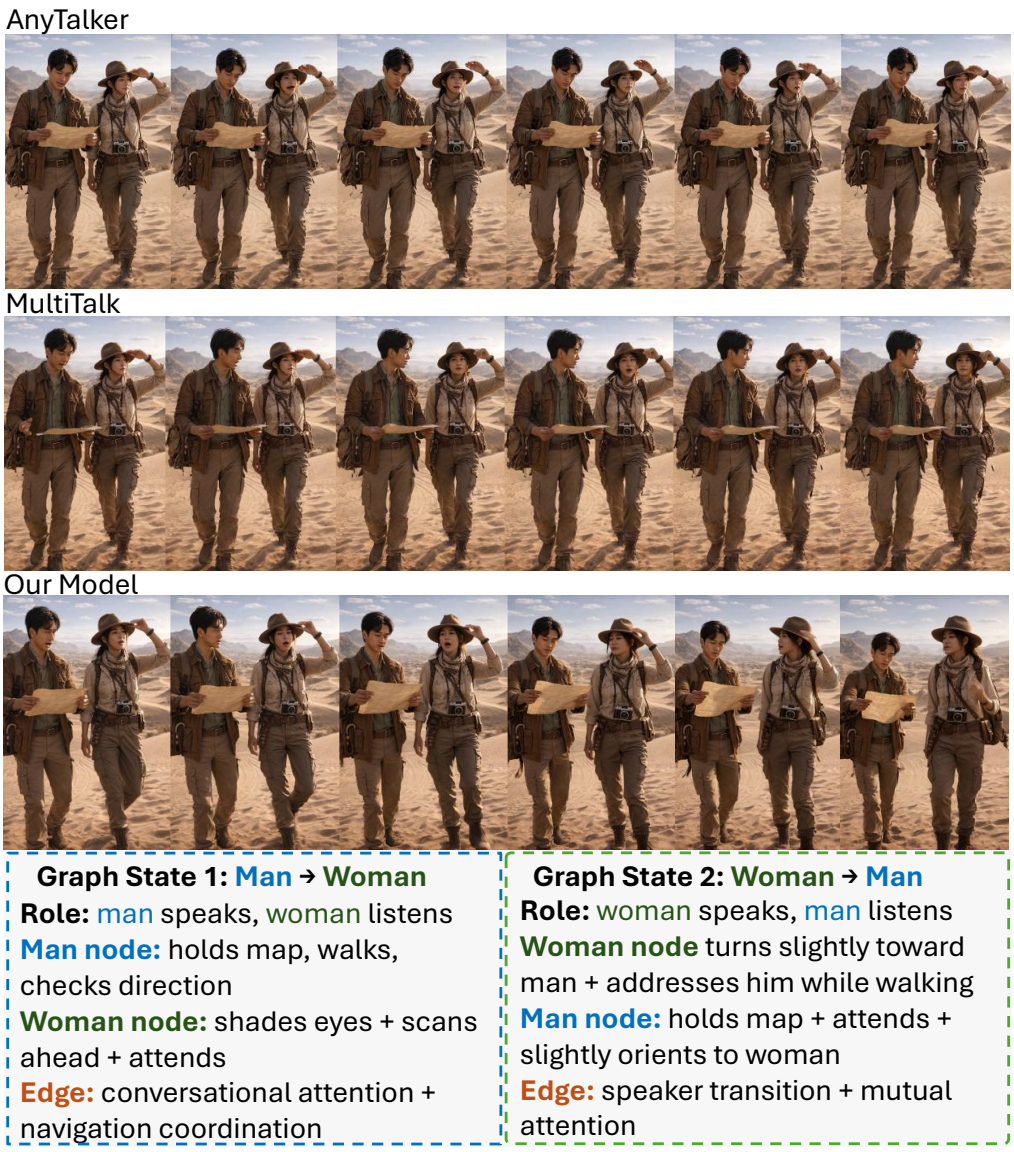

Figure 8: **Two-person conversational generation.** Our method preserves walking motion while adapting gaze, orientation, and responses across speaker turns.

incorrectly classifies the other avatar as speaking:

$$\text{Listener FPR} = \frac{N_{\text{listener predicted as speaking}}}{N_{\text{listener samples}}}. \tag{5}$$

Lower Listener FPR indicates fewer cases of unintended speech-related motion in the listening avatar.

D.2 Ablation Analysis

The quantitative ablation reported in the main paper isolates the contributions of the within- and cross-stream branches. Direct Wav2Vec2 conditioning obtains a Sync-C of 3.86, an Active-Speaker AP of 89.71%, and a Listener FPR of 45.52%. Adding the within-stream branch improves Sync-C to 4.19 and Active-Speaker AP to 90.46%, while Listener FPR changes only marginally, from 45.52% to 45.27%. This suggests that modeling local speech dynamics improves speech-conditioned motion and synchronization, but provides limited information for distinguishing local pauses from speaker transitions.

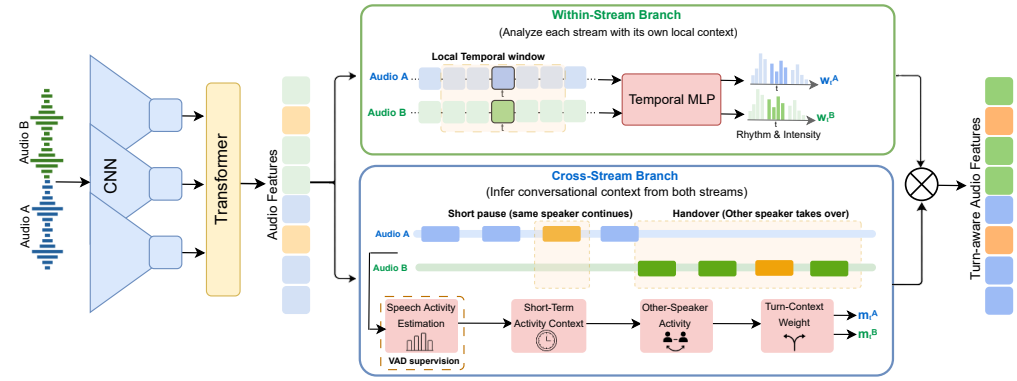

Figure 9: **Turn-Context Audio Adapter.** The within-stream branch models local speech dynamics to estimate stream-specific weights w^A and w^B, while the cross-stream branch uses activity across both streams to distinguish local pauses from speaker handovers and estimate turn-context gates m^A and m^B. The two are applied sequentially to produce turn-aware audio features.



