STREAMAV-BENCH: A COMPREHENSIVE BENCHMARK FOR STREAMING AUDIO-VIDEO GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Recent advancements in generative models are pushing video generation toward unbounded streaming audio-video generation for real-time interactive worlds. However, existing benchmarks primarily evaluate completed sequences and struggle to capture streaming properties. To bridge this gap, we introduce StreamAV-Bench, the first comprehensive benchmark tailored for streaming audio-video generation. StreamAV-Bench establishes a unified evaluation framework, including the progressive track for instruction adherence and long-horizon stability, and the interactive track for interactive response and state retention and reuse. With expert-verified evaluation cases across 32 fine-grained dimensions, we conduct an extensive evaluation of 13 representative systems. Our analysis reveals that current models suffer from temporal drift in progressive generation and responsiveness bottlenecks during interactive control. Based on a comprehensive failure analysis, we share insights to advance the development of native joint audio-video streaming models.

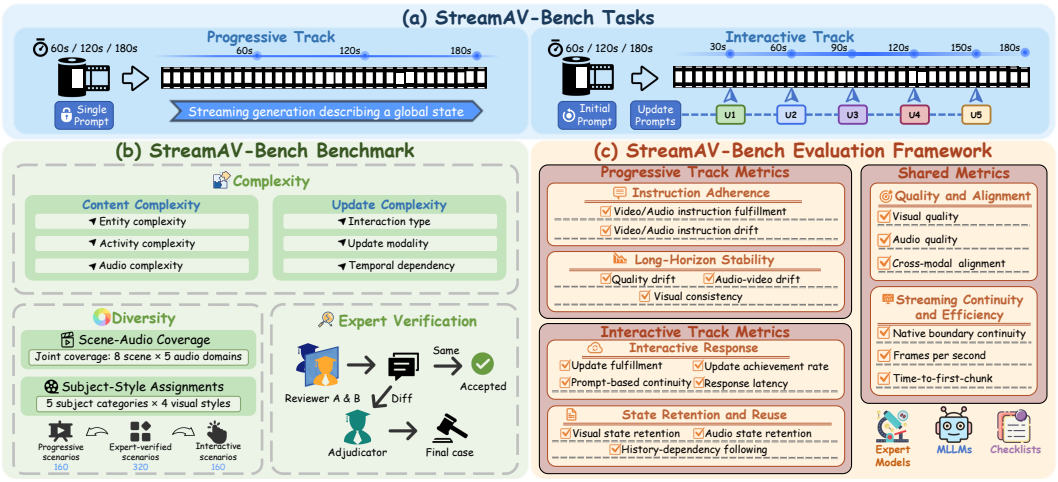

Figure 1: We present StreamAV-Bench, a comprehensive benchmark for streaming audio-video generation. (a) The benchmark includes a progressive track to assess instruction adherence and long-horizon stability, and an interactive track to measure interactive response alongside state retention and reuse. (b) The benchmark covers content complexity in both tracks and update complexity in the interactive track, with 320 expert-verified scenarios that span 8 scene domains, 5 audio domains, 5 subject categories, and 4 visual styles. (c) The evaluation is driven by a unified framework of 32 fine-grained dimensions, computed with expert models, MLLMs, and corresponding checklists.

1 INTRODUCTION

Recent advances have pushed video generation beyond short clips toward long-horizon and interactive audio-video generation, paving the way for the real-time interactive world. Built upon the

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.

streaming paradigm, recent systems (Su et al., 2026; Bai et al., 2026; Huang et al., 2026) generate these immersive worlds over unbounded durations. In this setting, audio-video sequences are generated progressively over an expanding time horizon. Users can interactively provide free-form prompts, while previously generated clips are fixed as committed history. Consequently, the model must maintain robust audio-video quality and cross-modal alignment, avoid temporal drift, execute dynamic switches in world states with rapid interactive response, and preserve boundary continuity as the generation progresses. These requirements introduce novel failure modes that cannot be adequately captured by standard evaluation protocols.

Current benchmarks struggle to evaluate the properties of streaming audio-video generation. While largely exploring long-horizon generation, previous video-centric benchmarks (Huang et al., 2025b; Zheng et al., 2025; Yuan et al., 2026; Liu et al., 2026a; Zhang et al., 2026) lack audio-video quality assessment. Meanwhile, existing audio-video generation benchmarks (Hua et al., 2025; Cao et al., 2025; Zhou et al., 2026b) focus on fundamental metrics (*e.g.*, audio-video quality, cross-modal alignment, and synchronization), with sparse attempts to explore long-horizon generation (Liu et al., 2026c) and narratives with scripted prompts (Wei et al., 2026). Collectively, these benchmarks focus on completed audio-video outputs, neglecting the aforementioned streaming properties and leaving streaming audio-video generation insufficiently evaluated. We summarize existing benchmarks in Table 1 to illustrate their specific focuses and their neglect of streaming properties.

In this paper, we introduce **StreamAV-Bench**, the first comprehensive benchmark for streaming audio-video generation. As shown in Fig. 1 (a), StreamAV-Bench includes a **progressive track** that uses a global prompt to evaluate progressively generated audio-video sequences for instruction adherence and long-horizon stability, and an **interactive track** that adopts an ordered prompt sequence for dynamic interaction to evaluate interactive response alongside state retention and reuse. As shown in Fig. 1 (b), the benchmark jointly considers the content complexity (*e.g.*, entity, activity, and audio complexity) in both tracks, and the update complexity (*e.g.*, interaction type, update modality, and temporal dependency) in the interactive track. Collectively, the benchmark comprises 320 scenarios evenly divided between the two tracks and spanning 8 scene domains, 5 audio domains, 5 subject categories, and 4 visual styles. To evaluate temporal scalability, we continuously generate each scenario for 180 seconds. All cases undergo expert verification to confirm prompt feasibility and eliminate semantic ambiguity.

Building on these two tracks and the resulting evaluation cases, StreamAV-Bench provides a unified framework with 32 fine-grained dimensions, as shown in Fig. 1 (c). Progressive metrics evaluate instruction adherence and long-horizon stability, interactive metrics evaluate interactive response alongside state retention and reuse, and shared metrics cover streaming continuity and efficiency as well as audio-video quality and alignment.

In summary, our main contributions are threefold:

- We introduce StreamAV-Bench, the first comprehensive benchmark tailored for streaming audio-video generation, featuring distinct progressive and interactive tracks.
- We curate high-quality evaluation cases through strict independent expert verification and design a unified framework with 32 dimensions for systematic assessment.
- We conduct an extensive evaluation of 13 representative generation systems, spanning native streaming audio-video models and cascaded T2V and V2A pipelines.

Our evaluation further reveals paradigm-specific strengths, dimension-dependent long-horizon degradation, and bottlenecks in update achievement, response latency, and state reuse, offering insights for developing native joint audio-video streaming models with long-horizon stability and real-time interactive capabilities.

2 Related Work

2.1 Streaming Audio-Video Generation

Recent audio-video generation employs either cascaded pipelines that add audio to generated video (Cheng et al., 2024; Shan et al., 2025) or joint models (Ruan et al., 2022; Liu et al., 2025a; Low et al., 2025; HaCohen et al., 2026). Beyond clip generation, video methods support multi-prompt

Table 1: Comparison with existing generative benchmarks. Our proposed StreamAV-Bench comprehensively evaluates the essential properties of streaming audio-video generation.

Benchmark	Generation Setting			Streaming Evaluation					
	AV Quality	Long Horizon	Scripted Prompt	Temporal Drift	Dynamic Switch	Interactive Response	State Retention & Reuse	Boundary Continuity	Streaming Efficiency
Long-horizon video benchmarks									
VBench-Long (Huang et al., 2025b)	✗	✓	✗	✗	✗	✗	✗	✗	✗
HeliosBench (Yuan et al., 2026)	✗	✓	✗	✓	✗	✗	✗	✗	✓
NarrLV (Feng et al., 2025b)	✗	✓	✗	✗	✗	✗	✗	✗	✗
MPVBench (Cai et al., 2025)	✗	✓	✓	✗	✗	✗	✗	✓	✗
NarraStream-Bench (Liu et al., 2026a)	✗	✓	✓	✗	✓	✗	✓	✓	✓
MBench (Zhang et al., 2026)	✗	✓	✓	✗	✓	✓	✗	✗	✗
Audio-video generation benchmarks									
VABench (Hua et al., 2025)	✓	✗	✗	✗	✗	✗	✗	✗	✗
T2AV-Compass (Cao et al., 2025)	✓	✗	✗	✗	✗	✗	✗	✗	✗
AVGen-Bench (Zhou et al., 2026b)	✓	✗	✗	✗	✗	✗	✗	✗	✗
LongAV-Compass (Liu et al., 2026c)	✓	✓	✓	✗	✗	✗	✗	✓	✗
MSAVBench (Wei et al., 2026)	✓	✗	✓	✗	✗	✗	✗	✗	✗
StreamAV-Bench (Ours)	✓	✓	✓	✓	✓	✓	✓	✓	✓

extension (Cai et al., 2025) and spatiotemporal completion (Cai et al., 2026), while streaming approaches reduce accumulated errors through causal (Yin et al., 2024) or self-rollout training (Huang et al., 2025a), block-wise processing (Kodaira et al., 2025), hierarchical contexts (Yuan et al., 2026), rolling contexts (Liu et al., 2025b), and bounded caches (Yang et al., 2025). Recent audio-video systems further explore synchronized streaming (Su et al., 2026; Huang et al., 2026) and controlled multi-shot or long-horizon generation (Liu et al., 2026b; Li et al., 2026), motivating systematic evaluation of progressive and interactive streaming capabilities.

2.2 Video Generation Benchmarks

Video benchmarks evaluate visual quality, temporal consistency, and prompt alignment (Huang et al., 2023), with human-aligned or broad-setting evaluations provided by FETV (Liu et al., 2023b), EvalCrafter (Liu et al., 2023a), and VBench++ (Huang et al., 2025b). Specialized benchmarks assess compositional instruction following, dynamics, and temporal state transitions (Sun et al., 2024; Liao et al., 2024; Feng et al., 2025a). For long-form or multi-prompt generation, existing benchmarks measure chunk-level degradation, event and narrative realization, cross-segment instruction following, and long-range memory (Zhang et al., 2025; Zheng et al., 2025; Liu et al., 2026a; Zhang et al., 2026). However, these works remain video-centric and omit the accompanying audio modality.

2.3 Audio-Video Generation Benchmarks

Audio-video benchmarks evaluate video and audio quality, cross-modal semantics, synchronization, instruction following, and perceptual realism (Mao et al., 2024; Hua et al., 2025; Cao et al., 2025; Zhou et al., 2026b). Other benchmarks target visually grounded sound and multi-talker settings (Zhou et al., 2026a; Dixit et al., 2025), multi-shot generation (Wei et al., 2026), or minute-long generation across diverse conditioning modalities (Liu et al., 2026c). Consequently, they evaluate completed outputs rather than the long-horizon and interactive properties of streaming systems.

3 StreamAV-Bench

3.1 Task Definition

Streaming audio-video generation models incrementally generate synchronized sequences over an expanding time horizon, with previously generated content becoming committed history that conditions subsequent generation. The prompt may remain fixed or be updated interactively, corresponding to progressive and interactive streaming, respectively. For the created audio-video world, we define its state as the audio-video content established by a prompt, which is expected to remain valid until modified by a subsequent update and may be reused by future instructions.

To provide a comprehensive evaluation, StreamAV-Bench evaluates streaming models across two tracks. The progressive track measures instruction adherence and long-horizon stability from a single global prompt over 180 seconds. The interactive track measures interactive response and state retention and reuse using an initial prompt followed by five updates at 30-second intervals,

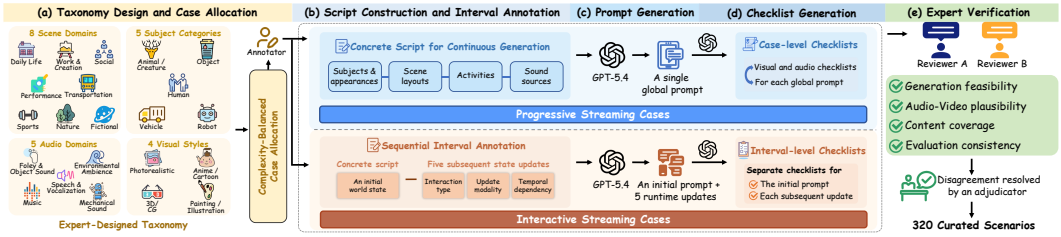

Figure 2: Overview of StreamAV-Bench construction. (a) Domain experts define the taxonomy and allocate complexity-balanced cases across both tracks. (b) Annotators script and annotate each case: the progressive track defines the entire sequence, while the interactive track specifies an interval sequence for dynamic state updates. (c) GPT-5.4 converts the annotations into one global prompt per progressive case and an initial prompt plus five runtime updates per interactive case. (d) Prompt-derived verifiable checklists evaluate case-level instruction adherence and stability, and interval-level interactive response, state retention, and reuse. (e) Two experts independently review the prompts and checklists, with disagreements adjudicated by a third expert, producing 320 curated scenarios.

each modifying visual content, audio content, or both. For both tracks, we additionally evaluate the same rollout at 60, 120, and 180 seconds to analyze performance across rollout horizons.

3.2 Benchmark Construction

To ensure a diverse and reliable benchmark for streaming audio-video generation, we employ a five-stage construction pipeline, as illustrated in Fig. 2.

Taxonomy design and case allocation. Domain experts define a taxonomy of 8 scene domains, 5 audio domains, 5 subject categories, and 4 visual styles, and construct 160 scenario themes, each instantiated in both tracks to yield 320 scenarios with balanced domain coverage. For both tracks, content complexity is defined along entity, activity, and audio dimensions; for each interactive scenario, update complexity further specifies how its five runtime updates are distributed across interaction type, update modality, and temporal dependency.

Script construction and interval annotation. For the allocated scenarios, annotators develop concrete scripts describing subjects and appearances, scene layouts, activities, and sound sources. Using this context, they produce structured annotations to organize the script over time. For the progressive track, these annotations bind the expected content across the entire sequence. For the interactive track, they define an initial world state and state updates for each subsequent 30-second interval.

Prompt generation. Based on annotated scripts, we employ an LLM (OpenAI, 2026) to generate a single global prompt per progressive case, describing the scene, visual style, subjects, ongoing activities, and audio. For interactive cases, it generates an initial prompt establishing the world and five textual updates for subsequent intervals, each specifying expected targets and state changes.

Checklist generation. We then utilize an LLM (OpenAI, 2026) to convert explicit audio-video requirements into measurable, verifiable checklists. Progressive cases associate visual and audio checklists with each prompt for case-level instruction adherence and stability, while interactive cases use separate criteria for the initial prompt and each subsequent update for interval-level evaluation.

