TAP-FOLEY: INFERENCE-TIME TARGET-ANCHORED PROJECTION FOR COUNTERFACTUAL VIDEO FOLEY GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Counterfactual Video Foley Generation replaces the *source* sound identity that a silent video implies with the *target* sound identity that a text prompt describes, while preserving the timing of visible events. Video-text-to-audio (VT2A) models struggle with this task because video conditioning carries not only the timing but also the source identity, which conflicts with the target prompt. Existing training-based methods suppress the source identity by adding a visual adapter or by training on video–text conflicts, yet it remains partially audible. Inference-time guidance methods suppress it more fully, by negative-source prompting with a fixed weight or by dropping video conditioning at an intermediate step, yet both degrade temporal alignment. We propose **TAP-Foley**, an inference-time guidance method that adaptively suppresses the source identity while retaining video conditioning throughout generation. At each step, it measures how strongly the video steers the prediction toward the source identity beyond what the target prompt alone would produce. It removes this excess along a single source–target axis with a scaled closed-form projection onto a target-anchored constraint, so the conflict between the two modalities is mitigated without dropping either. To reliably evaluate sound identity replacement and temporal alignment, we introduce VGGSound-Counterfactual, a benchmark for Counterfactual Video Foley Generation. Comprehensive experiments show that **TAP-Foley** achieves a more favorable trade-off between sound identity replacement and temporal alignment than the evaluated configurations of prior methods. At matched replacement, it improves temporal alignment by 20% over the strongest prior inference-time guidance method. Code, benchmark, and demos are available at `https://anonymous129108.github.io/TAP-Foley/`.

1 INTRODUCTION

Video Foley Generation synthesizes sound effects for a silent video so that acoustic events follow the visible action. *Counterfactual Video Foley Generation* preserves this timing while deliberately replacing the sound identity that the video implies. Figure 1 illustrates this setting with a silent video of a barking dog, paired with a source prompt "dog barking" naming the sound the video implies, and a target prompt "sheep bleating" conflicting with it. The generated output must sound like a sheep rather than a dog, yet each bleat must coincide with a visible bark. The task thus imposes two key requirements, *sound identity replacement* and *temporal alignment*.

Existing video-text-to-audio (VT2A) diffusion and flow-matching models (Cheng et al., 2025; Dai et al., 2026; Tian et al., 2026; Zhang et al., 2026; Jeong et al., 2025) struggle with this task because the video conditioning that carries the timing also steers generation toward the source identity. These models are trained to reconstruct each clip's original audio from its video and a text prompt describing that audio, so they learn both timing and identity from the video. In the counterfactual setting, however, the target prompt and the video indicate different sound identities. Because this conflict is not explicitly addressed during training, these models tend to either retain the source identity or produce an additive mixture of the two identities (Yang et al., 2026b; Lee et al., 2026).

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

Existing methods for Counterfactual Video Foley Generation force a trade-off between the two requirements. Among training-based methods, CAFA (Benita et al., 2025) adds a trainable visual adapter to a text-to-audio model and ControlFoley (Yang et al., 2026b) trains directly on video–text conflicts, yet the source identity remains audible in their outputs. Among inference-time guidance methods, negative-source classifier-free guidance steers the model's prediction at each denoising step away from the source prompt with a fixed weight, suppressing the source identity but degrading temporal alignment. CounterFlow (Lee et al., 2026) corrects a video-only prediction along a text-only source–target axis and drops video conditioning at an intermediate step. This drop prevents the video from reinforcing the source identity in later steps, but those steps then lose the video signal that carries the timing. Appendix A reviews related work in more detail.

We propose **TAP-Foley**, an inference-time guidance method built on *Target-Anchored Projection*, which mitigates the conflict between the video and the target prompt without dropping either modality. At each denoising step, the frozen VT2A backbone produces a joint prediction from the video and the target prompt, and two text-only predictions from the source and target prompts. The difference between the two text-only predictions defines a source–target contrast axis. The target-only prediction serves as the *anchor*: along this axis, the corrected prediction is constrained to lie no closer to the source identity than the anchor does. The distance by which the joint prediction lies past the anchor along the axis measures how strongly the video steers the prediction toward the source identity. **TAP-Foley** removes this excess with a scaled closed-form projection onto the constraint; the correction is zero when the video does not steer past the anchor and otherwise moves the joint prediction only along the contrast axis.

Evaluating this task is also difficult because existing benchmarks confound model failures with problems in the data. They draw clips from VGGSound (Chen et al., 2020), whose annotations do not reliably mark off-screen or multiple sound sources, and assign target prompts without regard to the temporal envelope of the sound the video implies. A low replacement score can then reflect a source prompt that does not describe the sound its video implies, and a low temporal alignment score can reflect an incompatible target envelope. We therefore build VGGSound-Counterfactual, in which each source prompt describes the sound its video implies and each target prompt has a temporal envelope compatible with the source. Multimodal large language models (MLLMs) screen each video for a single visible emitter, verify from video and audio jointly that the sound matches the source prompt, and label the temporal envelope of the source. Human reviewers examine every clip that these checks leave uncertain or reject, and correct low-confidence envelope labels.

Our contributions are threefold. (i) We propose **TAP-Foley**, a novel inference-time guidance method that adaptively suppresses the source identity while retaining video conditioning throughout generation. It mitigates the conflict between the video and text conditions without dropping either, through a scaled closed-form projection onto a target-anchored constraint. (ii) We introduce VGGSound-Counterfactual, a benchmark for Counterfactual Video Foley Generation that evaluates sound identity replacement and temporal alignment, in which each triplet pairs a video with a source prompt that describes its sound and a target prompt whose timing the video can support. (iii) We evaluate existing VT2A systems and inference-time guidance methods on this benchmark. Prior methods either leave much of the source identity audible or, at comparable replacement, degrade temporal alignment more than **TAP-Foley**.

2 METHOD

2.1 PROBLEM SETUP

Counterfactual Video Foley Generation imposes two requirements, sound identity replacement and temporal alignment. The input is a triplet (V, S, T), where V is a silent video, S is a source prompt naming the sound identity visually implied by V, and T is a target prompt that conflicts with S. The output is an audio clip for V. Sound identity replacement requires that the output be recognized as the target identity T rather than the source identity S. Temporal alignment requires that the acoustic onsets and temporal envelope of the output follow the visible actions in V.

TAP-Foley builds on a pretrained VT2A backbone $\boldsymbol{f}_\theta(\boldsymbol{z}, t; c_{\mathrm{vid}}, c_{\mathrm{txt}})$ that predicts the velocity of latent $\boldsymbol{z}$ at continuous timestep t. Diffusion and flow-matching models can both be described as predicting a velocity along a noise-to-data interpolation (Salimans & Ho, 2022; Lipman et al., 2023;

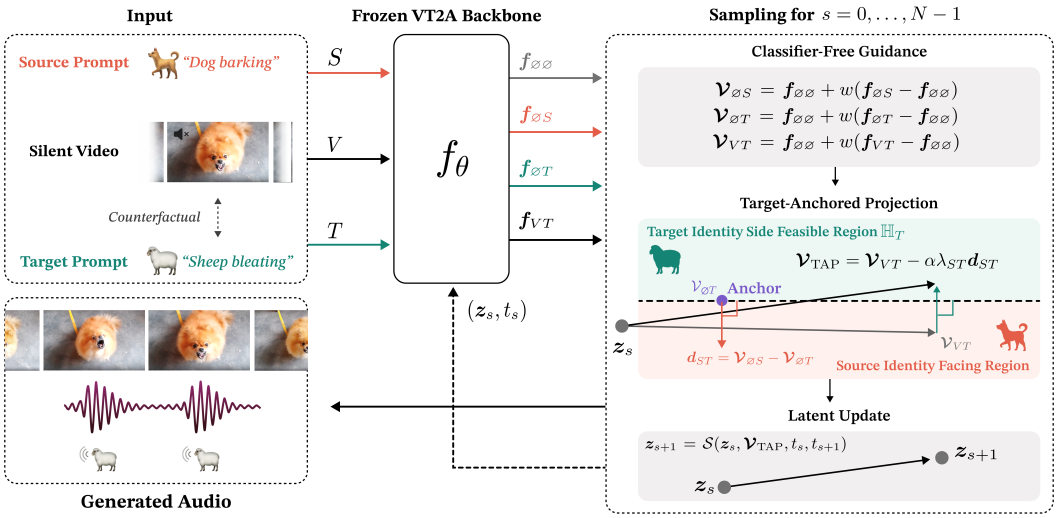

Figure 1: **Overview of TAP-Foley.** (Left) The inputs are a silent video, a source prompt, and a conflicting target prompt, and the output is an audio clip that matches the target prompt while preserving the timing of visible events. (Right) At every step, CFG combines four backbone outputs into the joint velocity $\boldsymbol{v}_{VT}$ and the text-only velocities $\boldsymbol{v}_{\varnothing S}$ and $\boldsymbol{v}_{\varnothing T}$, whose difference $\boldsymbol{d}_{ST}$ is the source–target contrast axis. Video conditioning typically places $\boldsymbol{v}_{VT}$ outside the half-space $\mathbb{H}_T$, on its source side (dashed boundary). **TAP-Foley** projects it along $-\boldsymbol{d}_{ST}$ onto $\mathbb{H}_T$ and scales this step by the correction scale α (green arrow), so that $\boldsymbol{v}_{\text{TAP}}$ (black arrow) lies inside $\mathbb{H}_T$. The native solver $\mathcal{S}$ of the backbone then advances the latent from $\boldsymbol{z}_s$ to $\boldsymbol{z}_{s+1}$ with $\boldsymbol{v}_{\text{TAP}}$.

Albergo et al., 2025), and **TAP-Foley** yields the same update under either parameterization. The backbone is conditioned on video $c_{\text{vid}} \in \{V, \varnothing\}$ and text $c_{\text{txt}} \in \{S, T, \varnothing\}$, where $\varnothing$ denotes the null condition of classifier-free guidance (CFG) (Ho & Salimans, 2022). At denoising step $s \in \{0, \ldots, N-1\}$ with latent $\boldsymbol{z}_s$ and timestep t_s, we write $\boldsymbol{f}_{c_{\text{vid}} c_{\text{txt}}} = \boldsymbol{f}_\theta(\boldsymbol{z}_s, t_s; c_{\text{vid}}, c_{\text{txt}})$ for the plain velocity prediction. For every condition pair $c = (c_{\text{vid}}, c_{\text{txt}})$, CFG combines the plain prediction $\boldsymbol{f}_c$ with the unconditional prediction $\boldsymbol{f}_{\varnothing\varnothing}$ at a shared guidance scale w to yield the guided velocity

$$\boldsymbol{v}_c = \boldsymbol{f}_{\varnothing\varnothing} + w\big(\boldsymbol{f}_c - \boldsymbol{f}_{\varnothing\varnothing}\big), \tag{1}$$

where calligraphic $\boldsymbol{v}$ with the corresponding subscripts denotes the guided counterpart of $\boldsymbol{f}$. The native solver $\mathcal{S}$ of the backbone advances the latent from Gaussian noise $\boldsymbol{z}_0$ to the audio latent $\boldsymbol{z}_N$ over N steps. Appendix B.1 gives the noise–velocity equivalence, and Appendix C.2 specifies the null condition of each backbone.

2.2 Target-Anchored Projection

Target-Anchored Projection (TAP) removes from the joint video–text velocity $\boldsymbol{v}_{VT}$ the steering toward the source identity that the video adds beyond what the target prompt alone produces. It measures this excess along a source–target contrast axis, relative to an anchor at the target-only velocity, and removes it with a closed-form projection.

Contrast axis from text-only velocities. Each denoising step of **TAP-Foley** calls the frozen VT2A backbone four times and forms three guided velocities, $\boldsymbol{v}_{VT}$, $\boldsymbol{v}_{\varnothing S}$, and $\boldsymbol{v}_{\varnothing T}$, with Equation 1. The joint video–text velocity $\boldsymbol{v}_{VT}$ is the *base velocity* to be corrected. Its video conditioning carries the timing of the visible events and steers the base velocity toward the source identity. The difference between the two text-only velocities $\boldsymbol{v}_{\varnothing S}$ and $\boldsymbol{v}_{\varnothing T}$ defines the *source–target contrast axis*

$$\boldsymbol{d}_{ST} = \boldsymbol{v}_{\varnothing S} - \boldsymbol{v}_{\varnothing T}. \tag{2}$$

The axis points from the target toward the source, so the correction moves the velocity along $-\boldsymbol{d}_{ST}$. When the backbone predicts the conditional mean velocities, the latent change induced by this cor-

Table 1: Inference-time guidance methods for VT2A written in the shared template $\boldsymbol{u} = \boldsymbol{b} - \gamma \boldsymbol{e}$. For CounterFlow, the arrow marks the intermediate step at which video conditioning is dropped, and its base uses its own video weight (Appendix B.4).

Method	Base $\boldsymbol{b}$	Direction $\boldsymbol{e}$	Weight γ	Base kept
Default CFG	$\boldsymbol{\mathcal{V}}_{VT}$	–	0	all steps
Negative-source CFG	$\boldsymbol{\mathcal{V}}_{VT}$	$\boldsymbol{\mathcal{V}}_{VS} - \boldsymbol{\mathcal{V}}_{\varnothing\varnothing}$	fixed, from w	all steps
CounterFlow	$\boldsymbol{\mathcal{V}}_{V\varnothing} \to \boldsymbol{\mathcal{V}}_{\varnothing\varnothing}$	$\boldsymbol{\mathcal{V}}_{\varnothing S} - \boldsymbol{\mathcal{V}}_{\varnothing T}$	fixed, per phase	until the drop step
TAP-Foley (ours)	$\boldsymbol{\mathcal{V}}_{VT}$	$\boldsymbol{\mathcal{V}}_{\varnothing S} - \boldsymbol{\mathcal{V}}_{\varnothing T}$	measured, $\alpha\lambda_{ST}$	all steps

rection at each solver step follows the gradient of the log ratio between the target and source densities. Appendix B.1 proves this correspondence.

