GENERATING VISUAL SOUNDS THE HARD WAY

Anonymous authors
Paper under double-blind review

ABSTRACT

We can often infer what an event should sound like simply by watching it. To generate such sounds from video, a model must identify which visual details matter acoustically and use them to guide synthesis. Yet many current video-to-audio models rely on captions for these cues and face a trade-off: removing captions degrades semantic and physical accuracy, while using them can worsen event timing. We introduce TacitV2A, which learns to guide audio generation from visual evidence without captions at inference. Captions serve only as a training signal: caption-guided self-distillation adapts its visual reasoner through a shared frozen audio generator, matching video-conditioned audio-velocity predictions to their stronger caption-conditioned counterparts. This alone lets the reasoner internalize much of the guidance that captions provide. We further refine the audio generator with reinforcement learning to better match paired real recordings. Across Foley-Bench, VGGSounder, and FlatSounds, quantitative and human evaluations show that TacitV2A is competitive with caption-conditioned models. On FlatSounds, it outperforms all external baselines in overall physical correctness and approaches its caption-conditioned teacher, matching its event coverage with lower timing error. Interactive results are available in the supplementary material.

1 INTRODUCTION

Watching a paddle strike a ball, we expect a brief impact at the moment of contact. The visible event suggests both what should be heard and when, without being described in words. These connections between sight and sound emerge early: infants match sounds to visual events (Spelke, 1976) and associate the appearance of wood and metal with their impact sounds (Ujiie et al., 2018). Video-to-audio (V2A) generation seeks to turn such visual evidence into sound (Iashin & Rahtu, 2021). This requires identifying the sounding objects (Arandjelovic & Zisserman, 2018), inferring properties that shape their sound (Bagad et al., 2026; Wang et al., 2025b), and following the events that make them audible (Owens et al., 2016).

Many recent models use captions alongside video to specify what should be heard (Cheng et al., 2025; Shan et al., 2025; Dai et al., 2026). Others obtain textual guidance through multimodal reasoning (Liu et al., 2025a; 2026a). In Figure 1, "a paddle hits a plastic ball" specifies the sound source, action, and material, while the video shows when each contact occurs. We call this the *easy way*: language supplies cues that the model would otherwise have to infer from visual evidence.

However, learning which visual details matter acoustically and how they should shape the sound is a central part of the task (Owens et al., 2016). Captions let a model partly bypass it, but at a cost. FlatSounds finds that captions generally improve semantic alignment and physical correctness but can degrade temporal synchronization (Li et al., 2026a), suggesting that models do not always align caption cues with the video's event structure. We therefore take the *hard way*: learning to infer from video what should be heard and when, without relying on captions to specify the sound.

Although several models already support caption-free inference (Wang et al., 2024b; Dai et al., 2026), learning effective visual guidance for sound generation remains challenging (Li et al., 2026a; Chen et al., 2025b). We study this problem in Cosmos3 (NVIDIA, 2026), which couples a multimodal reasoner with an audio generator. In our zero-shot comparisons, its caption-conditioned representations guide the generator more effectively than those from video alone (Tables 1 and 2). Since both settings use the same generator, this gap suggests that better visual conditioning could recover synthesis capabilities already available with captions. This motivates using the caption-

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

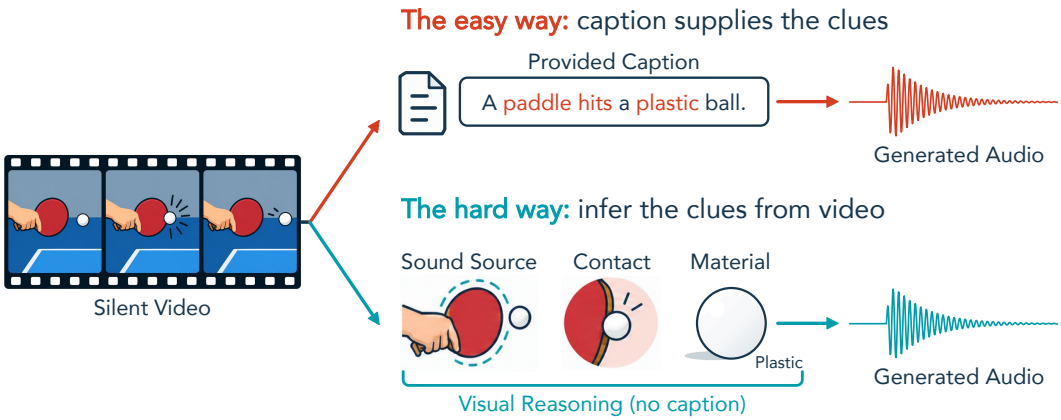


Figure 1: **Generating visual sounds the hard way.** Given a silent video (e.g., a paddle striking a ping-pong ball), the task is to generate the corresponding sound. The *easy way* (top) uses a caption to specify what should be heard. We take the *hard way* (bottom), learning to infer the sound-relevant cues (e.g., sound source, material, and contact timing) directly from the video and generate audio without captions at test time.

conditioned model as a teacher during training, so the visual reasoner learns to infer sound-relevant cues from video and encode them in a form the generator can use.

We introduce TacitV2A, a model for caption-free V2A generation trained through *caption-guided self-distillation*. Drawing on self-distillation with privileged information (Zhao et al., 2026), we train the visual reasoner so that, without captions, it steers the shared frozen audio generator toward the teacher's velocity predictions. Flow matching to real audio complements this supervision, keeping learning grounded in reference sounds. We then refine the generator itself with reinforcement learning (RL) (Zheng et al., 2026) to better match paired real recordings. At test time, the model uses only video and a shared system prompt, without an intermediate captioning stage.

We evaluate TacitV2A on FoleyBench, VGGSounder, and FlatSounds using automatic metrics and human judgments. With the generator frozen, self-distillation recovers much of the caption-conditioned teacher's generation performance, showing that better visual guidance can unlock synthesis capabilities already present in the model. After RL, TacitV2A becomes competitive with caption-conditioned baselines. In particular, on FlatSounds it approaches the teacher's physical correctness, matches its event coverage, and improves on its timing. Listeners also prefer it to every external caption-conditioned baseline in our listening study and favor its synchronization even over the teacher's. Our key contributions are:

- We present TacitV2A, which learns to infer sound-relevant cues from video and use them for audio generation, without per-sample captions or an intermediate captioning stage at inference.
- We develop caption-guided self-distillation to train the visual reasoner through a shared frozen generator's audio predictions, and further refine the generator with RL using audio-audio similarity to paired real recordings.
- We demonstrate competitive caption-free generation across three benchmarks through quantitative and human evaluations of sound quality, physical correctness, and temporal synchronization.

2 Related Work

Video-to-audio generation. Video-to-audio (V2A) generation has progressed from specific physical interactions (Owens et al., 2016) to diverse sound events using autoregressive models (Iashin & Rahtu, 2021; Viertola et al., 2025; Saito et al., 2025) as well as diffusion and flow models (Luo et al., 2023; Wang et al., 2024b;a; Zhang et al., 2026; 2025; Chen et al., 2026; Peng & Zhang, 2026; Hadjeres et al., 2026). Video-text-to-audio (VT2A) models additionally condition on captions (Cheng et al., 2025; Shan et al., 2025; Oh et al., 2026; Dai et al., 2026; Tian et al., 2026a; Tao et al., 2026), with extensions for finer control over content and timing (Chen et al., 2025c; Li et al., 2026c; Yang et al., 2026; Wang et al., 2026b; Hayakawa et al., 2026). Text can be supplied by automatic caption-

ing (Xing et al., 2024; Tran et al., 2026) or multimodal chain-of-thought reasoning (Liu et al., 2025a; 2026a). Although many models support caption-free generation, captions often improve semantic or physical accuracy (Dai et al., 2026; Li et al., 2026a). Other methods train an audio generator conditioned on representations from a frozen multimodal language model (Tian et al., 2026b). We instead adapt a visual reasoner (NVIDIA, 2026) through caption-guided self-distillation while keeping the audio generator frozen. Captions thereby provide training supervision for how visual evidence is used for sound generation, enabling inference without captions or an intermediate captioning stage.

Representation learning for generation. Visual and audio representation learning commonly uses contrastive alignment (Zhai et al., 2023; Bolya et al., 2025; Wang et al., 2026c), caption prediction (Li et al., 2022; Yu et al., 2022; Ghosh et al., 2025; Tseng et al., 2026), reconstruction (He et al., 2022; Tang et al., 2026), or latent prediction (Bardes et al., 2024). For generation, representations can serve as prediction targets (Chen et al., 2025a), guide diffusion training through feature alignment (Yu et al., 2025; Jiang et al., 2026; Wang et al., 2026a), or replace captions as conditions in unified models (Xie et al., 2026). Conversely, generative objectives can train conditional representations through image generation (Hudson et al., 2024; Liu et al., 2026c) and speech reconstruction (Wang et al., 2026d). This feedback can also come from a frozen image generator (Pan et al., 2024; Wang et al., 2025c). Self-distillation transfers predictions across views or models (Furlanello et al., 2018; Caron et al., 2021) and can internalize context (Snell et al., 2022) or privileged information (Zhao et al., 2026), with concurrent HPSD (Bu et al., 2026) using velocity matching on hybrid teacher-student trajectories. Building on these ideas, we use the model's more effective caption-conditioned representations to teach a visual reasoner how to guide sound generation. By matching the velocity predictions of a shared frozen audio generator, the student learns visual representations that recover this guidance directly from video.

Reinforcement learning for diffusion models. Reward and preference optimization refine diffusion and flow models using feedback on generated samples, through policy gradients over denoising trajectories (Black et al., 2024; Liu et al., 2025b; Xue et al., 2025), preference objectives (Wallace et al., 2024), or reward-guided prediction targets (Zhou et al., 2026). In text-to-audio generation, this feedback targets agreement with a description and perceptual quality. It can come from human preferences (Liao et al., 2024), text-audio similarity (Majumder et al., 2024; Hung et al., 2026), or automated assessments of event content, ordering, and audio quality (Wang et al., 2025d; Li et al., 2026b). Video-to-audio methods further incorporate feedback on synchronization with visible events (Liu et al., 2026a; Peng & Zhang, 2026). We use RL to bring generated sound closer to paired real recordings after learning visual conditioning through self-distillation. Specifically, we apply DiffusionNFT (Zheng et al., 2026) to the generator with an audio-audio similarity reward, keeping the learned visual reasoner fixed to retain its visual guidance.

3 TACITV2A

Given a silent video $\mathcal{V}$, we generate a soundtrack consistent with its visible events, without a per-sample caption at inference. Our model couples a visual reasoner with an audio generator. We first train the reasoner through a frozen generator using caption-guided self-distillation, then keep the learned reasoner fixed and refine the generator with RL against paired real recordings. We refer to our method as TacitV2A, since its visual representations encode sound-relevant knowledge that guides audio generation without being explicitly stated in captions. Figure 2 shows its architecture and self-distillation procedure.

3.1 VISUAL CONDITIONING FOR AUDIO GENERATION

We build on Cosmos3-Nano (NVIDIA, 2026), which couples an autoregressive reasoner $R_{\boldsymbol{\theta}}$ with a diffusion generator $G_{\boldsymbol{\varphi}}$. Here $\boldsymbol{\theta}$ denotes reasoner LoRA parameters and $\boldsymbol{\varphi}$ the pre-trained generator weights, with the base reasoner weights implicit in $R_{\boldsymbol{\theta}}$. Frozen video encoders provide two forms of conditioning: ViT features enter the reasoner, while video-VAE latents $\boldsymbol{x} = E_{\mathrm{V}}(\mathcal{V})$ condition the generator directly. We use Cosmos3 for conditional audio generation by keeping $\boldsymbol{x}$ clean and fixed throughout denoising. Only audio latents are generated and contribute to the training losses.

The reasoner processes the video features and a shared system prompt P to produce layer-wise keys and values, $\boldsymbol{h}_S(\mathcal{V}) = R_{\boldsymbol{\theta}}(E_{\mathrm{ViT}}(\mathcal{V}), P)$. Our *Audio-tuned* prompt directs attention to sound sources,

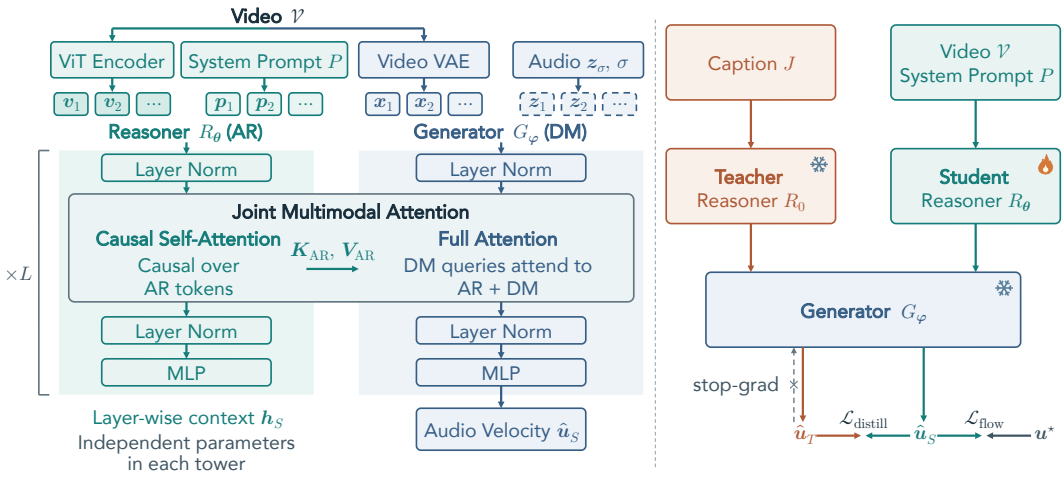


Figure 2: **TacitV2A architecture and caption-guided self-distillation.** Given a silent video and a shared system prompt, the autoregressive (AR) reasoner extracts sound-relevant information to guide a diffusion model (DM). The reasoner's keys and values condition audio generation through joint attention (left). We learn this visual conditioning using a frozen caption-conditioned teacher from the same base model (right). Both paths use the same frozen generator, clean video latents, and noisy audio inputs. We update only the student's reasoner LoRA adapters using teacher velocity matching ($\mathcal{L}_{\text{distill}}$) and flow matching to reference audio ($\mathcal{L}_{\text{flow}}$).

physical interactions, and event timing, with its instructions fixed during training and inference. We select it using both generated-audio quality and coverage of sound-relevant information in reasoner outputs (Appendix C.2).

At each layer ℓ, generator queries attend jointly to the reasoner's keys and values and those of the generator:

$$\boldsymbol{O}^{\ell}_{\text{DM}} = \text{Attn}_{\text{full}}\big(\boldsymbol{Q}^{\ell}_{\text{DM}}, [\boldsymbol{K}^{\ell}_{\text{AR}}; \boldsymbol{K}^{\ell}_{\text{DM}}], [\boldsymbol{V}^{\ell}_{\text{AR}}; \boldsymbol{V}^{\ell}_{\text{DM}}]\big)\,. \tag{1}$$

Here $\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}$ are query, key, and value tensors, $\boldsymbol{O}$ is the attention output, and AR and DM index the reasoner and generator. $[\cdot;\cdot]$ concatenates tokens, and $\text{Attn}_{\text{full}}$ denotes noncausal attention. The reasoner uses causal self-attention and does not attend to generator tokens, so its context depends only on the video and prompt. The generator accesses this context directly throughout denoising, without first decoding an intermediate caption.

3.2 Caption-Guided Self-Distillation for Visual Reasoner

In our zero-shot comparisons, we observe that the same pre-trained reasoner guides audio generation more effectively with captions than with visual input (Tables 1 and 2). A caption states sound-relevant cues explicitly, whereas visual conditioning requires the reasoner to infer them from the scene and encode them in a form the generator can use. We use the caption-conditioned model as a teacher to learn this visual conditioning while keeping the generator frozen.

For each training pair $(\mathcal{V}, A)$, a structured caption J provides the teacher context $\boldsymbol{h}_T = R_0(J)$, while the student uses the visual context $\boldsymbol{h}_S$. R_0 and $R_{\boldsymbol{\theta}}$ share the pre-trained reasoner weights, with LoRA adapters disabled for the teacher and enabled for the student. Their text and visual tokens have no direct correspondence, so we compare how their representations guide the same generator rather than matching the representations themselves.

A frozen audio-VAE encoder gives reference latents $\boldsymbol{z}_0 = E_{\text{A}}(A)$. We construct noisy audio and its target velocity by rectified flow:

$$\boldsymbol{z}_\sigma = (1-\sigma)\boldsymbol{z}_0 + \sigma\boldsymbol{\epsilon}, \qquad \boldsymbol{u}^\star = \boldsymbol{\epsilon} - \boldsymbol{z}_0, \tag{2}$$

where $\boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$ and $\sigma \in [0,1]$ is the audio noise level. Both paths receive identical clean video latents, noisy audio, and noise levels. Only the reasoner context differs in their audio-velocity predictions:

$$\hat{\boldsymbol{u}}_T = G_{\boldsymbol{\varphi}}(\boldsymbol{z}_\sigma, \sigma, \boldsymbol{x}; \boldsymbol{h}_T), \qquad \hat{\boldsymbol{u}}_S = G_{\boldsymbol{\varphi}}(\boldsymbol{z}_\sigma, \sigma, \boldsymbol{x}; \boldsymbol{h}_S). \tag{3}$$

The teacher therefore retains video conditioning through the generator, while its reasoner obtains sound-relevant guidance from the caption.

We train the visual reasoner with two complementary losses:

$$\mathcal{L} = \lambda_{\mathrm{FM}} \underbrace{\frac{1}{N_z}\,\mathbb{E}\Big[\|\hat{\boldsymbol{u}}_S - \boldsymbol{u}^\star\|_2^2\Big]}_{\mathcal{L}_{\mathrm{flow}}} + \lambda_{\mathrm{KD}} \underbrace{\frac{1}{N_z}\,\mathbb{E}\Big[\|\hat{\boldsymbol{u}}_S - \mathrm{sg}(\hat{\boldsymbol{u}}_T)\|_2^2\Big]}_{\mathcal{L}_{\mathrm{distill}}}, \tag{4}$$

where λ_{FM} and λ_{KD} weight the two losses, N_z counts audio latent elements per example, and sg stops gradients. The norms act on vectorized audio-velocity errors, and both losses operate only on audio positions.

Flow matching ($\mathcal{L}_{\mathrm{flow}}$) grounds learning in the recorded sound, limiting reliance on imperfect teacher predictions. Distillation ($\mathcal{L}_{\mathrm{distill}}$) uses the caption-conditioned teacher's audio-velocity predictions as targets, teaching the visual reasoner to guide the same generator from video.

We update only the reasoner LoRA adapters (Hu et al., 2022). Gradients from both losses pass through the frozen generator's joint attention to $\boldsymbol{h}_S$ and back into the reasoner. Appendix B.2 gives the paired conditioning dropout and adaptation details, and Appendix B.3 analyzes the objective.