Figure 10: **Effect of the Turn-Context Audio Adapter.** Direct Wav2Vec2 conditioning can produce unintended listener activation around pauses and speaker transitions. The full adapter better suppresses inactive-stream influence, resulting in cleaner turn transitions and more stable speaker–listener behavior.

Adding the cross-stream branch produces a substantially larger improvement. Relative to the within-stream variant, Sync-C increases from 4.19 to 5.76 and Active-Speaker AP from 90.46% to 94.38%, while Listener FPR decreases from 45.27% to 13.63%. The 31.64 percentage-point reduction in Listener FPR highlights the benefit of reasoning jointly over activity across streams: low activity in one stream can be interpreted differently depending on whether the competing stream remains inactive or becomes active. The full adapter therefore improves both speech synchronization and the temporal selectivity of audio conditioning across speaker transitions.

The qualitative comparison in Fig. 10 follows the same trend as the quantitative ablation. Direct Wav2Vec2 conditioning can activate the listening avatar around pauses and turn transitions, whereas the full adapter produces clearer turn transitions and reduces unintended listener activation. These observations further support the role of cross-stream context in determining *when* each audio stream should influence speech-driven motion.

Table 9: **Quantitative ablation of Audio-Vision Binding.** Disabling the learned compatibility bias weakens active-speaker identification and substantially increases unintended listener activation.

Method	Sync-C ↑	Active-Speaker AP (%) ↑	Listener FPR (%) ↓
w/o Compatibility Bias ($\lambda_b = 0$)	1.26	48.71	75.19
Full model	5.76	94.38	13.63

E Quantitative Analysis of Audio-Vision Binding

To quantify the contribution of Audio-Vision Binding, we disable the learned compatibility bias in Eq. 2 by setting $\lambda_b = 0$, while keeping all other model components and evaluation settings unchanged. We use the same evaluation protocol as for the Turn-Context Audio Adapter, reporting Sync-C, Active-Speaker AP, and Listener FPR.

As shown in Table 9, disabling the compatibility bias substantially degrades stream-to-avatar association. Active-Speaker AP decreases from 94.38% to 48.71%, while Listener FPR increases from 13.63% to 75.19%, indicating frequent unintended activation of the listening avatar. Sync-C also decreases from 5.76 to 1.26. These results complement the qualitative example in Fig. 5 and show that learned audio-visual compatibility is important for directing each audio stream toward the corresponding avatar.

F Controllability of Interaction-Guided Generation

Figure 11 provides an additional example of interaction-guided generation in a tennis scene. In the first graph state, the woman speaks while holding the ball and directing her attention toward the man, who remains in a listening state. After the turn transition, the man becomes the speaker, turns toward the woman, and addresses her, while the woman becomes the listener and continues to attend to him. The generated sequence follows the specified speaker-listener transition while preserving behaviors that are not required to change.

Such interactions can be difficult to specify using a single global text prompt because different avatar states must change at different conversational times while other behaviors remain unchanged. The interaction graph instead explicitly specifies *which avatar should change*, *what should change*, and *which behaviors should persist* across conversational states. This provides structured control over the evolution of the interaction while retaining the generative flexibility of the video model.

G Limitations

Our framework is currently trained and evaluated for two-person conversations; extending it to larger groups will require suitable multi-party data and further study of stream-to-avatar binding. Longer interactions involving repeated turn-taking across multiple conversational turns also remain to be explored. Video generation remains computationally expensive, requiring approximately 35 minutes to generate a 20 s video at 25 FPS on a single NVIDIA A100 GPU. Our interaction-control metrics rely on GPT-5.6 Sol as the primary fixed visual evaluator with predefined interaction requirements and evaluation prompts. Although our cross-evaluator analysis with InternVL3.5 shows consistent trends, MLLM judgments may still be affected by evaluator-specific biases and ambiguity in visually subtle behaviors such as gaze and short interaction transitions. Human validation of these metrics therefore remains an important direction for future work.

H Scene-Grounded Interaction Planner

We fine-tune the interaction planner $\mathcal{M}_P$ using GRPO Shao et al. (2024). For each conversational segment, the planner receives the reference image I, the static reference graph $\mathcal{G}^{(0)}$, the previous graph state $\mathcal{G}^{(j-1)}$, the audio-derived turn state ρ_j, and an optional text prompt $\mathcal{T}_{\text{text}}$. The planner samples a group of candidate graph edits, which are evaluated using task-specific interaction rewards. GRPO updates the planner to favor candidates with higher rewards relative to the sampled group. The overall training procedure is illustrated in Fig. 12.

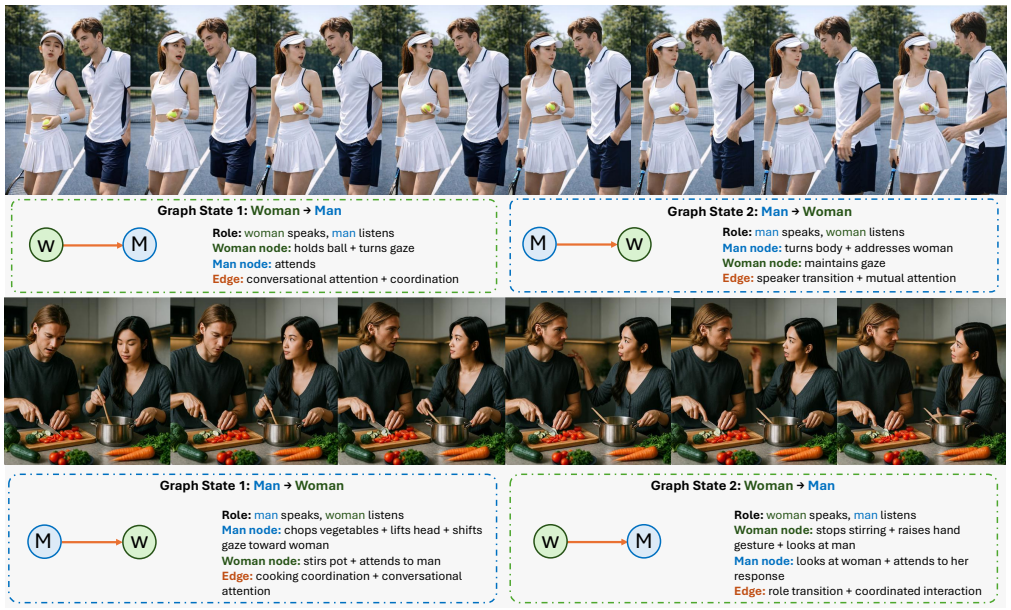

Figure 11: **Controllable interaction transition in a tennis scene.** The structured interaction graph specifies speaker–listener roles, avatar-specific actions, and their transitions across conversational states. ConfabMuse follows the requested state change while preserving behaviors that remain valid across the interaction.