Constraint anchored at the target-only velocity. The target-only velocity $\boldsymbol{\mathcal{V}}_{\varnothing T}$, which the backbone predicts when the target prompt alone decides the sound identity, is the *anchor* of our constraint. This anchor defines the *target-anchored half-space* along $\boldsymbol{d}_{ST}$ and the coefficient that measures its violation,

$$\mathbb{H}_T = \{\boldsymbol{u} : \langle \boldsymbol{u} - \boldsymbol{\mathcal{V}}_{\varnothing T},\, \boldsymbol{d}_{ST} \rangle \le 0\}\,, \qquad \lambda_{ST} = \max\left(\frac{\langle \boldsymbol{\mathcal{V}}_{VT} - \boldsymbol{\mathcal{V}}_{\varnothing T},\, \boldsymbol{d}_{ST} \rangle}{\|\boldsymbol{d}_{ST}\|_2^2},\, 0 \right), \tag{3}$$

where the boundary of $\mathbb{H}_T$ is the hyperplane through $\boldsymbol{\mathcal{V}}_{\varnothing T}$ with normal $\boldsymbol{d}_{ST}$ (Figure 1). The velocities $\boldsymbol{\mathcal{V}}_{VT}$ and $\boldsymbol{\mathcal{V}}_{\varnothing T}$ share the target prompt and differ only in the video, so the inner product $\langle \boldsymbol{\mathcal{V}}_{VT} - \boldsymbol{\mathcal{V}}_{\varnothing T}, \boldsymbol{d}_{ST} \rangle$ is the contribution of the video along the contrast axis. The coefficient λ_{ST} normalizes this contribution by the squared norm of $\boldsymbol{d}_{ST}$. The constraint is therefore violated, and λ_{ST} is positive, only when the video steers the joint velocity further toward the source than the target prompt alone does.

Projection onto the constraint. By Equation 3, λ_{ST} is the Euclidean-projection coefficient of $\boldsymbol{\mathcal{V}}_{VT}$ onto $\mathbb{H}_T$ (Appendix B.2), so

$$\boldsymbol{\mathcal{V}}_{\text{proj}} = \boldsymbol{\mathcal{V}}_{VT} - \lambda_{ST} \boldsymbol{d}_{ST} \tag{4}$$

is the point of $\mathbb{H}_T$ closest to $\boldsymbol{\mathcal{V}}_{VT}$, and it equals $\boldsymbol{\mathcal{V}}_{VT}$ when $\lambda_{ST} = 0$. **TAP-Foley** instead uses

$$\boldsymbol{\mathcal{V}}_{\text{TAP}} = \boldsymbol{\mathcal{V}}_{VT} - \alpha\lambda_{ST} \boldsymbol{d}_{ST} = \boldsymbol{\mathcal{V}}_{VT} + \alpha\big(\boldsymbol{\mathcal{V}}_{\text{proj}} - \boldsymbol{\mathcal{V}}_{VT}\big), \tag{5}$$

which scales this projection step by a single hyperparameter $\alpha \ge 1$, the *correction scale*, with $\alpha = 1$ recovering the projection itself. We scale it up because the projection alone leaves sound identity replacement weaker than the scaled update on both backbones (Appendix D.4).

The scaled update stays strictly inside $\mathbb{H}_T$ for every $\alpha > 1$ (Appendix B.2.1), and its departure from $\boldsymbol{\mathcal{V}}_{VT}$ is α times the distance from $\boldsymbol{\mathcal{V}}_{VT}$ to $\mathbb{H}_T$, so the gain in replacement comes at a cost in temporal alignment (Appendix D.4). We use $\alpha = 1.2$ because it empirically gives a moderate balance between sound identity replacement and temporal alignment. Appendix D.4 reports the sweep over α, and Algorithm 1 in Appendix B.3 summarizes the denoising loop of **TAP-Foley**.

2.3 Relation to Inference-Time Guidance Methods

Prior inference-time guidance methods and **TAP-Foley** share a common update format, $\boldsymbol{u} = \boldsymbol{b} - \gamma \boldsymbol{e}$, in which a base velocity $\boldsymbol{b}$ is corrected along a direction $\boldsymbol{e}$ with a weight γ. Table 1 writes each method in this form, and Appendix B.4 derives every row from its source. Four choices separate them, namely the conditioning of the base, the direction, whether γ is fixed or measured from the current prediction, and how long the base keeps its conditioning. Only CounterFlow changes its base and its fixed weight mid-denoising, at the transition step.

CounterFlow drops the video conditioning partway through denoising to manage the trade-off between identity replacement and temporal alignment, because the video conditioning carries both the source identity and the timing. **TAP-Foley** instead corrects the joint velocity in proportion to how far it lies past the anchor, so we expect identity replacement to succeed without dropping the video conditioning. The direction $\boldsymbol{d}_{ST}$ follows from the requirement that the correction move the prediction

from the source identity to the target identity, rather than merely away from the source. The axis $\boldsymbol{d}_{ST}$ has no video-conditioned term, whereas the negative-source CFG direction $\boldsymbol{\mathcal{V}}_{VS} - \boldsymbol{\mathcal{V}}_{\varnothing\varnothing}$ is expected to weaken the video conditioning that carries the timing, because $\boldsymbol{\mathcal{V}}_{VS}$ is video-conditioned.

Our contrast axis and our scaled projection have counterparts in guidance for text-to-image diffusion models, whereas our anchor departs from them. Our contrast axis $\boldsymbol{d}_{ST}$ takes the prompt-contrast form of Contrastive Guidance (Wu & De la Torre, 2024), which CounterFlow also uses, but we apply it to a different base velocity, the joint video–text velocity $\boldsymbol{\mathcal{V}}_{VT}$. Our scaled closed-form projection takes the form of Concept Removal Guidance (CRG; Choi et al., 2026), which bounds a score of how present a concept is by a user-set threshold, whereas we anchor the constraint at the target-only prediction, which specifies the identity to produce and needs no threshold. This anchor is the novel component, because those methods contrast two prompts within one modality, whereas here the source identity enters through the video and the target through the text. It lets a text-only axis measure this video-induced deviation, so **TAP-Foley** is the only row of Table 1 that pairs a multimodal base with a text-only direction and anchor. Replacing $\boldsymbol{d}_{ST}$ with the source-only direction $\boldsymbol{\mathcal{V}}_{\varnothing S} - \boldsymbol{\mathcal{V}}_{\varnothing\varnothing}$ of CRG or the anchor with the unconditional velocity collapses replacement (Section 4.4). Appendix B.5 relates these choices to further geometric and constrained guidance methods.

3 Experiments

3.1 Benchmark: VGGSound-Counterfactual

A benchmark for Counterfactual Video Foley Generation must satisfy three requirements on every triplet (V, S, T): 1) the source prompt S must exclusively describe the sound identity that V implies, because the identity metrics penalize residual evidence of S; 2) the prompts S and T must have compatible temporal envelopes since misaligned outputs can be a benchmark artifact rather than a method failure; 3) multiple target identities should be paired with each video clip to evaluate conflict resolution in diverse scenarios. Existing benchmarks meet at most one of these requirements. VGGSound-Sparse (Iashin et al., 2022), used by Benita et al. (2025), Lee et al. (2026), and Chen et al. (2025), includes clips with multiple or off-screen sources (Zverev et al., 2025) and pairs them with targets of arbitrary envelope. Its 12 categories further bound the ordered identity pairs at 132. VGGSound-TVC (Yang et al., 2026b) likewise lacks source prompt screening and leaves the target envelope unconstrained.

We introduce VGGSound-Counterfactual, which addresses all three requirements through semi-automatic (V, S) screening and temporally compatible target identity mapping. We first filter the VGGSound test set by its VGGSounder annotations for clips labeled with a single, synchronized audio–visual event. Because the VGGSound caption used as S names a single sound identity, modality-specific MLLMs screen V for a single visible source and the original audio for a single dominant sound source. An MLLM then jointly assesses the remaining clips for sound identity, temporal alignment, and whether the visible action causes the sound. Human reviewers finally examine every clip that the modality-specific screening marks uncertain or that the joint assessment rejects or marks uncertain, and audit a random sample of the accepted clips, leaving 1,189 clips. For each retained clip, an MLLM labels the local event envelope of the audio with one of five values, from sharp hit to continuous stream. It also labels the clip-level occurrence pattern with one of six values, from an isolated event to continuous activity. In the same call, it selects ten VGGSound captions that name a different sound identity with an exactly or compatibly matched envelope and occurrence pattern. Human reviewers correct the labels that the MLLM assigns with low confidence. The resulting benchmark contains 11,890 triplets spanning 212 source categories, 297 target categories, and 3,377 distinct ordered source–target pairs. Appendix C.1 describes the full curation and target-selection procedure, and Appendix C.1.3 reports the benchmark statistics.

3.2 Experimental Setup

Implementation details. We apply **TAP-Foley** to two frozen VT2A backbones, MMAudio (Cheng et al., 2025) and Omni2Sound (Dai et al., 2026), at their published step counts and guidance scales. MMAudio uses its 25-step Euler solver at guidance scale $w = 4.5$. Omni2Sound uses 100 steps at $w = 2.5$, and we replace its default stochastic solver with a deterministic one so

that guidance methods differ only in their update rule, not in the noise injected during sampling. We fix the correction scale $\alpha = 1.2$ at every step for both backbones in all tables, except for the sweep $\alpha \in \{1.0, 1.2, \ldots, 2.0\}$ reported in Figure 2 and Table 8. All metrics are computed on the first 8 s of each output. Appendix C.3.1 details the evaluation protocol.

Baselines. We compare **TAP-Foley** with existing VT2A systems and with other inference-time guidance methods. For the first comparison, we evaluate five publicly available VT2A systems as released. CAFA (Benita et al., 2025) and ControlFoley (Yang et al., 2026b) are designed for video–text identity conflicts, FoleyCrafter (Zhang et al., 2026) and ReWaS (Jeong et al., 2025) are text-to-audio models with a timing signal extracted from video, and AudioX (Tian et al., 2026) is a general-purpose VT2A model. We also evaluate each system with negative-source CFG at its native guidance scale, so that every system can exploit the source prompt S. For the second comparison, we use the same seed and a comparable number of function evaluations within each backbone, and compare **TAP-Foley** with negative-source CFG, CounterFlow (Lee et al., 2026), and Default CFG, which samples with the uncorrected joint velocity $\boldsymbol{\mathcal{V}}_{VT}$. CounterFlow uses its published guidance weights on MMAudio, rescaled on Omni2Sound to keep their ratios to the guidance scale w, and we sweep its transition step, at which it drops video conditioning, as a fraction of the total steps. Appendix C.2 details the inference setup of every compared system.

Evaluation metrics. We evaluate three aspects of each output, sound identity replacement, temporal alignment, and generation quality. Sound identity replacement is measured by CLAP similarity to the target prompt (CLAP-T) (Wu et al., 2023) and source–target classification accuracy (Clf. acc.) (Benita et al., 2025). FLAM (Wu et al., 2025), a frame-level audio–language model, scores the evidence for a prompt at each frame; we report the peak target evidence (FLAM-T), its excess over the peak source evidence (ΔFLAM), and the fraction of triplets with positive excess, denoted (+) Ratio (Lee et al., 2026). Temporal alignment is quantified by DeSync (Iashin et al., 2024), which estimates the audio–visual offset without reference to any prompt. Identity metrics do not indicate when the target sounds, and DeSync also rewards a synchronized reproduction of the source, so success requires both groups to score well. We also report Fréchet Audio Distance (FAD) (Kilgour et al., 2019) on PANNs embeddings (Kong et al., 2020), Inception Score (IS) (Salimans et al., 2016), and Audiobox Production Quality (PQ) (Tjandra et al., 2025) as quality references. FAD compares outputs with the original recordings and can rise after a successful replacement, so we do not rank methods by it. Appendix C.3 gives the formula and implementation of every metric.

4 Results

4.1 Main Results

Comparison with existing VT2A systems. Table 2(a) shows that **TAP-Foley** achieves strong sound identity replacement while preserving temporal alignment. On MMAudio, it reaches the best CLAP-T and FLAM-T, along with the second-best Clf. acc., ΔFLAM, and (+) Ratio. Its DeSync trails only ControlFoley, whose ΔFLAM of 0.089 is less than a quarter of the 0.384 of **TAP-Foley**.

In their default setting, all five systems retain the source identity with ΔFLAM at or below 0.194, whereas **TAP-Foley** reaches 0.384 on MMAudio without losing temporal alignment. Negative-source CFG raises ΔFLAM for every system but worsens DeSync as well, so every dashed segment in Figure 2(a) points the same way. Training on conflicts does not remove the trade-off, since CAFA and ControlFoley fail to suppress the source identity even while keeping their temporal alignment. Every baseline with ΔFLAM above 0.25 has DeSync above 0.7, whereas **TAP-Foley** reaches 0.384 and 0.299 at DeSync 0.427 and 0.510, and no system dominates any point of its sweep.

Comparison with inference-time guidance methods. We compare inference-time guidance methods within each backbone at a comparable number of function evaluations (NFE). Table 2(b) shows that **TAP-Foley** improves temporal alignment over CounterFlow by a large margin on both backbones, while its identity replacement differs by backbone. On MMAudio at $\alpha = 1.2$, CounterFlow scores higher on some identity replacement metrics, while **TAP-Foley** scores higher on audio quality and reduces DeSync from 0.649 to 0.427 (34%). On Omni2Sound, **TAP-Foley** outperforms

Table 2: Counterfactual Video Foley Generation on 11,890 VGGSound-Counterfactual triplets. Panel (b) uses comparable NFE, with doubled steps for Default CFG and negative-source CFG and the published transition for CounterFlow (Appendix C.2). Bold and underline mark the best and second-best values *across all rows in (a)* and *within each backbone in (b)*. †FAD is not ranked.