3.3 Reinforcement Learning for Audio Generator

After self-distillation, we freeze the learned reasoner and refine the generator with DiffusionNFT (Zheng et al., 2026). We add generator LoRA adapters initialized to preserve the distilled model's predictions and use the same video-audio training pairs. For each video, the current model generates K candidate waveforms $\tilde{A}_i$. We compare each candidate to the paired real recording A using the frozen LAION-CLAP audio encoder f_{C} (Wu et al., 2023):

$$s_i = \frac{f_{\mathrm{C}}(\tilde{A}_i)^\top f_{\mathrm{C}}(A)}{\|f_{\mathrm{C}}(\tilde{A}_i)\|_2\,\|f_{\mathrm{C}}(A)\|_2}. \tag{5}$$

This audio-audio reward uses the recording as a reference for how the event sounds.

Within each group, normalized rewards weight DiffusionNFT's positive and negative reconstruction terms on re-noised generated audio latents. The real recording determines the reward, while the generated candidate provides the reconstruction target. We update only generator LoRA adapters, using a velocity-space penalty to anchor predictions to the frozen distilled model. This refines audio synthesis while retaining the visual conditioning learned in the first stage. Appendix B.4 specifies the reward normalization, objective, and configuration. At test time, the model generates audio from the video and shared prompt, without captions or intermediate text generation.

4 Experiments

4.1 Experimental Setup

Training data. We collect video-audio pairs from eight sources: VGGSound (Chen et al., 2020), AudioCaps (Kim et al., 2019), AudioSet (Gemmeke et al., 2017), Panda-70M (Chen et al., 2024), YouTube-8M (Abu-El-Haija et al., 2016), Greatest Hits (Owens et al., 2016), Sound of Water (Bagad et al., 2026), and EPIC-SOUNDS (Huh et al., 2025). To obtain training examples in which the sound can be inferred from the video, we apply a two-stage filtering pipeline. The first stage removes speech and unrelated background music. The second stage checks audio quality and audio-visual alignment. After filtering, we retain 89,966 clips for both self-distillation and RL and 935 for validation, all eight seconds long. Neither split overlaps with the evaluation benchmarks. Per-dataset counts and filtering details are provided in Appendix C.1.

Model architecture. We use the pre-trained Cosmos3-Nano (NVIDIA, 2026). Its autoregressive reasoner and diffusion generator each have 36 transformer layers with a hidden dimension of 4,096. We retain its pre-trained Qwen3-VL vision encoder (Bai et al., 2025), the video VAE from Wan2.2 (Team Wan, 2025), and the ETTA-based audio VAE (Lee et al., 2025) for 48-kHz stereo audio. Section 3.1 describes how we use this model for video-conditioned audio generation. Additional architectural details are provided in Appendix B.1.

Table 1: Quantitative comparison on FoleyBench. "Cap." denotes caption conditioning. Bold marks the best result and underline the best caption-free result.

Method	Cap.	FD_{VGG} ↓	FD_{PANN} ↓	FD_{PaSST} ↓	KL ↓	IB ↑	DeSync ↓	$CLAP_{LAION}$ ↑	$CLAP_{MS}$ ↑
FoleyCrafter	✓	4.148	15.61	203.6	2.654	0.261	1.156	0.267	0.385
MMAudio	✓	1.763	**7.83**	85.1	2.438	0.303	**0.424**	**0.306**	0.479
Hunyuan-Foley	✓	2.063	13.10	129.8	2.480	**0.332**	0.452	0.268	0.411
ThinkSound	✓	4.552	12.92	153.8	2.799	0.211	0.603	0.176	0.336
PrismAudio	✓	3.770	10.39	152.2	2.364	0.239	0.494	0.257	0.414
Omni2Sound	✓	1.801	15.36	183.9	2.448	0.307	0.517	0.264	0.394
AudioX	✓	2.716	17.62	158.1	2.741	0.261	1.094	0.281	0.418
Audio-Omni	✓	3.488	14.03	145.1	2.660	0.218	1.050	0.254	0.414
Cosmos3	✓	2.439	11.37	128.2	2.334	0.251	0.829	0.293	0.471
FoleyCrafter	–	6.194	18.55	255.3	3.035	0.264	1.153	0.170	0.289
MMAudio	–	1.876	8.86	90.4	2.867	0.284	<u>0.439</u>	0.230	0.412
Hunyuan-Foley	–	3.018	18.76	193.2	3.028	0.321	0.506	0.127	0.289
ThinkSound	–	7.266	18.33	196.4	3.278	0.187	0.631	0.120	0.281
PrismAudio	–	8.656	17.97	216.9	2.757	0.192	0.485	0.171	0.341
Omni2Sound	–	2.470	16.89	210.5	2.848	0.289	0.588	0.192	0.324
AudioX	–	3.909	19.76	208.7	3.440	0.227	1.076	0.158	0.304
Audio-Omni	–	4.039	16.36	163.7	3.037	0.208	1.047	0.191	0.351
Cosmos3	–	5.016	17.56	199.2	2.606	0.216	0.793	0.205	0.382
TacitV2A (distilled)	–	2.536	12.45	148.8	2.361	0.266	0.798	0.284	0.466
+RL	–	**1.704**	<u>8.21</u>	**82.6**	**2.298**	<u>0.322</u>	0.549	<u>0.299</u>	**0.494**

Training configuration. Both training stages use LoRA with rank 16 and scaling parameter 32. We train the reasoner for 3 epochs with self-distillation and use the final checkpoint. The global batch size is 128, and AdamW (Loshchilov & Hutter, 2019) uses a constant learning rate of 5×10^{-5}. Teacher and student share conditioning dropout with probability 0.1, and the loss weights are $\lambda_{\mathrm{FM}} = 0.1$ and $\lambda_{\mathrm{KD}} = 1$. We then train the generator adapters with DiffusionNFT for 1,000 updates at a peak learning rate of 10^{-4}. Each RL update uses eight videos with eight audio candidates per video. For both benchmark evaluation and RL rollouts, we use 35 UniPC (Zhao et al., 2023) steps, classifier-free guidance (Ho & Salimans, 2021) scale 6, and flow shift 10. Adapter placement and remaining training settings are provided in Appendices B.2 and B.4.

Evaluation metrics. We use AV-Benchmark (Cheng et al., 2025) on FoleyBench (Dixit et al., 2026) and VGGSounder (Zverev et al., 2025): FD scores compare generated and reference audio distributions, while KL compares audio-event predictions. ImageBind (IB) (Girdhar et al., 2023) measures audio-visual semantic agreement, and CLAP measures audio-text agreement, distinct from our audio-audio RL reward. We assess synchronization using DeSync, which estimates audio-video offsets with Synchformer (Iashin et al., 2024). We also use FlatSounds (Li et al., 2026a) to evaluate temporal alignment and physical correctness. HitCov measures the fraction of annotated events matched, Timing Error measures mean absolute timing error over matched events, and Perfect Align measures the fraction of generations covering every annotated event. These event-level metrics complement DeSync. The physical metrics assess whether temporal-envelope, spectral, pitch, and room-acoustic properties vary as expected with changes in physical factors. We evaluate with three generation seeds on FoleyBench and VGGSounder and ten on FlatSounds (Appendix C).

Baselines. We compare TacitV2A with MMAudio (Cheng et al., 2025), ThinkSound (Liu et al., 2025a), PrismAudio (Liu et al., 2026a), Omni2Sound (Dai et al., 2026), AudioX (Tian et al., 2026a), Audio-Omni (Tian et al., 2026b), Hunyuan-Foley (Shan et al., 2025) and FoleyCrafter (Zhang et al., 2026) under caption-conditioned and caption-free settings. For ThinkSound and PrismAudio, we use caption+CoT in the captioned setting and generic CoT in the caption-free setting. We also include two Cosmos3 references (the captioned and caption-free Cosmos3 rows). The caption-conditioned teacher receives LLM-expanded benchmark captions and thus serves as an oracle reference for Cosmos3-based caption-then-generate pipelines (Appendix D.3), which TacitV2A aims to match without captioning. Zero-shot Cosmos3 with ViT features and the Audio-tuned prompt marks the starting point. We report TacitV2A after self-distillation ("distilled" in the tables) and after RL.

4.2 Comparison to Baselines

Distributional quality. As shown in Tables 1 and 2, removing captions from Cosmos3 worsens FD and KL on FoleyBench and VGGSounder. Self-distillation brings both close to those of the

Table 2: Quantitative comparison on VGGSounder. "Cap." denotes caption conditioning. Bold marks the best result and underline the best caption-free result.

Method	Cap.	FD_{VGG} ↓	FD_{PANN} ↓	FD_{PaSST} ↓	KL ↓	IB ↑	DeSync ↓	$CLAP_{LAION}$ ↑	$CLAP_{MS}$ ↑
FoleyCrafter	✓	3.235	20.81	196.3	1.976	0.283	1.207	0.233	**0.558**
MMAudio	✓	1.258	4.79	83.2	**1.443**	0.360	0.515	0.268	0.352
Hunyuan-Foley	✓	2.381	12.80	142.2	1.791	0.344	0.619	0.252	0.311
ThinkSound	✓	2.581	11.69	116.5	1.738	0.271	0.620	0.214	0.286
PrismAudio	✓	2.572	11.23	118.4	1.614	0.284	0.601	0.254	0.341
AudioX	✓	1.894	10.34	138.8	1.765	0.291	1.212	**0.293**	0.356
Audio-Omni	✓	3.038	13.18	157.4	1.808	0.261	1.170	0.255	0.328
Omni2Sound	✓	1.191	5.75	92.9	1.584	0.353	0.562	0.260	0.532
Cosmos3	✓	3.148	22.69	167.9	2.144	0.266	0.821	0.208	0.274
FoleyCrafter	–	4.401	25.10	218.5	2.268	0.299	1.195	0.166	0.498
MMAudio	–	**0.934**	5.10	83.9	1.772	0.354	**0.510**	0.231	0.318
Hunyuan-Foley	–	2.737	21.09	191.1	2.492	0.335	0.619	0.139	0.213
ThinkSound	–	4.531	18.02	142.1	2.337	0.236	0.638	0.153	0.231
PrismAudio	–	7.427	30.28	198.9	2.811	0.207	0.571	0.117	0.224
AudioX	–	2.368	18.33	195.6	2.906	0.261	1.203	0.166	0.240
Audio-Omni	–	3.130	14.24	162.8	2.074	0.261	1.161	0.213	0.290
Omni2Sound	–	1.157	5.66	90.0	1.752	0.361	0.563	0.233	<u>0.501</u>
Cosmos3	–	7.441	41.23	298.9	3.112	0.212	0.817	0.141	0.230
TacitV2A (distilled)	–	3.132	22.84	168.8	2.161	0.269	0.816	0.205	0.272
+RL	–	2.145	**4.72**	**80.2**	<u>1.449</u>	**0.364**	0.698	<u>0.266</u>	0.379

caption-conditioned teacher, even though the generator remains frozen. Because the teacher and student share this generator, the improvement comes from how the visual reasoner guides it. The generator can already synthesize these sounds, but the zero-shot reasoner cannot steer it toward them from video. By learning to infer and encode the cues that captions make explicit, the reasoner unlocks this existing capability. Subsequent RL then refines synthesis itself, making the final model competitive with caption-conditioned models across several audio feature spaces.

Semantic alignment. Closer agreement with the reference audio distribution does not establish that the generated sound fits each input video. We therefore examine audio-text agreement through CLAP alongside audio-visual agreement through IB. We find that self-distillation improves both over caption-free Cosmos3, bringing CLAP close to the caption-conditioned teacher's and even exceeding its IB on both benchmarks (Tables 1 and 2). These results suggest that the student can infer from video what captions tell the teacher. RL further improves both forms of agreement, with the final model achieving the highest MS-CLAP on FoleyBench and IB on VGGSounder among all evaluated models.

Physical correctness. We find that our final model improves on caption-free Cosmos3 and approaches its caption-conditioned counterpart in average physical correctness (Table 4). The improvement begins with self-distillation. Although it trains only the reasoner, the distilled model already surpasses all external baselines, including caption-conditioned ones. Since FlatSounds tests whether generated sounds reflect controlled physical changes, we interpret these gains as evidence that the reasoner learns to infer and use physical cues from video to guide sound generation, recovering part of the guidance supplied by captions. RL further improves physical correctness, yet our CLAP scores remain below several captioned systems (Table 3), illustrating that semantic agreement alone does not capture physical fidelity. Recovery also varies across properties. TacitV2A exceeds the teacher on pitch,

Table 3: Overall performance on FlatSounds. All baselines use captions, whereas ours are caption-free. Confidence is the overall physical confidence, HitCov and PerfAlign denote hit coverage and perfect-alignment rate, and Timing is the mean matched-hit error in milliseconds.

Method	Confidence↑	HitCov↑	Timing↓	PerfAlign↑	$CLAP_{LAION}$↑
FoleyCrafter	0.197	0.746	49.2	0.510	0.613
MMAudio	0.223	0.736	58.3	0.490	0.640
Hunyuan-Foley	0.312	0.902	44.7	0.693	**0.649**
ThinkSound	0.229	0.665	52.2	0.447	0.580
PrismAudio	0.299	0.927	47.8	0.731	0.615
AudioX	0.160	0.547	48.4	0.325	0.647
Audio-Omni	0.116	0.409	46.8	0.232	0.615
Omni2Sound	0.288	0.852	59.8	0.662	0.624
Cosmos3	**0.349**	**0.978**	36.5	0.793	0.608
TacitV2A (distilled)	0.322	0.972	34.3	0.790	0.607
+RL	0.347	**0.978**	**33.1**	**0.795**	0.628

Table 4: Per-property physical confidence on FlatSounds. "Cap." denotes caption conditioning, "Temporal" denotes temporal modulation, and "Avg." is the overall physical confidence. Bold marks the best result and underline the best caption-free result.

Method	Cap.	Attack ↑	Decay ↑	F0 ↑	Centroid ↑	Flux ↑	Rolloff ↑	Temporal ↑	RT60 ↑	DRR ↑	Avg. ↑
FoleyCrafter	✓	0.152	0.152	0.173	0.229	0.270	0.238	0.329	0.137	0.154	0.197
MMAudio	✓	0.214	0.167	0.229	0.256	0.255	0.225	0.262	0.240	0.184	0.223
Hunyuan-Foley	✓	0.265	0.312	0.324	0.335	0.407	0.309	0.284	0.244	0.261	0.312
ThinkSound	✓	0.247	0.192	0.200	0.266	0.176	0.302	0.298	0.234	0.150	0.229
PrismAudio	✓	**0.281**	0.305	0.281	0.318	0.319	0.310	0.281	0.294	0.227	0.299
AudioX	✓	0.145	0.095	0.170	0.180	0.232	0.177	0.202	0.129	0.158	0.160
Audio-Omni	✓	0.060	0.062	0.148	0.127	0.146	0.105	0.203	0.139	0.089	0.116
Omni2Sound	✓	0.255	0.249	0.333	0.310	0.290	0.261	0.263	0.336	0.209	0.288
Cosmos3	✓	0.229	**0.382**	0.324	**0.481**	**0.408**	**0.473**	0.301	0.204	0.239	**0.349**
FoleyCrafter	–	0.172	0.180	0.187	0.243	0.297	0.216	0.300	0.173	0.184	0.208
MMAudio	–	0.196	0.174	0.225	0.268	0.236	0.250	0.202	0.222	0.224	0.220
Hunyuan-Foley	–	0.231	0.308	0.253	0.299	<u>0.382</u>	<u>0.306</u>	0.316	0.267	0.294	0.288
ThinkSound	–	0.205	0.202	0.255	0.261	0.245	0.252	0.219	0.175	0.148	0.223
PrismAudio	–	<u>0.262</u>	0.258	0.284	0.304	0.304	0.283	0.242	0.252	0.245	0.282
AudioX	–	0.076	0.087	0.177	0.131	0.173	0.094	0.111	0.139	0.099	0.124
Audio-Omni	–	0.083	0.089	0.157	0.128	0.215	0.107	0.195	0.118	0.078	0.126
Omni2Sound	–	0.259	0.245	0.311	0.291	0.248	0.236	0.246	**0.339**	0.189	0.269
Cosmos3	–	0.231	0.296	0.244	0.329	0.320	0.283	0.300	0.275	0.351	0.287
TacitV2A (distilled)	–	0.203	0.339	0.282	0.387	0.356	0.297	0.387	0.282	0.381	0.322
+RL	–	0.229	<u>0.352</u>	**0.353**	<u>0.412</u>	0.362	0.293	**0.411**	0.295	**0.398**	<u>0.347</u>

temporal modulation, RT60, and DRR but trails it on decay and the spectral properties, so approaching the teacher on average still leaves some acoustic detail unresolved.

Temporal synchronization. For repeated impacts, a synchronized soundtrack must include a sound at each contact. DeSync estimates audio-video offsets, but a small offset can coexist with missing or reordered events (Sharma & Tapadiya, 2026). Our diagnostic comparison also reveals disagreements between DeSync rankings and human judgments (Appendix C.3). Many baselines (e.g., MMAudio) condition generation on Synchformer features, which also underlie DeSync, so the metric may favor them. We therefore complement DeSync with event recovery and timing measures, assessing both whether sounds are present and when they occur. On FoleyBench and VGGSounder, DeSync changes little after self-distillation and improves with RL, although several baselines retain lower scores (Tables 1 and 2). On FlatSounds, the distilled model already has lower timing error than the caption-conditioned teacher while nearly matching its event coverage. After RL, our model matches the teacher's event coverage and Perfect Align rate with the lowest timing error (Table 3). These results indicate that TacitV2A follows visible events as completely as the teacher and times them more precisely, without captions at inference. We provide further caption-free comparisons in Appendix D.2 and qualitative results on the supplementary project webpage (Appendix A).

4.3 Ablation Study and Analysis

Impact of self-distillation. To test the value of self-distillation, we vary the loss weights while holding the generator and all other training settings fixed (Table 5). We find that flow matching alone $(1:0)$ is consistently weaker, whereas adding the distillation term improves every reported metric. The teacher therefore provides useful guidance beyond what the student learns from paired audio alone. By contrast, training only the generator with LoRA fine-tuning or our RL recipe brings little improvement (Appendix D.1). These results support our central intuition: the teacher's text-conditioned representations already guide the generator effectively, so its audio predictions give the visual reasoner a reference for guiding the same generator from video. At the same time, distillation alone $(0:1)$ is weaker than the combined objective, indicating that it benefits from continued supervision by real recordings. The two objectives are therefore complementary.

Table 5: Ablation of the self-distillation objective on FoleyBench using representative metrics. All runs use the same caption-free student recipe.

$\lambda_{FM}:\lambda_{KD}$	FD_{PANN} ↓	IB ↑	DeSync ↓	$CLAP_{MS}$ ↑
1 : 0	16.92	0.223	0.809	0.392
1 : 1	14.68	0.249	0.802	0.431
0.1 : 1	**12.45**	**0.266**	**0.798**	**0.466**
0 : 1	16.35	0.230	0.805	0.402

Flow matching grounds sound prediction in the recorded audio, while distillation teaches the visual reasoner to recover the guidance that the teacher obtains from captions. We therefore retain flow matching alongside distillation and use the best-performing combined objective in the remaining experiments. The reasoner also describes more sound-relevant cues after self-distillation, despite receiving no text-generation supervision (Appendix D.4).