Expert verification. Two domain experts independently review prompts, checklists, and update targets against scripts and interval annotations for generation feasibility, audio-video plausibility, content coverage, and evaluation consistency. Cases are revised and re-reviewed for omitted script content, unsupported checklists or update targets, or updates conflicting with established states. A third domain expert, the adjudicator, resolves disagreements and produces 320 curated scenarios.

3.3 Benchmark Statistics

Content distribution. As shown in Fig. 3 (a)–(d), StreamAV-Bench contains 320 scenarios across 8 scene domains (32–48 each), 5 audio domains (48–80 each), 5 subject categories, and 4 visual styles,

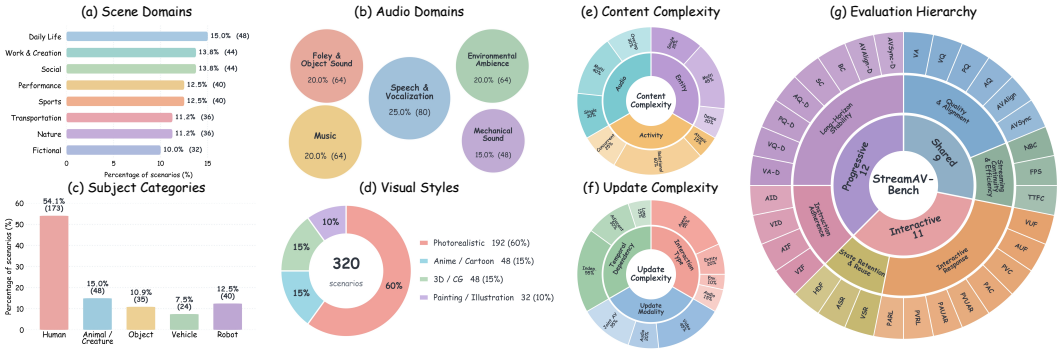

Figure 3: Detailed data statistics and the evaluation framework of StreamAV-Bench. (a)–(d) Distributions across 8 scene domains, 5 audio domains, 5 subject categories, and 4 visual styles. (e)–(f) Content and update complexity factors, respectively. (g) Overview of the evaluation framework, categorizing 32 fine-grained evaluation dimensions.

including 192 photorealistic and 128 non-photorealistic scenarios. These orthogonal combinations support robust evaluation across diverse audio-video settings.

Content and update complexity. As shown in Fig. 3 (e)–(f), both tracks characterize content complexity through *entity complexity*, *activity complexity*, and *audio complexity*, reflecting visual states, activity structures, and sound sources, respectively. The interactive track further characterizes update complexity through *interaction type*, *update modality*, and *temporal dependency*, capturing the requested operation, affected modalities, and reliance on prior states. These factors support systematic evaluation of long-horizon generation and runtime state manipulation.

3.4 Evaluation Framework

Our evaluation framework combines tailored expert models with checklist-guided Multimodal Large Language Model (MLLM) assessments across 32 fine-grained dimensions in six categories (Fig. 3 (g)). Progressive metrics evaluate instruction adherence and long-horizon stability, interactive metrics evaluate interactive response and state retention and reuse, and shared metrics cover streaming continuity and efficiency as well as quality and alignment.

Quality and alignment. For visual quality, visual aesthetics (*VA*) uses the LAION Aesthetic Predictor (LAION-AI, 2022) to measure appeal, while visual quality (*VQ*) employs Gemini 3.1 Pro (Google DeepMind, 2026) to assess visual fidelity, subject integrity, motion naturalness, and artifacts. For audio quality, production quality (*PQ*) uses Audiobox Aesthetics (Tjandra et al., 2025) to measure perceptual and production fidelity, while audio quality (*AQ*) employs Gemini 3.1 Pro to assess naturalness and artifacts. For cross-modal alignment, audio-video alignment (*AVAlign*) uses ImageBind similarity (Girdhar et al., 2023) to measure semantic correspondence, whereas audio-video synchronization (*AVSync*) uses Synchformer (Iashin et al., 2024) to estimate synchronization.

Instruction adherence. For each progressive case, we use Gemini 3.1 Pro with fixed visual and audio checklists derived from the global prompt. Video instruction fulfillment (*VIF*) and audio instruction fulfillment (*AIF*) assess how completely the corresponding requirements are realized on a 1–5 scale. Extending the endpoint comparison used by HeliosBench (Yuan et al., 2026), video instruction drift (*VID*) and audio instruction drift (*AID*) reuse the same checklists to measure the change in adherence between the first and last 30-second intervals.

Long-horizon stability. To evaluate long-horizon stability, the progressive track examines quality drift, visual consistency, and audio-video drift. Adapting the endpoint comparison used by HeliosBench (Yuan et al., 2026), we compute the absolute changes in visual aesthetics (*VA-D*), visual quality (*VQ-D*), production quality (*PQ-D*), and audio quality (*AQ-D*) between the first and last 30-second intervals. Subject consistency (*SC*) and background consistency (*BC*) follow VBench-Long (Huang et al., 2025b) to assess the temporal consistency of subject appearance and background scenes, respectively. Finally, *AVAlign-D* and *AVSync-D* measure endpoint changes in audio-video semantic correspondence and temporal synchronization.

Interactive response. The interactive track evaluates the model's capacity for dynamic state updates. Video update fulfillment (*VUF*) and audio update fulfillment (*AUF*) use Gemini 3.1 Pro to assess how completely each update's visual and audio targets are realized. During prompt updates, we calculate prompt-based visual and audio continuity (*PVC*, *PAC*) to further assess temporal transition smoothness. Finally, prompt-based visual and audio update achievement rates (*PVUAR* and *PAUAR*) measure the proportion of updates whose core targets are realized, while the corresponding response latencies (*PVRL* and *PARL*) measure how quickly successful realization occurs.

State retention and reuse. To evaluate whether required elements (*e.g.*, subject appearance and ongoing audio) are consistently preserved, visual state retention (*VSR*) and audio state retention (*ASR*) are introduced. *VSR* and *ASR* compare the intervals before and after each update while accounting for the changes requested by that update. To further evaluate state reuse, history-dependency following (*HDF*) measures whether the current instruction is correctly executed using relevant content from earlier interactions. All three metrics use Gemini 3.1 Pro on 1–5 scales.

Streaming continuity and efficiency. At the boundaries of consecutive model-native generation units, native boundary continuity (*NBC*) assesses technical continuity by detecting boundary artifacts such as black frames, repeated frames, freezing, and flashes. We report generation speed in frames per second (*FPS*) under each system's default inference configuration, and quantify startup latency as time-to-first-chunk (*TTFC*).

4 Experiments

4.1 Experimental Setup

Evaluated models. We comprehensively evaluate 13 systems across 2 generation paradigms. Native joint streaming audio-video models include PixVerse R1 (PixVerse, 2026), HappyOyster (Alibaba, 2026), and OmniForcing (Su et al., 2026). For cascaded pipelines, we pair HunyuanVideo-Foley (Shan et al., 2025) with 10 native streaming video models: Odyssey-2 (Odyssey, 2025), Self-Forcing (Huang et al., 2025a), LongLive (Yang et al., 2025), Rolling-Forcing (Liu et al., 2025b), Deep Forcing (Yi et al., 2025), MemFlow (Ji et al., 2025), Causal-Forcing (Zhu et al., 2026), Helios (Yuan et al., 2026), SWIFT (Tan et al., 2026), and IAMFlow (Liu et al., 2026a).

Implementation details. We evaluate open-source models using official checkpoints and inference pipelines on NVIDIA H800 GPUs, and commercial systems via official web interfaces. All models use default recommended configurations and receive identical inputs: a single global prompt for the progressive track and an ordered prompt sequence with updates every 30 seconds of generated content for the interactive track. This ensures that each update is introduced at the same content horizon across systems, while FPS and TTFC separately measure generation efficiency.

Evaluation protocol. Following Section 3.4, shared metrics are computed on both tracks, instruction adherence and long-horizon stability on the progressive track, and interactive response and state retention and reuse on the interactive track, using case-specific checklists for instruction-conditioned metrics. Observations are aggregated within each case before uniform averaging across cases.

4.2 Main Results

Table 2 summarizes the overall performance and highlights three findings about current streaming generation systems.

Performance is distributed across paradigms. Among the evaluated systems, native joint streaming audio-video models achieve the strongest audio fidelity and synchronization (*PQ*, *AQ*, and *AVSync*) and lead nine of the eleven interactive-track metrics. Cascaded pipelines based on native streaming T2V instead lead in visual quality and semantic alignment (*VA*, *VQ*, and *AVAlign*), seven of the eight long-horizon stability metrics, and streaming continuity (*NBC*). Instruction adherence remains mixed: native joint systems lead in video instruction fulfillment (*VIF*) and audio instruction drift (*AID*), whereas cascaded pipelines lead in audio instruction fulfillment (*AIF*) and video instruction drift (*VID*). Overall, no single system dominates the full evaluation suite.

High absolute performance does not ensure temporal stability. The results show that high generation quality or instruction fulfillment does not necessarily imply low temporal drift. IAMFlow

Table 2: Evaluation results on StreamAV-Bench. Metrics for quality, alignment, streaming continuity, and efficiency are evaluated on both tracks. Instruction adherence and long-horizon stability are evaluated on the progressive track, while interactive response and state retention and reuse are evaluated on the interactive track. The best and second-best results are highlighted in **bold** and underlined. Dashes (–) denote unavailable results.

Method	Quality and Alignment						Streaming Continuity and Efficiency		
	VA↑	VQ↑	PQ↑	AQ↑	AVAlign↑	AVSync↓	NBC↑	FPS↑	TTFC↓
Native joint streaming audio-video models									
PixVerse R1	0.529	2.428	6.348	2.810	0.234	**0.855**	–	–	–
HappyOyster	0.529	2.785	**6.817**	**3.157**	0.206	1.044	–	–	–
OmniForcing	0.554	2.374	6.351	2.528	0.119	1.423	98.808	12.799	3.998
Cascaded streaming audio-video pipelines									
Odyssey-2	0.531	2.820	6.426	2.659	0.260	0.946	–	–	–
Self-Forcing	0.585	2.753	6.453	2.623	0.256	0.919	94.359	15.628	1.278
LongLive	0.605	2.768	6.459	2.696	0.272	0.969	85.418	10.125	1.381
Rolling-Forcing	0.572	2.757	6.393	2.673	0.265	1.078	96.657	10.603	2.962
Deep Forcing	0.597	2.632	6.523	2.668	**0.280**	0.878	92.482	11.065	1.397
MemFlow	0.600	2.836	6.407	2.651	0.269	1.035	87.724	10.840	1.343
Causal-Forcing	0.556	2.397	6.383	2.714	0.280	1.093	**99.558**	15.421	1.284
Helios	0.470	2.176	6.241	2.521	0.200	0.927	83.657	10.231	8.059
SWIFT	**0.605**	2.735	6.483	2.680	0.270	0.970	87.043	12.359	1.496
IAMFlow	0.602	**2.840**	6.415	2.625	0.268	1.016	86.957	6.049	1.772

Method	Instruction Adherence				Long-Horizon Stability							
	VIF↑	AIF↑	VID↓	AID↓	VA-D↓	VQ-D↓	PQ-D↓	AQ-D↓	SC↑	BC↑	AVAlign-D↓	AVSync-D↓
Native joint streaming audio-video models												
PixVerse R1	2.976	2.511	0.484	0.425	0.032	0.253	0.421	0.626	0.907	0.930	0.065	0.579
HappyOyster	**3.565**	2.692	0.510	0.508	0.032	0.417	0.623	**0.411**	0.904	0.934	0.077	0.534
OmniForcing	2.423	2.343	0.572	**0.277**	0.083	0.273	0.429	0.506	0.962	0.953	0.081	0.479
Cascaded streaming audio-video pipelines												
Odyssey-2	3.106	2.778	0.500	0.566	0.040	0.463	0.362	0.456	0.976	0.966	0.064	0.548
Self-Forcing	3.019	2.810	0.373	0.457	0.025	0.323	0.341	0.488	0.981	0.969	0.054	0.476
LongLive	3.164	2.814	0.317	0.581	0.023	0.367	0.280	0.475	0.986	0.973	0.052	0.476
Rolling-Forcing	3.009	2.858	0.425	0.474	0.043	0.427	0.329	0.425	0.980	0.966	0.064	0.506
Deep Forcing	3.041	2.629	0.359	0.545	0.023	0.334	0.289	0.497	0.979	0.967	0.062	0.553
MemFlow	3.142	2.854	0.333	0.492	0.028	0.502	0.319	0.503	**0.986**	0.972	0.054	**0.474**
Causal-Forcing	2.834	2.729	0.364	0.615	0.030	**0.189**	0.320	0.509	0.952	0.946	0.058	0.491
Helios	2.977	2.712	0.694	0.566	0.149	0.306	0.649	0.513	0.967	0.967	0.115	0.514
SWIFT	3.152	**2.887**	0.305	0.515	**0.022**	0.411	**0.272**	0.463	0.986	0.973	**0.049**	0.501
IAMFlow	3.192	2.789	**0.302**	0.600	0.023	0.425	0.331	0.563	0.986	**0.973**	0.057	0.490

Method	Interactive Response								State Retention and Reuse		
	VUF↑	AUF↑	PVC↑	PAC↑	PVUAR↑	PAUAR↑	PVRL↓	PARL↓	VSR↑	ASR↑	HDF↑
Native joint streaming audio-video models											
PixVerse R1	2.573	**2.895**	**4.278**	2.544	0.350	0.466	12.992	10.352	2.430	**4.108**	2.461
HappyOyster	**2.975**	2.879	3.744	**3.509**	**0.518**	**0.488**	14.484	13.621	2.799	3.772	**2.817**
OmniForcing	1.848	2.242	4.161	3.369	0.053	0.316	13.295	**7.925**	2.624	3.388	2.487
Cascaded streaming audio-video pipelines											
Odyssey-2	2.408	2.244	3.919	1.749	0.274	0.308	18.758	15.733	2.836	2.707	2.554
Self-Forcing	2.237	2.339	3.912	1.784	0.185	0.334	14.164	12.369	2.851	2.813	2.348
LongLive	2.341	2.138	3.631	1.809	0.241	0.266	11.792	13.078	2.724	2.712	2.212
Rolling-Forcing	2.315	2.221	4.086	1.800	0.217	0.298	13.234	12.820	2.542	2.597	2.502
Deep Forcing	2.295	2.146	3.879	1.770	0.220	0.288	13.433	12.650	2.449	2.601	2.206
MemFlow	2.309	2.286	3.630	1.880	0.217	0.326	13.698	13.154	**2.961**	2.879	2.385
Causal-Forcing	2.284	2.165	4.218	1.715	0.219	0.283	14.032	12.680	2.218	2.468	2.357
Helios	2.253	2.461	3.317	1.644	0.197	0.354	14.043	12.207	2.405	1.985	2.478
SWIFT	2.357	2.102	3.615	1.856	0.246	0.276	**10.340**	12.094	2.609	2.705	2.322
IAMFlow	2.319	2.190	3.655	1.819	0.222	0.297	11.683	13.001	2.836	2.903	2.424

achieves the highest visual quality (*VQ*) score of 2.840 but a visual quality drift (*VQ-D*) of 0.425, whereas Causal-Forcing records the lowest *VQ-D* of 0.189 despite a substantially lower *VQ* score of 2.397. A similar pattern appears in audio instruction adherence: SWIFT achieves the highest audio instruction fulfillment (*AIF*) score of 2.887 but an audio instruction drift (*AID*) of 0.515, while OmniForcing records the lowest *AID* of 0.277 despite the lowest *AIF* score of 2.343. These results show that low drift may coexist with lower absolute performance, underscoring the need to evaluate generation quality and instruction fulfillment jointly with their temporal changes.