Method		Sound Identity Replacement					Temporal Alignment	Distribution /Quality		
		CLAP-T ↑	Clf. acc. ↑	FLAM-T ↑	ΔFLAM ↑	(+) Ratio ↑	DeSync ↓	IS ↑	PQ ↑	FAD†
(a) Comparison with existing VT2A systems										
Ours										
TAP-Foley	MMAudio	**0.296**	0.885	**0.959**	0.384	0.900	0.427	**20.64**	6.21	21.6
	Omni2Sound	0.241	0.829	0.887	0.299	0.821	0.510	19.83	6.31	24.5
Baseline										
ControlFoley		0.229	0.601	0.864	0.089	0.610	**0.316**	17.10	6.13	14.6
CAFA		0.243	0.786	0.942	0.194	0.785	0.685	12.21	6.36	22.3
FoleyCrafter		0.154	0.514	0.746	-0.004	0.478	1.244	10.74	6.06	30.4
ReWaS		0.005	0.443	0.344	-0.005	0.504	1.088	5.33	5.44	58.3
AudioX		0.187	0.475	0.840	-0.034	0.475	1.136	16.27	**6.53**	18.1
Baseline (with negative-source CFG)										
ControlFoley		0.255	**0.893**	0.910	**0.456**	**0.916**	0.754	15.66	5.89	23.6
CAFA		0.200	0.873	0.875	0.335	0.860	0.715	10.36	6.25	24.5
FoleyCrafter		0.105	0.732	0.662	0.281	0.742	1.264	8.08	6.02	40.8
ReWaS		-0.045	0.586	0.190	0.050	0.575	1.168	4.73	5.52	67.7
AudioX		0.217	0.845	0.873	0.329	0.853	1.233	14.79	6.19	25.2
(b) Comparison of inference-time guidance methods										
MMAudio										
TAP-Foley (ours)		**0.296**	0.885	**0.959**	0.384	0.900	0.427	**20.64**	**6.21**	21.6
CounterFlow		0.292	**0.916**	0.949	**0.475**	**0.927**	0.649	19.75	6.16	23.7
Default CFG		0.162	0.350	0.727	-0.145	0.335	**0.323**	16.39	**6.21**	10.8
Negative-source CFG		0.212	0.758	0.824	0.280	0.778	0.703	12.93	5.93	19.2
Omni2Sound										
TAP-Foley (ours)		**0.241**	**0.829**	**0.887**	**0.299**	**0.821**	0.510	**19.83**	**6.31**	24.5
CounterFlow		0.213	0.765	0.862	0.226	0.740	0.576	15.00	6.06	23.7
Default CFG		0.110	0.223	0.620	-0.267	0.195	**0.408**	18.58	6.27	14.2
Negative-source CFG		0.132	0.453	0.714	-0.029	0.438	0.642	11.35	6.05	27.1

CounterFlow on every ranked metric and reduces DeSync from 0.576 to 0.510 (11%). **TAP-Foley** also achieves higher ΔFLAM and lower DeSync than negative-source CFG on both backbones.

Figure 2(b) compares the α sweep of **TAP-Foley** with the transition sweep of CounterFlow. A later CounterFlow transition retains video conditioning for more denoising steps and lowers both DeSync and ΔFLAM. For every CounterFlow setting except the earliest MMAudio transition, whose ΔFLAM of 0.495 exceeds the sweep, a **TAP-Foley** sweep point matches its ΔFLAM within 0.002 or exceeds it while reducing DeSync by 12% to 20%, as Table 8 shows. At CounterFlow's published transition, the matched points $\alpha = 2.0$ on MMAudio and $\alpha = 1.0$ on Omni2Sound reduce DeSync by 19% and 20%, respectively.

Retaining video conditioning throughout denoising does not close this gap. At the 100% transition, which keeps video conditioning at every step, CounterFlow's DeSync falls to 0.579 on MMAudio and 0.539 on Omni2Sound, yet **TAP-Foley** at $\alpha = 1.8$ and $\alpha = 1.0$ still reaches higher ΔFLAM with 12% and 14% lower DeSync. The two methods also differ in their base $\boldsymbol{b}$ and weight rule (Table 1), and Section 4.4 shows that the per-step weight contributes to source suppression. **TAP-Foley** also keeps PQ on par with Default CFG on both backbones, whereas negative-source CFG lowers both PQ and IS. Relative to Default CFG, the timing cost of **TAP-Foley** grows with the correction scale, and DeSync rises from 0.394 at $\alpha = 1.0$ to 0.527 at $\alpha = 2.0$ on MMAudio. Appendix D.4 reports the full sweep.

Human evaluation. We conduct a listening test to check whether the metric gains in identity replacement and temporal alignment hold perceptually. A total of 17 participants rate the eight systems in Table 3 on 20 (V, S, T) triplets, scoring target identity recognition, source identity recognition, and temporal alignment on a scale of 1 to 5; Appendix D.1 details the protocol.

As shown in Table 3, within each backbone, **TAP-Foley** receives the highest target recognition and the lowest source recognition, together with the second-best temporal alignment. Default CFG and

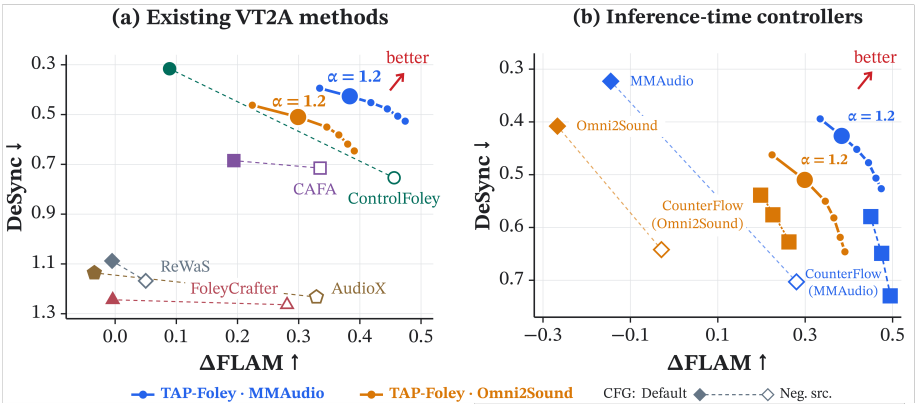

Figure 2: **Replacement–alignment trade-off on the 11,890 triplets.** The DeSync axis is reversed. Small connected circles trace the **TAP-Foley** sweep over $\alpha \in \{1.0, 1.2, \ldots, 2.0\}$, with $\alpha = 1.2$ enlarged. **(a)** Existing VT2A systems, with a dashed segment from the default (filled) to the negative-source setting (hollow). **(b)** Inference-time guidance methods at comparable NFE.

training-based systems fail to capture the target identity, to suppress the source identity, or both. CounterFlow instead falls short on temporal alignment, with the lowest temporal rating within each backbone. The ratings therefore reproduce the two failure modes of prior methods, and **TAP-Foley** alone is rated highly on both axes.

Table 3: Human evaluation on 20 triplets, as the mean and 95% CI over 17 participants on a 1–5 scale (Appendix D.1). Bold and underline mark the best and second-best values across (a) and within each backbone in (b).

(a) Existing VT2A systems

Method	Target ↑	Source ↓	Temporal ↑
Ours			
TAP-Foley (MMAudio)	**4.32** ±0.14	**1.24** ±0.08	<u>3.95</u> ±0.24
TAP-Foley (Omni2Sound)	<u>3.72</u> ±0.19	1.78 ±0.14	3.68 ±0.24
Baseline			
CAFA	3.19 ±0.21	<u>1.57</u> ±0.13	3.28 ±0.22
ControlFoley	2.49 ±0.15	2.89 ±0.13	**4.19** ±0.23

(b) Within-backbone guidance methods

Method	Target ↑	Source ↓	Temporal ↑
MMAudio			
TAP-Foley (ours)	**4.32** ±0.14	**1.24** ±0.08	<u>3.95</u> ±0.24
CounterFlow	<u>3.95</u> ±0.15	<u>1.33</u> ±0.06	3.00 ±0.21
Default CFG	2.08 ±0.13	3.41 ±0.15	**4.37** ±0.19
Omni2Sound			
TAP-Foley (ours)	**3.72** ±0.19	**1.78** ±0.14	<u>3.68</u> ±0.24
CounterFlow	<u>3.03</u> ±0.20	<u>2.50</u> ±0.13	3.54 ±0.24
Default CFG	1.49 ±0.10	4.01 ±0.16	**4.30** ±0.17

4.2 Onset-Level Evaluation

To assess generalization beyond VGGSound, we evaluate onset alignment on 1,280 (V, S, T) triplets from the Greatest Hits (Owens et al., 2016) dataset. The dataset includes short, impulsive hitting and scratching sounds, so each onset must land on the visible contact. **TAP-Foley** attains the highest onset recall and F1 among inference-time guidance methods on both backbones. It raises recall relative to CounterFlow from 0.705 to 0.851 on MMAudio and from 0.713 to 0.803 on Omni2Sound. Appendix D.2 details the evaluation protocol and the results.

4.3 Correction Dynamics

The projection-active ratio is the fraction of triplets whose joint velocity lies outside the target-anchored half-space at a given step. It stays above 96% on MMAudio and 87% on Omni2Sound at every step, even though each corrected velocity lies inside the half-space. The joint velocity therefore lies on the source side of the anchor at nearly every step, so the correction stays active throughout denoising rather than at a few steps. Figure 5 visualizes this steering in latent space and

Table 4: Ablation of the contrast axis, the anchor, and the weight of **TAP-Foley**. Each variant replaces $\boldsymbol{d}_{ST}$ with $\boldsymbol{e}$, the anchor $\boldsymbol{\mathcal{V}}_{\varnothing T}$ with $\boldsymbol{\mathcal{V}}_{\varnothing\varnothing}$, or the per-step weight with its mean $\alpha\bar{\lambda}_{ST}$ over the $\alpha = 1.2$ run, keeping the joint base. Bold marks the best value per backbone.

Variant	CLAP-T ↑	Clf. acc. ↑	FLAM-T ↑	ΔFLAM ↑	(+) Ratio ↑	DeSync ↓
MMAudio						
$\boldsymbol{e} = \boldsymbol{\mathcal{V}}_{VS} - \boldsymbol{\mathcal{V}}_{\varnothing\varnothing}$	0.228	0.780	0.860	0.275	0.795	0.696
$\boldsymbol{e} = \boldsymbol{\mathcal{V}}_{\varnothing S} - \boldsymbol{\mathcal{V}}_{\varnothing\varnothing}$	0.190	0.515	0.802	0.003	0.514	**0.330**
anchor $\boldsymbol{\mathcal{V}}_{\varnothing T} \to \boldsymbol{\mathcal{V}}_{\varnothing\varnothing}$	0.208	0.537	0.840	0.021	0.538	0.331
$\gamma = \alpha\lambda_{ST} \to \alpha\bar{\lambda}_{ST} = 0.77$	0.284	0.847	0.948	0.340	0.859	0.408
$\boldsymbol{e} = \boldsymbol{d}_{ST}$, anchor $\boldsymbol{\mathcal{V}}_{\varnothing T}$ **(ours)**	**0.296**	**0.885**	**0.959**	**0.384**	**0.900**	0.427
Omni2Sound						
$\boldsymbol{e} = \boldsymbol{\mathcal{V}}_{VS} - \boldsymbol{\mathcal{V}}_{\varnothing\varnothing}$	0.094	0.688	0.655	0.212	0.691	1.189
$\boldsymbol{e} = \boldsymbol{\mathcal{V}}_{\varnothing S} - \boldsymbol{\mathcal{V}}_{\varnothing\varnothing}$	0.124	0.384	0.670	-0.097	0.362	**0.418**
anchor $\boldsymbol{\mathcal{V}}_{\varnothing T} \to \boldsymbol{\mathcal{V}}_{\varnothing\varnothing}$	0.171	0.525	0.774	0.017	0.502	0.425
$\gamma = \alpha\lambda_{ST} \to \alpha\bar{\lambda}_{ST} = 1.82$	0.238	0.818	0.884	0.280	0.797	0.503
$\boldsymbol{e} = \boldsymbol{d}_{ST}$, anchor $\boldsymbol{\mathcal{V}}_{\varnothing T}$ **(ours)**	**0.241**	**0.829**	**0.887**	**0.299**	**0.821**	0.510

decomposes one correction step. Appendix D.3 details this figure, and Appendix D.6 shows case studies.

4.4 Ablations

We ablate the direction, the anchor, and the weight of **TAP-Foley** while keeping the other components fixed (Table 4). **(a) Contrast axis $\boldsymbol{d}_{ST}$.** Section 2.3 argued that the direction needs a target term and no video term, so we replace $\boldsymbol{d}_{ST}$ with $\boldsymbol{\mathcal{V}}_{VS} - \boldsymbol{\mathcal{V}}_{\varnothing\varnothing}$, which has a video term and no target term, and with $\boldsymbol{\mathcal{V}}_{\varnothing S} - \boldsymbol{\mathcal{V}}_{\varnothing\varnothing}$, which has neither. With the video term, ΔFLAM falls to 0.275 on MMAudio and 0.212 on Omni2Sound, and DeSync rises to 0.696 and 1.189. With neither term, DeSync stays near Default CFG at 0.330 and 0.418, but ΔFLAM collapses to 0.003 and -0.097. Removing the video term therefore restores timing, adding the target term restores replacement, and only $\boldsymbol{d}_{ST}$ improves both metrics on both backbones. **(b) Anchor $\boldsymbol{\mathcal{V}}_{\varnothing T}$.** The anchor defines what counts as excess, so we evaluate whether the unconditional velocity can replace the target-only prediction in Equation 3. Replacement collapses, with ΔFLAM at 0.021 on MMAudio and 0.017 on Omni2Sound, while DeSync improves to 0.331 and 0.425. The difference $\boldsymbol{\mathcal{V}}_{VT} - \boldsymbol{\mathcal{V}}_{\varnothing\varnothing}$ includes the target prompt's own shift, which cancels part of the video's steering toward the source, so the constraint measures a smaller violation and the anchor has to be target-only. **(c) Weight $\alpha\lambda_{ST}$.** The weight is measured per step, so we evaluate whether a constant of the same average size suppresses the source equally well by replacing $\alpha\lambda_{ST}$ with its mean $\alpha\bar{\lambda}_{ST}$ over the $\alpha = 1.2$ run. Target evidence remains similar, with CLAP-T and FLAM-T within 0.012 of ours, whereas Clf. acc., ΔFLAM, and (+) Ratio fall on both backbones, and DeSync is lower by 0.018 on MMAudio and by 0.007 on Omni2Sound, where the difference is not significant. The constant corrects by the same amount regardless of the violation and appears to leave more of the source audible, and its lower DeSync is consistent with a smaller correction (Appendix D.5).

5 Conclusion

We presented **TAP-Foley**, an inference-time guidance method for Counterfactual Video Foley Generation that mitigates the conflict between video and text conditioning without dropping either. At each step, it measures how strongly the video steers the joint prediction past a target-only anchor along a text-only source–target axis, and a closed-form projection with a correction scale α removes this excess. We also introduced VGGSound-Counterfactual, a benchmark for Counterfactual Video Foley Generation that evaluates sound identity replacement and temporal alignment. On this benchmark and two VT2A backbones, **TAP-Foley** achieves a more favorable replacement–alignment trade-off than the evaluated configurations of prior methods. Appendix E.1 discusses the limitations and future work.