Effect of RL. Motivated by Goodhart's law (Strathern, 1997), we use audio-audio CLAP similarity to paired recordings as the sole reward, avoiding direct optimization of text-audio evaluation scores. RL improves all reported metrics on FoleyBench and VGGSounder, spanning distributional quality, semantic alignment, and temporal synchronization, including metrics based on encoders not used for the reward (Tables 1 and 2). On FlatSounds, most physical properties improve, but decay, spectral flux, and RT60 gain little, and rolloff does not improve (Table 4). Qualitatively, RL changes sound texture (McDermott & Simoncelli, 2011) more than event content, making the generated audio sound closer to real recordings. This is consistent with the metrics: RL substantially lowers FD, whereas event coverage and timing on FlatSounds change little (Table 3). RL thus mainly refines how an event sounds in its environment, including reverberation and ambient sound (Li et al., 2024). The aim is fidelity to the recorded scene, rather than uniformly clean, studio-produced audio.

4.4 Human Evaluation

We conduct a pairwise listening study on 50 clips from FoleyBench, VGGSounder, and FlatSounds. Listeners judge how well each soundtrack fits the video overall and how well its sounds line up with visible actions. TacitV2A is preferred to every external baseline in Table 6 on both questions. Listeners also prefer its synchronization to the caption-conditioned teacher's, although they prefer the teacher overall. Without captions, our model thus follows visible events more closely than its teacher but still trails it in how well the soundtrack fits the scene as a whole. This synchronization preference agrees with the lower timing error on FlatSounds but not with DeSync, which favors MMAudio, supporting our use of event-level timing alongside DeSync. Appendix C.4 details the evaluation protocol.

Table 6: Human evaluation with Elo ratings. All baselines use captions, whereas ours is caption-free.

Method	Overall ↑	Sync ↑
Ground Truth	1198	1200
Cosmos3	**1136**	1068
Omni2Sound	980	782
Hunyuan-Foley	919	976
MMAudio	858	903
PrismAudio	776	695
Audio-Omni	673	813
FoleyCrafter	586	594
TacitV2A	1033	**1123**

5 Conclusion

We presented TacitV2A, which learns to infer sound-relevant cues from video and use them to guide audio generation without captions at test time. Caption-guided self-distillation trains only the visual reasoner through a shared frozen generator, while flow matching keeps learning grounded in real recordings. The resulting gains show that better visual guidance can unlock synthesis capabilities the generator already has. RL then refines the generator itself to better match paired real recordings. In quantitative and human evaluations, the final model is competitive with caption-conditioned systems in sound quality and synchronization, and its physical correctness on FlatSounds approaches that of its teacher. These results suggest that a generator guided by descriptions can teach a model to infer the same cues from what it sees.

Limitations and broader impacts. Our study focuses on non-speech sounds and does not address speech generation for talking-head videos. We validate self-distillation with a single pre-trained model, so its generality across architectures remains untested. Within this scope, recovery of physical properties is uneven, with decay and spectral properties still trailing the caption-conditioned teacher. Listeners also prefer the teacher overall while favoring TacitV2A for synchronization, suggesting that accurate timing alone does not make a soundtrack fit the scene as a whole. Some acoustic ambiguity is inherent to the input. Internal object structure and recording conditions may not be visible, so a plausible generated sound need not reproduce the original recording. Such audio can support audio-visual creation, but because it can sound convincing without matching the original, it may also change how viewers interpret footage of real events. We therefore encourage clear labeling of synthesized audio so that it is not mistaken for a video's original sound.

AI USE STATEMENT

We used generative AI tools for proofreading and polishing the manuscript. Our research pipeline also uses model-generated captions as the source of teacher inputs and for data screening, and LLMs for caption expansion, refinement of the shared system prompt, and evaluation of sound-relevant content in decoded reasoner outputs (Appendices C.1, B.1, C.2, and D.4). We have reviewed and revised the AI-assisted manuscript text and take full responsibility for the scientific claims, reported results, and all text and artifacts in this submission.

ETHICS STATEMENT

We use existing audio-visual datasets and filter training clips for visible sound-producing events, aiming to exclude speech and unrelated background music (Appendix C.1). These filters do not anonymize the footage or eliminate biases inherited from the data and pre-trained models. Our human evaluation asks participants to compare soundtrack fit and synchronization, and we report aggregate preferences and uncertainty (Appendix C.4). As discussed in Section 5, synthesized sound can change how viewers interpret footage of real events. Our supplementary project webpage distinguishes generated soundtracks from original recordings. We encourage users to label synthesized audio clearly and to respect the privacy of recorded individuals and the usage conditions of the datasets and models.

REPRODUCIBILITY STATEMENT

We describe the architecture and training objectives in Section 3, and report training and inference settings, baseline conditioning, and evaluation metrics in Section 4.1, with metric definitions, baseline details, and FlatSounds scoring in Appendix C. Appendix B details model conditioning, adapter configurations, and optimization procedures for both training stages, including reward computation. We document the source datasets, caption generation, split composition, and filtering criteria in Appendix C.1. Prompt refinement and evaluation of the visual reasoner are described in Appendices C.2 and D.4. Appendix C.3 explains event-timing measures and synchronization diagnostics, and Appendix C.4 specifies the human evaluation protocol, rating procedure, and uncertainty estimation. Qualitative results are provided on the supplementary project webpage (Appendix A). We will make the source code publicly available upon acceptance.

REFERENCES

Sami Abu-El-Haija, Nisarg Kothari, Joonseok Lee, Paul Natsev, George Toderici, Balakrishnan Varadarajan, and Sudheendra Vijayanarasimhan. YouTube-8M: A Large-Scale Video Classification Benchmark. *arXiv preprint arXiv:1609.08675*, 2016.

Relja Arandjelovic and Andrew Zisserman. Objects That Sound. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 435–451, 2018.

Piyush Bagad, Makarand Tapaswi, Cees G. M. Snoek, and Andrew Zisserman. The Sound of Water: Inferring Physical Properties from Pouring Liquids. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 48(9):11111–11123, September 2026.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-VL Technical Report. *arXiv preprint arXiv:2511.21631*, 2025.

Adrien Bardes, Quentin Garrido, Jean Ponce, Xinlei Chen, Michael Rabbat, Yann LeCun, Mido Assran, and Nicolas Ballas. Revisiting Feature Prediction for Learning Visual Representations from Video. *Transactions on Machine Learning Research*, 2024.

Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training Diffusion Models with Reinforcement Learning. In *International Conference on Learning Representations (ICLR)*, pp. 4965–4987, 2024.

Daniel Bolya, Po-Yao Huang, Peize Sun, Jang Hyun Cho, Andrea Madotto, Chen Wei, Tengyu Ma, Jiale Zhi, Jathushan Rajasegaran, Hanoona Bangalath, Junke Wang, Marco Monteiro, Hu Xu, Shiyu Dong, Nikhila Ravi, Shang-Wen Li, Piotr Dollár, and Christoph Feichtenhofer. Perception Encoder: The Best Visual Embeddings Are Not at the Output of the Network. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 60884–60937, 2025.

Ralph Allan Bradley and Milton E. Terry. Rank Analysis of Incomplete Block Designs: I. The Method of Paired Comparisons. *Biometrika*, 39(3–4):324–345, 1952.

Jiazi Bu, Pengyang Ling, Yujie Zhou, Yibin Wang, Yuhang Zang, Xuanlang Dai, Shengyuan Ding, Tianyi Wei, Xiaohang Zhan, Jiaqi Wang, Tong Wu, Dahua Lin, and Xingang Pan. HPSD: Hybrid-Policy Self-Distillation for Text-Image-to-Video Diffusion Models. *arXiv preprint arXiv:2608.13205*, 2026.

Yuqin Cao, Xiongkuo Min, Yixuan Gao, Wei Sun, and Guangtao Zhai. AGAV-Rater: Adapting Large Multimodal Model for AI-Generated Audio-Visual Quality Assessment. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, pp. 6615–6633, 2025.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging Properties in Self-Supervised Vision Transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9650–9660, 2021.

Honglie Chen, Weidi Xie, Andrea Vedaldi, and Andrew Zisserman. VGGSound: A Large-Scale Audio-Visual Dataset. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 721–725, 2020.

Jiuhai Chen, Zhiyang Xu, Xichen Pan, Yushi Hu, Can Qin, Tom Goldstein, Lifu Huang, Tianyi Zhou, Saining Xie, Silvio Savarese, Le Xue, Caiming Xiong, and Ran Xu. BLIP3-o: A Family of Fully Open Unified Multimodal Models-Architecture, Training and Dataset. Computer Vision in the Wild Workshop at the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2025a.

Ke Chen, Xingjian Du, Bilei Zhu, Zejun Ma, Taylor Berg-Kirkpatrick, and Shlomo Dubnov. HTS-AT: A Hierarchical Token-Semantic Audio Transformer for Sound Classification and Detection. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 646–650, 2022.

Liyang Chen, Hongkai Chen, Yujun Cai, Sifan Li, Qingwen Ye, and Yiwei Wang. Detecting and Mitigating Insertion Hallucination in Video-to-Audio Generation. *arXiv preprint arXiv:2510.08078*, 2025b.

Tsai-Shien Chen, Aliaksandr Siarohin, Willi Menapace, Ekaterina Deyneka, Hsiang-wei Chao, Byung Eun Jeon, Yuwei Fang, Hsin-Ying Lee, Jian Ren, Ming-Hsuan Yang, and Sergey Tulyakov. Panda-70M: Captioning 70M Videos with Multiple Cross-Modality Teachers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13320–13331, 2024.

Zehua Chen, Junyou Wang, Yuxuan Jiang, Zhenying Fang, Yusheng Dai, Jianfei Chen, Ziwei Liu, and Jun Zhu. Visual Representation Matters: Exploiting Temporal Differences in Video-to-Audio Generation. *arXiv preprint arXiv:2608.04902*, 2026.

Ziyang Chen, Prem Seetharaman, Bryan Russell, Oriol Nieto, David Bourgin, Andrew Owens, and Justin Salamon. Video-Guided Foley Sound Generation with Multimodal Controls. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18770–18781, 2025c.

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. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 28901–28911, 2025.

Yusheng Dai, Zehua Chen, Yuxuan Jiang, Qiuhong Ke, Jianfei Cai, and Jun Zhu. Omni2Sound: Towards Unified Video-Text-to-Audio Generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1661–1671, 2026.

Satvik Dixit, Koichi Saito, Zhi Zhong, Yuki Mitsufuji, and Chris Donahue. FoleyBench: A Benchmark for Video-to-Audio Models. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 14512–14516, 2026.

Tommaso Furlanello, Zachary Lipton, Michael Tschannen, Laurent Itti, and Anima Anandkumar. Born Again Neural Networks. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, pp. 1607–1616, 2018.

Jort F. Gemmeke, Daniel P. W. Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R. Channing Moore, Manoj Plakal, and Marvin Ritter. Audio Set: An Ontology and Human-Labeled Dataset for Audio Events. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 776–780, 2017.

Sreyan Ghosh, Arushi Goel, Jaehyeon Kim, Sonal Kumar, Zhifeng Kong, Sang-gil Lee, Chao-Han Huck Yang, Ramani Duraiswami, Dinesh Manocha, Rafael Valle, and Bryan Catanzaro. Audio Flamingo 3: Advancing Audio Intelligence with Fully Open Large Audio Language Models. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 41819–41886, 2025.

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. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15180–15190, 2023.

Lucas Goncalves, Prashant Mathur, Chandrashekhar Lavania, Metehan Cekic, Marcello Federico, and Kyu J. Han. Perceptual Evaluation of Audio-Visual Synchrony Grounded in Viewers' Opinion Scores. In *European Conference on Computer Vision (ECCV)*, pp. 288–305, 2024.

Kaisi Guan, Bingzi Zhang, Xihua Wang, Ying Ba, Xin Cheng, Yijing Chen, and Ruihua Song. FATE: Frame-Level Audio-Visual Temporal Embedding. *arXiv preprint arXiv:2608.01310*, 2026.

Gaëtan Hadjeres, Marc Ferras, Khaled Koutini, Benno Weck, Alexandre Bittar, Thomas Hummel, Zineb Lahrichi, Hakim Missoum, Joan Serrà, and Yuki Mitsufuji. Woosh: A Sound Effects Foundation Model. *arXiv preprint arXiv:2604.01929*, 2026.

Akio Hayakawa, Masato Ishii, Takashi Shibuya, and Yuki Mitsufuji. Step-by-Step Video-to-Audio Synthesis via Negative Audio Guidance. In *European Conference on Computer Vision (ECCV)*, pp. 681–698, 2026.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked Autoencoders Are Scalable Vision Learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16000–16009, 2022.

Jonathan Ho and Tim Salimans. Classifier-Free Diffusion Guidance. In *Neural Information Processing Systems (NeurIPS) Workshop on Deep Generative Models and Downstream Applications*, 2021.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-Rank Adaptation of Large Language Models. In *International Conference on Learning Representations (ICLR)*, 2022.

Yinming Huang, Shuyuan Tu, Xi Yan, Zihan Yang, Jianhua Han, Hang Xu, Kaihang Pan, Yu-Gang Jiang, and Zuxuan Wu. VA-Judger: Reward Modeling from Human Preference Feedback for Joint Video-Audio Generation. *arXiv preprint arXiv:2608.18607*, 2026.

Drew A. Hudson, Daniel Zoran, Mateusz Malinowski, Andrew K. Lampinen, Andrew Jaegle, James L. McClelland, Loic Matthey, Felix Hill, and Alexander Lerchner. SODA: Bottleneck Diffusion Models for Representation Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 23115–23127, 2024.

Jaesung Huh, Jacob Chalk, Evangelos Kazakos, Dima Damen, and Andrew Zisserman. EPIC-SOUNDS: A Large-Scale Dataset of Actions That Sound. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(11):9953–9965, 2025.

Chia-Yu Hung, Navonil Majumder, Zhifeng Kong, Ambuj Mehrish, Amir Zadeh, Chuan Li, Rafael Valle, Bryan Catanzaro, and Soujanya Poria. TangoFlux: Super Fast and Faithful Text to Audio Generation with Flow Matching and CLAP-Ranked Preference Optimization. In *International Conference on Learning Representations (ICLR)*, pp. 150793–150816, 2026.

Vladimir Iashin and Esa Rahtu. Taming Visually Guided Sound Generation. In *British Machine Vision Conference (BMVC)*, 2021.

Vladimir Iashin, Weidi Xie, Esa Rahtu, and Andrew Zisserman. Synchformer: Efficient Synchronization from Sparse Cues. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5325–5329, 2024.

Dengyang Jiang, Mengmeng Wang, Liuzhuozheng Li, Lei Zhang, Haoyu Wang, Wei Wei, Guang Dai, Yanning Zhang, and Jingdong Wang. Representation Alignment for Diffusion Transformers without External Components. In *International Conference on Learning Representations (ICLR)*, pp. 18884–18915, 2026.

Chris Dongjoo Kim, Byeongchang Kim, Hyunmin Lee, and Gunhee Kim. AudioCaps: Generating Captions for Audios in the Wild. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pp. 119–132, 2019.

Sang-Gil Lee, Zhifeng Kong, Arushi Goel, Sungwon Kim, Rafael Valle, and Bryan Catanzaro. ETTA: Elucidating the Design Space of Text-to-Audio Models. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, pp. 33157–33180, 2025.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. BLIP: Bootstrapping Language-Image Pre-Training for Unified Vision-Language Understanding and Generation. In *Proceedings of the 39th International Conference on Machine Learning (ICML)*, pp. 12888–12900, 2022.

Tingle Li, Renhao Wang, Po-Yao Huang, Andrew Owens, and Gopala Anumanchipalli. Self-Supervised Audio-Visual Soundscape Stylization. In *European Conference on Computer Vision (ECCV)*, pp. 20–40, 2024.

Tingle Li, Siddharth Gururani, Kevin J. Shih, Gantavya Bhatt, Sang-gil Lee, Zhifeng Kong, Arushi Goel, Gopala Anumanchipalli, and Ming-Yu Liu. Benchmarking Single-Factor Physical Video-to-Audio Generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1939–1949, 2026a.

Xiquan Li, Junxi Liu, Wenxi Chen, Haina Zhu, Ziyang Ma, and Xie Chen. Resonate: Reinforcing Text-to-Audio Generation via Online Feedback from Large Audio Language Models. In *Proceedings of Interspeech*, pp. 5000–5005, 2026b.

You Li, Dewei Zhou, Fan Ma, Fu Li, Dongliang He, and Yi Yang. FoleyDirector: Fine-Grained Temporal Steering for Video-to-Audio Generation via Structured Scripts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 29254–29264, 2026c.

Huan Liao, Haonan Han, Kai Yang, Tianjiao Du, Rui Yang, Qinmei Xu, Zunnan Xu, Jingquan Liu, Jiasheng Lu, and Xiu Li. BATON: Aligning Text-to-Audio Model Using Human Preference Feedback. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 4542–4550, 2024.

Huadai Liu, Kaicheng Luo, Jialei Wang, Wen Wang, Qian Chen, Zhou Zhao, and Wei Xue. ThinkSound: Chain-of-Thought Reasoning in Multimodal LLMs for Audio Generation and Editing. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 78526–78552, 2025a.

Huadai Liu, Kaicheng Luo, Wen Wang, Qian Chen, Peiwen Sun, Rongjie Huang, Xiangang Li, Jieping Ye, and Wei Xue. PrismAudio: Decomposed Chain-of-Thought and Multi-Dimensional Rewards for Video-to-Audio Generation. In *International Conference on Learning Representations (ICLR)*, pp. 25176–25204, 2026a.

Jie Liu, Gongye Liu, Jiajun Liang, Yangguang Li, Jiaheng Liu, Xintao Wang, Pengfei Wan, Di Zhang, and Wanli Ouyang. Flow-GRPO: Training Flow Matching Models via Online RL. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 40783–40818, 2025b.

Kai Liu, Wei Li, Lai Chen, Shengqiong Wu, Yanhao Zheng, Jiayi Ji, Fan Zhou, Jiebo Luo, Ziwei Liu, Hao (Scofield) Fei, and Tat-Seng Chua. JavisDiT: Joint Audio-Video Diffusion Transformer with Hierarchical Spatio-Temporal Prior Synchronization. In *International Conference on Learning Representations (ICLR)*, pp. 139160–139194, 2026b.

Zhiheng Liu, Weiming Ren, Haozhe Liu, Zijian Zhou, Shoufa Chen, Haonan Qiu, Xiaoke Huang, Zhaochong An, Fanny Yang, Aditya Patel, Viktar Atliha, Tony Ng, Xiao Han, Chuyan Zhu, Chenyang Zhang, Ding Liu, Juan-Manuel Perez-Rua, Sen He, Jürgen Schmidhuber, Wenhu Chen, Ping Luo, Wei Liu, Tao Xiang, Jonas Schult, and Yuren Cong. TUNA: Taming Unified Visual Representations for Native Unified Multimodal Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15740–15751, 2026c.

David Lopez-Paz, Léon Bottou, Bernhard Schölkopf, and Vladimir Vapnik. Unifying Distillation and Privileged Information. In *International Conference on Learning Representations (ICLR)*, 2016.

Ilya Loshchilov and Frank Hutter. Decoupled Weight Decay Regularization. In *International Conference on Learning Representations (ICLR)*, 2019.

Simian Luo, Chuanhao Yan, Chenxu Hu, and Hang Zhao. Diff-Foley: Synchronized Video-to-Audio Synthesis with Latent Diffusion Models. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 48855–48876, 2023.