Fast response does not guarantee successful interaction. The interactive-track results show that update achievement and response latency do not necessarily align. HappyOyster obtains the highest prompt-based visual and audio update achievement rates (*PVUAR* and *PAUAR*) of 0.518 and 0.488, respectively, but its corresponding response latencies (*PVRL* and *PARL*) are 14.484 and 13.621 seconds. In contrast, SWIFT achieves the lowest prompt-based visual response latency (*PVRL*) of 10.340 seconds but a *PVUAR* of only 0.246, while OmniForcing achieves the lowest prompt-based audio response latency (*PARL*) of 7.925 seconds but a *PAUAR* of 0.316. These results show that low

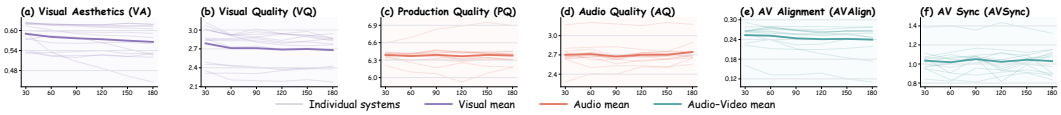

Figure 4: Temporal evolution of progressive-track quality and audio-video alignment over 180 seconds. Thin curves show individual systems, while thick curves show the mean across 13 systems. Higher is better for VA, VQ, PQ, AQ, and AVAlign, whereas lower is better for AVSync.

Table 3: Human alignment of selected evaluation metrics. We report Spearman rank correlations between per-model human win rates and the corresponding direction-adjusted benchmark scores.

Metric	VQ	AQ	VIF	AIF	VID	AID	VUF	AUF	PVC	PAC	PVUAR	PAUAR	PVRL	PARL	VSR	ASR	HDF
Spearman ρ	0.88	0.90	0.91	0.87	0.83	0.84	0.89	0.91	0.94	0.90	0.90	0.91	0.77	0.81	0.86	0.89	0.80

latency among achieved updates does not imply reliable update achievement; achievement rates and response latencies must therefore be interpreted jointly. Meanwhile, MemFlow records the highest visual state retention (*VSR*) score of 2.961, while no system exceeds 2.817 in history-dependency following (*HDF*), underscoring the difficulty of retaining and reusing state during interaction.

4.3 Analysis and Findings

Dimension-specific quality evolution. Over 180 seconds, we use degradation slope and trajectory variation to characterize overall direction and adjacent-segment fluctuations across six progressive-track dimensions. As shown in Fig. 4, *VA*, *AVAlign*, and *VQ* degrade in 13, 12, and 10 of the 13 systems, respectively, while *PQ*, *AQ*, and *AVSync* show mixed trends. Variation reveals instability hidden by the overall trends. Odyssey-2 and PixVerse R1 improve in *VQ* and *AQ*, respectively, yet fluctuate most in those dimensions, while SWIFT has both the steepest *AVSync* degradation and the largest variation. Overall, visual quality and semantic alignment degrade most consistently, whereas audio quality and temporal synchronization often fluctuate rather than decline monotonically.

Longer rollouts reveal track-specific bottlenecks. We compare performance at 60, 120, and 180 seconds across both tracks. In the progressive track, quality drift increases with duration. *VA-D* rises from 0.025 to 0.042 and *VQ-D* from 0.313 to 0.361, while alignment and synchronization drift change little beyond 120 seconds. In the interactive track, visual update performance remains stable, whereas audio update fulfillment and achievement decline. Response latencies increase by approximately 1.3 seconds, and audio state retention weakens while visual state retention remains stable. Consequently, progressive generation accumulates quality drift over longer rollouts, while repeated interaction exposes audio-side degradation and slower responses.

Content complexity mainly affects content realization rather than temporal consistency. Among the content-complexity factors, entity density causes the clearest visual degradation, reducing *VIF* from 3.090 to 2.816 and *VQ* from 2.760 to 2.651. Greater activity complexity also lowers visual quality, while multi-source audio poses the greatest challenge to audio instruction fulfillment. In contrast, subject and background consistency remain stable across these factors, indicating that increasing content complexity primarily affects content realization rather than temporal consistency.

Visual state changes and history reuse remain interaction bottlenecks. Visual and audio update fulfillment are comparable, yet visual updates have lower achievement rates (0.243 vs. 0.331) and longer response latencies (13.534 vs. 12.437 seconds). Direct actions are followed more reliably than state or environment changes, while history-dependency following declines with temporal distance. The primary bottlenecks are visual update achievement, state revision, and history reuse.

Qualitative failure analysis. As illustrated in Fig. 5, we identify 8 common failure modes in streaming generation. During progressive generation, models exhibit instruction drift, quality degradation, visual inconsistency, and audio-video drift. During interaction, they struggle to execute runtime updates, retain established states, or reuse earlier information. Abrupt temporal discontinuities emerge at prompt-update boundaries. Collectively, these observations underscore the bottlenecks in long-horizon stability, interactive response, state retention and reuse, and prompt-based continuity.



Figure 5: Qualitative failure cases. Common failure modes in streaming generation models.

4.4 Human Alignment

We conduct blind pairwise comparisons on 60 balanced cases covering both tracks and all 13 systems. Visual and audio quality (*VQ* and *AQ*) are validated on both tracks, instruction adherence on the progressive track, and interactive response and state retention and reuse on the interactive track. For each matched input and metric, three annotators compare anonymized outputs in randomized left-right order, with ties allowed. We aggregate human preferences into model-level win rates and report Spearman correlations with corresponding direction-adjusted benchmark scores. Table 3 shows strong human alignment across all 17 metrics, with correlations ranging from 0.77 to 0.94.

5 Insights and Future Directions

Our evaluation identifies stateful streaming as the central paradigm for future audio-video generation, where models should maintain a persistent world state, update it dynamically, and render each sequence with minimal latency.

Maintaining long-horizon stability. Long-horizon generation should maintain a persistent state rather than simply extend sequence duration. Future methods should preserve perceptual quality, instruction adherence, subject appearance, and audio-video alignment throughout the stream. Training and inference should jointly optimize absolute performance and long-horizon stability during unbounded generation rather than clip-level objectives.

Advancing reliable interaction. Each runtime update should modify the current world state rather than generate an independent one. The model must update specified audio-video content while preserving valid elements and reusing earlier information for rapid, seamless state transitions. This requires selective state-update mechanisms that jointly optimize update achievement, state retention and reuse, and response latency for robust free-form interaction.

Developing joint audio-video streaming. To construct a persistent world, current clip-level generation methods cannot fully resolve the challenges of real-time interaction. Future systems should generate audio and video jointly and incrementally from a shared and continuously updated latent state. This design must be coupled with system-level optimizations, such as state caching and coordinated audio-video scheduling, to minimize latency while maintaining synchronization.

Together, these directions shift the field from video clip generation toward continuously evolving audio-video world models that can be maintained, updated, and interacted with in real time.

6 Conclusion

We present StreamAV-Bench, a comprehensive benchmark for long-horizon and interactive streaming audio-video generation, comprising 320 expert-verified scenarios across progressive and interactive tracks. Its 32-dimensional framework evaluates quality and alignment, streaming continuity and efficiency, instruction adherence, long-horizon stability, interactive response, and state retention and reuse. Evaluating 13 native joint and cascaded systems under a unified protocol reveals failure modes overlooked by existing benchmarks, providing a standardized testbed for advancing streaming audio-video generation.

AI USE STATEMENT

GPT-5.4 was used during benchmark construction to generate natural-language prompts from human-authored structured scripts and to derive candidate visual and audio checklists. Each prompt and checklist was independently verified against the corresponding script and interval annotations by two domain experts, with disagreements resolved by a third expert. Gemini 3.1 Pro was used as the automated multimodal evaluator for the MLLM-based metrics described in the evaluation framework and supplementary material. Its model-level alignment with human preferences was assessed through the reported human study. AI assistance was also used to polish author-written text. The authors reviewed all AI-assisted content and take full responsibility for the final paper.

REFERENCES

Alibaba. Alibaba launches HappyOyster, a world model product for real-time immersive creation and interaction. Official product release, 2026. URL https://www.alizila.com/alibaba-launches-happyoyster-a-world-model-product-for-real-time-immersive-creation-and-interaction/. Accessed July 16, 2026.

Lichen Bai, Tianhao Zhang, Shitong Shao, Dingwei Tan, Qiyu Zhong, Zhengpeng Xie, Haopeng Li, Qinghao Huang, Dandan Shen, Tengjiao Ji, Wei Wang, Peicheng Wu, Yuxuan Zhao, Xiangyu Zhu, Welly Luo, Shurui Yang, and Zeke Xie. MaineCoon: Pursuing a real-time audio-visual social world model, 2026.

Minghong Cai, Xiaodong Cun, Xiaoyu Li, Wenze Liu, Zhaoyang Zhang, Yong Zhang, Ying Shan, and Xiangyu Yue. DiTCtrl: Exploring attention control in multi-modal diffusion transformer for tuning-free multi-prompt longer video generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7763–7772, 2025.

Minghong Cai, Qiulin Wang, Zongli Ye, Wenze Liu, Quande Liu, Weicai Ye, Xintao Wang, Pengfei Wan, Kun Gai, and Xiangyu Yue. VideoCanvas: Unified video completion from arbitrary spatiotemporal patches via in-context conditioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4475–4485, 2026.

Zhe Cao, Tao Wang, Jiaming Wang, Yanghai Wang, Yuanxing Zhang, Jiahao Wang, Jialu Chen, Miao Deng, Yubin Guo, Chenxi Liao, Yize Zhang, Zhaoxiang Zhang, and Jiaheng Liu. T2AV-Compass: Towards unified evaluation for text-to-audio-video generation, 2025.

Ho Kei Cheng, Masato Ishii, Akio Hayakawa, Takashi Shibuya, Alexander Schwing, and Yuki Mitsufuji. MMAudio: Taming multimodal joint training for high-quality video-to-audio synthesis, 2024.

Satvik Dixit, Koichi Saito, Zhi Zhong, Yuki Mitsufuji, and Chris Donahue. FoleyBench: A benchmark for video-to-audio models, 2025.

Weixi Feng, Jiachen Li, Michael Saxon, Tsu-jui Fu, Wenhu Chen, and William Yang Wang. TC-Bench: Benchmarking temporal compositionality in conditional video generation. In *Findings of the Association for Computational Linguistics: ACL 2025*, 2025a.

Xiaokun Feng, Haiming Yu, Meiqi Wu, Shiyu Hu, Jintao Chen, Chen Zhu, Jiahong Wu, Xiangxiang Chu, and Kaiqi Huang. NarrLV: Towards a comprehensive narrative-centric evaluation for long video generation, 2025b.

Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. ImageBind: One embedding space to bind them all, 2023.

Google DeepMind. Gemini 3.1 pro model card. https://deepmind.google/models/model-cards/gemini-3-1-pro/, 2026. Accessed: 2026-07-17.

Yoav HaCohen, Benny Brazowski, Nisan Chiprut, Yaki Bitterman, Andrew Kvochko, Avishai Berkowitz, et al. LTX-2: Efficient joint audio-visual foundation model, 2026.

Daili Hua, Xizhi Wang, Bohan Zeng, Xinyi Huang, Hao Liang, Junbo Niu, Xinlong Chen, Quanqing Xu, and Wentao Zhang. VABench: A comprehensive benchmark for audio-video generation, 2025.

Lianghua Huang, Zhi-Fan Wu, Wei Wang, Yupeng Shi, Mengyang Feng, et al. Wan-Streamer v0.1: End-to-end real-time interactive foundation models, 2026.

Xun Huang, Zhengqi Li, Guande He, Mingyuan Zhou, and Eli Shechtman. Self forcing: Bridging the train-test gap in autoregressive video diffusion. In *Advances in Neural Information Processing Systems*, 2025a.

Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, Yaohui Wang, Xinyuan Chen, Limin Wang, Dahua Lin, Yu Qiao, and Ziwei Liu. VBench: Comprehensive benchmark suite for video generative models, 2023.

Ziqi Huang, Fan Zhang, Xiaojie Xu, Yinan He, Jiashuo Yu, Ziyue Dong, Qianli Ma, Nattapol Chanpaisit, Chenyang Si, Yuming Jiang, Yaohui Wang, Xinyuan Chen, Ying-Cong Chen, Limin Wang, Dahua Lin, Yu Qiao, and Ziwei Liu. VBench++: Comprehensive and versatile benchmark suite for video generative models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025b. doi: 10.1109/TPAMI.2025.3633890.

Vladimir Iashin, Weidi Xie, Esa Rahtu, and Andrew Zisserman. Synchformer: Efficient synchronization from sparse cues, 2024.

Sihui Ji, Xi Chen, Shuai Yang, Xin Tao, Pengfei Wan, and Hengshuang Zhao. MemFlow: Flowing adaptive memory for consistent and efficient long video narratives, 2025.

Akio Kodaira, Tingbo Hou, Ji Hou, Markos Georgopoulos, Felix Juefei-Xu, Masayoshi Tomizuka, and Yue Zhao. StreamDiT: Real-time streaming text-to-video generation, 2025.

LAION-AI. Aesthetic predictor. `https://github.com/LAION-AI/aesthetic-predictor`, 2022. Accessed: 2026-07-26.

Haoran Li, Fredreic Li, Shichen Ma, Jie Huang, Yijun Liu, Jiaqi Shi, Yanwen Ma, Yaofeng Su, Xin Lu, Haoyu Wang, Xiaoxiao Ma, Guohui Zhang, Yaowei Li, Mingchen Zhong, Junhao Zhuang, Songchun Zhang, Weiyang Jin, Yuxuan Bian, Shiyi Zhang, Haojun Xu, Shuai Lu, Xin Han, Wei Tang, Tong He, Jiaqi Wang, Ping Luo, Haoyang Huang, Zeyue Xue, and Nan Duan. JoyAI-Echo: Pushing the frontier of long audio-visual generation. Technical report, 2026. URL `https://echo-team-joy-future-academy-jd.github.io/Echo-LongVideo-Page/`. Accessed July 16, 2026.