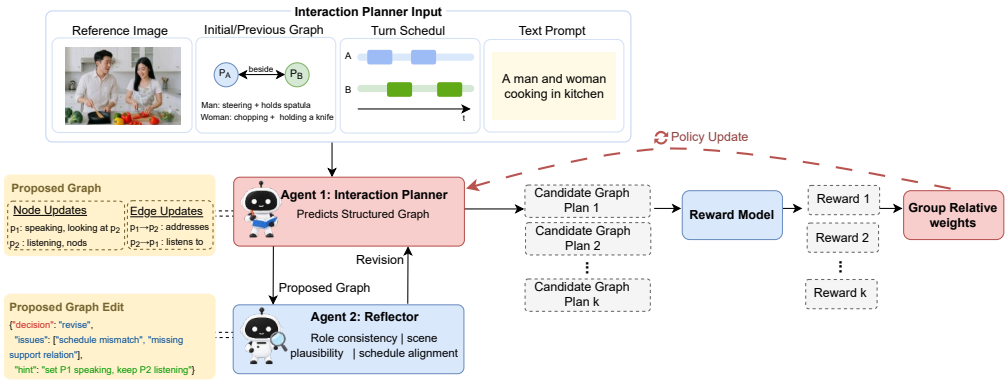

Figure 12: **GRPO training of the Scene-Grounded Interaction Planner.** Given the reference image, previous graph state, turn schedule, and optional text prompt, the planner samples candidate graph edits. The candidates are evaluated using task-specific interaction rewards, and their group-relative scores guide the planner update. A frozen reflector evaluates proposed edits for scene grounding and interaction consistency and provides revision feedback.

GRPO configuration. We LoRA-fine-tune Qwen3-VL-8B using AdamW with a learning rate of 1×10^{-5} for 5 epochs. For each input, we sample eight candidate graph edits for group-relative optimization. The reflector remains frozen throughout training.

For each candidate graph edit, the total reward is

$$R = \lambda_{\text{str}} R_{\text{str}} + \lambda_{\text{sch}} R_{\text{sch}} + \lambda_{\text{grd}} R_{\text{grd}} + \lambda_{\text{tmp}} R_{\text{tmp}} + \lambda_{\text{rel}} R_{\text{rel}}, \tag{6}$$

where the five terms evaluate structural validity, turn-schedule alignment, scene grounding, temporal persistence, and interaction-relation consistency, respectively. We set $\lambda_{\text{str}} = 0.10$, $\lambda_{\text{sch}} = 0.30$, $\lambda_{\text{grd}} = 0.25$, $\lambda_{\text{tmp}} = 0.20$, and $\lambda_{\text{rel}} = 0.15$ in all experiments.

H.1 Structural Validity

R_{str} measures whether the predicted segment-level graph follows the predefined graph schema and can be parsed and applied by the graph-update procedure. The expected structure contains the fields `graph_id`, `type`, `segment`, `based_on`, `node_updates`, `edge_updates`, and `scene_updates`. We check required fields, correct data types, valid avatar and graph identifiers, permitted role and relation labels, valid edge endpoints, and the absence of unknown fields. The reward is

$$R_{\text{str}} = g_{\text{parse}} \frac{1}{|\mathcal{C}_{\text{str}}|} \sum_{c \in \mathcal{C}_{\text{str}}} \mathbb{I}[c \text{ is satisfied}], \tag{7}$$

where $\mathcal{C}_{\text{str}}$ denotes the set of structural constraints and $g_{\text{parse}} = 1$ if the output is valid JSON and 0 otherwise. Malformed JSON therefore receives zero reward, while valid JSON with partial schema violations receives credit according to the fraction of satisfied constraints.

H.2 Turn-Schedule Alignment

R_{sch} measures whether the predicted graph follows the audio-derived speaker schedule for the current segment. We evaluate both the speaker-listener roles and their corresponding conversational relations.

Role alignment is computed as

$$R_{\text{role}} = \frac{\text{number of avatars with correct roles}}{\text{total number of avatars}}. \tag{8}$$

For our two-avatar setting, $R_{\text{role}} = 1$ when both roles are correct, 0.5 when one is correct, and 0 otherwise.

We additionally evaluate whether the predicted conversational edges agree with the relations implied by the scheduled roles, such as `speaking-to` and `listening-to`. Let $\mathcal{E}_j^*$ denote the expected conversational edges for segment j and $\hat{\mathcal{E}}_j$ the predicted edges. Edge alignment is measured as

$$R_{\text{edge}} = F_1(\hat{\mathcal{E}}_j, \mathcal{E}_j^*). \tag{9}$$

The final schedule-alignment reward is

$$R_{\text{sch}} = 0.7R_{\text{role}} + 0.3R_{\text{edge}}. \tag{10}$$

Speaker-listener roles receive greater weight because they are directly specified by the audio-derived turn schedule.

H.3 Scene Grounding

R_{grd} measures whether the predicted graph edit is supported by the reference image, static graph $\mathcal{G}^{(0)}$, and optional text prompt, without introducing unsupported actions, objects, relations, or scene changes. The reward combines structured attribute grounding with a plausibility score from the frozen reflector.

For structured grounding, R_{req} measures whether the required actions and relations specified by the graph and prompt are respected, while R_{sup} measures whether newly introduced graph attributes are supported by the available scene information. We combine them as

$$R_{\text{attr}} = 0.6R_{\text{req}} + 0.4R_{\text{sup}}. \tag{11}$$

The frozen reflector $\mathcal{M}_R$ additionally evaluates whether the proposed edit is visually plausible with respect to the reference scene, including pose and body consistency, action-object compatibility, physical interactions, and scene consistency. It produces a normalized score $R_{\text{VLM}} \in [0, 1]$. The final grounding reward is

$$R_{\text{grd}} = 0.6R_{\text{attr}} + 0.4R_{\text{VLM}}. \tag{12}$$

H.4 TEMPORAL PERSISTENCE

R_{tmp} measures whether the predicted graph changes only the attributes required by the current conversational segment while preserving information that should remain unchanged. We first compute

$$R_{\text{keep}} = \frac{\text{number of correctly preserved attributes}}{\text{number of attributes expected to remain unchanged}}, \tag{13}$$

where preserved attributes include appearance, ongoing actions, posture, held objects, and physical relations inherited from $\mathcal{G}^{(j-1)}$.