Navonil Majumder, Chia-Yu Hung, Deepanway Ghosal, Wei-Ning Hsu, Rada Mihalcea, and Soujanya Poria. Tango 2: Aligning Diffusion-Based Text-to-Audio Generations through Direct Preference Optimization. In *Proceedings of the 32nd ACM International Conference on Multimedia (ACM MM)*, pp. 564–572, 2024.

Josh H. McDermott and Eero P. Simoncelli. Sound Texture Perception via Statistics of the Auditory Periphery: Evidence from Sound Synthesis. *Neuron*, 71(5):926–940, 2011.

Aditya K. Menon, Ankit Singh Rawat, Sashank Reddi, Seungyeon Kim, and Sanjiv Kumar. A Statistical Perspective on Distillation. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*, pp. 7632–7642, 2021.

NVIDIA. Cosmos 3: Omnimodal World Models for Physical AI. *arXiv preprint arXiv:2606.02800*, 2026.

Hyun-Bin Oh, Yuhta Takida, Toshimitsu Uesaka, Tae-Hyun Oh, and Yuki Mitsufuji. PAVAS: Physics-Aware Video-to-Audio Synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14481–14491, 2026.

Andrew Owens, Phillip Isola, Josh McDermott, Antonio Torralba, Edward H. Adelson, and William T. Freeman. Visually Indicated Sounds. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2405–2413, 2016.

Xichen Pan, Li Dong, Shaohan Huang, Zhiliang Peng, Wenhu Chen, and Furu Wei. Kosmos-G: Generating Images in Context with Multimodal Large Language Models. In *International Conference on Learning Representations (ICLR)*, pp. 40959–40974, 2024.

Wenshuo Peng and Kaipeng Zhang. HarmoniDPO: Video-Guided Audio Generation via Preference-Optimized Diffusion. *International Journal of Computer Vision*, 134(2):77, 2026.

Yijie Qian, Juncheng Wang, Chao Xu, Huihan Wang, Yuxiang Feng, Yang Liu, Baigui Sun, Yong Liu, and Shujun Wang. Beyond Time Shifts: Adapting Omni-LLM as a Reference-Free Evaluator for Generative Audio-Visual Models. In *European Conference on Computer Vision (ECCV)*, pp. 39–58, 2026.

Koichi Saito, Julian Tanke, Christian Simon, Masato Ishii, Kazuki Shimada, Zachary Novack, Zhi Zhong, Akio Hayakawa, Takashi Shibuya, and Yuki Mitsufuji. SoundReactor: Frame-Level Online Video-to-Audio Generation. *arXiv preprint arXiv:2510.02110*, 2025.

Ashish Seth, Xinhao Mei, Changsheng Zhao, Varun Nagaraja, Ernie Chang, Gregory P. Meyer, Gael Le Lan, Yunyang Xiong, Vikas Chandra, Yangyang Shi, Dinesh Manocha, and Zhipeng Cai. EgoAVU: Egocentric Audio-Visual Understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15805–15814, 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. *arXiv preprint arXiv:2508.16930*, 2025.

Jai Kumar Sharma and Peeyush Tapadiya. What Do Audio-Visual Synchronization Metrics Actually Measure? European Conference on Computer Vision (ECCV) Workshop on Generative AI for Audio-Visual Content Creation (Gen4AVC), 2026.

SII-OpenMOSS Team. MOVA: Towards Scalable and Synchronized Video-Audio Generation. *arXiv preprint arXiv:2602.08794*, 2026.

Charlie Snell, Dan Klein, and Ruiqi Zhong. Learning by Distilling Context. *arXiv preprint arXiv:2209.15189*, 2022.

Tomáš Souček and Jakub Lokoč. TransNet V2: An Effective Deep Network Architecture for Fast Shot Transition Detection. In *Proceedings of the 32nd ACM International Conference on Multimedia (ACM MM)*, pp. 11218–11221, 2024.

Elizabeth Spelke. Infants' Intermodal Perception of Events. *Cognitive Psychology*, 8(4):553–560, 1976.

Marilyn Strathern. 'Improving Ratings': Audit in the British University System. *European Review*, 5(3):305–321, 1997.

Hao Tang, Chen-Wei Xie, Xiaoyi Bao, Tingyu Weng, Pandeng Li, Yun Zheng, and Liwei Wang. UniLiP: Adapting CLIP for Unified Multimodal Understanding, Generation and Editing. In *International Conference on Learning Representations (ICLR)*, pp. 60440–60465, 2026.

Ye Tao, Lupeng Liu, Xuenan Xu, Jiasun Feng, Jiarui Wang, Ying Qin, Shuiyang Mao, Wei Liu, and Shuai Wang. Foley-Omni: A Unified Multimodal Generation Model from Task-Level Audio Synthesis to Complete Video Soundtrack Generation. *arXiv preprint arXiv:2606.03672*, 2026.

Team Wan. Wan: Open and Advanced Large-Scale Video Generative Models. *arXiv preprint arXiv:2503.20314*, 2025.

Zeyue Tian, Zhaoyang Liu, Yizhu Jin, Ruibin Yuan, Liumeng Xue, Xu Tan, Qifeng Chen, Wei Xue, and Yike Guo. AudioX: A Unified Framework for Anything-to-Audio Generation. In *International Conference on Learning Representations (ICLR)*, pp. 110625–110651, 2026a.

Zeyue Tian, Binxin Yang, Zhaoyang Liu, Jiexuan Zhang, Ruibin Yuan, Hubery Yin, Qifeng Chen, Chen Li, Jing Lyu, Wei Xue, and Yike Guo. Audio-Omni: Extending Multi-Modal Understanding to Versatile Audio Generation and Editing. In *Proceedings of the SIGGRAPH 2026 Conference Papers*, pp. 1–10, 2026b.

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 Assessment for Speech, Music and Sound. In *IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pp. 1–8, 2025.

Thanh V. T. Tran, Ngoc-Son Nguyen, Luong Tran, Long-Khanh Pham, Paarth Neekhara, Shehzeen Hussain, and Van Nguyen. Precise Video-to-Audio Generation with Cross-Modal Alignment in Latent Space. In *European Conference on Computer Vision (ECCV)*, pp. 174–192, 2026.

Wei-Cheng Tseng, Xuanru Zhou, Mingyue Huo, Yiwen Shao, Hao Zhang, and Dong Yu. Revisiting Audio-Language Pretraining for Learning General-Purpose Audio Representation. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 34244–34263, 2026.

Yuta Ujiie, Wakayo Yamashita, Waka Fujisaki, So Kanazawa, and Masami K. Yamaguchi. Cross-modal Association of Auditory and Visual Material Properties in Infants. *Scientific Reports*, 8: 9301, 2018.

Ilpo Viertola, Vladimir Iashin, and Esa Rahtu. Temporally Aligned Audio for Video with Autoregression. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5, 2025.

Apoorv Vyas, Heng-Jui Chang, Cheng-Fu Yang, Po-Yao Huang, Luya Gao, Julius Richter, Sanyuan Chen, Matthew Le, Piotr Dollár, Christoph Feichtenhofer, Ann Lee, and Wei-Ning Hsu. Pushing the Frontier of Audiovisual Perception with Large-Scale Multimodal Correspondence Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 30172–30182, 2026.

Bram Wallace, Meihua Dang, Rafael Rafailov, Linqi Zhou, Aaron Lou, Senthil Purushwalkam, Stefano Ermon, Caiming Xiong, Shafiq Joty, and Nikhil Naik. Diffusion Model Alignment Using Direct Preference Optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8228–8238, 2024.

Heng Wang, Jianbo Ma, Santiago Pascual, Richard Cartwright, and Weidong Cai. V2A-Mapper: A Lightweight Solution for Vision-to-Audio Generation by Connecting Foundation Models. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pp. 15492–15501, 2024a.

Jun Wang, Xijuan Zeng, Chunyu Qiang, Ruilong Chen, Shiyao Wang, Le Wang, Wangjing Zhou, Pengfei Cai, Jiahui Zhao, Nan Li, Zihan Li, Yuzhe Liang, Xiaopeng Wang, Haorui Zheng, Ming Wen, Kang Yin, Yiran Wang, Nan Li, Feng Deng, Liang Dong, Chen Zhang, Di Zhang, and Kun Gai. Kling-Foley: Multimodal Diffusion Transformer for High-Quality Video-to-Audio Generation. *arXiv preprint arXiv:2506.19774*, 2025a.

Mengmeng Wang, Dengyang Jiang, Liuzhuozheng Li, Yucheng Lin, Guojiang Shen, Xiangjie Kong, Yong Liu, Guang Dai, and Jingdong Wang. SRA 2: Variational Autoencoder Self-Representation Alignment for Efficient Diffusion Training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 32978–32987, 2026a.

Renhao Wang, Haoran Geng, Tingle Li, Philipp Wu, Feishi Wang, Gopala Anumanchipalli, Trevor Darrell, Boyi Li, Pieter Abbeel, Jitendra Malik, and Alexei A. Efros. The Sound of Simulation: Learning Multimodal Sim-to-Real Robot Policies with Generative Audio. In *Proceedings of the 9th Conference on Robot Learning (CoRL)*, pp. 420–436, 2025b.

Shiyao Wang, Xijuan Zeng, Hui Wang, Shiwan Zhao, Feng Deng, Chen Zhang, and Yong Qin. FoleyGenEx: Unified Video-to-Audio Generation with Multi-Modal Control, Temporal Alignment, and Semantic Precision. In *Proceedings of Interspeech*, pp. 4052–4061, 2026b.

Wei-Yao Wang, Kazuya Tateishi, Shuyang Cui, Christian Simon, Takashi Shibuya, Shusuke Takahashi, and Yuki Mitsufuji. Omni-Interactive Universal Embedder. *arXiv preprint arXiv:2608.27044*, 2026c.

Wenxuan Wang, Quan Sun, Fan Zhang, Yepeng Tang, Jing Liu, and Xinlong Wang. Diffusion Feedback Helps CLIP See Better. In *International Conference on Learning Representations (ICLR)*, pp. 74327–74345, 2025c.

Yongqi Wang, Wenxiang Guo, Rongjie Huang, Jiawei Huang, Zehan Wang, Fuming You, Ruiqi Li, and Zhou Zhao. Frieren: Efficient Video-to-Audio Generation Network with Rectified Flow Matching. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 128118–128138, 2024b.

Yuancheng Wang, Zhenyu Tang, Yun Wang, Arthur Hinsvark, Yingru Liu, Yinghao Li, Kainan Peng, Junyi Ao, Mingbo Ma, Mike Seltzer, Qing He, and Xubo Liu. Scaling Speech Tokenizers with Diffusion Autoencoders. In *International Conference on Learning Representations (ICLR)*, pp. 113881–113904, 2026d.

Zehan Wang, Ke Lei, Chen Zhu, Jiawei Huang, Sashuai Zhou, Luping Liu, Xize Cheng, Shengpeng Ji, Zhenhui Ye, Tao Jin, and Zhou Zhao. T2A-Feedback: Improving Basic Capabilities of Text-to-Audio Generation via Fine-Grained AI Feedback. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 23535–23547, 2025d.

Yusong Wu, Ke Chen, Tianyu Zhang, Yuchen Hui, Taylor Berg-Kirkpatrick, and Shlomo Dubnov. Large-Scale Contrastive Language-Audio Pretraining with Feature Fusion and Keyword-to-Caption Augmentation. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5, 2023.

Ji Xie, Trevor Darrell, Luke Zettlemoyer, and XuDong Wang. Reconstruction Alignment Improves Unified Multimodal Models. In *International Conference on Learning Representations (ICLR)*, pp. 120095–120137, 2026.

Yazhou Xing, Yingqing He, Zeyue Tian, Xintao Wang, and Qifeng Chen. Seeing and Hearing: Open-Domain Visual-Audio Generation with Diffusion Latent Aligners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7151–7161, 2024.

Kaituo Xu, Yan Jia, Kai Huang, Junjie Chen, Wenpeng Li, Kun Liu, Feng-Long Xie, Xu Tang, and Yao Hu. FireRedASR2S: A State-of-the-Art Industrial-Grade All-in-One Automatic Speech Recognition System. *arXiv preprint arXiv:2603.10420*, 2026.

Zeyue Xue, Jie Wu, Yu Gao, Fangyuan Kong, Lingting Zhu, Mengzhao Chen, Zhiheng Liu, Wei Liu, Qiushan Guo, Weilin Huang, and Ping Luo. DanceGRPO: Unleashing GRPO on Visual Generation. *arXiv preprint arXiv:2505.07818*, 2025.

Jianxuan Yang, Xinyue Guo, Zhi Cheng, Kai Wang, Lipan Zhang, Jinjie Hu, Qiang Ji, Yihua Cao, Yihao Meng, Zhaoyue Cui, Mengmei Liu, Meng Meng, and Jian Luan. ControlFoley: Unified and Controllable Video-to-Audio Generation with Cross-Modal Conflict Handling. In *Proceedings of the 34th ACM International Conference on Multimedia (ACM MM)*, 2026.

Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. CoCa: Contrastive Captioners Are Image-Text Foundation Models. *Transactions on Machine Learning Research*, 2022.

Sihyun Yu, Sangkyung Kwak, Huiwon Jang, Jongheon Jeong, Jonathan Huang, Jinwoo Shin, and Saining Xie. Representation Alignment for Generation: Training Diffusion Transformers Is Easier Than You Think. In *International Conference on Learning Representations (ICLR)*, pp. 87400–87442, 2025.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid Loss for Language Image Pre-Training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 11975–11986, 2023.

Kang Zhang, Trung X. Pham, Suyeon Lee, Axi Niu, Arda Senocak, and Joon Son Chung. Model-Guided Dual-Role Alignment for High-Fidelity Open-Domain Video-to-Audio Generation. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 36224–36255, 2025.

Yiming Zhang, Yicheng Gu, Yanhong Zeng, Zhening Xing, Yuancheng Wang, Zhizheng Wu, Bin Liu, and Kai Chen. FoleyCrafter: Bring Silent Videos to Life with Lifelike and Synchronized Sounds. *International Journal of Computer Vision*, 134(1):46, 2026.

Siyan Zhao, Zhihui Xie, Mengchen Liu, Jing Huang, Guan Pang, Feiyu Chen, and Aditya Grover. Self-Distilled Reasoner: On-Policy Self-Distillation for Large Language Models. In *Proceedings of the 43rd International Conference on Machine Learning (ICML)*, 2026.

Wenliang Zhao, Lujia Bai, Yongming Rao, Jie Zhou, and Jiwen Lu. UniPC: A Unified Predictor-Corrector Framework for Fast Sampling of Diffusion Models. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 49842–49869, 2023.

Kaiwen Zheng, Huayu Chen, Haotian Ye, Haoxiang Wang, Qinsheng Zhang, Kai Jiang, Hang Su, Stefano Ermon, Jun Zhu, and Ming-Yu Liu. DiffusionNFT: Online Diffusion Reinforcement with Forward Process. In *International Conference on Learning Representations (ICLR)*, pp. 134129–134150, 2026.

Wei Zhou, Xiongwei Zhu, Lingdong Kong, Bo Chen, Lei Zhang, Yongyuan Liang, Xiaoxia Hou, Ye Tian, Xian Sun, Yingshuo Wang, Linfeng Li, Shengqiong Wu, Leigang Qu, Feng Li, Wei Liu, Julian McAuley, and Tat-Seng Chua. On-Policy Self-Distillation in Diffusion Models. *arXiv preprint arXiv:2608.24646*, 2026.

Daniil Zverev, Thaddäus Wiedemer, Ameya Prabhu, Matthias Bethge, Wieland Brendel, and A. Sophia Koepke. VGGSounder: Audio-Visual Evaluations for Foundation Models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1027–1037, 2025.

A Project Webpage

We provide an interactive project webpage in the supplementary material to illustrate the central question of this paper: whether a model can infer from video both what should be heard and when. We also include animations showing how caption-guided self-distillation trains the visual reasoner and RL then refines the audio generator (Section 3). We present qualitative results from the three benchmarks, with listening comparisons organized around the following points:

- **Dependence on caption guidance.** Comparing the same baseline with and without a caption reveals what textual guidance contributes. In a curated hammering example, removing the caption introduces background music unrelated to the visible action. This illustrates why simply removing captions does not ensure that generation is grounded in video.
- **Sound content and event timing.** A correct sound type does not ensure a sound at every visible contact. In a repeated-tapping example, a caption-conditioned baseline produces the intended sound but misses a tap that is preserved without the caption. This comparison illustrates the need to assess event timing alongside sound content.
- **Learning to generate from video.** We compare TacitV2A with the caption-conditioned Cosmos3 teacher, its caption-free counterpart, and external baselines. These comparisons let readers assess whether TacitV2A recovers the relevant sounds while following visible events, without captions at inference.

The webpage also presents benchmark metrics and human preference results, connecting these listening comparisons to the broader evaluation in Section 4 and Appendix D.

B Method Details

B.1 Visual Conditioning for Audio Generation

Cosmos3 architecture. Cosmos3 (NVIDIA, 2026) combines an autoregressive (AR) reasoner with a diffusion (DM) generator in a mixture-of-transformers architecture (Section 2.3.2 of the Cosmos3 report). The two towers have separate attention, feed-forward, and normalization parameters and interact only through attention. At each layer ℓ, each tower forms its own queries, keys, and values. The reasoner applies causal self-attention,

$$\boldsymbol{O}^{\ell}_{\mathrm{AR}} = \mathrm{Attn}_{\mathrm{causal}}\big(\boldsymbol{Q}^{\ell}_{\mathrm{AR}}, \boldsymbol{K}^{\ell}_{\mathrm{AR}}, \boldsymbol{V}^{\ell}_{\mathrm{AR}}\big)\,, \tag{6}$$

while generator queries attend jointly to the keys and values of both towers (Equation 1). The reasoner does not attend to generator tokens.

Inputs in our V2A setting. The reasoner reads ViT features of sampled frames in temporal order, followed by the system prompt tokens. Under causal attention, the prompt tokens attend to the preceding visual tokens, so the reasoner context depends on the video even though the prompt is shared. The generator receives clean video-VAE latents and noisy audio latents, and only the audio latents are generated. The action modality is unused, and the ViT and both VAEs remain frozen. We write $\boldsymbol{h}_S$ for the student's layer-wise reasoner keys and values, $\{(\boldsymbol{K}^{\ell}_{\mathrm{AR}}, \boldsymbol{V}^{\ell}_{\mathrm{AR}})\}^{L}_{\ell=1}$, taken after the model's normalization and positional encoding, and $\boldsymbol{h}_T$ for the corresponding collection from the caption-conditioned teacher. An audio query thus retrieves the reasoner context and the audio-visual latents in the same attention operation. The generator uses $\boldsymbol{h}_S$ directly, without a decoded caption, and $\boldsymbol{h}_S$ does not depend on the noisy audio.