AI USE STATEMENT

In this work, we used generative AI tools for research ideation and execution, for drafting and polishing the manuscript, for generating benchmark data, and for proving mathematical claims. LLM-based assistants helped brainstorm and refine research ideas, gave feedback on the method and the experimental design, helped implement the method and the evaluation pipeline, and helped run and manage the experiments. Multimodal large language models, namely Gemini 3.5 Flash and Qwen3-Omni-30B-A3B-Instruct, screened and verified the source clips, labeled their temporal descriptors, and selected the target prompts of VGGSound-Counterfactual (Appendix C.1). LLM-based assistants also helped derive and write the proofs in Appendix B. They drafted parts of the manuscript, edited it for clarity and readability, and checked and formatted the reference list. We did not use generative AI tools to search for or identify related work.

We have reviewed all AI-assisted work. The authors checked every reported result against the logged outputs, verified each step of the derivations, and reviewed and tested the code used in the experiments. Human reviewers examined every clip that the MLLM screening marked as uncertain or that the joint verification rejected, and they corrected low-confidence temporal labels. The authors revised every AI-drafted passage. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

REFERENCES

Michael S. Albergo, Nicholas M. Boffi, and Eric Vanden-Eijnden. Stochastic interpolants: A unifying framework for flows and diffusions. *Journal of Machine Learning Research*, 26(209):1–80, 2025. URL https://www.jmlr.org/papers/v26/23-1605.html.

Mohammadreza Armandpour, Ali Sadeghian, Huangjie Zheng, Amir Sadeghian, and Mingyuan Zhou. Re-imagine the negative prompt algorithm: Transform 2D diffusion into 3D, alleviate janus problem and beyond. *arXiv preprint arXiv:2304.04968*, 2023.

Yuanhao Ban, Ruochen Wang, Tianyi Zhou, Minhao Cheng, Boqing Gong, and Cho-Jui Hsieh. Understanding the impact of negative prompts: When and how do they take effect? In *Proceedings of the European Conference on Computer Vision*, pp. 190–206. Springer, 2024. doi: 10.1007/978-3-031-73024-5_12.

Roi Benita, Michael Finkelson, Tavi Halperin, Gleb Sterkin, and Yossi Adi. CAFA: A controllable automatic Foley artist. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15917–15926, 2025. doi: 10.1109/ICCV51701.2025.01477.

Manuel Brack, Felix Friedrich, Dominik Hintersdorf, Lukas Struppek, Patrick Schramowski, and Kristian Kersting. SEGA: Instructing text-to-image models using semantic guidance. In *Advances in Neural Information Processing Systems*, volume 36, pp. 25365–25389, 2023. doi: 10.52202/075280-1102.

Tim Brooks, Aleksander Holynski, and Alexei A. Efros. InstructPix2Pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18392–18402, 2023. doi: 10.1109/CVPR52729.2023.01764.

Honglie Chen, Weidi Xie, Andrea Vedaldi, and Andrew Zisserman. VGGSound: A large-scale audio-visual dataset. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 721–725, 2020. doi: 10.1109/ICASSP40776.2020.9053174.

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*, pp. 18770–18781, 2025. doi: 10.1109/CVPR52734.2025.01749.

Ho Kei Cheng, Masato Ishii, Akio Hayakawa, Takashi Shibuya, Alexander G. 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*, pp. 28901–28911, 2025. doi: 10.1109/CVPR52734.2025.02691.

Yoonseok Choi, Chaeyoung Oh, Hyunjun Choi, Seokin Seo, and Kee-Eung Kim. Concept removal guidance: Evidence-calibrated negative guidance for safe diffusion sampling. In *Proceedings of the 43rd International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*, 2026. URL https://openreview.net/forum?id=BSDvEh7oqy.

Jacob K. Christopher, Stephen Baek, and Ferdinando Fioretto. Constrained synthesis with projected diffusion models. In *Advances in Neural Information Processing Systems*, volume 37, pp. 89307–89333, 2024. doi: 10.52202/079017-2834.

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*, pp. 1661–1671, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Dai_Omni2Sound_Towards_Unified_Video-Text-to-Audio_Generation_CVPR_2026_paper.html.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Vladimir Iashin, Weidi Xie, Esa Rahtu, and Andrew Zisserman. Sparse in space and time: Audio-visual synchronisation with trainable selectors. In *Proceedings of the British Machine Vision Conference*. BMVA Press, 2022. URL https://bmvc2022.mpi-inf.mpg.de/0395.pdf.

Vladimir Iashin, Weidi Xie, Esa Rahtu, and Andrew Zisserman. Synchformer: Efficient synchronization from sparse cues. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 5325–5329, 2024. doi: 10.1109/ICASSP48485.2024.10448489.

Yujin Jeong, Yunji Kim, Sanghyuk Chun, and Jiyoung Lee. Read, watch and scream! Sound generation from text and video. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 17590–17598, 2025. doi: 10.1609/aaai.v39i17.33934.

Yilei Jiang, Zhen Wang, Yanghao Wang, Jun Yu, Yueting Zhuang, Jun Xiao, and Long Chen. FlowDC: Flow-based decoupling-decay for complex image editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 25757–25766, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Jiang_FlowDC_Flow-Based_Decoupling-Decay_for_Complex_Image_Editing_CVPR_2026_paper.html.

Kevin Kilgour, Mauricio Zuluaga, Dominik Roblek, and Matthew Sharifi. Fréchet audio distance: A reference-free metric for evaluating music enhancement algorithms. In *Proceedings of Interspeech*, pp. 2350–2354, 2019. doi: 10.21437/Interspeech.2019-2219.

Qiuqiang Kong, Yin Cao, Turab Iqbal, Yuxuan Wang, Wenwu Wang, and Mark D. Plumbley. PANNs: Large-scale pretrained audio neural networks for audio pattern recognition. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:2880–2894, 2020. doi: 10.1109/TASLP.2020.3030497.

Felix Koulischer, Johannes Deleu, Gabriel Raya, Thomas Demeester, and Luca Ambrogioni. Dynamic negative guidance of diffusion models. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=6p74UyAdLa.

Vladimir Kulikov, Matan Kleiner, Inbar Huberman-Spiegelglas, and Tomer Michaeli. FlowEdit: Inversion-free text-based editing using pre-trained flow models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 19721–19730, 2025. doi: 10.1109/ICCV51701.2025.01834.

Gyubin Lee, Junwon Lee, and Juhan Nam. CounterFlow: A two-phase inference-time sampling for counterfactual video Foley generation. *arXiv preprint arXiv:2605.18916*, 2026.

Junwon Lee, Jaekwon Im, Dabin Kim, and Juhan Nam. Video-Foley: Two-stage video-to-sound generation via temporal event condition for Foley sound. *IEEE Transactions on Audio, Speech and Language Processing*, 33:3813–3825, 2025. doi: 10.1109/TASLPRO.2025.3597477.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=PqvMRDCJT9t.

Nan Liu, Shuang Li, Yilun Du, Antonio Torralba, and Joshua B. Tenenbaum. Compositional visual generation with composable diffusion models. In *Proceedings of the European Conference on Computer Vision*, pp. 423–439. Springer, 2022. doi: 10.1007/978-3-031-19790-1_26.

Hila Manor and Tomer Michaeli. Zero-shot unsupervised and text-based audio editing using DDPM inversion. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 34603–34629, 2024. URL https://proceedings.mlr.press/v235/manor24a.html.

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*, pp. 2405–2413, 2016. doi: 10.1109/CVPR.2016.264.

Rodrigo Pérez-Dattari, Francisco Leiva, Andrea Testa, Leonel Rozo, Javier Ruiz del Solar, and Noémie Jaquier. Let the dynamics flow: Stable flow matching dynamical systems. *arXiv preprint arXiv:2606.03834*, 2026.

Ciara Rowles, Varun Jampani, Simon Donné, Shimon Vainer, Julian Parker, and Zach Evans. Foley Control: Aligning a frozen latent text-to-audio model to video. *arXiv preprint arXiv:2510.21581*, 2025.

Seyedmorteza Sadat, Otmar Hilliges, and Romann M. Weber. Eliminating oversaturation and artifacts of high guidance scales in diffusion models. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=e2ONKX6qzJ.

Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=TIdIXIpzhoI.

Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training GANs. In *Advances in Neural Information Processing Systems*, volume 29, 2016. URL https://proceedings.neurips.cc/paper_files/paper/2016/hash/8a3363abe792db2d8761d6403605aeb7-Abstract.html.

Patrick Schramowski, Manuel Brack, Björn Deiseroth, and Kristian Kersting. Safe latent diffusion: Mitigating inappropriate degeneration in diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22522–22531, 2023. doi: 10.1109/CVPR52729.2023.02157.

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.

Bowen Shi, Andros Tjandra, John Hoffman, Helin Wang, Yi-Chiao Wu, Luya Gao, Julius Richter, Matthew Le, Apoorv Vyas, Sanyuan Chen, Christoph Feichtenhofer, Piotr Dollár, Wei-Ning Hsu, and Ann Lee. SAM Audio: Segment anything in audio. In *Proceedings of the 43rd International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*, 2026. URL https://openreview.net/forum?id=Q4Cca8N7na.

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*, 2026. URL https://openreview.net/forum?id=qjJWxK3yWo.

Andros Tjandra, Yi-Chiao Wu, Baishan Guo, John Hoffman, Brian Ellis, Apoorv Vyas, Bowen Shi, Sanyuan Chen, Matt Le, Nick Zacharov, Carleigh Wood, Ann Lee, and Wei-Ning Hsu. Meta Audiobox aesthetics: Unified automatic quality assessment for speech, music, and sound. *arXiv preprint arXiv:2502.05139*, 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*, pp. 30172–30182, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Vyas_Pushing_the_Frontier_of_Audiovisual_Perception_with_Large-Scale_Multimodal_Correspondence_CVPR_2026_paper.html.

Yuancheng Wang, Zeqian Ju, Xu Tan, Lei He, Zhizheng Wu, Jiang Bian, and Sheng Zhao. AUDIT: Audio editing by following instructions with latent diffusion models. In *Advances in Neural Information Processing Systems*, volume 36, pp. 71340–71357, 2023. doi: 10.52202/075280-3124.

Chen Wu and Fernando De la Torre. Contrastive prompts improve disentanglement in text-to-image diffusion models. *arXiv preprint arXiv:2402.13490*, 2024.

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 *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 1–5, 2023. doi: 10.1109/ICASSP49357.2023.10095969.

Yusong Wu, Christos Tsirigotis, Ke Chen, Cheng-Zhi Anna Huang, Aaron Courville, Oriol Nieto, Prem Seetharaman, and Justin Salamon. FLAM: Frame-wise language-audio modeling. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 67719–67740, 2025. URL https://proceedings.mlr.press/v267/wu25ab.html.

Jeongyong Yang, Seunghwan Jang, and SooJean Han. SafeFlowMatcher: Safe and fast planning using flow matching with control barrier functions. In *International Conference on Learning Representations*, 2026a. URL https://openreview.net/forum?id=refcXHU1Nh.

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. *arXiv preprint arXiv:2604.15086*, 2026b.

Jaehong Yoon, Shoubin Yu, Vaidehi Patil, Huaxiu Yao, and Mohit Bansal. SAFREE: Training-free and adaptive guard for safe text-to-image and video generation. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=hgTFotBRKl.

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. doi: 10.1007/s11263-025-02649-3.

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*, pp. 1027–1037, 2025. doi: 10.1109/ICCV51701.2025.00103.

APPENDIX

A Supplement to the Introduction **15**

A.1 Related Work 15

B Supplement to the Method **16**

B.1 Density-Ratio Interpretation 16

B.2 Euclidean Projection onto the Target-Anchored Half-Space 17

B.3 Algorithm 18

B.4 Deriving $\boldsymbol{u} = \boldsymbol{b} - \gamma \boldsymbol{e}$ for Guidance Methods 18

B.5 Relation to Geometric Guidance and Constrained Generation 19

C Supplement to the Experiments **21**

C.1 Benchmark Construction 21

C.2 Compared Systems and Guidance Methods 27

C.3 Evaluation Metrics 28

D Supplement to the Results **30**

D.1 Human Evaluation Protocol 30

D.2 Onset-Level Evaluation on Greatest Hits 30

D.3 Projection Activity and Correction Geometry 32

D.4 Correction-Scale Sweep 33

D.5 Mean-Matched Constant Weight 34

D.6 Qualitative Case Studies 34

E Supplement to the Conclusion **36**

E.1 Limitations and Future Work 36

A Supplement to the Introduction

A.1 Related Work

Counterfactual Video Foley Generation draws on prior work in video-conditioned audio generation and in counterfactual Foley and audio editing. Appendix B.5 relates **TAP-Foley** to geometric guidance for generative models.

A.1.1 Video-Conditioned Audio Generation

VT2A models differ in how they obtain timing from the video, and this choice determines how strongly the video also fixes the sound identity. MMAudio (Cheng et al., 2025), Omni2Sound (Dai et al., 2026), and AudioX (Tian et al., 2026) learn joint audio–visual–text representations, so the same visual features supply both the timing and the identity of the sound. FoleyCrafter (Zhang et al., 2026) instead conditions a text-to-audio model on a visual onset signal, while ReWaS (Jeong et al., 2025) and Video-Foley (Lee et al., 2025) condition it on an energy trajectory estimated from the video. HunyuanVideo-Foley (Shan et al., 2025) fuses audio and video by joint attention and injects text by cross-attention to reduce the imbalance between the two conditions. Foley Control (Rowles et al., 2025) keeps a text-to-audio model frozen and adds video cross-attention, so that the prompt sets the semantics and the video refines the timing.

Training and evaluating against the original audio does not test whether a model follows a text prompt that conflicts with the video. These models learn to reproduce the original audio of each video and are scored against it, which rewards agreement with the observed sound. We therefore evaluate every system on sound identity replacement, including residual source evidence, and on temporal alignment together.