We further penalize unnecessary graph changes using

$$R_{\text{edit}} = \exp(-\alpha N_{\text{unjust}}), \tag{14}$$

where N_{unjust} is the number of graph changes not justified by the current interaction segment and $\alpha = 0.5$ controls the penalty strength. The final temporal-persistence reward is

$$R_{\text{tmp}} = 0.7R_{\text{keep}} + 0.3R_{\text{edit}}. \tag{15}$$

H.5 INTERACTION-RELATION CONSISTENCY

R_{rel} measures whether the roles, actions, and relations in the complete interaction graph are logically consistent. We define a set of rule-based consistency constraints. For example, if avatar P_1 `addresses` P_2, then P_1 should be the speaker; if P_2 `listens-to` P_1, then P_2 should be the listener and P_1 should be an active speaker. We further check that relations connect valid avatars, duplicate relations are absent, symmetric relations such as `coordinates-with` are represented consistently, mutually exclusive roles or incompatible actions are not assigned to the same avatar, and persistent physical relations are not removed without justification.

The reward is the fraction of consistency constraints that are satisfied:

$$R_{\text{rel}} = \frac{1}{|\mathcal{C}_{\text{rel}}|} \sum_{c \in \mathcal{C}_{\text{rel}}} \mathbb{I}[c \text{ is satisfied}], \tag{16}$$

where $\mathcal{C}_{\text{rel}}$ denotes the predefined set of relation and state consistency rules. This reward is computed directly from the structured graph and does not require an MLLM judgment.

I INTERACTION CONTROL EVALUATION PROTOCOL

This section describes the evaluation protocol used to measure whether a generated multi-person video follows the interaction states specified by the *interaction control graph*. The protocol evaluates four complementary aspects of interaction control: *whether the requested interaction state is satisfied*, *whether the requested state transitions occur*, *whether each avatar attends to the specified target*, and *whether the interaction evolves according to the intended temporal schedule*.

For all primary MLLM-based evaluations, we use GPT-5.6 Sol as a fixed visual evaluator. For each evaluation, the model is provided with the reference image to establish avatar identity, chronologically sampled frames from the generated video, and the avatar-specific roles, actions, attention targets, and interaction relations specified by the interaction control graph.

I.1 EVALUATION INPUTS AND GRAPH STATES

Each evaluation example includes a reference image, a generated video, a static reference graph $G^{(0)}$, a sequence of interaction graphs $\{G^{(1)}, \ldots, G^{(M)}\}$, and the audio-derived turn boundaries that specify when the interaction should change from one graph state to the next.

The reference graph $G^{(0)}$ identifies the participants and describes the initial scene. Each subsequent graph contains only the attributes that change during that interaction segment. To obtain the complete state used for evaluation, unchanged attributes are carried forward from the previous graph state. This is the same update procedure used by the interaction planner.

For example, if $G^{(2)}$ changes a participant's speaking state and gaze target but does not change an existing physical-support relation, the support relation is carried over from $G^{(1)}$. The generated video is therefore evaluated against the complete graph state, rather than only the attributes that were changed.

We then convert each complete graph state into individual requirements that can be checked separately in the video. This avoids combining several behaviors into a single judgment. The requirements are grouped into four categories: *role*, *action*, *gaze target*, and *interaction relation*. Static properties such as clothing, background appearance, and participant gender are not evaluated by these interaction-control metrics.

Example. Suppose a graph state contains the description

$$P_2 \text{ speaks while turning toward } P_1 \text{ and completing the landing.}$$

Rather than evaluating this description as a single requirement, we separate it into three independently verifiable requirements:

1. P_2 is speaking.
2. P_2 is oriented toward P_1.
3. P_2 completes the landing.

This allows each requested behavior to be evaluated separately, so failure of one requirement does not automatically determine the outcome of the others.

I.2 Interaction State Compliance Score

Interaction State Compliance Score (ISCS) measures how well the visible interaction in the generated video matches the interaction specified by a graph state. For each state, we convert the graph into individual requirements from four categories: *role*, *action*, *gaze target*, and *interaction relation*. GPT-5.6 evaluates each requirement as *yes*, *no*, or *uncertain*. A *yes* indicates that the requested behavior is visibly present, while *no* and *uncertain* are counted as unsatisfied.

For each category k, we compute the proportion of requirements that are satisfied:

$$C_k = \frac{N_k^{\text{yes}}}{N_k^{\text{total}}}, \tag{17}$$

where N_k^{yes} is the number of requirements judged as *yes*, and N_k^{total} is the total number of requirements evaluated in category k. For example, if a graph state contains four action requirements and three are judged as satisfied, then $C_{\text{action}} = 3/4 = 0.75$.

The overall ISCS is the average compliance across the four categories:

$$\text{ISCS} = \frac{C_{\text{role}} + C_{\text{action}} + C_{\text{gaze}} + C_{\text{relation}}}{4}. \tag{18}$$

If a graph state contains no requirement from a particular category, that category is excluded from the average. ISCS ranges from 0 to 1, where a higher score indicates better agreement between the generated interaction and the graph specification. We report ISCS as a percentage in the experimental tables.

ISCS Example. Consider a graph state from a campsite sequence in which P_1 is speaking while holding a tent pole and looking toward the tent frame, while P_2 is listening and attaching the fabric to the frame.

The graph state is converted into individual requirements, for example:

Category	Requirement	GPT-5.6
Role	P_1 is speaking	yes
Role	P_2 is listening	yes
Action	P_1 is holding the tent pole	yes
Action	P_2 is attaching the tent fabric	no
Gaze	P_1 looks toward the tent frame	yes
Relation	P_2 is coordinating with P_1	yes

The category scores are therefore

$$C_{\text{role}} = \frac{2}{2} = 1.0, \qquad C_{\text{action}} = \frac{1}{2} = 0.5,$$

$$C_{\text{gaze}} = \frac{1}{1} = 1.0, \qquad C_{\text{relation}} = \frac{1}{1} = 1.0.$$

The final Interaction State Compliance Score is

$$\text{ISCS} = \frac{1.0 + 0.5 + 1.0 + 1.0}{4} = 0.875 = 87.5\%. \tag{19}$$

In this example, most of the requested graph state is visible, but P_2 does not perform the specified fabric-attachment action. This lowers the action compliance while the role, gaze, and interaction relation remain correct.