Model configuration. For Cosmos3-Nano, the autoregressive reasoner and diffusion generator each have 36 transformer layers with a hidden dimension of 4,096, a feed-forward dimension of 12,288, 32 query heads, and 8 key-value heads. The two branches have independent attention, feed-forward, and normalization parameters. The pre-trained Qwen3-VL vision encoder (Bai et al., 2025) is a 27-layer ViT with a hidden dimension of 1,152, 16 attention heads, and 16×16 spatial patches. The video VAE from Wan2.2-TI2V-5B (Team Wan, 2025) produces 48-channel latents with $4\times$ temporal and 16×16 spatial compression, followed by 2×2 spatial patchification. The audio VAE, based on the ETTA architecture (Lee et al., 2025), encodes 48-kHz audio into 64-channel latents at 25 latent frames per second.

Zero-shot V2A inference. The Cosmos3-Nano generator is pre-trained for joint audio-visual generation, but not for generating audio alone from an entire observed video. We use it in this setting by encoding the video with the frozen video VAE, keeping all latents $\boldsymbol{x} = E_{\mathrm{V}}(\mathcal{V})$ clean, and initializing the audio latents with Gaussian noise. Sampling denoises only the audio, from $\sigma = 1$ to $\sigma = 0$ under the convention in Equation 2, and a frozen audio-VAE decoder converts the result into a waveform. We use classifier-free guidance with the settings in Section 4.1. The unconditional branch replaces the reasoner input with a null input that omits the Audio-tuned system prompt, the caption, and the ViT tokens. The generator still receives the clean video latents, so guidance amplifies the effect of the reasoner context. We use the same procedure after self-distillation and RL.

Caption-conditioned teacher. The caption-conditioned model gives the reasoner a structured caption instead of the video. A per-sample caption C is expanded by the Claude Opus 4.6 upsampler U into a JSON description $J = U(C)$ using the original Cosmos prompt-upsampling instructions.[1] For training clips, C is the caption we generate from each clip's video and audio (Appendix C.1). For evaluation, C is the original benchmark caption. The reasoner processes J, and the generator attends to its layer-wise keys and values throughout denoising while still receiving the clean video latents. A caption can supply much of the semantic answer: for example, "a paddle hits a plastic ball" names the source, interaction, and material that a caption-free model must infer from pixels. Strong generation under caption conditioning therefore does not show that the model can recover these cues from video. We use this model as the frozen teacher and as the captioned Cosmos3 reference in our tables.

System prompt. The system prompt keeps the 18 top-level fields of the original Cosmos JSON schema but directs their content toward sound-producing events. Its instructions emphasize the dominant visibly active source, acoustically relevant materials, contact or emission mechanisms, spatial relationships, and the onset, offset, and repetition of events. They also ask for consistent source identity over time, separate source motion from camera motion, and keep descriptions grounded in visible evidence, without unsupported speech, music, or offscreen events. When fine acoustic properties cannot be inferred reliably, the prompt favors a broader plausible sound category. The prompt contains no sample-specific labels or event descriptions, and only duration, frame rate, resolution, and aspect ratio are filled in automatically from the generation settings. We refine these instructions with all model weights frozen (Appendix C.2) and call the result the *Audio-tuned* system prompt, which stays fixed for training and inference. Appendix E reproduces it in full.

B.2 Self-Distillation Objective and Implementation

The generator already synthesizes sound well when the reasoner reads a detailed caption. With ViT features and the shared prompt instead, the reasoner must infer sound sources and physical interactions from visual features and encode them in a form the generator can use. We learn this with reasoner LoRA while keeping the generator frozen, which preserves its synthesis ability and focuses learning on the visual conditioning.

Teacher and student conditioning. We train on paired videos and audio with corresponding per-sample JSON captions $J = U(C)$, obtained by upsampling each clip's generated caption as in Appendix B.1. The teacher uses the frozen pre-trained reasoner R_0 with the sample's JSON caption J, giving $\boldsymbol{h}_T = R_0(J)$. Its reasoner receives text only, while the video remains available through the generator's VAE conditioning. The student uses the visual conditioning $\boldsymbol{h}_S$ from Section 3.1. Both views use the same frozen generator $G_{\boldsymbol{\varphi}}$. Here $\boldsymbol{\theta}$ denotes the trainable reasoner LoRA parameters, while $\boldsymbol{\varphi}$ remains fixed. The captions provide privileged supervision during training, while the student's inference inputs remain the video and shared prompt.

For classifier-free conditioning dropout, we draw one shared variable $d \sim \mathrm{Bernoulli}(p_{\mathrm{drop}})$ per example, with $p_{\mathrm{drop}} = 0.1$. When $d = 0$, both contexts are as defined above. When $d = 1$, both paths receive the null input of the unconditional branch (Appendix B.1), giving $\boldsymbol{h}_S = R_{\boldsymbol{\theta}}(\emptyset)$ and $\boldsymbol{h}_T = R_0(\emptyset)$, where $\emptyset$ denotes this null input. Conditioning dropout during training therefore matches the unconditional branch used for guidance at inference. Clean video-VAE latents remain

[1] https://github.com/nvidia/cosmos-framework/blob/main/docs/prompt_upsampling.md

available to both paths. Below, $\boldsymbol{h}_S$, $\boldsymbol{h}_T$, and their predictions include this paired dropout choice, with d suppressed in their notation.

Audio-space supervision. The teacher and student reasoner sequences differ in length and content, so their hidden states lack a direct tokenwise correspondence. We instead supervise their predictions at the generator's audio output positions, which are aligned for the same training example. We encode the reference waveform as $\boldsymbol{z}_0 = E_{\mathrm{A}}(A)$ and construct the noisy latent $\boldsymbol{z}_\sigma$ and target velocity $\boldsymbol{u}^\star$ using Equation 2. In both self-distillation and RL, audio noise levels follow the video branch's mode-sampling schedule (NVIDIA, 2026), with mode parameter 1.29 and rectified-flow shift 10. The teacher and student receive identical clean video latents $\boldsymbol{x}$, noisy audio latents $\boldsymbol{z}_\sigma$, and noise level σ. Their audio-velocity predictions are defined in Equation 3. Matching these predictions transfers the teacher's conditioning behavior in the space used to generate sound, without requiring alignment of the reasoner's text and visual tokens. Unlike on-policy or hybrid-policy self-distillation (Zhao et al., 2026; Bu et al., 2026), we supervise on forward-noised real audio rather than on student-generated states, and adapt only the conditioning pathway.

The expectations below average over training triples $(\mathcal{V}, A, J)$ and independent draws of the noise level, Gaussian noise, and conditioning dropout from the distributions specified above. The individual loss terms in Equation 4 are

$$\mathcal{L}_{\text{flow}} = \frac{1}{N_z}\,\mathbb{E}_{\mathcal{V},A,J,\sigma,\boldsymbol{\epsilon},d}\left[\|\hat{\boldsymbol{u}}_S - \boldsymbol{u}^\star\|_2^2\right], \tag{7}$$

$$\mathcal{L}_{\text{distill}} = \frac{1}{N_z}\,\mathbb{E}_{\mathcal{V},A,J,\sigma,\boldsymbol{\epsilon},d}\left[\|\hat{\boldsymbol{u}}_S - \mathrm{sg}(\hat{\boldsymbol{u}}_T)\|_2^2\right], \tag{8}$$

where sg denotes stop-gradient and N_z counts audio latent channels and time steps within each example. The norms act on the vectorized audio-velocity errors, so division by N_z averages the squared error over their elements. Both terms operate only on generated audio positions. The clean video conditioning is excluded from the loss. The flow term anchors the student to the observed audio, while the distillation term encourages agreement with the caption-conditioned teacher under the same noise realization. We use $\lambda_{\text{FM}} = 0.1$ and $\lambda_{\text{KD}} = 1$, giving $\mathcal{L} = 0.1\mathcal{L}_{\text{flow}} + \mathcal{L}_{\text{distill}}$. We selected this ratio from the FoleyBench ablation in Table 5 and kept it fixed for VGGSounder and FlatSounds.

Adaptation and gradient flow. We adapt only the reasoner's query, key, value, and output projections using LoRA (Hu et al., 2022), with rank 16 and scaling parameter 32. The pre-trained reasoner weights, generator, ViT, and both VAEs remain frozen. Gradients from both losses pass through the frozen generator's joint attention to $\boldsymbol{h}_S$ and update the reasoner adapters. Teacher and student share the base model: adapters are disabled for the teacher's no-gradient forward pass and enabled for the student's forward pass. Paired dropout changes the reasoner inputs as specified above, without dropping the video-VAE conditioning or changing which parameters are optimized. We train for three epochs with a global batch size of 128 and use the final checkpoint. AdamW (Loshchilov & Hutter, 2019) uses a constant learning rate of 5×10^{-5}, betas $(0.9, 0.99)$, and zero weight decay.

B.3 Why Caption-Guided Self-Distillation Can Help

Our teacher and student share pre-trained weights, but receive different information. A caption makes sound-relevant cues explicit to the teacher, whereas the student must obtain useful conditioning from visual features. This is a form of learning using privileged information (Lopez-Paz et al., 2016). The core idea is that caption guidance can provide more stable sound-prediction targets, while matching these targets through the frozen generator trains the reasoner to produce useful visual conditioning. The flow-matching term keeps learning anchored to recorded audio. Below, we explain this mechanism and show, in a simplified estimation problem, when the benefit of more stable targets outweighs the bias introduced by the teacher.

Learning conditioning through sound prediction. Let W collect the effective inputs available to the student: the reasoner input after conditioning dropout, clean video latents, noisy audio latents, noise level, and the dropout variable d. The effective reasoner input is the ViT features and shared prompt, including its filled-in generation settings, when $d = 0$, and the null input when $d = 1$. We

retain the notation $\boldsymbol{u}^\star$, $\hat{\boldsymbol{u}}_T$, and $\hat{\boldsymbol{u}}_S$ for the target, teacher, and student velocities, respectively. The trainable reasoner parameters $\boldsymbol{\theta}$ affect $\hat{\boldsymbol{u}}_S$ only through $\boldsymbol{h}_S$. For a single sampled example, we view the adapter parameters and layer-wise reasoner context as vectors. With $q = (\boldsymbol{z}_\sigma, \sigma, \boldsymbol{x})$, the chain rule for the half-squared distillation error gives

$$\nabla_{\boldsymbol{\theta}} \tfrac{1}{2} \|\hat{\boldsymbol{u}}_S - \mathrm{sg}(\hat{\boldsymbol{u}}_T)\|_2^2 = (D_{\boldsymbol{\theta}} \boldsymbol{h}_S)^\top \left(D_{\boldsymbol{h}_S} G_{\boldsymbol{\varphi}}(q; \boldsymbol{h}_S)\right)^\top (\hat{\boldsymbol{u}}_S - \hat{\boldsymbol{u}}_T)\,. \tag{9}$$

Here D denotes a Jacobian, and the error is unnormalized. Thus the teacher supervises the visual representation through changes that affect the frozen generator's velocity predictions. This comparison remains well defined when text and visual contexts have different token counts. It requires agreement in their effect on sound prediction, without requiring equal hidden states. Conditioning changes that leave the generator's output unchanged are not distinguished by this loss.

The target learned without captions. For the statistical analysis, we fix the teacher and consider the population distribution of video-audio-caption triples together with the noise and dropout sampling described above. All expectations below refer to this joint distribution. Assume finite second moments and nonnegative loss weights with $\lambda_{\mathrm{FM}} + \lambda_{\mathrm{KD}} > 0$. Let $\rho = \lambda_{\mathrm{KD}}/(\lambda_{\mathrm{FM}} + \lambda_{\mathrm{KD}})$. Completing the square in Equation 4 gives

$$\begin{aligned} \boldsymbol{Y}_\rho &= (1-\rho)\boldsymbol{u}^\star + \rho \hat{\boldsymbol{u}}_T, \\ \frac{\mathcal{L}}{\lambda_{\mathrm{FM}} + \lambda_{\mathrm{KD}}} &= \frac{1}{N_z}\, \mathbb{E}\, \|\hat{\boldsymbol{u}}_S - \boldsymbol{Y}_\rho\|_2^2 + \frac{\rho(1-\rho)}{N_z}\, \mathbb{E}\, \|\boldsymbol{u}^\star - \hat{\boldsymbol{u}}_T\|_2^2\,. \end{aligned} \tag{10}$$

The second term is independent of $\boldsymbol{\theta}$. For the following estimation identities, we omit the common factor $1/N_z$ and work with unnormalized squared errors. Define $\boldsymbol{m}(W) = \mathbb{E}[\boldsymbol{u}^\star \mid W]$ and $\boldsymbol{b}(W) = \mathbb{E}[\hat{\boldsymbol{u}}_T - \boldsymbol{u}^\star \mid W]$. Conditional least squares then yields

$$\mathbb{E}\|\hat{\boldsymbol{u}}_S - \boldsymbol{Y}_\rho\|_2^2 = \mathbb{E}\|\hat{\boldsymbol{u}}_S - (\boldsymbol{m}(W) + \rho \boldsymbol{b}(W))\|_2^2 + \mathbb{E}\|\boldsymbol{Y}_\rho - \mathbb{E}[\boldsymbol{Y}_\rho \mid W]\|_2^2. \tag{11}$$

To see this, substitute $\mathbb{E}[\boldsymbol{Y}_\rho \mid W] = \boldsymbol{m}(W) + \rho \boldsymbol{b}(W)$ and expand the square. The cross term vanishes because the residual has zero conditional mean. The unrestricted optimal predictor is therefore $\boldsymbol{m}(W) + \rho \boldsymbol{b}(W)$. Fluctuations of the blended target around this conditional mean contribute irreducible target variance. The student is not required to reconstruct every detail of the caption. The implemented student is further restricted to $\hat{\boldsymbol{u}}_S = G_{\boldsymbol{\varphi}}(q; \boldsymbol{h}_S(\mathcal{V}))$, with the prompt and dropout state implicit in $\boldsymbol{h}_S$. Its reasoner cannot depend on the noisy audio, and LoRA need not realize the unrestricted optimum.

When teacher targets can help. The statistical benefit of distillation depends on the tradeoff between teacher bias and target variance (Menon et al., 2021). To illustrate this tradeoff, fix W and let $\bar{\boldsymbol{Y}}_\rho$ average n independent draws from $\boldsymbol{Y}_\rho \mid W$. Here n counts conditional draws for an illustrative mean estimator, rather than the samples used in LoRA training. Its expected squared error is

$$\mathbb{E}\big[\|\bar{\boldsymbol{Y}}_\rho - \boldsymbol{m}(W)\|_2^2 \mid W\big] = \rho^2 \|\boldsymbol{b}(W)\|_2^2 + \frac{\mathrm{tr}\,\mathrm{Cov}(\boldsymbol{Y}_\rho \mid W)}{n}. \tag{12}$$

Flow targets alone give $\mathrm{tr}\,\mathrm{Cov}(\boldsymbol{u}^\star \mid W)/n$. Hence distillation helps this estimator when the reduction in target variance divided by n exceeds the introduced squared bias.

As an idealized sufficient condition, suppose a teacher observes both W and J and predicts $\hat{\boldsymbol{u}}_T = \mathbb{E}[\boldsymbol{u}^\star \mid W, J]$. Then $\boldsymbol{b}(W) = \boldsymbol{0}$, and conditional orthogonality gives

$$\mathrm{Cov}(\boldsymbol{Y}_\rho \mid W) = \mathrm{Cov}(\hat{\boldsymbol{u}}_T \mid W) + (1-\rho)^2 \mathbb{E}[\mathrm{Cov}(\boldsymbol{u}^\star \mid W, J) \mid W] \preceq \mathrm{Cov}(\boldsymbol{u}^\star \mid W). \tag{13}$$

Indeed, writing $\boldsymbol{u}^\star = \hat{\boldsymbol{u}}_T + (\boldsymbol{u}^\star - \hat{\boldsymbol{u}}_T)$ makes the two components conditionally uncorrelated. The teacher removes the residual variability not explained by its conditioning, while the blended target retains a fraction $1 - \rho$ of that residual. Under this condition, distillation preserves the population flow objective's minimizers without increasing target variance. Our actual teacher uses (q, J) and the paired dropout state rather than all of W, so caption conditioning and shared pre-trained weights alone do not imply this assumption.

The analysis clarifies the complementary roles of the two losses. Teacher predictions can reduce target variability, while flow matching limits the influence of teacher bias. With the generator frozen, both signals supervise the reasoner through the effect of its visual conditioning on sound prediction.

The analysis does not, however, determine the loss weights. With our chosen weights, $\rho \approx 0.91$, so the blended target places greater weight on teacher predictions while retaining supervision from recorded audio. Our ablation shows that this combination outperforms the other tested ratios, including distillation alone (Table 5). More generally, the analysis characterizes the training objective under the stated assumptions, while Section 4 evaluates the resulting model.

B.4 RL Details

Policy and adaptation. The initial policy is the model learned by caption-guided self-distillation. Its reasoner adapters, base model weights, ViT, and VAEs remain frozen. We introduce rank-16 LoRA adapters with scaling parameter 32 in the generator's query, key, value, and output projections, together with the audio input/output projections. The adapters' output matrices are initialized to zero, preserving the initial model's predictions. Only these generator adapters receive RL updates. We denote these adapters by $\boldsymbol{\eta}$ and the adapted generator by $G_{\boldsymbol{\varphi},\boldsymbol{\eta}}$, with the pre-trained parameters $\boldsymbol{\varphi}$ fixed. Disabling the generator adapters recovers $G_{\boldsymbol{\varphi}}$. The caption teacher and self-distillation objective are absent from this stage. Training uses the same filtered video-audio pairs as self-distillation.

Audio-audio reward. For each video $\mathcal{V}$ with reference waveform A, the current policy generates K audio latents $\tilde{\boldsymbol{z}}_{0,i}$ and decodes them into waveforms $\tilde{A}_i$. Generation uses the same guided sampler as evaluation (Section 4.1), so the reward scores samples produced as at test time. Let f_{C} denote the frozen LAION-CLAP audio encoder (Wu et al., 2023). We compute the audio-audio cosine similarity s_i in Equation 5. The reward encoder uses HTSAT-base (Chen et al., 2022) with feature fusion disabled. Waveforms are processed at 48 kHz by averaging channels, removing each clip's DC component, and retaining the first eight seconds, in that order. The CLAP frontend then applies its repeat-padding procedure to the ten-second embedding window. Audio-audio similarity is the sole reward. We do not directly optimize the audio-text CLAP scores used for evaluation, although the reward and LAION-CLAP evaluation share the same representation space.

We normalize scores within the K candidates for the same video and map the resulting advantages to optimality weights:

$$a_i = \frac{s_i - \bar{s}}{\mathrm{Std}(s_1, \ldots, s_K) + 10^{-4}}, \qquad r_i = \mathrm{sigmoid}(a_i/\tau)\,, \tag{14}$$

where $\bar{s}$ is the group mean, Std uses the sample standard deviation, and the temperature is $\tau = 0.3$. These weights determine the positive and negative components of the DiffusionNFT update. Rollout generation, decoding, and reward computation use no gradients: the generated latents and optimality weights are held constant when optimizing the policy.