A.1.2 Counterfactual Foley and Audio Editing

Prior work resolves the conflict between video and text either by training the generator or by changing its denoising. MultiFoley (Chen et al., 2025) trains a generator with text, audio, and video controls, so that a text prompt can alter the sound of a video. CAFA (Benita et al., 2025) adds a trainable visual adapter to a text-to-audio model, and ControlFoley (Yang et al., 2026b) trains on video–text conflicts directly. CounterFlow (Lee et al., 2026) keeps the pretrained weights fixed and uses a two-phase denoising update that drops video conditioning in the second phase. It exposes separate video, text, and CFG weights together with a transition step, all reported on MMAudio only, whereas **TAP-Foley** has the single scale α and runs unchanged on both backbones (Appendix C.2). **TAP-Foley** also keeps the weights fixed, but it retains the joint video–text branch at every step and corrects its velocity against a target-only anchor.

Our setting also differs from instruction-based audio editing. AUDIT (Wang et al., 2023) edits an observed waveform according to a text instruction, and ZETA (Manor & Michaeli, 2024) edits one zero-shot through DDPM inversion of a pretrained text-to-audio model. Counterfactual Video Foley Generation observes no source waveform at inference, so the source identity is implied only by the silent video and the source prompt.

B SUPPLEMENT TO THE METHOD

B.1 DENSITY-RATIO INTERPRETATION

Under idealized predictions, the TAP correction moves the latent along the gradient of the target-versus-source log-density ratio on both backbones. Writing $-\boldsymbol{d}_{ST} = \boldsymbol{\mathcal{V}}_{\varnothing T} - \boldsymbol{\mathcal{V}}_{\varnothing S}$, Equation 5 adds the target–source velocity difference to the joint prediction,

$$\boldsymbol{\mathcal{V}}_{\mathrm{TAP}} = \boldsymbol{\mathcal{V}}_{VT} + \gamma(\boldsymbol{\mathcal{V}}_{\varnothing T} - \boldsymbol{\mathcal{V}}_{\varnothing S}), \qquad \gamma = \alpha\lambda_{ST} \geq 0. \tag{6}$$

We first relate this velocity difference to the gradient of a density ratio and then trace how each backbone's solver adds the correction to the latent. The derivation assumes that the model predictions equal the conditional mean velocities defined below.

From velocity differences to a density ratio. We compare the target and the source through their log-density ratio. Let r increase from noise to data, and let $p_r(\boldsymbol{z} \mid c)$ denote the latent density at time r under text condition c. The log-density ratio

$$\ell_r(\boldsymbol{z}) = \log \frac{p_r(\boldsymbol{z} \mid T)}{p_r(\boldsymbol{z} \mid S)} \tag{7}$$

measures how much more likely a latent is under the target than under the source. Increasing ℓ_r favors the target relative to the source, which does not require the target density itself to increase. Conditions shared by both text-only branches, such as duration, are omitted from the notation.

Along a Gaussian interpolation path, the source–target difference of the conditional mean velocities is proportional to the gradient of ℓ_r. We consider the path

$$\boldsymbol{z}_r = a_r\boldsymbol{x} + b_r\boldsymbol{\epsilon}, \qquad \boldsymbol{x} \sim p(\boldsymbol{x} \mid c), \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}), \tag{8}$$

where $\boldsymbol{x}$ and $\boldsymbol{\epsilon}$ are independent. At interior times, we assume $a_r, b_r > 0$, $\dot{a}_r \geq 0$, and $\dot{b}_r \leq 0$, with at least one of the two derivative inequalities strict. The conditional mean velocity is

$$\boldsymbol{u}_c(\boldsymbol{z}, r) = \mathbb{E}\Big[\dot{a}_r\boldsymbol{x} + \dot{b}_r\boldsymbol{\epsilon} \mid \boldsymbol{z}_r = \boldsymbol{z}, c\Big]. \tag{9}$$

For differentiable densities under standard regularity conditions, the Gaussian noise identity $\mathbb{E}[\boldsymbol{\epsilon} \mid \boldsymbol{z}_r = \boldsymbol{z}, c] = -b_r\nabla_{\boldsymbol{z}} \log p_r(\boldsymbol{z} \mid c)$ gives (Lipman et al., 2023; Albergo et al., 2025)

$$\boldsymbol{u}_c = \frac{\dot{a}_r}{a_r}\boldsymbol{z} + \kappa_r\nabla_{\boldsymbol{z}} \log p_r(\boldsymbol{z} \mid c), \qquad \kappa_r = \frac{b_r}{a_r}(\dot{a}_r b_r - a_r\dot{b}_r) > 0. \tag{10}$$

The first term does not depend on the condition, so it cancels in the source–target difference,

$$\boldsymbol{u}_T - \boldsymbol{u}_S = \kappa_r\nabla_{\boldsymbol{z}}\ell_r(\boldsymbol{z}). \tag{11}$$

The velocity difference therefore points in the direction along which the target density grows relative to the source density.

CFG preserves this relation up to a positive scale when source and target share the guidance scale. With the unconditional velocity $\boldsymbol{u}_0$ and a shared scale $w > 0$, the guided velocities $\mathcal{U}_{\varnothing c} = \boldsymbol{u}_0 + w(\boldsymbol{u}_c - \boldsymbol{u}_0)$ satisfy

$$\mathcal{U}_{\varnothing T} - \mathcal{U}_{\varnothing S} = w\kappa_r\nabla_{\boldsymbol{z}}\ell_r(\boldsymbol{z}). \tag{12}$$

Here $\mathcal{U}$ denotes velocity with respect to the increasing noise-to-data coordinate r. A backbone's native prediction $\boldsymbol{\mathcal{V}}$ need not follow this orientation, so we next derive the latent correction for each solver.

MMAudio. On MMAudio, the TAP correction adds a nonnegative multiple of the log-density-ratio gradient to the latent. The backbone interpolates as $\boldsymbol{z}_r = r\boldsymbol{x} + (1-r)\boldsymbol{\epsilon}$, so its native velocity already follows increasing r with $\kappa_r = (1-r)/r$. Its Euler update $\boldsymbol{z}_{\mathrm{next}} = \boldsymbol{z} + \Delta r\,\boldsymbol{\mathcal{V}}$ with $\Delta r > 0$ is linear in the velocity. Starting from the same latent, the TAP and uncorrected joint updates therefore differ by

$$\Delta\boldsymbol{z}_{\mathrm{corr}} := \boldsymbol{z}^{\mathrm{TAP}}_{\mathrm{next}} - \boldsymbol{z}^{\mathrm{base}}_{\mathrm{next}} = \Delta r\,\gamma(\boldsymbol{\mathcal{V}}_{\varnothing T} - \boldsymbol{\mathcal{V}}_{\varnothing S}) = \Delta r\,\gamma w\frac{1-r}{r}\nabla_{\boldsymbol{z}}\ell_r(\boldsymbol{z}). \tag{13}$$

The coefficient $\Delta r\,\gamma w(1-r)/r$ is nonnegative for $0 < r < 1$.

Omni2Sound. On Omni2Sound, two sign reversals cancel, so the TAP correction follows the same log-density-ratio gradient as on MMAudio. The backbone parameterizes the path by an angle, $\boldsymbol{z}_\theta = \cos\theta\,\boldsymbol{x} + \sin\theta\,\boldsymbol{\epsilon}$, and predicts $\boldsymbol{\mathcal{V}} = d\boldsymbol{z}/d\theta$. Generation decreases θ from $\pi/2$ to 0, so the native velocity is oriented from data toward noise. The substitution $r = 1 - 2\theta/\pi$ gives

$$a_r = \sin(\pi r/2), \qquad b_r = \cos(\pi r/2), \qquad \mathcal{U} = -\frac{\pi}{2}\boldsymbol{\mathcal{V}}. \tag{14}$$

Hence $\kappa_r = (\pi/2)\cot(\pi r/2)$, and Equation 12 becomes

$$\boldsymbol{\mathcal{V}}_{\varnothing T} - \boldsymbol{\mathcal{V}}_{\varnothing S} = -w\cot(\pi r/2)\nabla_{\boldsymbol{z}}\ell_r(\boldsymbol{z}). \tag{15}$$

The native prediction difference thus points opposite to the gradient. The deterministic solver, however, multiplies the prediction by a negative coefficient. For an angular decrement $\delta > 0$, the update is $\boldsymbol{z}_{\text{next}} = \cos\delta\,\boldsymbol{z} - \sin\delta\,\boldsymbol{\mathcal{V}}$, so the additional latent change due to TAP is

$$\Delta\boldsymbol{z}_{\text{corr}} = -\sin\delta\,\gamma(\boldsymbol{\mathcal{V}}_{\varnothing T} - \boldsymbol{\mathcal{V}}_{\varnothing S}) = \sin\delta\,\gamma w\cot(\pi r/2)\nabla_{\boldsymbol{z}}\ell_r(\boldsymbol{z}). \tag{16}$$

The coefficient $\sin\delta\,\gamma w\cot(\pi r/2)$ is nonnegative for $0 < r < 1$ and $0 < \delta < \pi/2$.

Invariance to the prediction target. **TAP-Foley** yields the same update whether the backbone predicts the velocity or the noise. Substituting $\boldsymbol{x} = (\boldsymbol{z}_r - b_r\boldsymbol{\epsilon})/a_r$ into the conditional mean velocity gives

$$\boldsymbol{u}_c = \frac{\dot{a}_r}{a_r}\boldsymbol{z} + c_r\,\mathbb{E}[\boldsymbol{\epsilon} \mid \boldsymbol{z}_r = \boldsymbol{z}, c], \qquad c_r = \frac{a_r\dot{b}_r - \dot{a}_r b_r}{a_r} < 0. \tag{17}$$

The first term is identical for every condition at a shared latent, so each difference in Equation 2 and Equation 3 is the corresponding noise difference scaled by c_r. The coefficient λ_{ST} is therefore the same in either parameterization, and the corrected noise prediction maps to $\boldsymbol{\mathcal{V}}_{\text{TAP}}$ by the same relation. CFG preserves this relation because its weights sum to one.

B.2 Euclidean Projection onto the Target-Anchored Half-Space

The coefficient λ_{ST} in Equation 3 is the unique solution of a minimum-distance problem, which we state in a general form. Consider a base velocity $\boldsymbol{b} \in \mathbb{R}^D$, an anchor $\boldsymbol{a} \in \mathbb{R}^D$, and a nonzero normal $\boldsymbol{d} \in \mathbb{R}^D$, and the problem

$$\min_{\boldsymbol{u}\in\mathbb{R}^D} \frac{1}{2}\|\boldsymbol{u} - \boldsymbol{b}\|_2^2 \quad \text{subject to} \quad \langle \boldsymbol{u} - \boldsymbol{a}, \boldsymbol{d}\rangle \le 0. \tag{18}$$

Proposition 1 (unique minimum-distance solution). For $\boldsymbol{d} \neq \boldsymbol{0}$, the unique solution of Equation 18 is

$$\boldsymbol{u}^\star = \boldsymbol{b} - \mu^\star\boldsymbol{d}, \qquad \mu^\star = \max\left(\frac{\langle \boldsymbol{b} - \boldsymbol{a}, \boldsymbol{d}\rangle}{\|\boldsymbol{d}\|_2^2}, 0\right). \tag{19}$$

Proof. The objective is strictly convex and the constraint is affine. The point $\boldsymbol{a} - \boldsymbol{d}$ is strictly feasible for $\boldsymbol{d} \neq \boldsymbol{0}$, so the Karush–Kuhn–Tucker (KKT) conditions are necessary and sufficient, and strict convexity makes the solution unique. With a multiplier $\mu \ge 0$, the Lagrangian is

$$\mathcal{L}(\boldsymbol{u}, \mu) = \frac{1}{2}\|\boldsymbol{u} - \boldsymbol{b}\|_2^2 + \mu\,\langle \boldsymbol{u} - \boldsymbol{a}, \boldsymbol{d}\rangle\,. \tag{20}$$

Stationarity gives

$$\nabla_{\boldsymbol{u}}\mathcal{L} = \boldsymbol{u} - \boldsymbol{b} + \mu\boldsymbol{d} = \boldsymbol{0}, \qquad \text{hence} \qquad \boldsymbol{u} = \boldsymbol{b} - \mu\boldsymbol{d}. \tag{21}$$

Let $c = \langle \boldsymbol{b} - \boldsymbol{a}, \boldsymbol{d}\rangle$. Substituting $\boldsymbol{u}$ into the constraint gives primal feasibility $c - \mu\|\boldsymbol{d}\|_2^2 \le 0$. The remaining KKT conditions are dual feasibility $\mu \ge 0$ and complementary slackness $\mu\big(c - \mu\|\boldsymbol{d}\|_2^2\big) = 0$. If $c \le 0$, the base velocity is already feasible, and $\mu^\star = 0$ with $\boldsymbol{u}^\star = \boldsymbol{b}$ satisfies all conditions. If $c > 0$, primal feasibility excludes $\mu = 0$, so the constraint is active and $\mu^\star = c/\|\boldsymbol{d}\|_2^2$. The two cases together give Equation 19. □

Setting $\boldsymbol{b} = \boldsymbol{\mathcal{V}}_{VT}$, $\boldsymbol{a} = \boldsymbol{\mathcal{V}}_{\varnothing T}$, and $\boldsymbol{d} = \boldsymbol{d}_{ST}$ recovers Equation 3. If $\boldsymbol{d}_{ST} = \boldsymbol{0}$, every velocity satisfies the constraint and the minimizer is $\boldsymbol{b}$ itself. Algorithm 1 therefore applies no correction when $\|\boldsymbol{d}_{ST}\|_2^2 = 0$.

B.2.1 Feasibility of the Update at Correction Scale α

Scaling the projection by any $\alpha \geq 1$ keeps the update inside the half-space. **TAP-Foley** uses

$$\boldsymbol{u}_\alpha = \boldsymbol{b} - \alpha\mu^\star \boldsymbol{d}, \qquad \alpha \geq 1, \tag{22}$$

in place of the minimum-distance point $\boldsymbol{u}^\star$. If $c \leq 0$, then $\mu^\star = 0$ and $\boldsymbol{u}_\alpha = \boldsymbol{b}$ remains feasible. If $c > 0$, then $\mu^\star = c/\|\boldsymbol{d}\|_2^2$ and

$$\langle \boldsymbol{u}_\alpha - \boldsymbol{a}, \boldsymbol{d} \rangle = c - \alpha\mu^\star \|\boldsymbol{d}\|_2^2 = (1-\alpha)c \leq 0. \tag{23}$$

In the notation of Equation 5, this constraint value equals $(1-\alpha)\lambda_{ST}\|\boldsymbol{d}_{ST}\|_2^2$. For an active constraint, $\alpha = 1$ places the update on the boundary and $\alpha > 1$ moves it into the interior on the target side.