Mingxiang Liao, Hannan Lu, Xinyu Zhang, Fang Wan, Tianyu Wang, Yuzhong Zhao, Wangmeng Zuo, Qixiang Ye, and Jingdong Wang. Evaluation of text-to-video generation models: A dynamics perspective. In *Advances in Neural Information Processing Systems*, 2024.

Jinzhuo Liu, Jiangning Zhang, Wencan Jiang, Yabiao Wang, Dingkang Liang, Zhucun Xue, Ran Yi, and Yong Liu. Advancing narrative long video generation via training-free identity-aware memory, 2026a.

Kai Liu, Wei Li, Lai Chen, Shengqiong Wu, Yanhao Zheng, Jiayi Ji, Fan Zhou, Jiebo Luo, Ziwei Liu, Hao Fei, and Tat-Seng Chua. JavisDiT: Joint audio-video diffusion transformer with hierarchical spatio-temporal prior synchronization, 2025a.

Kaiqi Liu, Yunyao Mao, Ziqi Cai, Zheng Geng, Jing Wang, Qiulin Wang, Xintao Wang, Pengfei Wan, Kun Gai, Shuchen Weng, and Boxin Shi. MAVIN: Multi-shot audio-visual generation with customized narrative control, 2026b.

Kunhao Liu, Wenbo Hu, Jiale Xu, Ying Shan, and Shijian Lu. Rolling forcing: Autoregressive long video diffusion in real time, 2025b.

Tengfei Liu, Yang Shi, Xuanyu Zhu, Jiafu Tang, Liu Yang, Qixun Wang, Zhuoran Zhang, Yuqi Tang, et al. LongAV-Compass: Towards unified evaluation of minute-scale audio-visual generation across T2AV, I2AV, and V2AV, 2026c.

Yaofang Liu, Xiaodong Cun, Xuebo Liu, Xintao Wang, Yong Zhang, Haoxin Chen, Yang Liu, Tieyong Zeng, Raymond Chan, and Ying Shan. EvalCrafter: Benchmarking and evaluating large video generation models, 2023a.

Yuanxin Liu, Lei Li, Shuhuai Ren, Rundong Gao, Shicheng Li, Sishuo Chen, Xu Sun, and Lu Hou. FETV: A benchmark for fine-grained evaluation of open-domain text-to-video generation. In *Advances in Neural Information Processing Systems*, 2023b.

Chetwin Low, Weimin Wang, and Calder Katyal. Ovi: Twin backbone cross-modal fusion for audio-video generation, 2025.

Yuxin Mao, Xuyang Shen, Jing Zhang, Zhen Qin, Jinxing Zhou, Mochu Xiang, Yiran Zhong, and Yuchao Dai. TAVGBench: Benchmarking text to audible-video generation. In *Proceedings of the 32nd ACM International Conference on Multimedia*, 2024.

Odyssey. Introducing Odyssey-2: A general-purpose world model. Official technical release, 2025. URL https://odyssey.ml/introducing-odyssey-2.

OpenAI. Introducing GPT-5.4. https://openai.com/index/introducing-gpt-5-4/, 2026. Accessed: 2026-07-17.

PixVerse. PixVerse launches R1: A real-time world model that redefines AI video generation. Official technical release, 2026. URL https://pixverse.ai/en/blog/pixverse-launches-r1-real-time-world-model. Accessed July 16, 2026.

Ludan Ruan, Yiyang Ma, Huan Yang, Huiguo He, Bei Liu, Jianlong Fu, Nicholas Jing Yuan, Qin Jin, and Baining Guo. MM-Diffusion: Learning multi-modal diffusion models for joint audio and video generation, 2022.

Sizhe Shan, Qiulin Li, Yutao Cui, Miles Yang, Yuehai Wang, Qun Yang, Jin Zhou, and Zhao Zhong. HunyuanVideo-Foley: Multimodal diffusion with representation alignment for high-fidelity foley audio generation, 2025.

Yaofeng Su, Yuming Li, Zeyue Xue, Jie Huang, Siming Fu, Haoran Li, Ying Li, Zezhong Qian, Haoyang Huang, and Nan Duan. OmniForcing: Unleashing real-time joint audio-visual generation, 2026.

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, 2024.

Shanwen Tan, Hao Li, Jingtao Zhang, Xiaosong Jia, Xue Yang, Shaofeng Zhang, and Yanyong Zhang. SWIFT: Prompt-adaptive memory for efficient interactive long video generation, 2026.

Andros Tjandra, Yi-Chiao Wu, Baishan Guo, John Hoffman, Brian Ellis, Apoorv Vyas, Bowen Shi, Sanyuan Chen, Matt Le, Nick Zacharov, Carleigh Wood, Ann Lee, and Wei-Ning Hsu. Meta Audiobox Aesthetics: Unified automatic quality assessment for speech, music, and sound, 2025.

Yujie Wei, Yujin Han, Zhekai Chen, Yongming Li, Kaixun Jiang, Zhihang Liu, Quanhao Li, et al. MSAVBench: Towards comprehensive and reliable evaluation of multi-shot audio-video generation, 2026.

Shuai Yang, Wei Huang, Ruihang Chu, Yicheng Xiao, Yuyang Zhao, Xianbang Wang, Muyang Li, Enze Xie, Yingcong Chen, Yao Lu, Song Han, and Yukang Chen. LongLive: Real-time interactive long video generation, 2025.

Jung Yi, Wooseok Jang, Paul Hyunbin Cho, Jisu Nam, Heeji Yoon, and Seungryong Kim. Deep forcing: Training-free long video generation with deep sink and participative compression, 2025.

Tianwei Yin, Qiang Zhang, Richard Zhang, William T. Freeman, Fredo Durand, Eli Shechtman, and Xun Huang. From slow bidirectional to fast autoregressive video diffusion models, 2024.

Shenghai Yuan, Yuanyang Yin, Zongjian Li, Xinwei Huang, Xiao Yang, and Li Yuan. Helios: Real real-time long video generation model, 2026.

Shengjun Zhang, Zhang Zhang, Simin Huang, Zhenyu Tang, Hanyang Wang, Chensheng Dai, Min Chen, Yifan Li, Yuxin Li, Yingjie Chen, Hao Liu, Chen Li, Jing Lyu, and Yueqi Duan. MBench: A comprehensive benchmark on memory capability for video world models, 2026.

Zeyu Zhang, Jinyuan Mao, Shuning Chang, Yuanyu He, Yizeng Han, Jiasheng Tang, Fan Wang, and Bohan Zhuang. BIFE: Better interaction, fewer errors for minute-long video generation, 2025.

Xiangqing Zheng, Chengyue Wu, Kehai Chen, and Min Zhang. LoCoT2V-Bench: Benchmarking long-form and complex text-to-video generation, 2025.

Yanghao Zhou, Haitian Li, Rexar Lin, Heyan Huang, Jinxing Zhou, Changsen Yuan, Tian Lan, Ziqin Zhou, Yudong Li, Jiajun Xu, Jingyun Liao, YiMing Cheng, Xuefeng Chen, Xian-Ling Mao, and Yousheng Feng. MTAVG-bench: A diagnostic benchmark for multi-talker dialogue-centric audio-video generation, 2026a.

Ziwei Zhou, Zeyuan Lai, Rui Wang, Yifan Yang, Zhen Xing, Yuqing Yang, Qi Dai, Lili Qiu, and Chong Luo. AVGen-Bench: A task-driven benchmark for multi-granular evaluation of text-to-audio-video generation, 2026b.

Hongzhou Zhu, Min Zhao, Guande He, Hang Su, Chongxuan Li, and Jun Zhu. Causal forcing: Autoregressive diffusion distillation done right for high-quality real-time interactive video generation, 2026.

STREAMAV-BENCH: A COMPREHENSIVE BENCHMARK FOR STREAMING AUDIO-VIDEO GENERATION SUPPLEMENTARY MATERIAL

Anonymous authors
Paper under double-blind review

A BENCHMARK DETAILS

A.1 TASK SETTINGS

Both tracks contain 160 scenarios and generate 180-second audio-video outputs from a single rollout. The progressive track evaluates instruction adherence and long-horizon generation stability using a 180-second sequence generated from a single prompt describing a global state. The interactive track evaluates interactive response alongside state retention and reuse using an ordered prompt sequence, where the initial prompt defines the starting scenario and five subsequent prompts are provided every 30 seconds to update visual content, audio content, or both. For both tracks, we additionally evaluate the first 60, 120, and 180 seconds of the same rollout.

A.2 BENCHMARK CONSTRUCTION

Taxonomy design and scenario allocation. We construct 160 scenario themes and use every theme exactly once in each track. The two track instances of a theme share its semantic scope but are independently realized under their respective temporal protocols. Before script construction, each scenario is assigned a scene domain, audio domain, subject category, visual style, and entity, activity, and audio complexity levels. These allocations balance semantic coverage and content complexity. For the interactive track, the five runtime updates are further distributed across interaction type, update modality, and temporal dependency.

Script construction and interval annotation. For each allocated scenario, annotators develop a structured script describing the scene, visual style, primary subjects, activities, and sound sources, with the assigned complexity levels controlling their number and organization. Progressive scripts define coherent, open-ended audio-video processes that can continue throughout a 180-second rollout. Interactive scripts instead jointly plan the initial world and all five runtime updates as a single state trajectory, rather than constructing the updates independently.

Each runtime update expresses one primary interaction intent and is annotated with its interaction type, update modality, temporal dependency, and, when applicable, the earlier update on which it depends. The complete sequence is required to be causally ordered and state-consistent: established subjects, objects, activities, sounds, and states persist until an update explicitly changes them. An adjacent update depends on the immediately preceding runtime update, whereas a long-range update depends on a non-adjacent earlier runtime update.

Prompt generation. Based on the structured scripts and interval annotations, GPT-5.4 (OpenAI, 2026) generates natural-language prompts. Progressive scenarios yield one self-contained global prompt. Interactive scenarios yield a self-contained initial prompt followed by five concise runtime updates for subsequent intervals. All prompts follow a common, generation-oriented style.

Checklist generation and expert verification. Following prompt generation, GPT-5.4 uses fixed checklist-construction templates to derive modality-specific visual and audio checklists. Each interactive runtime update defines its evaluation target; joint audio-video updates are split into visual and audio targets while preserving their original semantics. For every adjacent or long-range update, the corresponding earlier prompt is associated with the target.

We assign two domain experts as independent reviewers and a third as an adjudicator. The reviewers examine the scripts, prompts, checklists, update targets, and temporal dependencies. They verify consistency with the assigned taxonomy and complexity design, update order, update modality, and temporal dependencies, and assess generation feasibility, audio-video plausibility, content coverage, temporal and cross-update state consistency, and evaluation consistency. Cases requiring correction are revised and reviewed again. The adjudicator resolves any disagreements, producing the final expert-verified set of 320 scenarios. All evaluation materials are finalized before model evaluation.

A.3 Benchmark Statistics

Table A1 reports the semantic distribution over the 320 scenarios. StreamAV-Bench spans 8 scene and 5 audio domains, 5 subject categories, and 4 visual styles, including 192 photorealistic and 128 non-photorealistic scenarios. Subject categories are assigned by the primary subject of each scenario, while audio domains identify the principal audio requirement.

Table A1: Semantic distribution over the 320 scenarios.

Dimension	Distribution
Scene domain	Daily Life 48; Work & Creation 44; Social 44; Performance 40; Sports 40; Transportation 36; Nature 36; Fictional 32
Audio domain	Speech & Vocalization 80; Foley & Object Sound 64; Mechanical Sound 48; Environmental Ambience 64; Music 64
Subject category	Human 173; Animal/Creature 48; Object 35; Vehicle 24; Robot 40
Visual style	Photorealistic 192; Anime/Cartoon 48; 3D/CG 48; Painting/Illustration 32

Content complexity. Both tracks characterize content complexity along three dimensions. *Entity complexity* measures the number of primary subjects that must remain individually recognizable: Single contains one primary subject, Multi contains two or three, and Dense contains four or more. Primary subjects may be people, animals, vehicles, robots, or other persistent focal entities rather than all visible objects. Each track contains 56 Single, 72 Multi, and 32 Dense scenarios. *Activity complexity* characterizes activity structure: Atomic contains one independent primary activity, Relational contains one dominant activity involving an explicit subject–subject or subject–object relation, and Concurrent contains multiple distinguishable activity threads occurring simultaneously. Each track contains 24 Atomic, 96 Relational, and 40 Concurrent scenarios. *Audio complexity* characterizes the number and temporal organization of salient sound sources: Single-Source contains one primary active source, Multi-Source contains multiple sources with little or no substantial overlap, and Overlapping contains multiple distinguishable sources active simultaneously. Each track contains 48 Single-Source, 64 Multi-Source, and 48 Overlapping scenarios. For interactive scenarios, these labels describe the initial world established before runtime updates.

Update complexity. The interactive track further characterizes each runtime update by what changes, which modality changes, and how much interaction history is required. *Interaction type* distinguishes agent actions, entity-state changes, environment changes, and audio changes; *update modality* distinguishes video-only, audio-only, and joint audio-video updates; and *temporal dependency* distinguishes independent updates from those relying on the immediately preceding or a non-adjacent earlier update. Across 800 updates, interaction types comprise 440 agent actions, 160 entity-state changes, 80 environment changes, and 120 audio changes; update modalities comprise 360 video-only, 160 audio-only, and 280 joint audio-video updates; temporal dependencies comprise 440 independent, 240 adjacent, and 120 long-range updates.

B Evaluation Framework Details

Our evaluation framework combines tailored expert models with Multimodal Large Language Model (MLLM) assessments guided by case-specific checklists. The 32 metrics are organized into six categories. The progressive track primarily measures instruction adherence and long-horizon stability, while the interactive track focuses on interactive response and state retention and reuse. Shared metrics evaluate streaming continuity and efficiency as well as quality and alignment on both tracks. For

each metric, we summarize the evaluated property, evaluator, score construction, and direction. Unless stated otherwise, observations are first aggregated within each case and then averaged uniformly across cases.

B.1 Quality and Alignment

We assess generated audio-video content along three complementary dimensions: visual quality, audio quality, and cross-modal alignment. The six metrics in this category are computed over six non-overlapping 30-second intervals and then averaged within each case.

(1) Visual Aesthetics (VA). VA measures perceptual visual appeal using the LAION Aesthetic Predictor (LAION-AI, 2022). Interval scores are normalized to $[0, 1]$ and averaged; higher is better.

(2) Visual Quality (VQ). VQ evaluates visual fidelity, subject integrity, motion naturalness, and artifact control using Gemini 3.1 Pro (Google DeepMind, 2026). The four dimensions are averaged on the original 1–5 scale; higher is better.

(3) Production Quality (PQ). PQ measures perceptual and production fidelity using the production-quality head of Audiobox Aesthetics (Tjandra et al., 2025). We report the original evaluator score; higher is better.