Forward-process predictions. For each generated latent, we draw fresh Gaussian noise and one noise level using the same video-branch mode-sampling schedule as self-distillation, then form $\tilde{\boldsymbol{z}}_{\sigma_i,i} = (1-\sigma_i)\tilde{\boldsymbol{z}}_{0,i} + \sigma_i \boldsymbol{\epsilon}_i$. Let $q_i = (\tilde{\boldsymbol{z}}_{\sigma_i,i}, \sigma_i, \boldsymbol{x})$. On this same noisy input, we evaluate three audio velocities:

$$\begin{aligned} \boldsymbol{u}_i &= G_{\boldsymbol{\varphi},\boldsymbol{\eta}}(q_i; \boldsymbol{h}_S), \\ \boldsymbol{u}_i^{\mathrm{old}} &= \mathrm{sg}(G_{\boldsymbol{\varphi},\boldsymbol{\eta}_{\mathrm{old}}}(q_i; \boldsymbol{h}_S))\,, \\ \boldsymbol{u}_i^{\mathrm{ref}} &= \mathrm{sg}(G_{\boldsymbol{\varphi}}(q_i; \boldsymbol{h}_S))\,. \end{aligned} \tag{15}$$

All three use $\boldsymbol{h}_S$ from the same frozen, distilled visual reasoner, so the objective below acts only on the conditional velocity. The unconditional branch used for guidance during rollouts shares the generator adapters but receives no separate constraint. The old policy's adapters $\boldsymbol{\eta}_{\mathrm{old}}$ are copied from the current policy every ten updates. The fixed reference is the distilled model before RL, implemented by disabling the generator adapters in the old-policy copy while retaining its reasoner adapters. Rollouts use the current policy, while the old policy anchors the positive and negative corrections defined below.

We follow DiffusionNFT (Zheng et al., 2026) to construct positive and negative reconstruction predictions. Let $\boldsymbol{u}_i^{\mathrm{target}} = \boldsymbol{\epsilon}_i - \tilde{\boldsymbol{z}}_{0,i}$ be the forward velocity target. Define the correction

Table 7: DiffusionNFT training hyperparameters.

Setting	Value
Video/audio window	8 s, video at 24 fps, audio at 48 kHz
Video conditioning	256p video-VAE input, 16 ViT frames
Trainable LoRA	Generator attention and audio projections, rank 16, scaling 32
Parallelism and batch	8 FSDP ranks, 1 video per rank per update
Rollout group	$K = 8$ per video, 64 candidates per global update
Sampler	35 UniPC (Zhao et al., 2023) steps, guidance 6, flow shift 10
Objective	$\beta = 1$, $c = 5$, $\kappa = 1$, sigmoid optimality map with $\tau = 0.3$
Old-policy update	Copy the current policy every 10 updates
Optimizer	AdamW, peak learning rate 10^{-4}
AdamW moments and decay	$(0.9, 0.99)$, $\epsilon = 10^{-8}$, weight decay 0.05
Schedule	50-step warmup, cosine cycle of 500,000 steps
Learning-rate multipliers	Initial 10^{-6}, peak 1, cycle minimum 0.5
Updates and gradient clipping	1,000 updates, gradient norm limit 1
Precision	bfloat16 model, float32 loss computation
Checkpoint interval	Every 50 updates

$\boldsymbol{\delta}_i = \beta(\boldsymbol{u}_i - \boldsymbol{u}_i^{\text{old}})$, where $\beta > 0$ is the correction scale, giving the implicit velocities

$$\begin{aligned} \boldsymbol{u}_i^{+} &= \boldsymbol{u}_i^{\text{old}} + \boldsymbol{\delta}_i = (1-\beta)\boldsymbol{u}_i^{\text{old}} + \beta \boldsymbol{u}_i, \\ \boldsymbol{u}_i^{-} &= \boldsymbol{u}_i^{\text{old}} - \boldsymbol{\delta}_i = (1+\beta)\boldsymbol{u}_i^{\text{old}} - \beta \boldsymbol{u}_i. \end{aligned} \tag{16}$$

Training objective. Under our rectified-flow convention, $\tilde{\boldsymbol{z}}_{0,i} = \tilde{\boldsymbol{z}}_{\sigma_i,i} - \sigma_i \boldsymbol{u}_i^{\text{target}}$. The reconstruction residuals therefore satisfy

$$\boldsymbol{d}_i^{\pm} = \tilde{\boldsymbol{z}}_{\sigma_i,i} - \sigma_i \boldsymbol{u}_i^{\pm} - \tilde{\boldsymbol{z}}_{0,i} = -\sigma_i(\boldsymbol{u}_i^{\pm} - \boldsymbol{u}_i^{\text{target}}). \tag{17}$$

Their average squared error is $\|\boldsymbol{d}_i^{\pm}\|_2^2/N_z = \sigma_i^2 \|\boldsymbol{u}_i^{\pm} - \boldsymbol{u}_i^{\text{target}}\|_2^2/N_z$, connecting reconstruction to velocity matching. Following the adaptive weighting in Section 3.3 of DiffusionNFT, we divide each mean squared reconstruction error by its detached mean absolute reconstruction error, floored at 10^{-5}:

$$e_i^{\pm} = \frac{\|\boldsymbol{d}_i^{\pm}\|_2^2/N_z}{\text{sg}\big(\max\{\|\boldsymbol{d}_i^{\pm}\|_1/N_z, 10^{-5}\}\big)}. \tag{18}$$

Both norms act on the N_z audio latent elements within each sample, with $\|\cdot\|_1$ denoting the sum of absolute values. With reconstruction-loss scale c and reference-penalty weight κ, we obtain

$$\mathcal{L}_{\text{NFT}} = \frac{c}{\beta K} \sum_{i=1}^{K} \left[r_i e_i^{+} + (1-r_i) e_i^{-} \right] + \frac{\kappa}{K N_z} \sum_{i=1}^{K} \left\| \boldsymbol{u}_i - \boldsymbol{u}_i^{\text{ref}} \right\|_2^2, \tag{19}$$

with $\beta = 1$, $c = 5$, and $\kappa = 1$. The second term is a velocity-space reference penalty. This loss is computed per video and averaged over videos in the update. The paired real audio determines the reward, while the generated latent provides the reconstruction target. We compute the loss in float32 using only audio positions. Video latents remain clean conditioning throughout.

Initialization and hyperparameters. Table 7 summarizes the RL hyperparameters. We reset the optimizer and learning-rate schedule, disable classifier-free conditioning dropout so that every rollout group retains its video and fixed prompt, and use the final checkpoint after 1,000 updates.

Interpretation of the NFT update. The implemented objective is Equation 19. The derivation below isolates the basic NFT correction before adaptive reconstruction weighting and the reference penalty. For this analysis, consider candidate groups drawn from a fixed old policy and re-noised as above. Expectations average over video-audio pairs, the candidate group, and fresh noising. To interpret the direction of the update, first consider the unnormalized squared-velocity objective $\mathcal{L}_{\text{vel}} = \mathbb{E}[\ell_i]$, where

$$\ell_i = r_i \|\boldsymbol{u}_i^{+} - \boldsymbol{u}_i^{\text{target}}\|_2^2 + (1-r_i)\|\boldsymbol{u}_i^{-} - \boldsymbol{u}_i^{\text{target}}\|_2^2. \tag{20}$$

Writing $\boldsymbol{b}_i = \boldsymbol{u}_i^{\text{old}} - \boldsymbol{u}_i^{\text{target}}$ and completing the square gives

$$\begin{aligned} \ell_i &= \|\boldsymbol{b}_i\|_2^2 + \|\boldsymbol{\delta}_i\|_2^2 + 2(2r_i - 1)\langle \boldsymbol{b}_i, \boldsymbol{\delta}_i \rangle \\ &= \|\boldsymbol{\delta}_i - (2r_i - 1)(\boldsymbol{u}_i^{\text{target}} - \boldsymbol{u}_i^{\text{old}})\|_2^2 + C_i, \end{aligned} \tag{21}$$

where $C_i = 4r_i(1-r_i)\|\boldsymbol{b}_i\|_2^2$ is constant with respect to the trainable velocity. Thus $r_i > 1/2$ and $r_i < 1/2$ favor opposite corrections.

For the population interpretation, fix the noisy input $y = (\tilde{\boldsymbol{z}}_\sigma, \sigma, \boldsymbol{x}, \boldsymbol{h}_S)$ and assume $\boldsymbol{u}^{\mathrm{old}}(y) = \mathbb{E}[\boldsymbol{u}^{\mathrm{target}} \mid y]$ under old-policy samples. Writing r for a generic candidate's group-normalized optimality weight, define

$$\alpha = \mathbb{E}[r \mid y], \qquad \boldsymbol{\mu}^+ = \frac{\mathbb{E}[r\boldsymbol{u}^{\mathrm{target}} \mid y]}{\alpha}, \qquad \boldsymbol{\mu}^- = \frac{\mathbb{E}[(1-r)\boldsymbol{u}^{\mathrm{target}} \mid y]}{1-\alpha}.$$

Then $\boldsymbol{u}^{\mathrm{old}} = \alpha\boldsymbol{\mu}^+ + (1-\alpha)\boldsymbol{\mu}^-$, and the optimal correction is

$$\begin{aligned}\boldsymbol{\delta}^*(y) &= \mathbb{E}[(2r-1)(\boldsymbol{u}^{\mathrm{target}} - \boldsymbol{u}^{\mathrm{old}}) \mid y] \\ &= 2\alpha(\boldsymbol{\mu}^+ - \boldsymbol{u}^{\mathrm{old}}) = 2(1-\alpha)(\boldsymbol{u}^{\mathrm{old}} - \boldsymbol{\mu}^-). \end{aligned} \tag{22}$$

Hence $\boldsymbol{u}^{\mathrm{opt}} = \boldsymbol{u}^{\mathrm{old}} + 2\boldsymbol{\Delta}/\beta$, with $\boldsymbol{\Delta} = \alpha(\boldsymbol{\mu}^+ - \boldsymbol{u}^{\mathrm{old}})$, recovering the original reinforcement direction. This population result concerns the basic objective under the stated old-policy assumption. Our training instead uses current-policy rollouts, adaptive reconstruction weighting, and a reference penalty. The result therefore provides an idealized interpretation rather than an exact population optimum for Equation 19.

C Experimental Details

Evaluation. We follow the official evaluation protocol of each benchmark and use three generation seeds on FoleyBench and VGGSounder and ten on FlatSounds. On FoleyBench and VGGSounder, $\mathrm{FD}_{\mathrm{VGG}}$, $\mathrm{FD}_{\mathrm{PANN}}$, and $\mathrm{FD}_{\mathrm{PaSST}}$ are computed with VGGish, PANNs, and PaSST embeddings following AV-Benchmark (Cheng et al., 2025). $\mathrm{CLAP}_{\mathrm{LAION}}$ and $\mathrm{CLAP}_{\mathrm{MS}}$ use LAION-CLAP and MS-CLAP. FlatSounds (Li et al., 2026a) evaluates nine properties: attack time, decay rate, fundamental frequency (F0), spectral centroid, flux, and rolloff, temporal modulation, reverberation time (RT60), and direct-to-reverberant ratio (DRR). Each test case specifies how some of these properties should change, either between two videos that differ in one physical factor or across events within one video. A property's confidence is the fraction of seeds showing the expected trend, with each seed's vote weighted by the mean of its hit coverage and CLAP score. For a video pair, both scores take the lower value of the two videos. The overall physical confidence first averages the confidences of the properties tested in each case and then averages over cases. Because cases test different subsets of properties, it differs from the mean of the per-property columns in Table 4. The CLAP score is the LAION-CLAP similarity between the generated audio and the benchmark caption, rescaled from $[-1, 1]$ to $[0, 1]$, and its average is reported as $\mathrm{CLAP}_{\mathrm{LAION}}$ in Table 3.

Baselines. All baselines use their official implementations and default inference settings. Captioned baselines receive the original benchmark captions, since the upsampled JSON captions exceed the input length of their text encoders. For ThinkSound and PrismAudio, the captioned setting uses the caption and a chain-of-thought (CoT) derived from it. In the caption-free setting, we remove the caption and replace the CoT with the same instruction for every clip: "Generate the sound that matches the visible actions in the video, synchronized with them, with only subtle ambient background." This generic CoT carries no sample-specific information.

C.1 Training Data and Filtering

Sources and splits. We collect candidate clips from eight sources (Table 8) and keep clips whose sound can be inferred from the video using a two-stage pipeline. Stage 1 removes clips with speech, unrelated background music, static video, or little audio activity. Stage 2 filters the remaining clips by audio quality, audio-visual and caption agreement, temporal synchronization, signal integrity, and shot continuity. We keep the first eight seconds of longer videos. The final selection contains 90,901 eight-second clips that do not overlap across sources: 89,966 training clips (199.92 hours), used for both self-distillation and RL, and 935 validation clips (2.08 hours).

We exclude the official test splits of all source datasets. In particular, VGGSounder (Zverev et al., 2025) re-annotates the VGGSound test split, which is excluded from our VGGSound data. FoleyBench (Dixit et al., 2026) curates clips from FineVideo and LVBench on YouTube and V3C1

Table 8: Dataset composition across the two filtering stages. Selected total is the sum of the training and validation splits after the second stage.

Dataset	Candidates	First-stage pool	Train	Validation	Selected total
VGGSound (Chen et al., 2020)	197,956	106,555	25,022	271	25,293
AudioCaps (Kim et al., 2019)	100,794	44,266	9,181	94	9,275
AudioSet (Gemmeke et al., 2017)	284,678	189,639	43,441	433	43,874
Panda-70M (Chen et al., 2024)	278,359	1,612	211	1	212
YouTube-8M (Abu-El-Haija et al., 2016)	254,676	40,182	7,709	81	7,790
Greatest Hits (Owens et al., 2016)	3,852	3,836	937	9	946
Sound of Water (Bagad et al., 2026)	563	563	108	1	109
EPIC-SOUNDS (Huh et al., 2025)	23,265	23,085	3,357	45	3,402
Total	1,144,143	409,738	89,966	935	90,901

on Vimeo, and FlatSounds (Li et al., 2026a) consists of independently recorded indoor videos of controlled physical interactions. Because FoleyBench and several of our sources are drawn from YouTube, we also compared YouTube video IDs and found no shared videos. Neither our training nor validation split overlaps with these three benchmarks.

Stage 1: broad filtering. For each automatically collected clip, we generate separate video and audio captions and run FireRedAED audio-event detection (Xu et al., 2026) and visual instrument classification. A clip is kept only if its video is not static, it contains no speech, and it either shows an instrument or contains no background music. Clips must also be at least eight seconds long and have valid captions and detector outputs. A video caption is invalid if it is empty, reads "None," or matches a "no visual change" pattern. Detected music or speech triggers a caption-based review: any positive music ratio r_m triggers a background-music review for clips without a visible instrument, and any positive speech ratio r_s triggers a speech review for clips that show an instrument or pass the background-music check. A missing speech decision falls back to $r_s > 0$, a missing background-music decision falls back to $r_m > 0$ for clips without an instrument, and a missing static-video decision leads to rejection. For the manually verified Greatest Hits, Sound of Water, and EPIC-SOUNDS data, we only check caption validity.

For the five web-video sources (VGGSound, AudioCaps, AudioSet, Panda-70M, and YouTube-8M), we also remove near-silent clips. We decode each audio track as 16-kHz mono PCM with floating-point amplitudes, where an absolute amplitude of 1 corresponds to digital full scale. Let $x[n]$ denote waveform samples and $A_{\text{peak}} = \max_n |x[n]|$ their peak amplitude before normalization. When $A_{\text{peak}} > 10^{-5}$, we peak-normalize the waveform as $\tilde{x}[n] = 0.95x[n]/(A_{\text{peak}} + 10^{-5})$. We divide the normalized waveform into unwindowed 25-ms frames (400 samples) with a 10-ms hop (160 samples), retaining only complete frames. For frame k with samples $\tilde{x}_k[n]$, we compute its RMS amplitude and level as

$$r_k = \sqrt{\max\left(\frac{1}{400}\sum_{n=0}^{399}\tilde{x}_k[n]^2, 10^{-12}\right)}, \qquad L_k = 20\log_{10}(r_k/1.0)\,\text{dBFS}.$$

The dBFS reference amplitude is 1.0, and a frame is active if $L_k > -45$ dBFS. We retain a clip only if all three conditions hold:

$$A_{\text{peak}} \geq 0.007, \qquad L_{50} \geq -80\,\text{dBFS}, \qquad \rho_{\text{active}} \geq 0.2, \tag{23}$$

where L_{50} is the median frame level and ρ_{active} is the active-frame fraction. Because frame levels are measured after per-clip peak normalization, they reflect relative activity and silent intervals rather than original absolute loudness. Only the A_{peak} threshold uses the original amplitude, corresponding to approximately -43.1 dBFS peak.

Stage 2: quality and alignment filtering. Following MOVA (SII-OpenMOSS Team, 2026), we caption the video and audio of each clip separately and merge the two descriptions into one caption with an LLM. This caption is the per-sample caption C that is upsampled into the teacher's JSON input J (Appendix B.1), and it is also used for filtering. We then score each clip that passes Stage 1 with several models. Audiobox Aesthetics (Tjandra et al., 2025) scores production quality (PQ), content usefulness (CU), content enjoyment (CE), and production complexity (PC). AGAV-Rater (Cao et al., 2025) assesses audio quality, audio-visual consistency, and overall quality, and

Table 9: Second-stage filtering thresholds, applied in the listed order after checking data validity and score availability. P_{25} and P_{15} denote the 25th- and 15th-percentile cutoffs. Additional conditions for clips with visible instruments are described in the text.

Measure	Keep condition	Cutoff type
Silence fraction	< 0.80	Fixed
Audiobox PQ	> 5.0	Fixed
Audiobox CU	> 4.5	Fixed
Audiobox CE	> 2.5	Fixed
AGAV audio quality	≥ 11.33594	P_{25}
AGAV audio-visual consistency	≥ 19.14063	P_{25}
AGAV overall quality	≥ 13.06250	P_{25}
Signal health	≥ 0.92268	P_{15}
Caption integrity	≥ 1.0	P_{15}
PE-AV audio-video	≥ 0.06058	P_{15}
PE-AV audio-caption	≥ 0.03484	P_{15}
PE-AV video-caption	≥ 0.10426	P_{15}
PE-AV audio-visual-caption	≥ 0.04534	P_{15}
JavisScore (when required)	≥ 0.19383	P_{15}
Interior TransNetV2 event probability	≤ 0.5 for all events	Fixed

PE-AV (Vyas et al., 2026) measures agreement across audio, video, and captions. JavisScore (Liu et al., 2026b) assesses temporal audio-visual alignment by aggregating ImageBind similarities over corresponding temporal windows, emphasizing poorly matched moments. It uses the same ImageBind representations as our IB evaluation metric. TransNetV2 (Souček & Lokoč, 2024) detects shot changes.

Table 9 lists the thresholds in the order they are applied. Silence, Audiobox scores, and shot changes use fixed cutoffs, and the remaining scores use the 25th or 15th percentile of their distributions. Because audio-quality scores tend to favor music, clips labeled as showing an instrument must also satisfy $\mathrm{CE} \geq 4.5$ and $\mathrm{PC} \geq 3.5$. These criteria do not apply when the instrument label is unknown.

C.2 System Prompt Refinement

We refine the system prompt on 300 selected videos from Kling-Audio-Eval (Wang et al., 2025a), which are distinct from our training data. Starting from the original Cosmos schema, a GPT-5.6 agent iteratively revises the instructions. Only the prompt changes, while the model weights, visual inputs, and generation procedure stay fixed. Each candidate prompt is scored in two ways: AV-Benchmark evaluates the audio generated with it, and coverage tests evaluate the sound-relevant content in the reasoner's decoded output.