GPT-5.6 prompt for graph-state compliance

```
You are an evaluator of generated multi-person interaction videos.

Your task is to determine whether each supplied interaction-graph
    claim is
visually satisfied in the corresponding video segment.

INPUT:
1. A reference image used only to identify the participants.
2. Participant descriptions mapping P1, P2, ... to the visible people.
3. Chronologically ordered frames sampled from one video segment.
4. A list of atomic graph claims to evaluate.

EVALUATION RULES:

- Use the reference image only to establish participant identity; do
    not use it
  as evidence that a behavior is satisfied in the generated video.
- Judge each claim only from the supplied video frames.
- Do not assume that a claim is correct because it is included in the
    graph.
- Evaluate every claim independently and evaluate only the participant
     named
  in that claim.
- Return "yes" only when the requested behavior is clearly supported
    by the
  sampled frames from the segment.
- For a behavior that is expected to persist during the segment, such
    as
  speaking/listening role, gaze target, an ongoing action, posture, or
     a
  physical relation, require the behavior to be consistently visible
    across
  the relevant sampled frames. A brief occurrence in one frame is not
```

```
  sufficient.
- For an action claim, require visible evidence of the specified
    action and,
  when the claim describes motion or continuation, use the
    chronological
  change across frames rather than a single frame.
- For a speaking claim, require visible temporal evidence consistent
    with
  speaking, such as repeated mouth or facial motion across the sampled
     frames.
- For a listening claim, require the participant not to exhibit
    competing
  speaking behavior and to remain visually engaged in the interaction.
- For a gaze or attention claim, use visible head and eye orientation
    relative
  to the specified participant or object.
- For a physical-relation claim such as supporting, holding, or
    touching,
  require visible body configuration or contact consistent with that
    relation.
- Do not treat scene similarity, participant appearance, or spatial
    proximity
  alone as evidence that an interaction claim is satisfied.
- Return "no" when the supplied frames provide clear evidence that the
     claim
  is not satisfied.
- Return "uncertain" when the evidence is insufficient, ambiguous,
    strongly
  occluded, or cannot be determined reliably.
- Do not guess.

For every supplied claim, return exactly one verdict:
"yes", "no", or "uncertain".

Do not add, remove, merge, or reinterpret claims.

Return JSON only in the following format:

{
  "claims": [
    {
      "id": "<claim_id>",
      "verdict": "yes|no|uncertain",
      "evidence": "<brief description of the visual evidence>"
    }
  ]
}
```

I.3 Interaction Transition Success

Interaction Transition Success (ITS) measures whether the attributes that are required to change between two consecutive graph states are correctly updated in the generated video. This differs from Interaction State Compliance Score (ISCS), which evaluates the complete interaction state within each video segment, including the specified roles, actions, attention targets, and relations. In contrast, ITS focuses only on the properties that change from one state to the next. For example, if P_1 changes from speaking to listening while the remaining interaction properties stay unchanged, ISCS evaluates the full state in both segments, whereas ITS evaluates specifically whether the speaking-to-listening change occurs.

For each transition from $G^{(j-1)}$ to $G^{(j)}$, we first reconstruct the complete state of both graphs by inheriting attributes that were not updated. We then compare the corresponding participant attributes and interaction relations. Each value that differs between the two states is treated as one requested

edit, while unchanged values are ignored. For example, P_1: `speaking` $\rightarrow$ `listening` is one edit, and P_2: `object` $\rightarrow$ `P1` as the attention target is another. For each requested edit, GPT-5.6 is given frames from the source segment $G^{(j-1)}$ and the target segment $G^{(j)}$. The edit is counted as successful only if the source segment shows the previous value and the target segment shows the requested new value for the same participant or relation. GPT-5.6 returns `yes`, `no`, or `uncertain`. Only `yes` is counted as successful. Let $N_{\text{total}}^{\text{edit}}$ denote the total number of requested graph edits and $N_{\text{success}}^{\text{edit}}$ the number of successful edits. We compute

$$\text{ITS} = \frac{N_{\text{success}}^{\text{edit}}}{N_{\text{total}}^{\text{edit}}}. \tag{20}$$

ITS lies in $[0, 1]$ and is reported as a percentage. It measures whether the requested change occurs, but not how close that change is to the scheduled turn boundary. Transition timing is evaluated separately in Sec. I.5

ITS Example. Consider the same campsite scenario used in the ISCS example. In the first interaction state, P_1 is speaking while holding a tent pole and looking toward the tent frame, while P_2 is listening, attaching the tent fabric, and looking toward the frame. In the next state, the conversational roles change: P_1 becomes the listener and turns toward P_2, while P_2 becomes the speaker, stops attaching the fabric, turns toward P_1, and points toward a ground stake.

The requested graph edits include:

- P_1: speaking $\rightarrow$ listening;
- P_2: listening $\rightarrow$ speaking;
- P_1 visual attention target: tent frame $\rightarrow$ P_2;
- P_2 attention target: tent frame $\rightarrow$ P_1;
- P_1 action: holding the tent pole upright $\rightarrow$ pausing the setup while listening;
- P_2 action: attaching the tent fabric $\rightarrow$ pointing toward the ground stake while speaking.

For example, consider the action edit

$$P_2: \quad \text{attaching tent fabric} \rightarrow \text{pointing toward the ground stake.}$$

This edit is successful if P_2 is seen attaching the fabric in the first interaction state and pointing toward the ground stake in the next state. It is unsuccessful if P_2 continues attaching the fabric, performs a different action, or if the pointing action is performed by P_1 instead.

Similarly, consider the role edit

$$P_1: \quad \text{speaking} \rightarrow \text{listening.}$$

This edit is successful if P_1 is the speaker in the first interaction state and becomes the listener in the next. If P_1 continues speaking after the state change, the edit is unsuccessful.

These examples illustrate that ITS evaluates whether the specific properties that are required to change between two consecutive graph states actually change in the generated video.