(4) Audio Quality (AQ). AQ evaluates audio naturalness and artifact control using Gemini 3.1 Pro. The two dimensions are averaged on a 1–5 scale; higher is better.

(5) Audio-Video Alignment (AVAlign). AVAlign measures semantic correspondence between generated audio and visual content using cosine similarity in the shared ImageBind embedding space (Girdhar et al., 2023). Similarities are averaged over intervals; higher is better.

(6) Audio-Video Synchronization (AVSync). AVSync measures temporal synchronization using the absolute audio-video offset predicted by Synchformer (Iashin et al., 2024). Offsets are averaged over intervals and reported in seconds; lower is better.

B.2 Streaming Continuity and Efficiency

(7) Native Boundary Continuity (NBC). NBC measures technical continuity at observable model-native generation-unit boundaries by detecting black frames, repeated frames, freezing, and flashes. To this end, NBC adopts the algorithmic transition detector and scoring function of LongAV-Compass (Liu et al., 2026b). Let r_b, r_d, t_f, and n_f denote the black-frame ratio, repeated-frame ratio, maximum freeze duration in seconds, and flash count, respectively. The boundary score is

$$\mathrm{NBC} = \mathrm{clip}(100 - (70r_b + 45r_d + 12t_f + 8n_f), 0, 100). \tag{A1}$$

For each system, we identify the boundaries between consecutive native generation units and use the same centered 4-second window whenever the full window is available. This captures artifacts introduced as generation proceeds from one unit to the next. Scores are averaged over boundaries within each case and then over all cases in both tracks. Higher scores indicate better continuity.

(8) Frames per Second (FPS). FPS measures generation throughput in output frames per second under each system's default recommended configuration. Higher is better.

(9) Time-to-First-Chunk (TTFC). TTFC measures the elapsed time from receiving a generation request until the system produces the first output chunk. Lower is better.

B.3 Instruction Adherence

For each progressive case, fixed visual and audio checklists derived from the global prompt define the requirements used by all instruction-adherence metrics.

(10–11) Video and Audio Instruction Fulfillment (VIF/AIF). VIF and AIF measure how completely the visual and audio requirements are realized over the full output. VIF covers scene, visual style, subjects, and activity, while AIF covers sound source, sound content, and temporal relations. Gemini 3.1 Pro scores each applicable criterion on a 1–5 completion scale, and criterion scores are averaged within each case; higher is better.

(12–13) Video and Audio Instruction Drift (VID/AID). VID and AID measure changes in visual and audio instruction adherence between the first and last intervals using the same fixed criteria. Following the endpoint comparison of HeliosBench (Yuan et al., 2026), for each matched criterion c we compute

$$D_c = \left| s_c^{(6)} - s_c^{(1)} \right|. \tag{A2}$$

VID and AID average D_c over matched visual and audio criteria, respectively. Both lie in $[0, 4]$; lower is better.

B.4 Long-Horizon Stability

The progressive track evaluates quality drift, visual consistency, and audio-video drift over the full generation horizon. Endpoint-based drift metrics compare the first and last 30-second intervals.

(14–17) Quality Drift. VA-D, VQ-D, PQ-D, and AQ-D measure the absolute endpoint change of their corresponding shared quality metrics:

$$Q\text{-D} = \left| q^{(6)} - q^{(1)} \right|. \tag{A3}$$

Differences are computed per case and then averaged; lower is better.

(18–19) Subject Consistency (SC) and Background Consistency (BC). SC and BC measure the temporal consistency of subject appearance and background scenes, respectively, following the official VBench-Long protocol (Huang et al., 2025b). Scores are reported on the original evaluator scale; higher is better.

(20–21) Audio-Video Drift (AVAlign-D/AVSync-D). AVAlign-D and AVSync-D measure absolute endpoint changes in audio-video semantic correspondence and temporal synchronization. They use the original AVAlign and AVSync units; lower is better.

B.5 Interactive Response

The interactive track evaluates whether systems realize runtime updates correctly, smoothly, and promptly. Metrics are evaluated for the five runtime updates introduced at 30, 60, 90, 120, and 150 seconds. VUF, AUF, PVUAR, PAUAR, PVRL, and PARL follow update-modality applicability, while PVC and PAC apply to all five updates.

(22–23) Video and Audio Update Fulfillment (VUF/AUF). VUF and AUF characterize visual and audio target realization within each post-update interval. For history-dependent updates, these current-interval measurements are complemented by HDF, which evaluates source-specific correctness from the annotated source and current intervals. Gemini 3.1 Pro assigns a 1–5 fulfillment score to each applicable target, and update scores are averaged within each case; higher is better.

(24) Prompt-Based Visual Continuity (PVC). PVC measures local visual continuity during prompt updates. It combines a deterministic transition score for black frames, repetition, freezing, and flashes with a Gemini 3.1 Pro judgment of unintended breaks, deformation, and disappearance, while allowing requested changes. The deterministic component applies the NBC artifact detector to each prompt update. Let S_{T} denote the resulting 0–100 transition score. We map it to the 1–5 scale as $S_{\mathrm{alg}} = 1 + 4S_{\mathrm{T}}/100$ and fuse it with the MLLM score:

$$S_{\mathrm{PVC}} = 0.70 S_{\mathrm{alg}} + 0.30 S_{\mathrm{MLLM}}. \tag{A4}$$

Scores are averaged over updates; higher is better.

(25) Prompt-Based Audio Continuity (PAC). PAC measures local audio continuity during prompt updates using Gemini 3.1 Pro. It penalizes unintended dropouts, hard cuts, repetition, clicks, clipping, and abrupt volume changes while allowing requested changes. Scores use a 1–5 scale and are averaged over updates; higher is better.

(26–27) Prompt-Based Visual and Audio Update Achievement Rates (PVUAR/PAUAR). PVUAR and PAUAR measure the fractions of applicable visual and audio updates whose core targets are clearly realized. A target counts as achieved only when the evaluator identifies a clear

Table A2: Evaluated systems. "Native joint" denotes native joint streaming audio-video generation, and "Cascaded" denotes a cascaded streaming audio-video pipeline.

System	Generation paradigm	Access
PixVerse R1	Native joint	Web
HappyOyster	Native joint	Web
OmniForcing	Native joint	Open
Odyssey-2	Cascaded	Web
Self-Forcing	Cascaded	Open
LongLive	Cascaded	Open
Rolling-Forcing	Cascaded	Open
Deep Forcing	Cascaded	Open
MemFlow	Cascaded	Open
Causal-Forcing	Cascaded	Open
Helios	Cascaded	Open
SWIFT	Cascaded	Open
IAMFlow	Cascaded	Open

realization time; merely beginning a change is insufficient. The case-level visual rate is

$$\mathrm{PVUAR} = \frac{\#\{\text{applicable visual updates with an achieved target}\}}{\#\{\text{applicable visual updates}\}}, \tag{A5}$$

and PAUAR is defined analogously over applicable audio updates. Rates lie in $[0, 1]$; higher is better.

(28–29) Prompt-Based Visual and Audio Response Latency (PVRL/PARL). Conditional on target achievement, PVRL and PARL measure how promptly the requested visual and audio targets are realized following an update, respectively. Measurements use a 0.5-second resolution. Latencies are averaged over successful updates within each case and then over cases with at least one achieved update; lower is better.

B.6 State Retention and Reuse

This category evaluates whether systems preserve states that should remain unchanged and correctly reuse relevant information from earlier interactions. HDF is computed only when its required source state is established.

(30) Visual State Retention (VSR). VSR measures whether subject appearance and the surrounding environment remain consistent across an update after accounting for requested visual changes. Gemini 3.1 Pro assigns 1–5 retention scores; higher is better.

(31) Audio State Retention (ASR). ASR measures whether recurring music, ambience, and other persistent sounds remain consistent when they are not targeted by the update. Gemini 3.1 Pro assigns a 1–5 retention score; higher is better.

(32) History-Dependency Following (HDF). HDF measures whether an adjacent or long-range update is correctly executed using the state established by its annotated earlier interaction. Gemini 3.1 Pro assigns a 1–5 score over established source states. Adjacent and long-range instances are combined within each case in proportion to their applicable updates; higher is better. Source-state establishment rates are reported in Table A5.

C Implementation Details

C.1 Evaluated Systems

We evaluate 13 systems across two generation paradigms, as summarized in Table A2. PixVerse R1 (PixVerse, 2026), HappyOyster (Alibaba, 2026), and OmniForcing (Su et al., 2026) are native joint streaming audio-video models. Cascaded pipelines pair HunyuanVideo-Foley (Shan et al., 2025) with native streaming video models: Odyssey-2 (Odyssey, 2025), Self-Forcing (Huang et al., 2025a), LongLive (Yang et al., 2025), Rolling-Forcing (Liu et al., 2025), Deep Forcing (Yi et al.,

2025), MemFlow (Ji et al., 2025), Causal-Forcing (Zhu et al., 2026), Helios (Yuan et al., 2026), SWIFT (Tan et al., 2026), and IAMFlow (Liu et al., 2026a). Open-source systems use official checkpoints and inference pipelines on NVIDIA H800 GPUs. Commercial systems are accessed via official web interfaces. All models are deployed with their default recommended configurations. Because the official web interfaces do not expose model-native generation-unit boundaries, NBC is unavailable for PixVerse R1, HappyOyster, and Odyssey-2. Web-interface timings conflate model inference with interface and network overhead and are therefore not directly comparable with local inference measurements. We therefore do not report FPS or TTFC for these systems.

C.2 MLLM Evaluation Settings

We use Gemini 3.1 Pro for all MLLM-based metrics. Videos are uniformly sampled at 2 FPS, except for PVC, which uses 6 FPS to assess short transitions around each prompt update. Depending on the metric, the evaluator receives video, audio, or synchronized audio-video clips.

VQ and AQ are evaluated over 30-second intervals, while VIF and AIF assess the complete 180-second output. VSR and ASR compare the intervals immediately before and after each update, and HDF compares the annotated source interval with the current interval. PVC and PAC assess visual and audio continuity at prompt updates, while VUF/AUF and PVRL/PARL evaluate whether and when the requested targets are realized. The MLLM scoring criteria and checklist-construction prompts are provided in Appendix F.

C.3 Interactive Continuation Protocol

For Self-Forcing (Huang et al., 2025a), Rolling-Forcing (Liu et al., 2025), Deep Forcing (Yi et al., 2025), Causal-Forcing (Zhu et al., 2026), and OmniForcing (Su et al., 2026), we use a shared runtime-update protocol based on the continuation strategies of Infinity-RoPE (Yesiltepe et al., 2026) and LongLive (Yang et al., 2025). The initial prompt starts the rollout; subsequent prompts become active at fixed 30-second boundaries while previously generated frames remain unchanged. At each boundary, conditioning switches to the active prompt and generation continues from the existing history.

C.4 Cascaded Audio Generation

Following prior long-form video-to-audio evaluation practice (Cheng et al., 2025; Zhang et al., 2025; Wei et al., 2026), each cascaded system pairs its streaming video generator with HunyuanVideo-Foley (Shan et al., 2025). Audio generation follows the same prompt schedule as the video stream. Each audio segment is conditioned on the corresponding video segment and the prompt active at that time, and the resulting segments are concatenated for evaluation.

C.5 Generation Controls

To ensure a consistent evaluation protocol, all systems process identical benchmark inputs: a single global prompt for the progressive track, and an ordered prompt sequence with fixed 30-second intervals for the interactive track. Outputs are converted to a common audio-video format before evaluation.

D Additional Experimental Results

D.1 Initial-World Alignment

The core interactive evaluation focuses on runtime-update response, state retention, and reuse. To complement these core dimensions, we additionally evaluate whether each system establishes the audio-video world requested by the initial prompt before the first runtime update. We apply the same VIF/AIF framework to checklists derived from the initial prompt over the first interval. Scores are averaged over applicable criteria within each case and then uniformly across the 160 interactive cases. Table A3 reports the resulting per-system scores.

Table A3: Initial-world alignment results on the interactive track. Initial VIF and Initial AIF reuse the VIF/AIF 1–5 scale. The best and second-best results are highlighted in **bold** and underlined.

System	Initial VIF↑	Initial AIF↑
Native joint streaming audio-video models		
PixVerse R1	2.954	2.467
HappyOyster	**3.364**	2.492
OmniForcing	2.527	2.299
Cascaded streaming audio-video pipelines		
Odyssey-2	3.084	2.643
Self-Forcing	3.011	2.638
LongLive	3.130	2.675
Rolling-Forcing	3.077	2.686
Deep Forcing	3.083	2.654
MemFlow	3.117	2.645
Causal-Forcing	2.930	2.590
Helios	2.908	2.710
SWIFT	3.131	**2.753**
IAMFlow	3.100	2.722

HappyOyster attains the highest Initial VIF, whereas SWIFT attains the highest Initial AIF. No single system leads both modalities, indicating complementary strengths in establishing the requested starting world.

D.2 Update Achievement and Response Latency

Update achievement measures how often requested changes are realized, while response latency measures how promptly successful realization occurs. Because PVRL and PARL are conditioned on achieved targets, lower latency does not by itself imply stronger overall responsiveness. Table A4 contrasts the system with the highest update achievement rates against those attaining the lowest visual and audio response latencies. HappyOyster obtains *PVUAR* and *PAUAR* scores of 0.518 and 0.488, with corresponding *PVRL* and *PARL* values of 14.484 and 13.621 seconds. SWIFT achieves a lower *PVRL* of 10.340 seconds but a *PVUAR* of 0.246, while OmniForcing achieves the lowest *PARL* of 7.925 seconds but a *PAUAR* of 0.316.

Table A4: Prompt-based visual and audio update achievement rates and response latencies for three representative systems. PVRL and PARL are computed over achieved targets.

System	PVUAR↑	PVRL↓	PAUAR↑	PARL↓
HappyOyster	0.518	14.484	0.488	13.621
SWIFT	0.246	10.340	0.276	12.094
OmniForcing	0.053	13.295	0.316	7.925

HappyOyster realizes a larger fraction of requested updates, whereas the lower visual or audio latency of SWIFT and OmniForcing is measured over a substantially smaller successful subset. The failure-penalized comparison in Section D.4 evaluates achievement and latency jointly.

D.3 History-Dependency Source Establishment

History-dependency following can be evaluated only when the required state is present in its annotated earlier interaction. Table A5 reports source-state establishment rates for adjacent (HDF-A) and long-range (HDF-L) dependencies, together with the corresponding HDF scores over established sources.

Table A5: History-dependency source-state establishment rates and conditional HDF scores. Establishment rates are reported on $[0, 1]$, while HDF-A and HDF-L scores use the original 1–5 scale over established sources. The best and second-best results are highlighted in **bold** and underlined.