Coverage tests. We adapt the Multimodal Context Graph (MCG) representation of EgoAVU (Seth et al., 2026), which connects visible objects and actions with audible sounds. For each of the 300 videos, we construct reference atoms and MCG relations from its dataset captions (Figure 3). Atoms describe entities, actions, interactions, environments, sounds, and acoustic properties. Relations connect them into physically and temporally grounded events, such as an action producing a sound or a sound having an acoustic attribute. An LLM judge extracts atoms and relations from the reasoner's decoded JSON output and matches them to the references. *Atom coverage* is the fraction of reference atoms recovered, and *MCG-relation coverage* is the fraction of reference relations recovered. Decoding is used only for this diagnostic, while the generator conditions directly on $\boldsymbol{h}_S$.

Selection and result. We select the final *Audio-tuned* prompt using both AV-Benchmark and coverage, and keep it fixed for all training and inference. Table 10 compares it with the original Cosmos prompt under the same frozen model and ViT conditioning. Audio-tuned lowers all three FD scores and DeSync and raises IB, LAION-CLAP, and both coverage scores, while MS-CLAP is nearly unchanged. It therefore provides a stronger starting point for reasoner adaptation.

C.3 Reliability of Audio-Visual Synchronization Metrics

What DeSync measures. DeSync predicts a single audio-video offset with Synchformer (Iashin et al., 2024; Cheng et al., 2025) and reports its magnitude. A global offset captures whether the

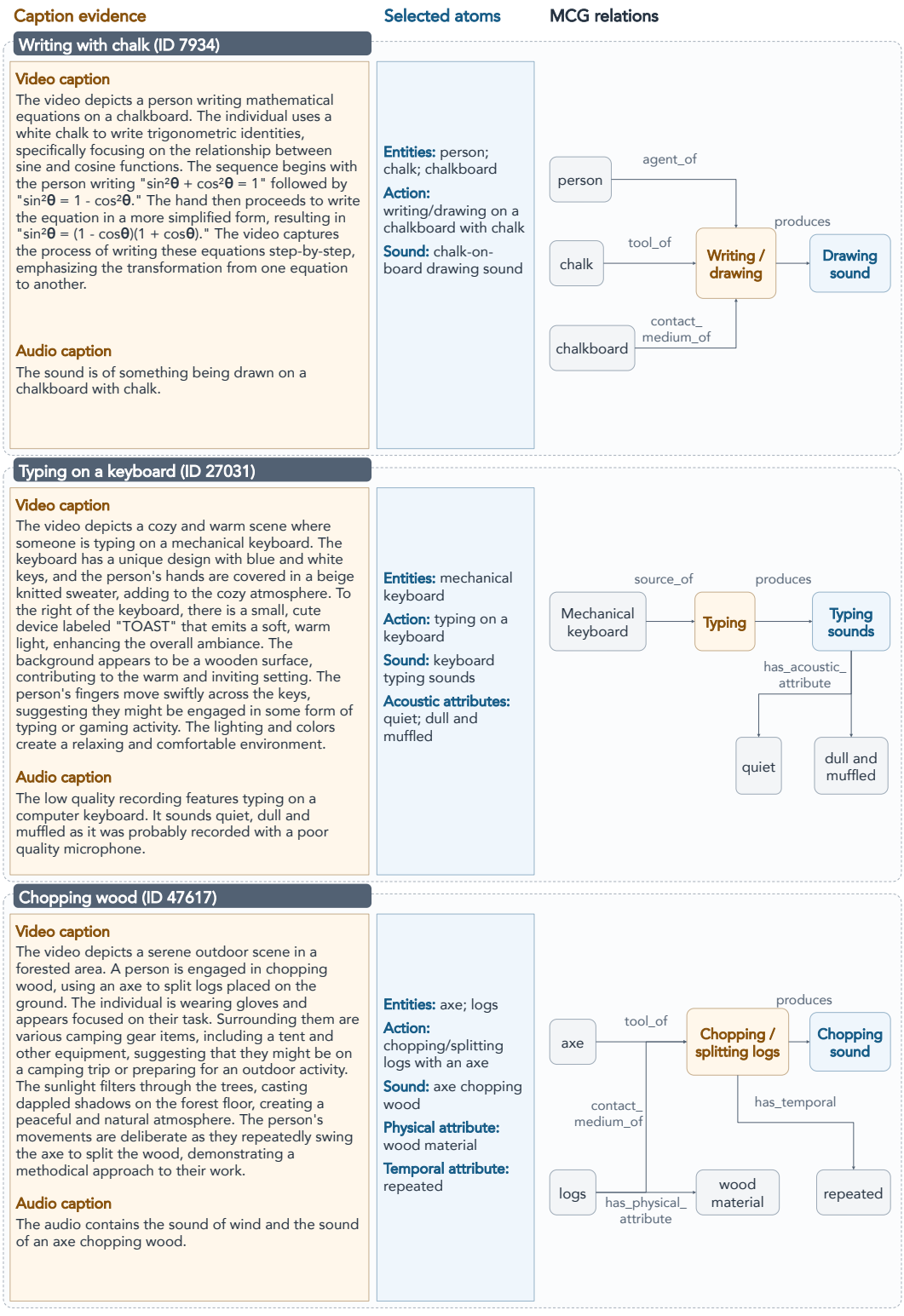


Figure 3: **Examples of reference atoms and MCG relations.** For three Kling-Audio-Eval videos, we show the original video and audio captions, selected sound-relevant atoms, and relations connecting entities, actions, sounds, and their attributes. These references are used to assess the coverage of decoded reasoner outputs during system prompt refinement.

Table 10: System prompt choice for zero-shot V2A on 300 selected Kling-Audio-Eval videos. Both rows use ViT conditioning and the same pre-trained Cosmos3 model. Atoms and Relations denote atom and MCG-relation coverage of decoded reasoner outputs, respectively.

System prompt	FD_{VGG} ↓	FD_{PANN} ↓	FD_{PaSST} ↓	IB ↑	DeSync ↓	$CLAP_{LAION}$ ↑	$CLAP_{MS}$ ↑	Atoms ↑	Relations ↑
Cosmos original	9.671	30.744	458.541	0.221	0.877	0.191	**0.353**	0.501	0.383
Audio-tuned	**3.991**	**27.283**	**369.329**	**0.249**	**0.779**	**0.204**	0.350	**0.615**	**0.510**

Table 11: Human synchronization preferences and automatic scores for three configurations on 28 FoleyBench clips. Within each clip, the three configurations are re-ranked and scored 1, 0.5, and 0 for first, second, and third place. Human preference is the mean score across clips.

Configuration	Human preference ↑	FATE ↑	JavisScore ↑	BTS ↑	DeSync ↓	VA-Judger B ↑
Cosmos3 (caption)	**0.571**	0.137	0.164	-4.825	0.718	7.036
MMAudio	0.518	**0.157**	**0.254**	$\mathbf{-3.289}$	**0.346**	**7.095**
Cosmos3 (zero-shot)	0.411	0.132	0.149	-3.993	0.757	6.976

soundtrack is shifted as a whole, but not whether individual sounds are missing, reordered, or attached to the wrong action. An independent audit supports this distinction (Sharma & Tapadiya, 2026): DeSync tracks controlled time shifts well (Kendall $\tau = 0.84$), but tracks fragment shuffling less well ($\tau = 0.27$) and agrees weakly with PEAVS (Goncalves et al., 2024) ($\tau = 0.07$), an automatic proxy for human ratings. We therefore check DeSync against human judgments and complement it with event-level timing metrics.

Comparison with human judgments. We use 28 FoleyBench clips with clear physical events and limited background noise, and compare three configurations: caption-conditioned Cosmos3, MMAudio, and zero-shot Cosmos3 with the Audio-tuned prompt. Human preference scores are recomputed from the original per-clip synchronization rankings, restricted to these three configurations. We compare them with DeSync and four other automatic scores. FATE (Guan et al., 2026) averages timestamp-aligned similarities between frame-level audio and video embeddings. JavisScore (Liu et al., 2026b) aggregates local ImageBind similarities, emphasizing poorly matched moments. BTS (Qian et al., 2026) is an omni-LLM adapted to predict human preferences. VA-Judger B (Huang et al., 2026) is the audio-visual consistency score of a trained multimodal judge, which covers more than timing. DeSync is averaged over the first and last 4.8-second segments, and BTS uses a five-second input.

Humans rank caption-conditioned Cosmos3 highest for synchronization, whereas all five automatic scores favor MMAudio (Table 11). With 28 clips and no significance test, this comparison only shows that automatic rankings can disagree with human judgments. The larger listening study likewise finds weak agreement between DeSync and synchronization preferences (Appendix D.5, Table 17).

Event-level timing on FlatSounds. FlatSounds provides manually verified timestamps for individual sound-producing events, so timing can be checked event by event (Li et al., 2026a). HitCov is the fraction of annotated events matched by a generated onset, Timing Error is the mean absolute error of matched onsets in milliseconds, and Perfect Align is the fraction of generations that match every annotated event. Higher HitCov and Perfect Align and lower Timing Error are better. These metrics measure whether each sound is present and when it occurs, which a global offset cannot capture. Because they require clearly identifiable events, they complement rather than replace DeSync on arbitrary videos. They are also separate from the physical-correctness metrics in Table 4, where temporal modulation measures amplitude modulation rather than event timing.

C.4 Human Evaluation Protocol

We describe the protocol for the study in Section 4.4. Detailed results are reported in Appendix D.5.

Clips and systems. We select 50 clips with clearly audible on-screen events, comprising 19 FoleyBench clips, 16 VGGSounder clips, and 15 FlatSounds source clips. The selection covers sustained

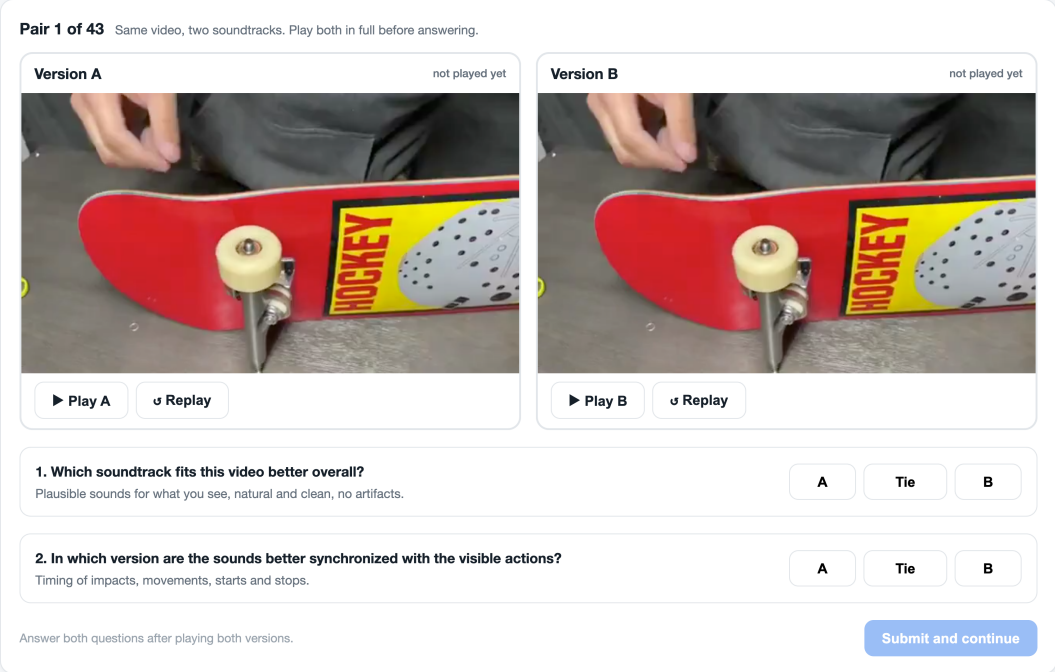


Figure 4: Pairwise human evaluation interface. Raters view the same video with two soundtracks and judge overall fit and synchronization separately, choosing A, B, or Tie for each question. Both versions must be played in full before a response can be submitted.

sounds and discrete impacts. Each clip is paired with nine soundtracks: its ground-truth recording, the outputs of the six external baselines in Table 6, the caption-conditioned Cosmos3 teacher, and the final caption-free TacitV2A model after RL. We omit ThinkSound and AudioX to keep the complete pairwise comparison tractable. The baselines receive captions to test our model against their caption-conditioned generation capabilities. This study therefore evaluates the final systems under their stated conditioning settings, rather than isolating the effect of removing captions.

Pairwise protocol. We evaluate all 36 soundtrack pairs for each clip, yielding 1,800 distinct pairs. Three different raters evaluate each pair, giving 5,400 pairwise evaluations from 139 participants. Each evaluation contains two judgments: which soundtrack fits the video better overall, and which is better synchronized with its visible actions. Raters view the same video with the two soundtracks, must play both to completion, and choose A, B, or Tie for each question. The overall-fit instructions ask raters to consider plausibility, naturalness, clean sound, and the absence of artifacts, while the synchronization instructions concern the timing of impacts, movements, and starts and stops (Figure 4). Raters do not see the captions. All soundtracks are loudness-normalized to -23 LUFS and trimmed to a common duration within each clip, and the left-right order is randomized to reduce presentation cues. Each session also includes three attention checks that pair the ground-truth recording with a soundtrack from a different clip. Some participants left before finishing their sessions, so the number of completed checks (407) is not a multiple of three. Raters answered 406 of these 407 checks correctly.

Ratings and uncertainty. We fit a separate Bradley-Terry model (Bradley & Terry, 1952) for each question, counting ties as half a win for each system. The fitted strengths are expressed on an Elo scale, where a 100-point difference corresponds to an expected win rate of approximately 0.64. Using a normal approximation, we report symmetric 95% confidence intervals as the fitted rating ± 1.96 bootstrap standard errors, estimated from 10,000 clip-level resamples (Table 16). Win rates in the table are empirical comparison outcomes, distinct from the fitted ratings. Ground truth is included as a reference and excluded when highlighting the best generated result. After mapping randomized A/B choices to system identities, unanimous agreement is 45.3% for overall fit and 44.0% for synchronization, with ties in 6.9% and 8.3% of individual responses, respectively.

Table 12: Additional ablations initialized from zero-shot Cosmos3 on FoleyBench. Direct RL applies NFT to the generator without reasoner distillation, while LoRA fine-tuning trains generator adapters by flow matching on paired recordings. Both keep the zero-shot visual reasoner fixed. We include the distilled results from Table 1 for comparison. Bold marks the best result.

Method	FD_{VGG} ↓	FD_{PANN} ↓	FD_{PaSST} ↓	KL ↓	IB ↑	DeSync ↓	$CLAP_{LAION}$ ↑	$CLAP_{MS}$ ↑
Cosmos3 (zero-shot)	5.016	17.56	199.2	2.606	0.216	**0.793**	0.205	0.382
Direct RL	5.071	17.43	200.6	2.621	0.214	0.799	0.206	0.380
Direct generator LoRA	4.987	17.69	198.6	2.614	0.215	0.797	0.205	0.381
TacitV2A (distilled)	**2.536**	**12.45**	**148.8**	**2.361**	**0.266**	0.798	**0.284**	**0.466**

Comparison with automatic metrics. For each question, we compute the Spearman rank correlation between the human ratings in Table 16 and the automatic scores of each benchmark across the eight generated systems (Table 17). The automatic scores come from the full benchmark runs in Tables 1, 2, and 3, whereas the human ratings are fitted on the selected clips from all three benchmarks. For the metrics shared by FoleyBench and VGGSounder, we report the arithmetic mean of the two correlations, giving each benchmark equal weight. FlatSounds is reported separately. We reverse lower-is-better metrics so that positive correlation indicates agreement.

D Additional Experiments

D.1 Additional Ablations

We test whether adapting the generator directly can replace reasoner self-distillation. Both controls start from caption-free Cosmos3, keep its visual reasoner fixed, and train only the generator. Direct RL applies DiffusionNFT (Zheng et al., 2026) with the same method and configuration as our RL stage (Appendix B.4), differing only in its starting point. LoRA fine-tuning uses the same generator adapters but trains them by flow matching on paired recordings instead of the reward, for three epochs as in reasoner training.

We find that direct RL and generator LoRA fine-tuning lead to small, mixed changes relative to zero-shot Cosmos3 (Table 12). Neither gives a clear improvement under the tested setup. For direct RL, we also observe little improvement in the training reward. We conjecture that weak visual conditioning produces candidates with similar errors, so their reward differences provide limited guidance for correcting them. Even with supervision from reference recordings, generator LoRA fine-tuning makes little progress. This suggests that the difficulty is not specific to reward learning: further training of the generator does little to compensate for the reasoner's weak visual guidance in these experiments.

Training the reasoner with flow matching alone is also weaker than self-distillation on all reported metrics (Table 5). This supports the value of teacher supervision beyond training the reasoner on paired audio alone. With caption-guided self-distillation, we improve generation while keeping the generator frozen. Generator RL then brings further gains (Table 1). The generator therefore still has room to improve when its visual guidance is stronger. These results support our choice to first teach the reasoner to infer sound-relevant cues from video, then refine the generated audio against paired recordings.

D.2 Caption-Free Comparison on FlatSounds

We extend Table 3 with results for baselines without per-sample captions and additionally include Frieren (Wang et al., 2024b), a model designed for video-only audio generation. For models that also support captions, we use the caption-free settings described in Section 4.1.

For baselines evaluated with and without captions, we find that removing captions lowers matched-hit timing error in most paired comparisons (Tables 3 and 13). Event coverage and Perfect Align also improve for several models, although both decline for PrismAudio, Omni2Sound, and Cosmos3, and coverage declines slightly for AudioX. Lower timing error therefore does not always mean that a model follows visible events more completely. These timing gains come with lower CLAP in every paired comparison and lower average physical confidence for most baselines (Table 4). This

Table 13: Caption-free comparison on FlatSounds, extending Table 3. No model receives a per-sample caption. Cosmos3 (zero-shot) is the starting point of TacitV2A, using ViT features and the shared Audio-tuned prompt. Confidence is the overall physical confidence, HitCov and PerfAlign denote hit coverage and perfect-alignment rate, and Timing is the mean matched-hit error in milliseconds.

Method	Confidence ↑	HitCov ↑	Timing ↓	PerfAlign ↑	$\text{CLAP}_{\text{LAION}}$ ↑
FoleyCrafter	0.208	0.834	46.3	0.607	0.550
MMAudio	0.220	0.758	56.7	0.515	0.616
Hunyuan-Foley	0.288	0.915	43.9	0.705	0.608
ThinkSound	0.223	0.750	51.5	0.511	0.553
PrismAudio	0.282	0.919	47.4	0.717	0.551
AudioX	0.124	0.538	45.7	0.338	0.533
Audio-Omni	0.126	0.539	46.7	0.353	0.563
Omni2Sound	0.269	0.850	57.9	0.659	0.600
Frieren	0.162	0.603	48.0	0.392	0.543
Cosmos3 (zero-shot)	0.287	0.928	37.1	0.741	0.599
TacitV2A (distilled)	0.322	0.972	34.3	0.790	0.607
+RL	**0.347**	**0.978**	**33.1**	**0.795**	**0.628**

contrast is consistent with the findings of FlatSounds (Li et al., 2026a) and suggests that the semantic guidance supplied by captions is not always combined effectively with the video's event structure. Captions can help specify what should be heard without improving when it is heard, while removing them can improve timing without preserving their semantic and physical guidance.