GPT-5.6 prompt for graph-edit evaluation

```
You are evaluating whether a requested interaction change is realized
    between
two consecutive segments of a generated multi-person video.

You are given:
1. A reference image used only to establish participant identity.
2. Chronologically ordered frames sampled from the SOURCE segment
   $G^{(j-1)}$.
```

```
3. Chronologically ordered frames sampled from the TARGET segment
   $G^{(j)}$.
4. A list of atomic graph edits describing the required change from
    the
   SOURCE segment to the TARGET segment.

For each edit:

- Use the reference image only to identify participants; do not use it
     as
  evidence that an edit is satisfied.
- Check whether the specified SOURCE condition is visibly present in
    the
  frames from $G^{(j-1)}$.
- Check whether the specified TARGET condition is visibly present in
    the
  frames from $G^{(j)}$.
- Return "yes" only when both conditions are satisfied for the same
    requested
  participant: the SOURCE condition in $G^{(j-1)}$ and the TARGET
    condition
  in $G^{(j)}$.
- The SOURCE and TARGET conditions must represent the requested change
    , rather
  than two visually equivalent states.
- A partial change, a different target behavior, or a change performed
     by the
  wrong participant does not satisfy the edit.
- Do not require the visible transition to occur exactly at the
    boundary
  between $G^{(j-1)}$ and $G^{(j)}$; transition timing is evaluated
  separately.
- Return "no" if the SOURCE condition is clearly present but the
    requested
  TARGET condition is not realized in $G^{(j)}$, or if a different
    target
  behavior is produced.
- Return "uncertain" if either condition cannot be determined reliably
     from
  its corresponding segment because of insufficient visual evidence,
  ambiguity, or occlusion.
- Do not infer success from the graph description itself.
- Do not add, remove, merge, or reinterpret the supplied edits.
- Do not guess.

For every supplied edit, return exactly one verdict:
"yes", "no", or "uncertain".

Return JSON only:

{
  "edits": [
    {
      "id": "<edit_id>",
      "verdict": "yes|no|uncertain",
      "evidence": "<brief description of the observed source and
    target states>"
    }
  ]
}
```

I.4 GAZE-TO-TARGET ACCURACY

For each participant and graph state, the interaction graph specifies a *visual attention target* through the `attend` attribute. The target may be another participant or a visible object in the scene. For example, the graph may specify that P_1 should attend to P_2, or to the cutting board. *Gaze-to-Target Accuracy* (GTA) measures how consistently the generated video follows these specified targets. At each evaluation point, GPT-5.6 receives a sequence of video frames and checks whether each participant is visibly oriented toward the target specified by the graph state scheduled at that time.

GPT-5.6 determines the attended target primarily from visible head orientation and, when sufficiently clear, eye direction. A judgment is counted as correct only when there is sufficient visual evidence that the participant is attending to the specified target. Spatial proximity alone is not sufficient. For example, standing beside a pot does not count as attending to the pot if the participant is visibly looking toward another person. For each participant with a specified attention target, GPT-5.6 returns *yes*, *no*, or *uncertain*. A *yes* is counted as correct, while *no* and *uncertain* are counted as incorrect. GTA is computed as

$$\mathrm{GTA} = \frac{N^{\mathrm{gaze}}_{\mathrm{correct}}}{N^{\mathrm{gaze}}_{\mathrm{total}}}, \tag{21}$$

where $N^{\mathrm{gaze}}_{\mathrm{correct}}$ is the number of evaluations in which the specified attention target is correctly observed, and $N^{\mathrm{gaze}}_{\mathrm{total}}$ is the total number of evaluated participant-time pairs. GTA ranges from 0 to 1 and is reported as a percentage.

GTA Example. Suppose the interaction graph specifies where each participant should be visually oriented:

$$\text{Segment 1:} \qquad P_1 \rightarrow \text{cutting board},$$

$$\text{Segment 2:} \qquad P_1 \xrightarrow{\text{look toward}} P_2.$$

During Segment 1, GPT-5.6 checks at each evaluation point whether P_1 is visibly attending to the cutting board. During Segment 2, it checks whether P_1 is visibly attending to P_2.

Thus, the expected attention target changes with the scheduled graph state, and GTA measures how often the generated video follows the target specified for each evaluation point.

Relation to ISCS. GTA differs from the gaze component of ISCS in how frequently the behavior is evaluated. ISCS checks whether the requested gaze behavior is represented within the graph segment as a whole. GTA instead checks the specified attention target repeatedly throughout the segment. It therefore penalizes cases where a participant attends to the correct target only briefly, changes attention too early, or reaches the target later than scheduled.

GPT-5.6 prompt for gaze-to-target evaluation

```
You are evaluating the visible attention direction of participants in
    a generated interaction video.

You are given:
1. A reference image defining P1, P2, and any additional participants.
2. A short sequence of chronological video frames.
3. One participant.
4. One requested attention target.

Determine whether the participant is visibly attending to the
    requested target.

Use the following rules:

- Use both head orientation and visible eye direction.
```

```
- When the target is another participant, verify that the evaluated
  participant is oriented toward that person's visible location.
- When the target is an object, verify orientation toward the visible
    object.
- Proximity to a target is not sufficient.
- Looking elsewhere, looking at the camera, or looking at a different
    object
  is incorrect.
- If attention direction cannot be determined reliably, return "
    uncertain".
- Do not infer the answer from the graph text.

Return JSON only:

{
  "participant": "<participant_id>",
  "target": "<target>",
  "verdict": "yes|no|uncertain",
  "evidence": "<brief visual evidence>"
}
```

I.5 Temporal Interaction Evaluation

ISCS and ITS determine whether the requested interaction states are present and whether the requested changes are carried out, respectively, but they do not measure whether these changes occur at the scheduled time. We therefore evaluate temporal interaction control using *Temporal State Alignment (TSA)*, *Transition Timing Error (TTE)*, and *Final-State Completion (FSC)*.

I.5.1 Temporal State Alignment

Temporal State Alignment (TSA) measures the fraction of evaluation windows in which the generated video shows the graph state assigned to that window. It therefore evaluates state correctness throughout the video, rather than only around a transition boundary. For each $0.2\,\mathrm{s}$ evaluation window, GPT-5.6 receives the corresponding video frames together with short descriptions of the graph states defined for that video. GPT-5.6 is not told which graph state should be active at that time or where the audio-derived transition boundaries occur. Instead, it independently selects the graph state that best matches the visible interaction.