System	Source Establishment		Conditional HDF	
	Adj.↑	Long↑	Adj.↑	Long↑
PixVerse R1	0.610	0.653	2.508	2.435
HappyOyster	**0.741**	<u>0.715</u>	**2.885**	**2.718**
OmniForcing	0.323	0.358	<u>2.641</u>	2.231
Odyssey-2	0.459	0.453	2.631	2.489
Self-Forcing	0.568	0.606	2.445	2.126
LongLive	0.612	0.655	2.212	2.167
Rolling-Forcing	0.607	0.595	2.614	2.138
Deep Forcing	0.560	0.555	2.278	2.088
MemFlow	0.571	0.580	2.377	<u>2.500</u>
Causal-Forcing	0.537	0.559	2.363	2.357
Helios	0.536	0.544	2.491	2.357
SWIFT	<u>0.616</u>	**0.716**	2.265	2.348
IAMFlow	0.565	0.610	2.496	2.216

HappyOyster leads adjacent source establishment and both conditional HDF scores, while SWIFT attains the highest long-range source establishment. OmniForcing ranks second in conditional HDF-A despite the lowest adjacent establishment rate, showing that source establishment and dependency following capture distinct failure modes.

D.4 Success- and Coverage-Adjusted Diagnostics

We define success-adjusted response latencies (SA-PVRL/SA-PARL) and coverage-adjusted HDF scores (CA-HDF-A/CA-HDF-L) by assigning fixed penalties to unachieved updates and unestablished dependency sources.

For modality $m \in \{\mathrm{V}, \mathrm{A}\}$, let $\mathcal{U}_c^m$ be the applicable updates in case c, $a_{c,u}^m$ indicate whether the target is achieved, and $\ell_{c,u}^m$ be its response latency when achieved. Let $\mathcal{C}_m$ denote cases containing at least one such update. With the fixed observation horizon $T = 30$ seconds, we define

$$\widetilde{\ell}_{c,u}^m = \begin{cases} \ell_{c,u}^m, & a_{c,u}^m = 1, \\ T, & a_{c,u}^m = 0, \end{cases} \qquad L_{\mathrm{SA}}^m = \frac{1}{|\mathcal{C}_m|} \sum_{c \in \mathcal{C}_m} \frac{1}{|\mathcal{U}_c^m|} \sum_{u \in \mathcal{U}_c^m} \widetilde{\ell}_{c,u}^m. \tag{A6}$$

This yields success-adjusted PVRL and PARL (SA-PVRL/SA-PARL), for which unachieved targets receive the observation-window cap rather than being excluded. These use the same latency definition as PVRL and PARL.

For a dependency class $d \in \{\text{HDF-A}, \text{HDF-L}\}$, let $z_{c,j}^d$ indicate whether the source state for check j is established and let $s_{c,j}^d \in [1, 5]$ be the corresponding conditional score. We use $\mathcal{J}_c^d$ for the checks in case c and $\mathcal{C}_d$ for cases containing at least one such check, and define

$$\widetilde{s}_{c,j}^d = \begin{cases} (s_{c,j}^d - 1)/4, & z_{c,j}^d = 1, \\ 0, & z_{c,j}^d = 0, \end{cases} \qquad S_{\mathrm{CA}}^d = \frac{1}{|\mathcal{C}_d|} \sum_{c \in \mathcal{C}_d} \frac{1}{|\mathcal{J}_c^d|} \sum_{j \in \mathcal{J}_c^d} \widetilde{s}_{c,j}^d. \tag{A7}$$

This yields CA-HDF-A and CA-HDF-L for the two dependency classes. Both scores lie on $[0, 1]$.

Table A6: Sensitivity diagnostics on the interactive track. SA-PVRL and SA-PARL are measured in seconds and assign the 30-second observation-window cap to unachieved targets. CA-HDF metrics combine source-state establishment with normalized conditional HDF quality. The best and second-best results are highlighted in **bold** and underlined.

System	SA-PVRL↓	SA-PARL↓	CA-HDF-A↑	CA-HDF-L↑
PixVerse R1	<u>24.011</u>	**20.855**	0.232	<u>0.234</u>
HappyOyster	**21.991**	<u>21.861</u>	**0.354**	**0.306**
OmniForcing	29.128	23.046	0.136	0.109
Odyssey-2	26.838	25.611	0.196	0.168
Self-Forcing	27.081	24.156	0.206	0.169
LongLive	25.575	25.490	0.186	0.187
Rolling-Forcing	26.355	24.941	<u>0.250</u>	0.155
Deep Forcing	26.420	25.024	0.181	0.150
MemFlow	26.503	24.433	0.200	0.216
Causal-Forcing	26.497	25.117	0.184	0.188
Helios	26.841	23.712	0.202	0.181
SWIFT	25.150	25.195	0.192	0.232
IAMFlow	25.976	24.936	0.213	0.183

HappyOyster obtains the lowest SA-PVRL and the highest CA-HDF-A and CA-HDF-L, while PixVerse R1 obtains the lowest SA-PARL.

D.5 Temporal Evolution

We evaluate progressive-track quality and audio-video alignment over six consecutive 30-second intervals. For each case c, let $q_{c,i}$ denote its score in interval i and $t_i \in \{30, 60, \ldots, 180\}$ the corresponding time in seconds. Degradation slope captures the overall direction across all segments, while trajectory variation measures adjacent-segment fluctuations. We first fit each case trajectory using

$$\hat{\beta}_c = \frac{\sum_{i=1}^{6}(t_i - \bar{t})(q_{c,i} - \bar{q}_c)}{\sum_{i=1}^{6}(t_i - \bar{t})^2}. \tag{A8}$$

The case-level degradation slope is $-100\hat{\beta}_c$ for higher-is-better dimensions and $100\hat{\beta}_c$ for AVSync, such that a positive value always indicates deterioration and the unit is score change per 100 seconds. We additionally compute case-level trajectory variation as

$$\mathrm{TV}_c = \frac{1}{5}\sum_{i=1}^{5} |q_{c,i+1} - q_{c,i}| \,. \tag{A9}$$

Both statistics are then averaged uniformly across cases for each system. TV remains in each dimension's native score unit.

Visual aesthetics (*VA*), audio-video alignment (*AVAlign*), and visual quality (*VQ*) degrade in 13, 12, and 10 of the 13 systems, respectively. By contrast, production quality (*PQ*), audio quality (*AQ*), and audio-video synchronization (*AVSync*) exhibit mixed directions, degrading in 9, 6, and 7 systems. Trajectory variation further reveals instability concealed by the overall trend: Odyssey-2 and PixVerse R1 show improving *VQ* and *AQ* trends (-0.056 and -0.019), yet the largest variations in their respective dimensions (0.423 and 0.508). Meanwhile, SWIFT exhibits both the steepest *AVSync* degradation (0.062) and the largest *AVSync* variation (0.536). Streaming evolution is therefore dimension-specific: visual quality and semantic alignment degrade most consistently, whereas audio quality and temporal synchronization often fluctuate rather than decline monotonically.

Rollout-horizon analysis. We compare performance at 60, 120, and 180 seconds across both tracks, using the same 180-second rollouts. For progressive cases, the endpoint drift of metric m at horizon h is

$$D_c^m(h) = \left| q_{c,k(h)}^m - q_{c,1}^m \right|, \qquad k(60) = 2,\ k(120) = 4,\ k(180) = 6. \tag{A10}$$

For interactive cases, the three horizons aggregate the first one, three, and five runtime updates, respectively. Across these prefixes, 60, 142, and 160 cases contain at least one audio update, while history-dependent updates account for 0%, 31.0%, and 45.0%, respectively. Interactive comparisons therefore capture both accumulated interaction and changing update composition. Scores are first aggregated within each case and then averaged across cases and systems. The resulting values are reported in Table A7.

Table A7: Rollout-horizon analysis at 60, 120, and 180 seconds, averaged uniformly across the 13 systems. Progressive values report prefix endpoint drift, while interactive values aggregate all runtime updates introduced by each horizon. PVRL and PARL are reported in seconds.

Progressive track				Interactive track			
Metric	60 s	120 s	180 s	Metric	60 s	120 s	180 s
VA-D↓	0.025	0.036	0.042	VUF↑	2.347	2.327	2.347
VQ-D↓	0.313	0.347	0.361	AUF↑	2.481	2.340	2.332
PQ-D↓	0.295	0.356	0.382	PVUAR↑	0.233	0.231	0.243
AQ-D↓	0.449	0.460	0.495	PAUAR↑	0.361	0.333	0.331
AVAlign-D↓	0.059	0.067	0.065	PVRL↓	12.283	13.268	13.534
AVSync-D↓	0.495	0.521	0.509	PARL↓	11.153	12.116	12.437
				VSR↑	2.606	2.619	2.637
				ASR↑	3.022	2.952	2.895

Averaged across the 13 systems, the progressive track exhibits increasing quality drift: visual aesthetics drift (*VA-D*) rises from 0.025 to 0.036 and 0.042, while visual quality drift (*VQ-D*) increases from 0.313 to 0.347 and 0.361. Production and audio quality drift (*PQ-D* and *AQ-D*) follow the same pattern, whereas audio-video alignment and synchronization drift (*AVAlign-D* and *AVSync-D*) change little beyond 120 seconds. In the interactive track, visual update fulfillment (*VUF*) and achievement (*PVUAR*) remain stable, whereas audio update fulfillment (*AUF*) decreases from 2.481 to 2.340 and 2.332, and audio update achievement (*PAUAR*) decreases from 0.361 to 0.333 and 0.331. Visual and audio response latencies (*PVRL* and *PARL*) both increase by approximately 1.3 seconds. Audio state retention (*ASR*) also declines from 3.022 to 2.952 and 2.895, while visual state retention (*VSR*) remains stable.

Controlled interaction-depth analysis. To assess robustness to changes in evaluated cases and temporal-dependency mix, we perform two complementary controls. First, we recompute cumulative audio metrics on the fixed cohort of 60 cases containing an audio update by 60 seconds. Second, we evaluate each turn using only independent audio updates. The results are reported in Table A8.

Table A8: Controlled analyses of interactive audio performance, averaged uniformly across the 13 systems. The fixed-cohort analysis uses the same 60 audio-applicable cases at all horizons. The turn-wise analysis includes only independent audio updates. P1–P5 denote update positions, and N is the number of applicable benchmark updates before system-level aggregation.

Fixed 60-case cohort				Independent audio updates					
Metric	60 s	120 s	180 s	Metric	P1	P2	P3	P4	P5
AUF↑	2.481	2.373	2.363	AUF↑	2.481	2.146	2.400	2.445	2.258
PAUAR↑	0.361	0.338	0.338	PAUAR↑	0.361	0.278	0.342	0.364	0.311
PARL↓	11.153	11.551	11.632	PARL↓	11.153	12.021	12.777	12.358	13.822
ASR↑	2.734	2.743	2.662	N	60	26	64	50	22

Fixing the audio-applicable cohort retains lower *AUF*, *PAUAR*, and *ASR* and higher *PARL* at 180 seconds than at 60 seconds, although the differences are smaller than those on the full set. Among independent audio updates, *AUF* and *PAUAR* are lower and *PARL* is higher at P5 than at P1, while intermediate turns remain non-monotonic. Controls indicate that changing update composition explains part, but not all, of the audio-side trend. Progressive generation accumulates quality drift over longer rollouts, while repeated interaction exposes audio-side degradation and slower responses.

D.6 Content and Update Complexity Results

We analyze the progressive track across entity, activity, and audio complexity. Entity density produces the clearest visual degradation: visual instruction fulfillment (*VIF*) decreases from 3.090 for single-entity cases to 2.816 for dense cases, while visual quality (*VQ*) decreases monotonically from 2.760 to 2.651. Activity complexity reveals a similar visual bottleneck, with *VQ* decreasing by 0.129 from atomic to concurrent activities without a corresponding decline in audio instruction fulfillment (*AIF*). Multi-source cases attain the lowest *AIF* (2.523). In contrast, subject consistency (*SC*) and background consistency (*BC*) remain nearly unchanged across all three factors. Increasing content complexity therefore primarily disrupts the realization of required visual content, while subject appearance and background scenes remain comparatively stable.

For the interactive track, visual and audio update fulfillment (*VUF* and *AUF*) are comparable (2.347 vs. 2.332). However, visual updates have lower achievement rates than audio updates (*PVUAR*/*PAUAR*: 0.243/0.331) and respond more slowly (*PVRL*/*PARL*: 13.534/12.437 seconds). Direct agent actions also attain higher *VUF* (2.418) than entity-state and environment changes (2.274 and 2.283). For history-dependent updates, *HDF* reaches only 2.477 for adjacent and 2.321 for long-range dependencies. The primary interaction bottlenecks therefore lie in reliably realizing visual changes, revising established states, and reusing earlier information.

E Human Alignment

We conduct blind pairwise comparisons on a balanced subset of 60 cases covering both tracks and all 13 systems. Visual and audio quality (*VQ* and *AQ*) are validated across both tracks, whereas instruction adherence metrics are evaluated on the progressive track, and interactive response and state retention and reuse metrics on the interactive track. For shared metrics, the 60 cases comprise 30 from each track; track-specific metrics use the corresponding 30-case subset. For each metric, we sample 120 paired output comparisons from all 13 systems using balanced random pairing. Each pair is independently evaluated by three annotators with randomized left-right order and ties allowed. This produces 2,040 pairwise tasks and 6,120 individual judgments across the 17 metrics. Majority vote determines the final label. Human preferences are aggregated into model-level win rates, with a tie contributing 0.5 to each system. We report the Spearman correlation between human win rates and the corresponding direction-adjusted benchmark scores.

Table A9: Human alignment of selected evaluation metrics. We report Spearman rank correlations between per-model human win rates and the corresponding direction-adjusted benchmark scores.

Metric	VQ	AQ	VIF	AIF	VID	AID	VUF	AUF	PVC	PAC	PVUAR	PAUAR	PVRL	PARL	VSR	ASR	HDF
Spearman ρ	0.88	0.90	0.91	0.87	0.83	0.84	0.89	0.91	0.94	0.90	0.90	0.91	0.77	0.81	0.86	0.89	0.80

As shown in Table A9, all 17 metrics show strong model-level human alignment, with correlations ranging from 0.77 to 0.94. Prompt-based visual continuity (*PVC*) attains the highest correlation, whereas visual and audio response latencies (*PVRL* and *PARL*) show lower correlations than the corresponding achievement and fulfillment metrics.

F Evaluation and Construction Prompts

This section provides the fixed prompts used for checklist construction, update decomposition, and MLLM evaluation. Curly-brace placeholders denote case-specific content.