Among the caption-free models, TacitV2A achieves the highest reported physical confidence, event coverage, Perfect Align, and CLAP, together with the lowest timing error (Table 13). Self-distillation accounts for most of these event-level gains over zero-shot Cosmos3, and RL adds smaller improvements. We find this combination informative because its timing advantage is accompanied by more complete event recovery and stronger physical and semantic agreement. The paired baseline comparisons show that removing captions alone does not consistently improve these dimensions together. Our results support learning visual guidance that recovers the sound-relevant cues supplied by captions while following the events in the video.

D.3 Caption-then-Generate Baseline

The caption-conditioned teacher uses oracle captions, which are not available for arbitrary videos. A practical caption-conditioned system must instead caption each video automatically before generating audio. We therefore evaluate such a caption-then-generate cascade as an additional baseline, built on the same model so that it differs from the teacher only in its caption source. Zero-shot Cosmos3 first decodes a JSON caption from each FlatSounds video using the Audio-tuned system prompt. We then pass this caption through the upsampler U used for the teacher's captions (Appendix B.1), and the caption-conditioned model generates audio from the resulting JSON caption.

The cascade falls short of the teacher with oracle captions (Table 14), indicating that caption-conditioned generation is sensitive to caption quality. TacitV2A needs no captioning stage. After self-distillation, it trails the cascade slightly in average physical confidence, and after RL it outperforms the cascade and approaches the oracle teacher. The two approaches also differ in which properties they recover. The cascade retains the spectral properties that captions describe well, whereas TacitV2A is stronger on temporal modulation, RT60, and DRR even before RL. Zero-shot Cosmos3 already exceeds both caption-conditioned variants on RT60 and DRR, suggesting that visual conditioning conveys room acoustics that captions leave implicit. The gain in temporal modulation instead emerges with self-distillation. Learning visual guidance directly thus removes the dependence on caption quality at inference while retaining such cues.

D.4 What Does the Visual Reasoner Learn?

We ask whether training the reasoner through audio generation also improves its ability to describe sound-producing events. To test this, we compare zero-shot Cosmos3 with TacitV2A by disabling or enabling the learned reasoner LoRA adapters. Both receive 16 uniformly sampled frames and the fixed Audio-tuned prompt, without captions, on 300 selected FoleyBench videos. We decode



Figure 5: **Audio descriptions decoded from the visual reasoner.** Given the same video and Audio-tuned system prompt, we compare outputs from zero-shot Cosmos3 and TacitV2A. These examples illustrate differences in descriptions of sound presence, action-linked sounds, acoustic texture, and event progression. The descriptions are decoded for analysis and do not condition audio generation.

Table 14: Per-property physical confidence on FlatSounds for Cosmos3 with different caption sources and for TacitV2A. The cascade baseline conditions Cosmos3 on automatically generated captions, and the teacher on oracle captions. Columns follow Table 4.

Method	Caption	Attack ↑	Decay ↑	F0 ↑	Centroid ↑	Flux ↑	Rolloff ↑	Temporal ↑	RT60 ↑	DRR ↑	Avg. ↑
Cosmos3 (zero-shot)	–	0.231	0.296	0.244	0.329	0.320	0.283	0.300	0.275	0.351	0.287
Cosmos3 (cascade)	Automatic	**0.235**	0.336	0.303	0.478	**0.421**	**0.481**	0.295	0.208	0.206	0.329
Cosmos3 (teacher)	Oracle	0.229	**0.382**	0.324	**0.481**	0.408	0.473	0.301	0.204	0.239	**0.349**
TacitV2A (distilled)	–	0.203	0.339	0.282	0.387	0.356	0.297	0.387	0.282	0.381	0.322
+RL	–	0.229	0.352	**0.353**	0.412	0.362	0.293	**0.411**	**0.295**	**0.398**	0.347

greedily with a 4,096-token limit. These descriptions are used only for this diagnostic and never enter audio generation.

Using the coverage protocol of Appendix C.2, Claude Opus 4.6 evaluates all texts against the same reference atoms and relations derived from the original dataset captions. References include explicit and entailed content, but only explicit statements receive credit. This tests whether the reasoner describes sounds, rather than only actions from which sounds could be inferred. We also score the original captions and their Opus-expanded versions used to condition the teacher. The original caption can score below its expansion because the latter makes more implicit reference content explicit. As a control, we pair the descriptions with references from other clips (Table 15).

After self-distillation, the reasoner recovers more reference content across every measure. Zero-shot Cosmos3 often describes visible objects while treating the reference event as silent. TacitV2A more often connects those objects to their actions and expected sounds, as reflected in both cue

Table 15: Sound-relevant content decoded from the visual reasoner before and after self-distillation. We measure reference coverage on 300 selected FoleyBench videos, with caption inputs and shuffled references for comparison. Caption expansion can increase coverage by making implicit reference content explicit. Source measures coverage of sound-emitting entities, and Present measures whether the reference event is described as audible. Timing measures temporal cue coverage, not timestamp accuracy. Atoms and Relations summarize atom and MCG-relation coverage. Higher is better, and bold highlights the better result between the two reasoners.

Text evaluated	Source	Action	Sound	Material	Timing	Present	Atoms	Relations
Original caption	0.87	0.98	0.66	0.95	0.92	0.58	0.90	0.80
Expanded caption	1.00	1.00	1.00	0.99	1.00	0.97	1.00	0.99
Shuffled reference	0.02	0.01	0.02	0.02	0.01	0.03	0.02	0.01
Cosmos3 zero-shot	0.46	0.30	0.31	0.56	0.16	0.43	0.47	0.26
TacitV2A	**0.65**	**0.53**	**0.56**	**0.68**	**0.45**	**0.70**	**0.65**	**0.48**

Table 16: Detailed human evaluation. Ratings are Elo-scaled Bradley-Terry strengths with symmetric 95% confidence intervals. Win is the fraction of a system's comparisons it wins. TacitV2A Win is the fraction TacitV2A wins against that row's system. Both win rates count ties as half. All generated baselines receive captions, while TacitV2A is caption-free. Bold highlights the best generated result in the Rating and Win columns.

	Overall fit			Synchronization		
Method	Rating	Win	TacitV2A Win	Rating	Win	TacitV2A Win
Ground Truth	1198 ± 16	0.83	0.28	1200 ± 19	0.83	0.38
Cosmos3	$\mathbf{1136 \pm 19}$	**0.77**	0.32	1068 ± 21	0.70	0.62
Omni2Sound	980 ± 18	0.59	0.67	782 ± 26	0.35	0.83
Hunyuan-Foley	919 ± 22	0.51	0.71	976 ± 19	0.59	0.75
MMAudio	858 ± 26	0.44	0.68	903 ± 23	0.50	0.80
PrismAudio	776 ± 27	0.34	0.78	695 ± 23	0.25	0.88
Audio-Omni	673 ± 26	0.23	0.90	813 ± 19	0.38	0.86
FoleyCrafter	586 ± 29	0.15	0.91	594 ± 28	0.15	0.94
TacitV2A	1033 ± 20	0.65	–	$\mathbf{1123 \pm 22}$	**0.76**	–

and relation coverage. Figure 5 illustrates these differences in the audio descriptions decoded from the reasoner. Low coverage with shuffled references supports that these descriptions capture content specific to each clip. Together, these findings suggest that the reasoner becomes better at interpreting visible objects and actions in terms of the sounds they produce.

We find this result informative because the reasoner receives no text-generation loss. Its adapters learn through teacher audio-velocity predictions and flow matching to real recordings, while the generator stays frozen. This audio supervision also improves the sound-relevant content expressed through the language head, providing complementary evidence for the reasoner's role in the generation gains. The same reasoner is retained in our final model, since subsequent RL updates only the generator.

The descriptions remain incomplete, especially for temporal cues, and coverage does not penalize unsupported additions. TacitV2A can still infer sounds absent from the reference, such as knife taps in an industrial mixer clip. This diagnostic shows improved recovery of relevant content, while event timing in generated audio is assessed separately in Table 3.

D.5 Additional Human Evaluation Results

Table 16 reports the full ratings, confidence intervals, and win rates for the study in Section 4.4. The evaluation protocol and scoring procedure are described in Appendix C.4.

Overall fit and synchronization. Ground truth ranks first on both questions. Among generated systems, the caption-conditioned teacher is preferred overall and TacitV2A is preferred for synchronization. Both outperform the other caption-conditioned systems on each question. These rankings also hold when we fit ratings separately for each benchmark. The teacher and TacitV2A show the

Table 17: Spearman rank correlation between automatic metrics and the human ratings in Table 16 across eight generated systems. For FoleyBench and VGGSounder, we average the correlations computed with each benchmark's scores. Lower-is-better metrics are reversed, so positive values indicate agreement. Bold marks the largest correlation in each row within each benchmark group.

Judgment	FoleyBench + VGGSounder (mean)								FlatSounds		
	FD_{VGG}	FD_{PANN}	FD_{PaSST}	KL	IB	DeSync	$CLAP_{LAION}$	$CLAP_{MS}$	Confidence	HitCov	PerfAlign
Overall	**0.52**	0.30	0.49	0.44	0.42	0.26	0.29	0.08	**0.86**	0.73	0.71
Sync	0.48	0.40	**0.64**	0.37	0.38	0.14	0.40	0.11	**0.69**	0.51	0.52

same difference in their direct comparisons (Table 16): listeners prefer TacitV2A's timing but the teacher's soundtrack overall. The overall question also asks whether sounds are plausible for the scene, natural, and free of artifacts, which this study does not score separately.

Agreement with automatic metrics. Table 17 summarizes how closely the automatic metrics follow human preferences, using the comparison procedure in Appendix C.4.

On FlatSounds, physical confidence closely follows overall preference. HitCov and PerfAlign also agree with both human judgments, although less strongly. Their agreement with overall preference suggests that covering and aligning the visible events matters to how well a soundtrack fits the scene, beyond the separate question about synchronization.

Across FoleyBench and VGGSounder, DeSync has weak average agreement with synchronization preferences, while FD_{PaSST} agrees more closely. This suggests that a lower estimated audio-video offset does not necessarily correspond to better perceived synchronization. Together with the FlatSounds results, this supports checking whether individual events are covered and aligned, alongside the global offset. The mean correlations summarize system rankings and can hide differences between benchmarks. They complement listening judgments rather than replace them.

E AUDIO-TUNED SYSTEM PROMPT

The following box reproduces the complete *Audio-tuned* system prompt template in JSON format. The semantic instructions are shared across videos, while duration, frame rate, resolution, and aspect ratio are filled automatically from the generation settings before tokenization. The template keeps the fields of the original Cosmos schema, including person attributes such as gender, age, and skin tone. The instructions ask the reasoner to infer these attributes only when they are visible and relevant, and to leave them empty otherwise. Its refinement procedure and comparison with the original Cosmos prompt are reported in Appendix C.2.

Audio-tuned System Prompt

```
{
  "subjects": [
    {
      "description": "Name the dominant visibly active sound source
  first, using a specific familiar category. Include secondary
  sources only when visibly active; do not give silent background
  objects equal acoustic priority.",
      "appearance_details": "Describe only visible properties that
  change the expected sound, such as material, contact surface,
  rigidity, hollowness, wetness, tension, moving parts, or vocal
  posture. Use specific material or mechanism detail only when
  visually clear; otherwise keep it broad.",
      "relationship": "Describe spatial and physical relationships
  among sound-producing objects: what touches, strikes, scrapes,
  rolls, slides, splashes, vibrates, accelerates, opens, closes, or
  moves relative to what.",
```

Audio-tuned System Prompt (continued)

```
    "location": "Describe where each subject is in the frame and
scene, including foreground/background placement and likely
distance from camera when visible.",
    "relative_size": "State size and sound-producing scale for
active sources; use tiny clicks, hand-scale impacts, body motion,
vehicle engines, or environmental sounds only when supported.",
    "orientation": "Infer object orientation, motion direction,
trajectory, and contact angle when visually relevant to sound
timing or timbre.",
    "pose": "For people, animals, tools, vehicles, or articulated
objects, describe pose/configuration only when it affects visible
physical action or sound generation.",
    "action": "Begin each visibly active source with a short
ordinary source-event phrase, such as footsteps on gravel or metal
striking wood, only when supported by the video. Then preserve
visible onset, offset, and repetition and name one characteristic
sound; omit speculation and long attribute lists.",
    "state_changes": "Across the full clip, describe the clearest
visible change of the dominant active source or its contact target,
including start, stop, or repetition. Keep source identity
consistent. Do not substitute camera motion, a passive observer, or
an unrelated stationary foreground object for the changing source.
If no source-linked change is visible, say so briefly.",
    "clothing": "Infer clothing only if visible and acoustically
relevant, such as fabric rustle, shoe contact, protective gear, or
accessories; otherwise leave empty.",
    "expression": "Infer expression only if visible and relevant
to vocal or body-action timing. For animals, include open-mouth,
alert, aggressive, distressed, excited, or calling posture when
visible. Do not add human speech unless visibly justified.",
    "gender": "Infer only if visually clear and relevant;
otherwise leave empty.",
    "age": "Infer only if visually clear and relevant; otherwise
leave empty.",
    "skin_tone_and_texture": "Infer only if visible and relevant;
otherwise leave empty.",
    "facial_features": "Infer only if visible and relevant;
otherwise leave empty.",
    "number_of_subjects": "Infer the count of this subject type
when visible or acoustically relevant.",
    "number_of_arms": "Infer only for visible humans or characters
when relevant to action timing; otherwise use 0 or leave empty.",
    "number_of_legs": "Infer only for visible humans, animals, or
characters when relevant to gait/contact timing; otherwise use 0 or
leave empty."
  }
],
"background_setting": "Infer the visually grounded environment and
acoustic space: indoor/outdoor, room size, hard or soft surfaces,
open air, water, road, kitchen, workshop, sports field, crowd,
traffic, machinery, reverberation, absorption, and persistent
ambient sound sources visible in the scene.",
"lighting": {
  "conditions": "Describe lighting conditions only as visual
evidence for environment, time, visibility, surface material, or
contact interpretation.",
  "direction": "Describe light direction only when it clarifies
object geometry, contact, surface shape, shadow position, or
motion.",
```

Audio-tuned System Prompt (continued)

```
  "shadows": "Describe shadows when they help infer spatial
contact, grounding, collision, distance, motion, or hidden object
relation.",
  "illumination_effect": "Describe visible highlights,
reflections, glare, translucency, wetness, or specular cues that
reveal material and likely sound quality."
},
"aesthetics": {
  "composition": "Describe framing and arrangement of acoustically
important objects, including repeated patterns, contact points, and
whether the action is centered, close, distant, occluded, or
partially offscreen.",
  "color_scheme": "Describe colors only when useful for
identifying materials, liquids, surfaces, environment, or object
semantics.",
  "mood_atmosphere": "Describe the realistic scene atmosphere
implied by visual evidence, such as quiet indoor demonstration,
busy street, workshop, vehicle motion, sports action, rain/water
scene, or tense impact sequence.",
  "patterns": "Describe repeated visual patterns that imply
repeated sound rhythms, such as footsteps, hammer hits, wheel
rotation, dripping, bouncing, clapping, glass taps, repeated barks
or animal calls, mouth-opening cycles, mechanical cycles, or
traffic flow."
},
"cinematography": {
  "camera_motion": "Separate camera motion from source motion;
camera shake, panning, or tracking alone must not create a sound
event.",
  "framing": "Describe shot scale and field of view, especially
whether sound-producing contacts are close-up, mid-shot, wide,
offscreen, or partially cropped.",
  "camera_angle": "Describe viewpoint and angle when it affects
spatial interpretation, occlusion, contact timing, or likely
acoustic perspective.",
  "depth_of_field": "Describe focus depth only when it affects
which objects/actions should dominate the audio.",
  "focus": "Describe what visual objects or contact points are in
focus and therefore likely primary acoustic evidence.",
  "lens_focal_length": "Infer lens perspective only if useful for
spatial distance, scale, speed, or acoustic perspective."
},
"style_medium": "Infer the medium as live-action, screen
recording, animation, CGI, game footage, surveillance, phone video,
dashcam, first-person camera, or other visible style. Audio should
match the visible medium but remain physically plausible.",
"artistic_style": "Infer visual style only when it affects
expected sound realism. Prefer realistic non-musical Foley unless
the visual style clearly implies stylization.",
"context": "In one concise sentence, link the dominant visible
source, observed action, contact or emission, relevant target or
medium, and familiar sound produced. Mention environment only when
it affects sound.",
"actions": [
  {
    "time": "Give an approximate onset-to-offset interval for the
dominant visibly active source's contact, motion, or emission;
include only the active phase visible in the clip.",
    "description": "For each clearly visible source event, write
one short natural sentence naming the active source, observed
action/contact or emission, and one familiar sound class supported
```

Audio-tuned System Prompt (continued)

```
  by those cues. Use a broader familiar class when fine timbre is
  uncertain; use visual evidence only and never source audio."
    }
  ],
  "text_and_signage_elements": [
    {
      "text": "Transcribe visible text, labels, captions, signs,
  overlays, or category words when present.",
      "category": "Classify each element as physical_in_scene,
  scene_sign, ui_text, vehicle_graphic, logo, or label; use the
  closest supported category.",
      "appearance": "Describe visible font, color, size, and style
  only when useful for identifying the element.",
      "spatial_temporal": "Describe where the text appears and when
  it is visible in the clip.",
      "context": "Use visible text only as supporting evidence for
  object identity or a likely sound class; text alone must not create
  an unrelated or offscreen sound."
    }
  ],
  "segments": [
    {
      "segment_index": "Use a zero-based integer segment index.",
      "time_range": "Split the clip when visible acoustic events,
  materials, object states, camera perspective, environment, or
  motion intensity change.",
      "description": "Describe the compact visual-physical state of
  this segment.",
      "key_changes": "Record only changes that alter sound: source
  action/contact, onset or offset, repetition, speed or force,
  emission posture, or acoustic distance. Keep the same
  source-to-sound identity across adjacent segments.",
      "camera": "Describe camera behavior only if it affects audio
  interpretation, synchronization, distance, or spatial perspective."
    }
  ],
  "transitions": [],
  "temporal_caption": "Write a concise chronological caption of
  visible source-action-sound events. Preserve source identity and
  align contacts or emissions, repetition, and intensity changes to
  visible timing; omit decorative scene details.",
  "audio_description": "Generate realistic audio from the visible
  event likely to dominate sound. Return complete JSON;
  audio_description is one string linking the active source and
  relevant visible target or medium through its action/contact or
  emission to the familiar sound produced. Keep uncertainty broad;
  never replace supported activity with silence or generic ambience.
  Add only visible source-bound secondary events. Omit unsupported
  music, speech, and offscreen events.",
  "resolution": {
    "H": "Infer or record the video height in pixels when
  available.",
    "W": "Infer or record the video width in pixels when available."
  },
  "aspect_ratio": "Infer or record the visible aspect ratio, such as
  16,9, 9,16, 4,3, 3,4, or 1,1.",
  "duration": "Infer or record the clip duration in seconds.",
  "fps": "Infer or record the video frame rate."
}
```