For an active constraint, the update with $\alpha > 1$ departs from $\boldsymbol{b}$ by α times the minimum distance. Its distance from $\boldsymbol{b}$ is $\alpha\mu^\star\|\boldsymbol{d}\|_2$, so it is not the solution of Equation 18. At $\alpha = 1.2$, the correction exceeds the smallest feasible correction by 20% in exchange for stronger replacement. Appendix D.4 reports the resulting trade-off between replacement and temporal alignment over α.

B.3 Algorithm

Algorithm 1 summarizes the denoising loop of **TAP-Foley** on a frozen backbone. At each step, the four backbone queries share the current latent $\boldsymbol{z}_s$ and timestep t_s, and they differ only in the video and text conditions. Inner products and norms are taken over the vectorized latent of each sample, so every triplet receives its own coefficient λ_{ST}. The check $q > 0$ handles the degenerate case $\boldsymbol{d}_{ST} = \boldsymbol{0}$ of Appendix B.2, in which no correction is applied. The native solver $\mathcal{S}$ then advances the latent by one step with the corrected velocity $\boldsymbol{\mathcal{V}}_{\mathrm{TAP}}$, so **TAP-Foley** inherits the solver of each backbone unchanged.

Algorithm 1 TAP-Foley denoising with Target-Anchored Projection

Require: Frozen velocity predictor $\boldsymbol{f}_\theta$, decoder D, native solver $\mathcal{S}$, time grid $\{t_s\}_{s=0}^{N}$, guidance scale w, correction scale $\alpha = 1.2$
Require: Silent video V, source prompt S, target prompt T, noise $\boldsymbol{z}_0 \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$
Ensure: Generated audio $D(\boldsymbol{z}_N)$

for $s = 0, \ldots, N-1$ **do**
  for $(c_{\mathrm{vid}}, c_{\mathrm{txt}}) \in \{(\varnothing,\varnothing), (V,T), (\varnothing,S), (\varnothing,T)\}$ **do**
    $\boldsymbol{f}_{c_{\mathrm{vid}}c_{\mathrm{txt}}} \leftarrow \boldsymbol{f}_\theta(\boldsymbol{z}_s, t_s; c_{\mathrm{vid}}, c_{\mathrm{txt}})$ ▷ $\varnothing$: null condition
  end for
  $\boldsymbol{\mathcal{V}}_{VT} \leftarrow \boldsymbol{f}_{\varnothing\varnothing} + w(\boldsymbol{f}_{VT} - \boldsymbol{f}_{\varnothing\varnothing})$
  $\boldsymbol{\mathcal{V}}_{\varnothing S} \leftarrow \boldsymbol{f}_{\varnothing\varnothing} + w(\boldsymbol{f}_{\varnothing S} - \boldsymbol{f}_{\varnothing\varnothing})$
  $\boldsymbol{\mathcal{V}}_{\varnothing T} \leftarrow \boldsymbol{f}_{\varnothing\varnothing} + w(\boldsymbol{f}_{\varnothing T} - \boldsymbol{f}_{\varnothing\varnothing})$
  $\boldsymbol{d}_{ST} \leftarrow \boldsymbol{\mathcal{V}}_{\varnothing S} - \boldsymbol{\mathcal{V}}_{\varnothing T}$; $q \leftarrow \|\boldsymbol{d}_{ST}\|_2^2$; $\lambda_{ST} \leftarrow 0$
  if $q > 0$ **then**
    $\lambda_{ST} \leftarrow \max(\langle \boldsymbol{\mathcal{V}}_{VT} - \boldsymbol{\mathcal{V}}_{\varnothing T}, \boldsymbol{d}_{ST} \rangle / q, 0)$
  end if
  $\boldsymbol{\mathcal{V}}_{\mathrm{TAP}} \leftarrow \boldsymbol{\mathcal{V}}_{VT} - \alpha\lambda_{ST}\boldsymbol{d}_{ST}$
  $\boldsymbol{z}_{s+1} \leftarrow \mathcal{S}(\boldsymbol{z}_s, \boldsymbol{\mathcal{V}}_{\mathrm{TAP}}, t_s, t_{s+1})$
end for
return $D(\boldsymbol{z}_N)$

B.4 Deriving $\boldsymbol{u} = \boldsymbol{b} - \gamma\boldsymbol{e}$ for Guidance Methods

Each guidance method in Table 1 reduces to a base velocity minus a weighted correction direction, $\boldsymbol{u} = \boldsymbol{b} - \gamma\boldsymbol{e}$, once its update rule is written with the guided velocities of Equation 1. This section derives $\boldsymbol{b}$, $\boldsymbol{e}$, and γ for each method at the shared guidance scale w. Every derivation uses the identity $\boldsymbol{\mathcal{V}}_c - \boldsymbol{\mathcal{V}}_{\varnothing\varnothing} = w(\boldsymbol{f}_c - \boldsymbol{f}_{\varnothing\varnothing})$, which holds because $\boldsymbol{\mathcal{V}}_{\varnothing\varnothing} = \boldsymbol{f}_{\varnothing\varnothing}$.

Default CFG. Default CFG drives the solver with the joint guided velocity, so $\boldsymbol{b} = \boldsymbol{\mathcal{V}}_{VT}$ and $\gamma = 0$.

Negative-source CFG. Negative-source CFG guides from the source-prompted prediction toward the target-prompted prediction under the same video (Table 1),

$$\boldsymbol{u} = \boldsymbol{f}_{VS} + w\big(\boldsymbol{f}_{VT} - \boldsymbol{f}_{VS}\big) = \boldsymbol{\mathcal{V}}_{VT} - (w-1)\big(\boldsymbol{f}_{VS} - \boldsymbol{f}_{\varnothing\varnothing}\big). \tag{24}$$

Because $\boldsymbol{\mathcal{V}}_{VS} - \boldsymbol{\mathcal{V}}_{\varnothing\varnothing} = w(\boldsymbol{f}_{VS} - \boldsymbol{f}_{\varnothing\varnothing})$, this rule takes the template form with $\boldsymbol{b} = \boldsymbol{\mathcal{V}}_{VT}$, $\boldsymbol{e} = \boldsymbol{\mathcal{V}}_{VS} - \boldsymbol{\mathcal{V}}_{\varnothing\varnothing}$, and the fixed weight $\gamma = (w-1)/w$. Its direction contains the video contribution and makes no reference to the target prompt. The contrast-axis ablation of Section 4.4 substitutes this direction for $\boldsymbol{d}_{ST}$, together with the source-only direction $\boldsymbol{\mathcal{V}}_{\varnothing S} - \boldsymbol{\mathcal{V}}_{\varnothing\varnothing}$ of CRG, which removes the video term but still omits the target prompt.

CounterFlow. CounterFlow (Lee et al., 2026) switches between two update rules at a transition step, each with its own weights. Before the transition, it combines a video-only term with weight w_{vid} and a text contrast term with weight w_{txt},

$$\boldsymbol{u} = \boldsymbol{f}_{\varnothing\varnothing} + w_{\text{vid}}\big(\boldsymbol{f}_{V\varnothing} - \boldsymbol{f}_{\varnothing\varnothing}\big) + w_{\text{txt}}\big(\boldsymbol{f}_{\varnothing T} - \boldsymbol{f}_{\varnothing S}\big). \tag{25}$$

After the transition, it drops the video term and applies a second-phase weight w_{cfg}, so that $\boldsymbol{u} = \boldsymbol{f}_{\varnothing\varnothing} + w_{\text{cfg}}(\boldsymbol{f}_{\varnothing T} - \boldsymbol{f}_{\varnothing S})$. Both phases correct along $\boldsymbol{e} = \boldsymbol{d}_{ST}$, because $\boldsymbol{d}_{ST} = \boldsymbol{\mathcal{V}}_{\varnothing S} - \boldsymbol{\mathcal{V}}_{\varnothing T} = w(\boldsymbol{f}_{\varnothing S} - \boldsymbol{f}_{\varnothing T})$. The base is the video-only guided velocity at scale w_{vid} in the first phase and the unconditional velocity $\boldsymbol{\mathcal{V}}_{\varnothing\varnothing}$ in the second. The weight is fixed at $\gamma = w_{\text{txt}}/w$ before the transition and at $\gamma = w_{\text{cfg}}/w$ after it. Our runs set $w_{\text{cfg}} = w$, which gives $\gamma = 1$ in the second phase. Appendix C.2 lists the weights on each backbone.

TAP-Foley. **TAP-Foley** keeps $\boldsymbol{b} = \boldsymbol{\mathcal{V}}_{VT}$ and $\boldsymbol{e} = \boldsymbol{d}_{ST}$ at every step and measures $\gamma = \alpha\lambda_{ST}$ from the constraint violation in Equation 3. Its weight therefore changes with the current latent at every step.

B.5 Relation to Geometric Guidance and Constrained Generation

The scaled closed-form projection of **TAP-Foley** follows Concept Removal Guidance (CRG; Choi et al., 2026). CRG applies a closed-form half-space projection with a scaling factor to remove concepts in text-to-image diffusion. It bounds a concept-presence score by a user-set threshold along a negative-prompt direction. This direction, from the unconditional to the negative-concept prediction, is $\boldsymbol{\mathcal{V}}_{\varnothing S} - \boldsymbol{\mathcal{V}}_{\varnothing\varnothing}$ in our notation, which Table 4 evaluates inside the target-anchored constraint. SAFREE (Yoon et al., 2025) also suppresses concepts by projection, but it projects prompt token embeddings away from an unsafe-concept subspace rather than correcting the prediction. **TAP-Foley** instead anchors its constraint at the target-only velocity, which specifies the identity to produce. It therefore replaces an identity rather than removing a concept, and it needs no threshold.

Geometric guidance changes the direction of a generative update as well as its magnitude. Perp-Neg (Armandpour et al., 2023) keeps only the component of negative-prompt guidance that is perpendicular to the positive-prompt guidance. APG (Sadat et al., 2025) splits the CFG update into components parallel and orthogonal to the conditional prediction. Contrastive Guidance (Wu & De la Torre, 2024) guides along the difference between two prompts that differ in a few tokens, the construction that the source–target contrast axis of **TAP-Foley** and CounterFlow follows. InstructPix2Pix (Brooks et al., 2023) splits CFG into separately weighted image and text terms, and a similar split isolates the video contribution $\boldsymbol{\mathcal{V}}_{VT} - \boldsymbol{\mathcal{V}}_{\varnothing T}$ that **TAP-Foley** constrains. Composable Diffusion (Liu et al., 2022) adds and subtracts conditional predictions to compose or negate concepts, whereas **TAP-Foley** bounds its subtraction by a constraint anchored at the target-only prediction.

The effect of negative and semantic guidance varies over the denoising trajectory. Ban et al. (2024) show that negative prompts take effect with a delay and remove concepts by cancellation in the latent. This behavior bears on whether a correction should act at every step or only in some phases of denoising. Safe Latent Diffusion (Schramowski et al., 2023), SEGA (Brack et al., 2023), and Dynamic Negative Guidance (Koulischer et al., 2025) make negative or semantic guidance depend on the denoising state, which the per-step weight of **TAP-Foley** shares.

Flow editing and constrained generation act directly on the velocity or the sample. FlowEdit (Kulikov et al., 2025) edits a sample through the difference of velocities under the source and target prompts, and FlowDC (Jiang et al., 2026) extends flow-based editing to complex edits. Constrained flow methods further restrict the velocity with control-barrier formulations or admissible

half-spaces (Yang et al., 2026a; Pérez-Dattari et al., 2026). Projected Diffusion Models (Christopher et al., 2024) instead project the sample onto a constraint set during denoising, whereas **TAP-Foley** projects the predicted velocity. These methods treat inference-time control as a geometric operation on the update rather than a change in CFG strength alone, a view that **TAP-Foley** shares.

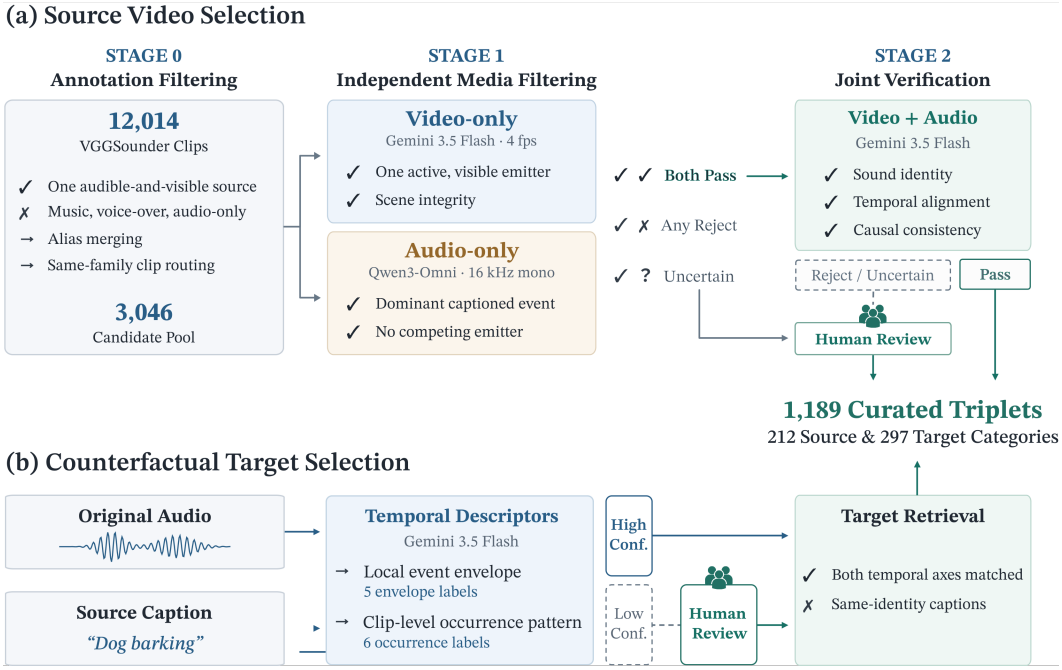

Figure 3: **Construction of VGGSound-Counterfactual.** **(a)** Source-video curation. Stage 0 screens the 12,014 VGGSounder clips by their annotations and yields 3,046 candidates. Stage 1 checks the silent video and the original audio independently and combines the two decisions with a reject-first rule. Stage 2 verifies the video and audio jointly, and human review resolves every uncertain fusion decision and every negative or uncertain joint decision, leaving 1,189 source clips from 212 categories. **(b)** Counterfactual target construction. The source audio and caption are mapped to a local event envelope and a clip-level occurrence pattern, and ten targets with a different sound identity and exact or compatible timing are selected from the 309 VGGSound captions, giving 11,890 triplets over 297 target categories.