F.1 Checklist Construction and Update Decomposition

Visual Instruction Checklist Template

```
# Task

Given a final audio-video generation prompt, construct a structured
checklist for Video Instruction Fulfillment (VIF).

The checklist contains four fields:

Scene
The environment or setting of the generated content, including specified
locations, surroundings, and important scene elements.

Visual Style
The specified visual style, artistic form, rendering style, or visual
medium.

Subjects
The central subjects specified in the prompt, including their number,
type or identity, and explicit visual attributes.

Activity
The primary activities specified in the prompt, including relevant
subject-subject or subject-object interactions.

# Instructions

- Generate one concise Yes/No question for each applicable field.
- Use only visual requirements explicitly stated in the prompt.
- Each question should be self-contained and objectively verifiable
  from the generated video.
- Closely related requirements within the same field may be combined.
- Do not introduce requirements that are not stated in the prompt.
- Phrase each question so that the expected answer is "Yes".
- If no applicable requirement exists for a field, return null.

Return strict JSON with no additional text.

# Output Format

{
  "video_instruction_following": {
    "scene": "<English question or null>",
    "visual_style": "<English question or null>",
    "subjects": "<English question or null>",
    "activity": "<English question or null>"
  }
}

Final generation prompt:

{generation_prompt}
```

Audio Instruction Checklist Template

```
# Task

Given a final audio-video generation prompt, construct a structured
checklist for Audio Instruction Fulfillment (AIF).

The checklist contains three fields:

Sound Source
The specified agent, entity, object, mechanism, or environmental source
responsible for producing a sound.

Sound Content
The specified audible events or sound content, including explicit
acoustic properties when applicable.

Temporal Relation
The specified temporal relationships among sounds, such as overlap,
simultaneity, separation, alternation, intermittency, or ordering.
```

```
# Instructions

- Generate one concise Yes/No question for each applicable field.
- Use only auditory requirements explicitly stated in the prompt.
- Each question should be self-contained and objectively verifiable
  from the generated audio-video content.
- Closely related requirements within the same field may be combined.
- Keep Sound Source focused on who or what produces the sound, and
  Sound Content focused on what is heard.
- Do not introduce requirements that are not stated in the prompt.
- Phrase each question so that the expected answer is "Yes".
- If no applicable requirement exists for a field, return null.

Return strict JSON with no additional text.

# Output Format

{
  "audio_instruction_following": {
    "sound_source": "<English question or null>",
    "sound_content": "<English question or null>",
    "temporal_relation": "<English question or null>"
  }
}

Final generation prompt:

{generation_prompt}
```

Joint Audio-Video Update Splitter

```
# Task

Split a Joint Audio-Video runtime update into a visual update and an audio
update.

# Instructions

- The visual update should contain the explicitly requested visual change.
- The audio update should contain the explicitly requested audio change.
- Preserve the original wording and semantic details whenever possible.
- Do not rewrite beyond what is necessary for the split or infer any new
  content.

Return strict JSON with no additional text.

# Output Format

{
  "visual_update": "<English visual update>",
  "audio_update": "<English audio update>"
}

Joint Audio-Video runtime update:

{runtime_update}
```

F.2 MLLM JUDGE PROMPTS

F.2.1 QUALITY

VQ Judge

```
# Task

Evaluate the visual quality of the provided generated video.

Assess four aspects: visual fidelity, subject integrity, motion naturalness,
and visual artifacts.

Assign a score from 1 to 5 to each aspect.
```

```
# Scoring Criteria

Visual Fidelity

1 = Visual quality is severely degraded by blur, exposure errors,
    compression, corruption, or other major failures.
2 = Visual quality is poor, with frequent degradation that noticeably
    affects viewing.
3 = Visual quality is acceptable but contains noticeable quality defects.
4 = Visual quality is good overall, with only minor imperfections.
5 = Visual content is sharp, clear, properly exposed, and well rendered.


Subject Integrity

1 = Main subjects are severely deformed or difficult to recognize.
2 = Subjects are recognizable but frequently distorted, unstable, or
    structurally broken.
3 = Subjects are mostly recognizable, with noticeable but tolerable
    identity or geometry inconsistencies.
4 = Subjects remain stable and structurally intact, with only minor defects.
5 = Subjects are consistently clear, complete, and visually coherent.


Motion Naturalness

1 = Motion is severely broken, physically incoherent, or impossible to
    meaningfully assess.
2 = Motion is largely stiff, incorrect, or visibly unnatural.
3 = Motion is understandable but contains noticeable jitter, irregularity,
    or unnatural dynamics.
4 = Motion is natural and coherent overall, with only minor local issues.
5 = Motion is smooth, fluent, and physically plausible.


Visual Artifacts

1 = Severe generation artifacts dominate the visual content.
2 = Obvious artifacts occur frequently and substantially disrupt viewing.
3 = Artifacts are noticeable but the video remains generally viewable.
4 = Only minor or occasional artifacts are visible.
5 = No obvious generation artifacts are observed.

# Instructions

Score the four aspects independently and provide a brief overall reason.

Return strict JSON with no additional text.

# Output Format

{
  "visual_fidelity": <1 | 2 | 3 | 4 | 5>,
  "subject_integrity": <1 | 2 | 3 | 4 | 5>,
  "motion_naturalness": <1 | 2 | 3 | 4 | 5>,
  "visual_artifacts": <1 | 2 | 3 | 4 | 5>,
  "reason": "<short reason>"
}
```

AQ Judge

```
# Task

Evaluate the audio quality of the provided generated audio-video clip.

Assess two aspects: audio naturalness and audio artifacts.

Assign a score from 1 to 5 to each aspect.

# Scoring Criteria

Audio Naturalness

1 = The audio sounds highly unnatural, implausible, or strongly synthetic.
2 = The audio remains recognizable but sounds noticeably artificial, with
    poor spatial plausibility or weak acoustic fit to the scene.
3 = The audio is generally plausible, though noticeable synthetic qualities
```

```
    or imperfect scene fit remain.
4 = The audio sounds natural, clear, and appropriate to the scene, with only
    minor unnaturalness.
5 = The audio is highly natural, clear, spatially plausible, and convincing
    within the generated scene.


Audio Artifacts

1 = Severe technical artifacts such as clipping, buzzing, dropouts, glitches,
    abrupt silence, or repetitive loops strongly disrupt the audio.
2 = Obvious artifacts occur frequently and noticeably interfere with listening.
3 = Artifacts are noticeable, but the audio remains generally usable.
4 = Only minor or occasional artifacts are present.
5 = The audio is clean and stable, with no obvious technical artifacts.

# Instructions

Score the two aspects independently and provide a brief overall reason.

Do not evaluate visual quality or frame-level synchronization, and score artifacts only
     for technical audio defects.

Return strict JSON with no additional text.

# Output Format

{
  "audio_naturalness": <1 | 2 | 3 | 4 | 5>,
  "audio_artifacts": <1 | 2 | 3 | 4 | 5>,
  "reason": "<short reason>"
}
```

F.2.2 Instruction Fulfillment and Drift

VIF Judge

```
Evaluate visual instruction fulfillment for this video clip.

Judge whether the visible semantic content requested by each criterion is
clearly expressed in the clip. Use only visual evidence.

Criteria:
{criteria}

Please rate the fulfillment of each criterion on a 5-point Likert scale:
- 1: No Fulfillment (The requested content is absent or clearly incorrect).
- 2: Weak Fulfillment (Only limited evidence is present, with major missing or incorrect
     elements).
- 3: Partial Fulfillment (The criterion is recognizable but only partly realized, with
     noticeable deficiencies).
- 4: Strong Fulfillment (The criterion is mostly realized, with only minor missing or
     incorrect details).
- 5: Full Fulfillment (The criterion is clearly and completely realized).

You must respond ONLY with a valid JSON object using the following structure:
{
  "answers": [
    {
      "id": "<criterion_id>",
      "reason": "A short explanation for your rating.",
      "score": <an integer between 1 and 5>
    }
  ]
}
```

AIF Judge

```
Evaluate audio instruction fulfillment for this synchronized audio-video clip.

Judge whether the audible semantic content requested by each criterion is
clearly expressed in the clip. Base each score on audible evidence.
Use visual content only when necessary to identify an explicitly requested
```

```
sound source, event condition, or temporal/causal relation. Do not include
visual fulfillment or general audio-video synchronization in the score.

Criteria:
{criteria}

Please rate the fulfillment of each criterion on a 5-point Likert scale:
- 1: No Fulfillment (The requested content is absent or clearly incorrect).
- 2: Weak Fulfillment (Only limited evidence is present, with major missing or incorrect
     elements).
- 3: Partial Fulfillment (The criterion is recognizable but only partly realized, with
     noticeable deficiencies).
- 4: Strong Fulfillment (The criterion is mostly realized, with only minor missing or
     incorrect details).
- 5: Full Fulfillment (The criterion is clearly and completely realized).

You must respond ONLY with a valid JSON object using the following structure:
{
  "answers": [
    {
      "id": "<criterion_id>",
      "reason": "A short explanation for your rating.",
      "score": <an integer between 1 and 5>
    }
  ]
}
```

VID Endpoint Judge

```
Evaluate visual instruction alignment for this video clip.

Judge how well the visible semantic content requested by each criterion aligns
with the content of the clip. Use only visual evidence.

Criteria:
{criteria}

Please rate the alignment of each criterion on a 5-point Likert scale:
- 1: No Alignment (The requested content is absent or clearly incorrect).
- 2: Weak Alignment (Only limited matching evidence is present, with major omissions or
     deviations).
- 3: Partial Alignment (The criterion is recognizable but only partly matched).
- 4: Strong Alignment (The criterion is mostly matched, with only minor discrepancies).
- 5: Full Alignment (The criterion is clearly and completely matched).

You must respond ONLY with a valid JSON object using the following structure:
{
  "answers": [
    {
      "id": "<criterion_id>",
      "reason": "A short explanation for your rating.",
      "score": <an integer between 1 and 5>
    }
  ]
}
```

AID Endpoint Judge

```
Evaluate audio instruction alignment for this synchronized audio-video clip.

Judge how well the audible semantic content requested by each criterion aligns
with the content of the clip. Base each score on audible evidence. Use visual
content only when necessary to identify an explicitly requested sound source,
event condition, or temporal/causal relation. Do not include visual alignment
or general audio-video synchronization in the score.

Criteria:
{criteria}

Please rate the alignment of each criterion on a 5-point Likert scale:
- 1: No Alignment (The requested content is absent or clearly incorrect).
- 2: Weak Alignment (Only limited matching evidence is present, with major omissions or
     deviations).
- 3: Partial Alignment (The criterion is recognizable but only partly matched).
```

```
- 4: Strong Alignment (The criterion is mostly matched, with only minor discrepancies).
- 5: Full Alignment (The criterion is clearly and completely matched).

You must respond ONLY with a valid JSON object using the following structure:
{
  "answers": [
    {
      "id": "<criterion_id>",
      "reason": "A short explanation for your rating.",
      "score": <an integer between 1 and 5>
    }
  ]
}
```

F.2.3 RUNTIME-UPDATE FULFILLMENT

VUF Judge

```
# Task

Evaluate Video Update Fulfillment (VUF) for the provided 30-second
synchronized audio-video interval after a runtime update.

Judge only the visual change explicitly requested by the current update.
Use audio only when necessary to identify an explicitly requested event
condition or visual consequence. Do not turn audio fulfillment or general
audio-video synchronization into part of the VUF score. Do not penalize
unrelated visual-quality defects unless they make the requested visual result
absent, incorrect, or unjudgeable.

Modality-specific visual update:
{visual_update}

# Fulfillment Scale

1 = No Fulfillment: The requested visual change is absent, or the main result
    clearly contradicts the update.
2 = Weak Fulfillment: There is an update-related attempt, preparation, or weak
    response, but the core visual target is not achieved.
3 = Partial Fulfillment: The core visual interaction is recognizable but
    incomplete, or important requested details are incorrect.
4 = Strong Fulfillment: The core visual target is correctly achieved, with
    only minor missing or incorrect details.
5 = Full Fulfillment: The core visual target, explicit semantic details, and
    any applicable requested causal or temporal relation are fully realized.

# Temporal Localization

- The evaluation timeline spans 30.0 seconds. The visual stream is sampled at
  2 FPS.
- Timestamp 0.0 is the first post-update frame. The only valid observed visual
  timestamps are 0.0, 0.5, ..., 29.5 seconds.
- onset_latency_s is the first sampled timestamp at which a clear,
  instruction-directed visual change begins.
- target_achievement_latency_s is the first sampled timestamp at which the
  core visual target is clearly achieved. It is not the time when an action
  naturally ends.
- If no clear onset occurs, set started=false and both latency fields to null.
- If onset occurs but the core target is never achieved, keep
  target_achievement_latency_s=null.

Inspect the full 30-second interval and ensure the fulfillment score, started,
both latency fields, timestamped evidence, and reason are mutually consistent
on the 2 FPS grid.

Return strict JSON with no additional text.

# Output Format

{
  "metric": "VUF",
  "started": <true | false>,
  "fulfillment_score": <1 | 2 | 3 | 4 | 5>,
  "onset_latency_s": <number | null>,
  "target_achievement_latency_s": <number | null>,
  "evidence": "<short timestamped visual evidence>",
```

```
  "reason": "<short reason>"
}
```

AUF Judge

```
# Task

Evaluate Audio Update Fulfillment (AUF) for the provided 30-second
synchronized audio-video interval after a runtime update.

Judge only the audio-applicable change explicitly requested by the current
update. Use video only when necessary to identify an explicitly requested
sound source, event condition, or audio consequence. Do not turn general
audio-video synchronization or visual fulfillment into part of the AUF score.

Modality-specific audio update:
{audio_update}

# Fulfillment Scale

1 = No Fulfillment: The requested audio change is absent, or the main result
    clearly contradicts the update.
2 = Weak Fulfillment: There is an update-related attempt or weak audio
    response, but the core audio target is not achieved.
3 = Partial Fulfillment: The core audio interaction is recognizable but
    incomplete, or important requested details are incorrect.
4 = Strong Fulfillment: The core audio target is correctly achieved, with
    only minor missing or incorrect details.
5 = Full Fulfillment: The core audio target, explicit semantic details, and
    any applicable requested causal or temporal relation are fully realized.

# Temporal Localization

- The evaluation timeline spans 30.0 seconds. Audio is continuous and
  synchronized with a visual stream sampled at 2 FPS.
- Timestamp 0.0 is the start of the post-update interval. Report audio times
  only on the shared 0.5-second grid: 0.0, 0.5, ..., 29.5 seconds.
- onset_latency_s is the first time at which a clear, instruction-directed
  audio change begins.
- target_achievement_latency_s is the first time at which the core audio
  target is clearly achieved. It is not the natural ending time of a sound.
- If no clear onset occurs, set started=false and both latency fields to null.
- If onset occurs but the core target is never achieved, keep
  target_achievement_latency_s=null.

Return strict JSON with no additional text.

# Output Format

{
  "metric": "AUF",
  "started": <true | false>,
  "fulfillment_score": <1 | 2 | 3 | 4 | 5>,
  "onset_latency_s": <number | null>,
  "target_achievement_latency_s": <number | null>,
  "evidence": "<short timestamped audio evidence>",
  "reason": "<short reason>"
}
```