Let $\hat{g}_t$ denote the graph state selected by GPT-5.6 for evaluation window t, and let g_t^* denote the graph state scheduled for that window. TSA is the fraction of evaluation windows in which the selected state matches the scheduled state:

$$\mathrm{TSA} = \frac{1}{N} \sum_{t=1}^{N} \mathbf{1}\left[\hat{g}_t = g_t^*\right], \tag{22}$$

where N is the total number of evaluation windows and $\mathbf{1}[\cdot]$ is an indicator function that equals 1 when the selected and scheduled states match, and 0 otherwise. GPT-5.6 may also return *neither* when the visible interaction does not match any candidate graph state, or *uncertain* when there is insufficient visual evidence to select a state. Both outcomes are counted as mismatches.

TSA penalizes interaction changes that occur either too early or too late. For example, if $G^{(2)}$ becomes visible while $G^{(1)}$ is still scheduled, that window is counted as a mismatch. Once $G^{(2)}$ is scheduled, the same behavior is counted as correctly aligned.

TSA Example. Suppose the interaction schedule contains two graph states:

$G^{(1)}$: P1 speaks while holding the tent pole and P2 attaches the fabric,

$G^{(2)}$: P2 speaks while pointing toward a ground stake and P1 listens.

Assume that $G^{(1)}$ is scheduled for the first three evaluation windows and $G^{(2)}$ for the next three. At each window, GPT-5.6 independently selects the graph state that best matches the visible interaction:

Window	Time	Scheduled	GPT-5.6	Match
1	0.0 s	$G^{(1)}$	$G^{(1)}$	✓
2	0.2 s	$G^{(1)}$	$G^{(1)}$	✓
3	0.4 s	$G^{(1)}$	$G^{(2)}$	×
4	0.6 s	$G^{(2)}$	$G^{(2)}$	✓
5	0.8 s	$G^{(2)}$	$G^{(2)}$	✓
6	1.0 s	$G^{(2)}$	$G^{(2)}$	✓

In Window 3, GPT-5.6 already identifies $G^{(2)}$, even though $G^{(1)}$ is still scheduled. The generated interaction therefore changes state earlier than required, and this window is counted as incorrect. The other five windows match the scheduled graph state, giving

$$\mathrm{TSA} = \frac{5}{6} = 0.833 = 83.3\%. \tag{23}$$

GPT-5.6 prompt for temporal-state identification

```
You are evaluating the temporal interaction state visible in a
    generated video.

You are given:
1. A reference image defining participant identity.
2. A short chronological sequence of video frames centered at one
    evaluation
   time.
3. Descriptions of the possible interaction states.

IMPORTANT:
You are NOT given the expected state for this time.
Determine the state only from the visible video evidence.

Compare the candidates using:
- participant actions,
- body posture,
- interaction relations,
- support/contact configuration,
- gaze or attention when visible,
- speaking/listening behavior when visually resolvable,
- progression of the described interaction.

Choose a graph state only when its defining interaction is visibly
    supported.

If none of the candidate states describes the visible interaction,
    return
"neither".

If two states cannot be distinguished reliably from the supplied
    frames,
return "uncertain".

Do not use the order of the candidate descriptions as evidence.
Do not assume that the video follows the intended graph.
```

```
Return JSON only:

{
  "state": "<graph_state_id|neither|uncertain>",
  "evidence": "<brief description of the visible interaction>"
}
```

I.5.2 Transition Timing Error

Transition Timing Error (TTE) evaluates only the timing of a change between two consecutive graph states. It compares the scheduled transition time with the time at which the generated video first shows the new state consistently. TTE is the absolute difference between these two times. Let τ_j denote the scheduled transition time from $G^{(j)}$ to $G^{(j+1)}$, determined by the audio-derived turn boundary. We identify when the corresponding interaction change becomes visible in the generated video using the same temporal-state predictions employed for TSA. TTE therefore measures adherence to the prescribed interaction schedule rather than agreement with ground-truth action timing.

To avoid treating a state that appears in only one evaluation window as the actual transition, we define the observed transition time $\hat{\tau}_j$ as the first evaluation time t at which GPT-5.6 identifies the new state $G^{(j)}$ at both t and the following evaluation window. The transition timing error is then computed as

$$\mathrm{TTE}_j = |\hat{\tau}_j - \tau_j| \,. \tag{24}$$

A lower TTE indicates more accurate transition timing. If the requested next state is never identified by GPT-5.6 in two consecutive evaluation windows, we report the transition as *Not Realized* (N/R) rather than assigning a transition time.

TTE Example. Consider a two-person sequence with the following audio-derived turn schedule:

$$\text{P1 speaks: } 0-4.0\,\text{s}, \qquad \text{P2 speaks: } 4.0-8.0\,\text{s}.$$

The corresponding graph states are

$$G^{(1)}: \text{ P1 speaks while holding the tent pole and P2 attaches the fabric,}$$
$$G^{(2)}: \text{ P2 speaks while pointing toward a ground stake and P1 listens.}$$

The scheduled transition from $G^{(1)}$ to $G^{(2)}$ therefore occurs at the audio-derived turn boundary

$$\tau_1 = 4.0\,\text{s}.$$

TTE is computed from the state predictions obtained during the TSA evaluation. Suppose GPT-5.6 assigns the following graph states to the 0.2 s evaluation windows around the boundary:

Evaluation Time	Selected State
3.8 s	$G^{(1)}$
4.0 s	$G^{(1)}$
4.2 s	$G^{(2)}$
4.4 s	$G^{(2)}$

The new state $G^{(2)}$ is first selected at 4.2 s and is also selected in the following window. According to our two-consecutive-window rule, the observed transition time is therefore

$$\hat{\tau}_1 = 4.2\,\text{s}.$$

The Transition Timing Error is

$$\text{TTE}_1 = |\hat{\tau}_1 - \tau_1| = |4.2 - 4.0| = 0.2\,\text{s}. \tag{25}$$

Thus, the observed interaction-state change occurs $0.2\,\text{s}$ after the scheduled turn boundary. The target time τ_1 is obtained from the audio turn schedule, while $\hat{\tau}_1$ is derived from the graph-state predictions produced during TSA evaluation.

I.5.3 Final-State Completion

Final-State Completion (FSC) measures whether the generated video reaches the interaction state specified at the end of the graph sequence. From the final graph state, we extract the requested final actions, postures, and interaction relations. Static properties, such as appearance and background, are not evaluated.