C Supplement to the Experiments

C.1 Benchmark Construction

This section details the curation summarized in Section 3.1. It covers source-video selection, target-prompt construction, the statistics of the resulting benchmark, and the MLLM prompts. Figure 3 summarizes the first two parts.

C.1.1 Source-Video Selection

Curation objective. A valid source clip must provide both a clean source event and a timing that a target can reuse. It shows one identifiable sound-producing source, contains its captioned event as the dominant sound, and contains no competing physical emitter. Its original audio must also align with the visible action in onset, offset, and repetition. These requirements exclude clips whose apparent synchronization comes from unrelated or off-screen audio, so the original audio can serve as a temporal reference.

We build the benchmark from VGGSounder (Zverev et al., 2025) with a three-stage reject-first protocol. VGGSounder re-annotates the VGGSound test split (Chen et al., 2020) with 29,509 annotation rows over 12,014 unique clips and 309 sound categories. The protocol screens the annotations (Stage 0), verifies the silent video and the original audio independently (Stage 1), and verifies the clip jointly with human review (Stage 2), as Figure 3(a) summarizes.

Relation-aware annotation filtering. Stage 0 forms the candidate pool without treating every multi-label clip as multi-source. It rejects clips flagged for background music, voice-over, or a static image, clips with an audio-only label or without an audible-and-visible label, and clips whose audible-and-visible labels cannot come from one source. Conversely, it merges four taxonomy aliases such as *dog barking* and *dog bow-wow*. It also routes clips covered by one of 27 same-source family relations, such as the vocalizations of a dog, to media verification instead of accepting them automatically. This policy retains 2,228 strict single-label clips, recovers 50 alias clips, and routes 768 family clips. These 3,046 clips enter content-based filtering.

Independent media filtering. Stage 1 checks visual validity and audio purity separately, so that a failure in one modality is not hidden by the other. Gemini 3.5 Flash inspects the silent video at 4 fps for scene integrity, source identifiability, sound-producing activity, and exactly one active physical emitter. Independently, Qwen3-Omni-30B-A3B-Instruct receives only the source caption and the first audio stream as mono 16-kHz PCM, without denoising, normalization, silence trimming, or source separation. It tests whether the captioned event is present and dominant and whether a competing emitter is absent. A reject-first intersection combines the two branches, so evidence from one modality cannot override a decisive failure in the other. Appendix C.1.4 lists every curation prompt.

Joint verification and human review. Stage 2 verifies each remaining clip jointly from video and audio, and human reviewers decide the uncertain and rejected cases. Gemini 3.5 Flash judges the original audio–video clip jointly for sound identity, temporal alignment, and causal consistency. Human reviewers examine every clip that the Stage 1 fusion leaves uncertain and every negative or uncertain Stage 2 decision, and their decision takes precedence. Human reviewers also audit a random sample of the accepted clips. The resulting source set contains 1,189 clips from 212 source categories.

C.1.2 Temporally Compatible Target Prompts

Temporal descriptors. We select targets by two categorical temporal descriptors, so that each of the ten target captions of a clip changes the sound identity while remaining realizable under the source timing (Figure 3(b)). Random class substitution would often pair a sustained source with an impulsive target. The local event envelope takes one of five values, namely sharp hit, short burst, ringing decay, sustained event, and continuous stream. The clip-level occurrence pattern takes one of six values, namely isolated once, sparse irregular, repeated regular, repeated irregular, dense texture, and continuous without clear onsets. Both descriptors are needed because matching only one can leave event duration or recurrence incompatible. For example, a regular sequence of impacts calls for a target with both an impact envelope and regular repetition.

Labeling and target selection. Gemini 3.5 Flash assigns both descriptors from the clip audio and selects the ten targets in the same call. It receives the mono 16-kHz audio at temperature 0, with a text prompt that contains the source caption, the 309 canonical VGGSound captions (Chen et al., 2020), and a few worked examples. The prompt excludes the source caption, its punctuation variants and synonyms, and any caption with the same audible identity. It prefers targets that match the source on both descriptors, allows a compatible mismatch otherwise, and asks for diverse and recognizable identities. The model also predicts both descriptors of each target, so every target carries an explicit exact or compatible label. A validator rejects any output that repeats the source caption, duplicates a target, leaves the vocabulary, or returns the wrong number of targets. Human reviewers correct the descriptors that the model assigns with low confidence.

C.1.3 Benchmark Statistics

Source clips span seven sound families with imbalanced support (Figure 4(a)). Instruments contribute 449 of the 1,189 clips and the catch-all Other family contributes 334, so the two together cover 65.9%. Water or weather is the smallest family with 21 clips. The filters select clips without rebalancing families, so the composition follows the VGGSound categories that pass verification.

The 11,890 triplets span 212 source categories, 297 target categories, and 3,377 distinct ordered source–target pairs. The 1,189 source clips come from 1,171 original videos, because some videos

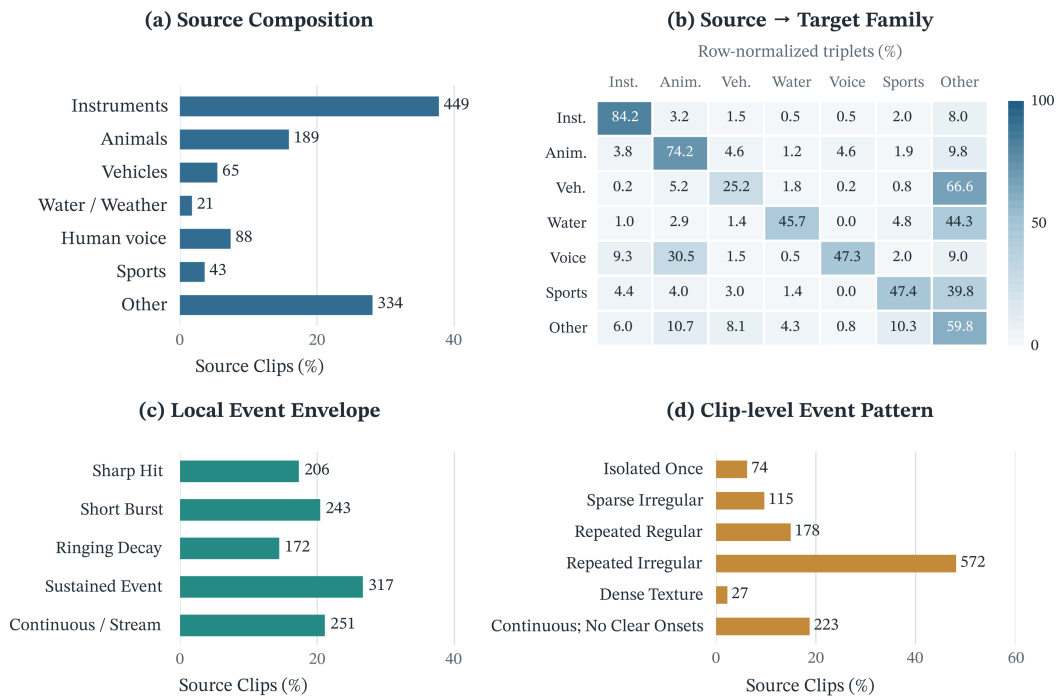

Figure 4: **Composition of VGGSound-Counterfactual. (a)** Source clips per sound family over the 1,189 retained clips. **(b)** Family of the target prompt given the family of the source, as a percentage of the 11,890 triplets within each row. **(c)** Local event envelope and **(d)** clip-level occurrence pattern of the source audio, over the same 1,189 clips. Bar labels give clip counts. Other is a catch-all family, so its row and column mix unrelated identities.

contribute more than one 10-s segment. The triplets contain 814 distinct ten-target sets. Source categories hold a median of four clips, with a range of 1 to 34 and 52 singleton categories.

Temporal compatibility concentrates target prompts within the source family (Figure 4(b)). Instrument sources draw 84.2% of their targets from instruments, and animal sources draw 74.2% from animals. Sounds that share a local envelope and an occurrence pattern often belong to the same family, and the MLLM likely also prefers targets that sound similar to the source, so the selection favors that family. We did not rebalance the family distribution, because counterfactual Foley requests in practice rarely leave the sound family of the source and a within-family target is the harder replacement. Within-family targets still change the sound identity, because the selection excludes the source caption and every caption with the same audible identity. The dog-to-sheep example of Figure 1 illustrates such a within-family replacement. Vehicle sources instead send 66.6% of their targets to Other, which mixes unrelated identities and is not a semantic family.

The two temporal descriptors differ in balance (Figure 4(c,d)). Each local envelope value holds between 14.5% and 26.7% of clips, so no envelope dominates. The occurrence pattern is concentrated instead, with repeated irregular on 572 clips and dense texture on 27. The benchmark therefore covers all five envelope values while favoring irregular repetition at the clip level.

The onset-level evaluation of Section 4.2 complements this distribution. Repeated regular patterns cover only 178 clips here, whereas every Greatest Hits clip consists of discrete stick strikes with reference onsets. The two evaluations therefore cover complementary timing regimes.

C.1.4 MLLM Prompts

We list each curation prompt verbatim. Braces mark the fields filled for each clip, and Stage 0 uses deterministic rules without a prompt. Every Gemini call uses Gemini 3.5 Flash with a medium thinking level and a JSON response schema that fixes the keys of the listed output.

Stage 1, silent video. Gemini receives the silent video at 4 fps together with the system prompt and the user prompt below. The response schema allows only yes, no, or uncertain for each of the six queries, each with one evidence sentence. A clip is rejected when any structural flag is yes or any validity criterion is no, and it goes to review when any answer is uncertain.

```
You are a strict video-dataset curator for a counterfactual video Foley benchmark.

You receive a SILENT video and a reference caption naming the intended visible sound-producing
     event. Judge only visible evidence in the video. The audio track has been removed. Never infer
      that an event occurs merely because it is named in the caption.

Inspect the complete video, including its beginning, middle, and end. In one response, evaluate
     exactly the following six queries. The first three are structural failure flags; the last
     three are source-validity criteria. Judge every query independently from observable visual
     evidence and do not add broader rejection criteria.

Structural failure flags:
1. intro_or_outro: A temporally contiguous prefix or suffix is a distinct intro or outro segment
     that replaces the main scene. Full-frame title cards, channel intros, talking-head lead-ins,
     distinct opening or closing scenes, and outro cards count. Text superimposed over an otherwise
      visible ongoing scene|including subtitles, labels, timestamps, watermarks, logos, and social-
     media handles|does not count.
2. unrelated_internal_scene: A caption-unrelated scene interrupts the relevant content inside the
     clip. A normal camera-angle change within the same scene or subject does not count.
3. multiple_scenes_in_frame: Editing places multiple different scenes in one frame, such as split-
     screen, picture-in-picture, or a collage. Multiple objects in one natural physical scene do
     not count for this structural flag; evaluate sound-source multiplicity under query 6.

Replay or looping, jump cuts, discontinuous editing, multi-take montage, and ordinary camera-angle
     changes are not structural failures by themselves. They matter only when they satisfy one of
     the definitions above.

Source-validity criteria:
4. captioned_source_identifiability: Is at least one physical sound-producing source instance
     described by the caption visually identifiable? Answer "yes" when a corresponding object or
     source instance is clearly identifiable; "no" when it is absent or only a different source is
     visible; "uncertain" when it is too small, occluded, or visually ambiguous. A group that
     matches the caption may satisfy identifiability, but it must still fail the single-instance
     requirement in query 6.
5. sound_producing_activity: Is at least one identified captioned source instance visibly
     performing an activity or operating state consistent with producing the captioned sound?
     Answer "yes" when this is clearly visible; "no" when every identified captioned source is
     visibly inactive or its activity contradicts the caption; "uncertain" when the source or
     activity state cannot be judged. If captioned_source_identifiability is not "yes", this answer
      must be "uncertain", never a guessed "no".
6. unique_active_sound_source: Does the clip contain exactly one independently identifiable
     physical sound-producing instance or emitter? Answer "no" whenever two or more distinct
     instances are visibly active or plausibly producing the captioned sound, even when every
     instance belongs to the same semantic class and matches the same reference caption. Examples
     that must be "no" include two barking dogs, multiple lions roaring, multiple musicians playing
      the same instrument, multiple people making the same vocal sound, and multiple machines of
     the same type operating. A plural or group-valued caption does not waive this exactly-one-
     instance requirement. Also answer "no" when any additional source of a different class is
     active and plausibly audible. Answer "yes" only when the visual evidence supports exactly one
     physical emitter and no other active plausibly audible source. Answer "uncertain" when the
     number of independently operating emitters cannot be determined reliably; never assume one
     merely because overlapping instances are hard to distinguish. Multiple visible components of
     one inseparable physical emitter or event do not by themselves count as multiple sources.

For every query, answer exactly "yes", "no", or "uncertain" and provide one short sentence citing
     observable visual evidence. Use "uncertain" rather than guessing. Never infer or describe
     unheard audio. Do not output a final keep/reject decision or numeric confidence.

Return only the structured response required by the supplied schema.
```

```
Sample ID: "{sample_id}"
Reference caption: "{caption}"

Review the attached silent video over its full duration. Evaluate the three structural failure
     flags and three source-validity criteria in one response using the system definitions.
```