F.2.4 PROMPT-BASED CONTINUITY

PVC Judge

```
# Task

Evaluate the visual transition stability of the provided 4-second video clip,
capturing the local transition associated with a runtime update.

Current runtime update:
{current_update}

Judge whether the video remains visually continuous and technically stable
across the boundary. First identify the visual changes allowed by the update,
```

```
then penalize only unintended defects that are separable from those changes.

Inspect for:
- black frames, flashes, flickering, freezing, or duplicate-frame stalls;
- abrupt visual discontinuity or generation break;
- abnormal subject deformation;
- broken or discontinuous action;
- unintended object or subject disappearance.

# Stability Scale

1 = No Stability: A severe generation break or corruption occurs at the
    boundary, making the transition unusable.
2 = Low Stability: Major boundary defects are clearly visible and strongly
    disrupt continuity.
3 = Moderate Stability: The transition remains understandable, but noticeable
    discontinuity, deformation, or local instability is present.
4 = High Stability: The transition is coherent overall, with only minor
    boundary artifacts.
5 = Full Stability: The transition is visually continuous and stable, with no
    obvious unintended boundary defect.

# Diagnostic Fields

- generation_break: true only if an unintended abrupt discontinuity,
  black/flash/freeze event, or broken action occurs at the boundary.
- deformation: true only if an unintended structural distortion of a subject
  or object occurs at the boundary.
- object_disappear: true only if an unintended subject or object disappearance
  occurs at the boundary.

The diagnostic fields support failure analysis and do not replace the score.
Return strict JSON with no additional text.

# Output Format

{
  "score": <1 | 2 | 3 | 4 | 5>,
  "generation_break": <true | false>,
  "deformation": <true | false>,
  "object_disappear": <true | false>,
  "reason": "<short reason>"
}
```

PAC Judge

```
# Task

Evaluate the audio transition stability of the provided 4-second audio clip,
capturing the local transition associated with a runtime update.

Current runtime update:
{current_update}

Judge whether the audio remains continuous and technically stable across the
boundary. First identify the audio changes allowed by the update, then penalize
only unintended defects that are separable from those changes.

Inspect for:
- unintended audio dropouts or abrupt silence;
- unintended hard cuts or broken ongoing sound;
- unintended repetition, looping, or stutter;
- spurious clicks, pops, buzzing, or clipping;
- unintended sudden volume jumps or other obvious boundary artifacts.

# Stability Scale

1 = No Stability: A severe audio break or corruption occurs at the boundary,
    making the transition unusable.
2 = Low Stability: Major boundary defects are clearly audible and strongly
    disrupt continuity.
3 = Moderate Stability: The transition remains understandable, but noticeable
    discontinuity, dropout, or artifact is present.
4 = High Stability: The transition is coherent overall, with only minor
    boundary artifacts.
5 = Full Stability: The transition is aurally continuous and stable, with no
    obvious unintended boundary defect.
```

```
# Diagnostic Fields

- audio_break: true only if an unintended dropout, abrupt silence, hard cut,
  broken ongoing sound, repetition, loop, or stutter occurs at the boundary.
- audio_artifact: true only if an unintended click, pop, buzz, clipping event,
  sudden volume jump, or comparable technical artifact occurs at the boundary.

The diagnostic fields support failure analysis and do not replace the score.
Return strict JSON with no additional text.

# Output Format

{
  "score": <1 | 2 | 3 | 4 | 5>,
  "audio_break": <true | false>,
  "audio_artifact": <true | false>,
  "reason": "<short reason>"
}
```

F.2.5 STATE RETENTION

VSR Judge

```
# Task

Evaluate Visual State Retention (VSR) across two consecutive 30-second video
intervals.

The first video is from before the current runtime update.
The second video is from after the update.

Current runtime update:
{current_update}

Changes explicitly requested by the current update are allowed and must not
be penalized. Judge only visual states that should remain unchanged.

Score these fields independently:

- subject_retention: identity, appearance, and unmodified subject states or
  ongoing activities.
- environment_retention: scene, background, and their unmodified visual
  appearance.

# Retention Scale

1 = No Retention: Applicable states are broadly lost, replaced, or contradicted.
2 = Weak Retention: Only limited prior state is retained, with major
    unauthorized changes.
3 = Partial Retention: The main state remains recognizable, but noticeable
    drift, loss, or incorrect changes occur.
4 = Strong Retention: Applicable states are mostly preserved, with only minor
    local deviations.
5 = Full Retention: All observable applicable non-target states are correctly
    preserved.

Return strict JSON with no additional text.

# Output Format

{
  "subject_retention": {
    "score": <1 | 2 | 3 | 4 | 5>,
    "reason": "<short reason>"
  },
  "environment_retention": {
    "score": <1 | 2 | 3 | 4 | 5>,
    "reason": "<short reason>"
  }
}
```

ASR Judge

```
# Task

Evaluate Audio State Retention (ASR) across two consecutive 30-second audio
intervals.

The first audio clip is from before the current runtime update.
The second audio clip is from after the update.

Current runtime update:
{current_update}

Audio changes explicitly requested by the current update are allowed and must
not be penalized. Judge whether applicable non-target audio states such as
music, ambience, device sounds, and other recurring sounds are preserved.

# Retention Scale

1 = No Retention: Applicable audio states are broadly lost, replaced, or
    contradicted.
2 = Weak Retention: Only limited prior audio state is retained, with major
    unauthorized changes.
3 = Partial Retention: The main audio state remains recognizable, but
    noticeable drift, loss, or incorrect changes occur.
4 = Strong Retention: Applicable audio states are mostly preserved, with only
    minor local deviations.
5 = Full Retention: All observable applicable non-target audio states are
    correctly preserved.

Return strict JSON with no additional text.

# Output Format

{
  "audio_retention_score": <1 | 2 | 3 | 4 | 5>,
  "reason": "<short reason>"
}
```

F.2.6 History Dependency

HDF Judge

```
# Task

Evaluate History-Dependency Following (HDF) for the current runtime update.

The first audio-video clip is from the annotated source interval.
The second audio-video clip is from the current interval.

Source prompt:
{source_prompt}

Current runtime update:
{current_update}

First determine whether the entity, state, or earlier effect required by the
current update is visibly or audibly established in the source clip.

- If it is not established, set source_state_established=false and
  dependency_following_score=null.
- If it is established, score how correctly the current interaction uses that
  exact source.
- Ignore unrelated differences between the two clips.

Judge only the source-specific dependency required by the current update; do
not score overall cross-clip similarity or unrelated state retention.

# Following Scale

1 = No Following: The specified source is not used, or the current interaction
    uses the wrong entity, state, or earlier effect.
2 = Weak Following: Only a weak or highly uncertain source-specific relation is
    present, with major errors.
3 = Partial Following: The dependency is recognizable but incomplete, or its
```

```
    key reference contains noticeable errors.
4 = Strong Following: The specified source is used correctly, with only
    minor detail errors.
5 = Full Following: The current interaction is clearly and completely grounded
    in the specified source.

Return strict JSON with no additional text.

# Output Format

{
  "source_state_established": <true | false>,
  "dependency_following_score": <1 | 2 | 3 | 4 | 5 | null>,
  "reason": "<short reason>"
}
```

G Example Benchmark Cases

The following examples illustrate the prompt structure and temporal dependencies used in the two evaluation tracks.

G.1 Progressive Example

A representative progressive case depicts a sailor crossing an open bay with a border collie aboard. Its global prompt combines continuous steering and sail trimming with wind-driven boat motion and a layered maritime soundscape. Table A10 gives the complete prompt.

Table A10: Complete 180-second prompt example for the progressive track.

Time	Prompt
0–180 s	A photorealistic compact sailing yacht crosses a breezy open bay beneath broken afternoon clouds, with taut white sails, a weathered deck, rolling blue water, and a distant low coastline; aboard, a windbreaker-clad adult sailor continuously steers at the helm, monitors the wind, and trims the working lines as the yacht heels and rises over the chop, while a distinct, alert black-and-white border collie in a red flotation vest remains settled in the cockpit and passively rides with the yacht's motion. The ratcheting winch and creaking rigging remain prominent over the simultaneous sounds of rushing wind, snapping canvas, and water striking the hull.

G.2 Interactive Example

A representative interactive case follows a lone badminton player practicing on a photorealistic indoor court. Table A11 gives the complete prompt schedule. The updates alter equipment use, court position, and training actions.

Table A11: Complete 180-second prompt schedule example for the interactive track.

Time	Prompt
0 s	In a photorealistic indoor badminton court with a net, shuttlecocks, and a courtside bench holding a spare racket, a lone badminton player in training clothes holds a racket and practices solo serves from the service court. The player calls out each serve count while brief exertion grunts and sharp shoe squeaks overlap the words.
30 s	The player places the spare racket from the courtside bench against the near net post.
60 s	The player picks up the just-placed spare racket with a short exertion grunt.
90 s	Facing the opposite court, the player raises a free hand and calls, "Ready!"
120 s	The player runs to the rear service line.
150 s	Using the spare racket picked up earlier, the player strikes an overhead clear with a crisp impact and a forceful grunt.

REFERENCES

Alibaba. Alibaba launches HappyOyster, a world model product for real-time immersive creation and interaction. Official product release, 2026. URL https://www.alizila.com/alibaba-launches-happyoyster-a-world-model-product-for-real-time-immersive-creation-and-interaction/. Accessed July 16, 2026.

Xin Cheng, Xihua Wang, Yihan Wu, Yuyue Wang, and Ruihua Song. LoVA: Long-form video-to-audio generation. In *2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5, 2025. doi: 10.1109/ICASSP49660.2025.10888085.

Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. ImageBind: One embedding space to bind them all, 2023.

Google DeepMind. Gemini 3.1 pro model card. https://deepmind.google/models/model-cards/gemini-3-1-pro/, 2026. Accessed: 2026-07-17.

Xun Huang, Zhengqi Li, Guande He, Mingyuan Zhou, and Eli Shechtman. Self forcing: Bridging the train-test gap in autoregressive video diffusion. In *Advances in Neural Information Processing Systems*, 2025a.

Ziqi Huang, Fan Zhang, Xiaojie Xu, Yinan He, Jiashuo Yu, Ziyue Dong, Qianli Ma, Nattapol Chanpaisit, Chenyang Si, Yuming Jiang, Yaohui Wang, Xinyuan Chen, Ying-Cong Chen, Limin Wang, Dahua Lin, Yu Qiao, and Ziwei Liu. VBench++: Comprehensive and versatile benchmark suite for video generative models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025b. doi: 10.1109/TPAMI.2025.3633890.

Vladimir Iashin, Weidi Xie, Esa Rahtu, and Andrew Zisserman. Synchformer: Efficient synchronization from sparse cues, 2024.

Sihui Ji, Xi Chen, Shuai Yang, Xin Tao, Pengfei Wan, and Hengshuang Zhao. MemFlow: Flowing adaptive memory for consistent and efficient long video narratives, 2025.

LAION-AI. Aesthetic predictor. https://github.com/LAION-AI/aesthetic-predictor, 2022. Accessed: 2026-07-26.

Jinzhuo Liu, Jiangning Zhang, Wencan Jiang, Yabiao Wang, Dingkang Liang, Zhucun Xue, Ran Yi, and Yong Liu. Advancing narrative long video generation via training-free identity-aware memory, 2026a.

Kunhao Liu, Wenbo Hu, Jiale Xu, Ying Shan, and Shijian Lu. Rolling forcing: Autoregressive long video diffusion in real time, 2025.

Tengfei Liu, Yang Shi, Xuanyu Zhu, Jiafu Tang, Liu Yang, Qixun Wang, Zhuoran Zhang, Yuqi Tang, et al. LongAV-Compass: Towards unified evaluation of minute-scale audio-visual generation across T2AV, I2AV, and V2AV, 2026b.

Odyssey. Introducing Odyssey-2: A general-purpose world model. Official technical release, 2025. URL https://odyssey.ml/introducing-odyssey-2.

OpenAI. Introducing GPT-5.4. https://openai.com/index/introducing-gpt-5-4/, 2026. Accessed: 2026-07-17.

PixVerse. PixVerse launches R1: A real-time world model that redefines AI video generation. Official technical release, 2026. URL https://pixverse.ai/en/blog/pixverse-launches-r1-real-time-world-model. Accessed July 16, 2026.

Sizhe Shan, Qiulin Li, Yutao Cui, Miles Yang, Yuehai Wang, Qun Yang, Jin Zhou, and Zhao Zhong. HunyuanVideo-Foley: Multimodal diffusion with representation alignment for high-fidelity foley audio generation, 2025.

Yaofeng Su, Yuming Li, Zeyue Xue, Jie Huang, Siming Fu, Haoran Li, Ying Li, Zezhong Qian, Haoyang Huang, and Nan Duan. OmniForcing: Unleashing real-time joint audio-visual generation, 2026.

Shanwen Tan, Hao Li, Jingtao Zhang, Xiaosong Jia, Xue Yang, Shaofeng Zhang, and Yanyong Zhang. SWIFT: Prompt-adaptive memory for efficient interactive long video generation, 2026.

Andros Tjandra, Yi-Chiao Wu, Baishan Guo, John Hoffman, Brian Ellis, Apoorv Vyas, Bowen Shi, Sanyuan Chen, Matt Le, Nick Zacharov, Carleigh Wood, Ann Lee, and Wei-Ning Hsu. Meta Audiobox Aesthetics: Unified automatic quality assessment for speech, music, and sound, 2025.

Yujie Wei, Yujin Han, Zhekai Chen, Yongming Li, Kaixun Jiang, Zhihang Liu, Quanhao Li, et al. MSAVBench: Towards comprehensive and reliable evaluation of multi-shot audio-video generation, 2026.

Shuai Yang, Wei Huang, Ruihang Chu, Yicheng Xiao, Yuyang Zhao, Xianbang Wang, Muyang Li, Enze Xie, Yingcong Chen, Yao Lu, Song Han, and Yukang Chen. LongLive: Real-time interactive long video generation, 2025.

Hidir Yesiltepe, Tuna Han Salih Meral, Adil Kaan Akan, Kaan Oktay, and Pinar Yanardag. Infinity-RoPE: Action-controllable infinite video generation emerges from autoregressive self-rollout. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026.

Jung Yi, Wooseok Jang, Paul Hyunbin Cho, Jisu Nam, Heeji Yoon, and Seungryong Kim. Deep forcing: Training-free long video generation with deep sink and participative compression, 2025.

Shenghai Yuan, Yuanyang Yin, Zongjian Li, Xinwei Huang, Xiao Yang, and Li Yuan. Helios: Real real-time long video generation model, 2026.

Yehang Zhang, Xinli Xu, Xiaojie Xu, Doudou Zhang, Li Liu, and Ying-Cong Chen. Orchestrating audio: Multi-agent framework for long-video audio synthesis. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, 2025. URL https://aclanthology.org/2025.emnlp-main.1133/.

Hongzhou Zhu, Min Zhao, Guande He, Hang Su, Chongxuan Li, and Jun Zhu. Causal forcing: Autoregressive diffusion distillation done right for high-quality real-time interactive video generation, 2026.