We evaluate the last $0.6\,\text{s}$ of the video using three consecutive $0.2\,\text{s}$ windows. GPT-5.6 checks each final-state requirement in each window. A requirement is counted as satisfied if it is observed in at least two of the three windows. Let $\mathcal{F}$ denote the set of final-state requirements. We compute

$$\text{FSC} = \frac{1}{|\mathcal{F}|} \sum_{f \in \mathcal{F}} s_{\text{final}}(f), \tag{26}$$

where $s_{\text{final}}(f) = 1$ if requirement f is observed in at least two of the three final evaluation windows, and 0 otherwise. For example, in the partner-dance sequence in Fig. 6, simply beginning to lower the lifted avatar is not sufficient. The final state is counted as completed only when the requested landing configuration is visibly reached.

FSC Example. Consider the final graph state in the campsite example, where P2 is speaking and pointing toward a ground stake while P1 listens. We extract the following final-state requirements:

F_1 P1 is listening.
F_2 P2 is speaking.
F_3 P2 is pointing toward the ground stake.
F_4 P1 has paused the tent setup while listening.

GPT-5.6 checks each requirement in the three $0.2\,\text{s}$ evaluation windows covering the final $0.6\,\text{s}$ of the video:

Requirement	W_1	W_2	W_3	Satisfied
F_1: P1 listening	✓	✓	✓	✓
F_2: P2 speaking	✓	×	✓	✓
F_3: P2 pointing	×	✓	✓	✓
F_4: P1 pauses setup	×	×	✓	×

A requirement is counted as satisfied when it is observed in at least two of the three windows. In this example, F_1, F_2, and F_3 are satisfied, while F_4 is not. Therefore,

$$\text{FSC} = \frac{3}{4} = 0.75 = 75\%. \tag{27}$$

GPT-5.6 prompt for final-state completion

```
You are evaluating whether a generated interaction has reached its
    requested
```

```
final state.

You are given:
1. A reference image defining participant identity.
2. Chronological frames sampled from the end of the generated video.
3. Atomic claims describing the requested final interaction state.

For each final-state claim:

- Judge whether the requested final condition is visibly reached.
- Do not give credit merely because the video begins moving toward the
    target.
- A partial transition is not equivalent to completing the final state
    .
- The requested participant, posture, action, or relation must be
    visibly
  correct.
- For physical interaction claims, require visible geometric/body
    evidence.
- Ignore general image quality and scene similarity.
- If the final condition cannot be determined reliably, return "
    uncertain".

Return JSON only:

{
  "claims": [
    {
      "id": "<claim_id>",
      "verdict": "yes|no|uncertain",
      "evidence": "<brief visual evidence>"
    }
  ]
}
```

I.5.4 Temporal Interaction Fidelity

TSA, TTE, and FSC measure different aspects of temporal interaction control. TSA measures whether the interaction state shown at each evaluation time matches the scheduled graph state, TTE measures how far the observed state transition occurs from the scheduled turn boundary, and FSC measures whether the requested final interaction state is reached.

Because TSA and FSC are higher-is-better scores, whereas TTE is an error measured in seconds, we convert TTE into a higher-is-better score called Transition Timing Accuracy (TTA). For transition j, let τ_j denote the scheduled turn boundary. We first compute the maximum possible timing error from this boundary within the evaluated video:

$$\Delta_j^{\max} = \max\left(\tau_j - t_{\text{start}}, t_{\text{end}} - \tau_j\right). \tag{28}$$

We then compute TTA as

$$\text{TTA}_j = 1 - \min\left(\frac{\text{TTE}_j}{\Delta_j^{\max}}, 1\right). \tag{29}$$

A transition that occurs close to the scheduled turn boundary therefore receives a high TTA score, while a larger timing error results in a lower score. If the requested transition is not observed according to the two-consecutive-window rule, we set $\text{TTA}_j = 0$. For videos with more than one transition, TTA is averaged across all transitions.

Temporal Interaction Fidelity (TIF) combines TSA, TTA, and FSC:

Table 10: **Cross-evaluator robustness of interaction-control metrics.** The same generated videos, sampled frames, interaction requirements, and evaluation prompts are evaluated using GPT-5.6 and InternVL3.5. Although absolute scores vary across evaluators, the ranking of the compared methods is preserved for all metrics. Higher is better.

Evaluator	Method	ISCS (%) ↑	ITS (%) ↑	GTA (%) ↑	TIF (%) ↑
GPT-5.6	AnyTalker	56.3	28.0	80.0	22.7
	Playmate2	60.4	32.0	82.7	38.1
	MultiTalk	90.6	62.0	93.2	87.9
	ConfabMuse	**93.8**	**81.0**	**94.6**	**92.8**
InternVL3.5	AnyTalker	48.1	25.0	75.3	24.6
	Playmate2	57.4	30.0	78.9	26.1
	MultiTalk	89.7	73.0	89.1	81.5
	ConfabMuse	**91.5**	**78.0**	**92.6**	**88.3**

$$\boxed{\mathrm{TIF} = \frac{\mathrm{TSA} + \mathrm{TTA} + \mathrm{FSC}}{3}} \tag{30}$$

TSA, TTA, and FSC are all defined on the $[0, 1]$ range and are averaged with equal weights. TIF is then reported as a percentage. A high TIF therefore requires the generated video to show the correct interaction states over time, change state close to the scheduled turn boundary, and reach the requested final state. Showing the requested interaction only briefly or at the wrong time is therefore not sufficient to obtain a high TIF.

I.6 Cross-Evaluator Robustness

To examine the sensitivity of the proposed interaction-control metrics to the choice of MLLM evaluator, we repeat the evaluation using the open-source InternVL3.5 8B model. We use the same generated videos, sampled frames, interaction requirements, and evaluation prompts as in the GPT-5.6 Sol evaluation.

Although the two evaluators differ in their absolute score calibration, they produce the same ordering of the four compared methods across ISCS, ITS, GTA, and TIF. ConfabMuse obtains the highest score under both evaluators for all four metrics. The method-level Spearman rank correlation between GPT-5.6 Sol and InternVL3.5 8B is $\rho = 1.0$ for each metric. While this analysis is limited to four compared methods, the consistent ordering suggests that the comparative trends are not specific to a single MLLM evaluator.