Stage 1, audio only. Qwen3-Omni-30B-A3B-Instruct receives the first audio stream as mono 16-kHz PCM with the prompt below and decodes greedily. The prompt is the only text input, and the caption is inserted as a JSON string. Any no answer rejects the clip, and any uncertain answer sends it to review.

```
You are evaluating audio for a single-source Foley dataset. You have access only to the supplied
     audio and the reference caption; no video, frames, filename semantics, or visual evidence is
     available. Reference caption: "{caption}"

Answer three criteria independently from audible evidence.

1. captioned_sound_presence: Is the sound event described by the caption audibly present? yes =
     identifiable audible evidence; no = absent or a different sound; uncertain = too faint, brief,
      masked, or ambiguous.
2. captioned_sound_dominance: Is the captioned sound the dominant foreground event over the clip or
      during its relevant active intervals? yes = principal foreground sound; no = another event
     consistently dominates or it is only minor background; uncertain = dominance changes or cannot
      be judged. If captioned_sound_presence is not yes, this answer MUST be uncertain. Low-level
     room tone, wind, or recording noise may coexist with a dominant event.
3. no_competing_audible_source: Does the audio support exactly one independently distinguishable
     physical sound-producing instance or emitter? yes = the audible evidence supports exactly one
     emitter and no other semantically identifiable source; no = two or more independent emitters
     are audible; uncertain = the number of emitters cannot be judged reliably from the mix. Two or
      more emitters MUST be no even when every emitter belongs to the same semantic class and
     matches the same reference caption. Examples that MUST be no include two or more dogs barking,
      multiple lions roaring, multiple musicians playing the same instrument, multiple people
     making the same vocal sound, or multiple engines operating. A plural or group-valued caption
     does not waive the exactly-one-instance requirement. Repeated or overlapping events, distinct
     spatial positions, different timbres, or independent onset patterns may support multiple
     emitters; when they do, answer no. Do not assume one merely because a mono or dense mix makes
     counting difficult; answer uncertain instead. Echo, reverberation, or multiple sound
     components produced by one physical emitter do not alone count as multiple sources. Background
      music, voice-over, and any identifiable source of a different class are also competing
     sources. Do not count low-level unidentifiable noise.

For each criterion, give a concise evidence string describing what is or is not heard. Do not infer
      from the caption alone. Do not provide a dataset membership decision, confidence, score,
     Markdown, or extra keys. Return exactly one JSON object in this schema:
{"answers":{"captioned_sound_presence":{"answer":"yes|no|uncertain","evidence":"..."},"
     captioned_sound_dominance":{"answer":"yes|no|uncertain","evidence":"..."},"
     no_competing_audible_source":{"answer":"yes|no|uncertain","evidence":"..."}}}
```

Stage 2, joint audio and video. Gemini receives the original clip with its audio at 4 fps together with the system prompt and the user prompt below. A yes answer keeps the clip provisionally, and a no or uncertain answer sends it to human review.

```
You are a strict audio-video dataset curator for a counterfactual video Foley benchmark.

You receive an ORIGINAL VIDEO WITH ITS ORIGINAL AUDIO and a reference caption naming the intended
     visible and audible event. Inspect the complete clip, including its beginning, middle, and end
     . Evaluate exactly one joint criterion: audio_video_correspondence.

Question: Does the audible captioned event correspond temporally and causally to the visible
     captioned event in this clip?

Answer "yes" only when all of the following hold:
- the audible identity matches the captioned visible source;
- audible onsets, offsets, repetitions, impacts, or continuous operation are compatible with the
     visible activity;
- there is no strong evidence that the audio comes from an unrelated scene, off-screen event,
     dubbing error, or temporally shifted track.

Answer "no" when any clear joint mismatch holds:
- audible events systematically occur without corresponding visible activity;
- visible sound-producing actions systematically lack the expected audible events;
- repeated impacts or onsets clearly disagree with visible repetitions;
- the audio belongs to a different source or scene;
- editing, dubbing, or temporal displacement breaks causal correspondence.

Answer "uncertain" when correspondence cannot be judged reliably, including continuous events
     without a reliable visual onset marker, occlusion or camera motion that prevents temporal
     judgment, weak visible articulation, or a possible mismatch below the temporal resolution of
     the supplied video sampling.

Natural acoustic propagation delay, reverberation, and ordinary codec latency are not mismatches
     unless large enough to break event-level correspondence. Judge only joint audio-video
     correspondence. Stage 1 already evaluated independent visual validity and audio purity; do not
      rescue or re-evaluate those filters. Do not infer correspondence from the caption alone.

Return exactly "yes", "no", or "uncertain", one concise evidence sentence grounded in both what is
     seen and what is heard, and relevant time intervals in seconds when a mismatch or ambiguity is
      localized. Use an empty interval list when no localized interval is needed. Do not output a
```

```
     final dataset membership decision, numeric confidence, hidden reasoning, Markdown, or extra
     fields.

Return only the structured response required by the supplied schema.
```

```
Sample ID: "{sample_id}"
Reference caption: "{caption}"

Review the attached original audio-video clip over its full duration. Evaluate only
     audio_video_correspondence using the system definition.
```

Table 5: Few-shot records inserted as {examples} in the target-selection prompt. Each record is serialized as one JSON object with the listed axes, high confidence, and the ten targets as exact matches. The reason field of every target reads "*target* can reproduce the demonstrated attacks and clip-level pattern." Targets are separated by ·.

Source caption	Axis 1 Axis 2	Audio evidence	Ten targets
playing acoustic guitar	ringing_decay repeated_irregular	Several plucked notes recur at uneven intervals, each with a resonant decay.	playing electric guitar · playing ukulele · playing bass guitar · playing steel guitar, slide guitar · playing mandolin · playing banjo · playing harp · playing zither · playing sitar · playing violin, fiddle
playing badminton	sharp_hit repeated_irregular	Countable racket impacts recur throughout the clip with uneven gaps.	playing tennis · playing table tennis · playing squash · playing volleyball · basketball bounce · golf driving · playing darts · striking pool · mouse clicking · people finger snapping
chopping wood	sharp_hit repeated_irregular	Dry impacts with very short tails recur many times at non-periodic intervals.	hammering nails · cap gun shooting · woodpecker pecking tree · playing snare drum · playing castanets · machine gun shooting · typing on typewriter · people clapping · playing table tennis · bowling impact

Temporal labels and target selection. Gemini receives the clip audio as mono 16-kHz PCM with the prompt below at temperature 0. The field {vocabulary} is the JSON list of the 309 VGGSound captions, and {examples} holds the three few-shot records of Table 5. The validator described above then checks every returned target against this vocabulary.

```
Analyze primarily the supplied audio, using the source caption only to identify which audible event
      matters.
Source sample_id: {sample_id}
Source caption (copy exactly): {source_caption}

Axis 1, shape of one event (choose exactly one):
- sharp_hit: near-instant attack with little body or tail
- short_burst: bounded brief event with audible body but no prominent ring
- ringing_decay: bounded excitation followed by salient resonant decay
- sustained_event: one bounded event held for a meaningful duration, with onset and offset
- continuous_or_stream: ongoing flow, drone, texture, or stream without a useful single-event
     envelope

Axis 2, how events populate the whole clip (choose exactly one):
- isolated_once: one salient occurrence
- sparse_irregular: a small number of loose separated occurrences; repetition is not dominant
- repeated_regular: clearly recurring events with approximately regular spacing
- repeated_irregular: clearly recurring, countable events with irregular spacing; repetition is
     dominant
- dense_texture: many rapid or overlapping micro-events merging into a texture
- continuous_no_clear_onsets: persistent activity without separable salient onsets

Return exactly ten ranked counterfactual targets. Copy every target verbatim from the frozen
     vocabulary below. Exclude the source itself, punctuation variants, synonyms, and the same
     audible identity. Prefer exact two-axis matches; otherwise use compatible. Keep identities
     diverse and recognizable. Report expected axes and exact/compatible for every target. Do not
     invent or repair vocabulary entries. The audio_evidence must describe concrete heard
     properties of this query clip (such as count, spacing, duration, attack, or decay).

Few-shot JSON examples (format demonstrations; do not copy their targets blindly):
{examples}

Frozen vocabulary:
```

```
{vocabulary}

Return one JSON object only, with exactly these keys and structure:
{"sample_id":"{sample_id}","source_caption":"{source_caption}","source_temporal":{"axis1":"...","
     axis2":"...","confidence":"low|medium|high","audio_evidence":"one concise clip-grounded
     sentence"},"targets":[{"rank":1,"target_caption":"...","axis1":"...","axis2":"...","
     temporal_match":"exact|compatible","reason":"one concise sentence"}, ... rank 10]}
Do not include markdown, hidden reasoning, or extra keys.
```

C.2 Compared Systems and Guidance Methods

Table 6: Denoising settings of the compared systems. **TAP-Foley** and CounterFlow use the backbone settings in the first two rows, while Default CFG and negative-source CFG double the steps in the within-backbone comparison. Every system uses seed 42. w is the guidance scale of each system's own CFG, which its negative-source variant also uses. Output is the native duration and sample rate before the 8-s evaluation crop.

System	Checkpoint	Solver	Steps	w	Output
MMAudio (Cheng et al., 2025)	large-44k v2	Euler	25	4.5	8 s, 44.1 kHz
Omni2Sound (Dai et al., 2026)	released	deterministic v-ODE	100	2.5	8 s, 16 kHz
CAFA (Benita et al., 2025)	AV-CLIP adapter	DPM++ 3M SDE	50	7.0	10 s, 44.1 kHz
ControlFoley (Yang et al., 2026b)	large-44k	Euler	25	4.5	8 s, 44.1 kHz
FoleyCrafter (Zhang et al., 2026)	Auffusion with adapters	released	25	7.5	8 s, 16 kHz
ReWaS (Jeong et al., 2025)	AudioLDM-M with energy control	DDIM	200	3.0	5 s, 16 kHz
AudioX (Tian et al., 2026)	AudioX-MAF	DPM++ 3M SDE	100	7.0	8 s, 44.1 kHz

Backbones for TAP-Foley. **TAP-Foley** runs on MMAudio large-44k v2 and Omni2Sound at the published step count and guidance scale of each backbone (Table 6). MMAudio keeps its Euler solver, and Omni2Sound replaces its default stochastic solver with the deterministic v-ODE solver. Each backbone uses one guidance scale for the joint, source-only, and target-only branches, which matches the equal-scale assumption of Appendix B.1. The native time grids are $t_s = s/N$ for MMAudio and $t_s = 1 - s/N$ for Omni2Sound. On MMAudio, the text-only branches replace the visual and synchronization features with the learned null embeddings. On Omni2Sound, the joint branch receives the CLIP visual features, the Synchformer features, and the target prompt. Its text-only branches follow the text-to-audio input format of the backbone, which omits the CLIP features and zero-pads the Synchformer features while keeping the duration tokens.

Seeds. All systems and guidance methods use seed 42. On both backbones, every guidance method thus starts a triplet from the same initial noise, and the ten targets of a clip share this noise as well. Differences between guidance methods on a backbone therefore do not stem from the initial noise.

Existing VT2A systems. We run the five existing VT2A systems from their public checkpoints with their released solvers, step counts, and guidance scales. The negative-source variant of each system replaces the null branch of its CFG with the same video and the source prompt, so that the two branches differ only in the text. Each variant thus applies the update $\boldsymbol{u} = \boldsymbol{f}_{VS} + w(\boldsymbol{f}_{VT} - \boldsymbol{f}_{VS})$ of Equation 24 to the native prediction of the system, whether velocity or noise, at its native scale w.

Guidance methods at comparable NFE. The guidance-method comparison of Table 2(b) equalizes the number of function evaluations (NFE) within each backbone as closely as the native step counts allow. **TAP-Foley** queries the backbone four times per step, so it uses 100 NFE on MMAudio and 400 NFE on Omni2Sound. Default CFG and negative-source CFG query it twice per step, so they run 50 and 200 steps to reach the same budget. CounterFlow keeps its native step count and queries four branches before its transition and three after it, which totals 92 and 368 NFE at the published transition, 8% fewer than **TAP-Foley**. Its transition sweep gives 88 and 352 NFE at 52% of the steps and 100 and 400 NFE at 100%, where the second phase is empty. All Omni2Sound guidance methods in this comparison use the deterministic v-ODE solver.

CounterFlow configuration. CounterFlow uses its published configuration on MMAudio and a ratio-preserving transfer of it on Omni2Sound. On MMAudio, the configuration uses 25 steps,

transition step 17, $w_{\text{vid}} = 3.0$, $w_{\text{txt}} = 5.0$, and $w_{\text{cfg}} = 4.5$. On Omni2Sound, we transfer this configuration to the 100-step solver and the guidance scale $w = 2.5$. The transfer keeps the two-phase update, the transition fraction, and the ratio of each weight to the guidance scale, which gives

$$N_{\text{trans}} = 100 \cdot \tfrac{17}{25} = 68, \qquad w_{\text{cfg}} = 2.5 \cdot \tfrac{4.5}{4.5} = 2.5,$$
$$w_{\text{vid}} = 2.5 \cdot \tfrac{3.0}{4.5} \approx 1.667, \qquad w_{\text{txt}} = 2.5 \cdot \tfrac{5.0}{4.5} \approx 2.778. \tag{26}$$

We additionally sweep the transition step over 52%, 68%, and 100% of the steps with these weights fixed. The sweep uses steps 13, 17, and 25 on MMAudio and steps 52, 68, and 100 on Omni2Sound.

C.3 Evaluation Metrics

C.3.1 Evaluation Protocol

All metrics evaluate the first 8 s of each output, because the compared systems generate outputs of different lengths. The native outputs last 10 s for CAFA, 8 s for the other systems, and 5 s for ReWaS (Table 6). Outputs shorter than 8 s keep their decoded length, so ReWaS is scored on its full 5-s output.

We report per-triplet metrics as means over the 11,890 triplets and compute FAD and IS over the full set of outputs.

C.3.2 Sound Identity Replacement

Identity metrics test target realization and source suppression, but they do not locate either sound in time. Let a denote the generated audio, and let S and T denote the source and target prompts.

CLAP similarity and classification. CLAP-T and Clf. acc. score the output against the prompts in the LAION-CLAP embedding space (Wu et al., 2023). With the normalized audio and text embeddings E_a and E_t, we define

$$C_T = \langle E_a(a), E_t(T) \rangle, \qquad C_S = \langle E_a(a), E_t(S) \rangle. \tag{27}$$

We use the HTSAT-base checkpoint trained on music, speech, and AudioSet, with audio resampled to 48 kHz. CLAP-T reports C_T, and higher is better. Source–target classification accuracy is the mean of $\mathbf{1}[C_T > C_S]$ (Benita et al., 2025), with ties counted as failures. It is a binary comparison between the two supplied prompts rather than a classification over the full sound vocabulary. Neither score rules out a source–target mixture, because a mixture can still place C_T above C_S.

FLAM evidence and contrast. FLAM-T, ΔFLAM, and the (+) Ratio score the peak frame-level evidence for the target and its excess over the source. Let $f(a, q, k)$ denote the frame-level evidence of FLAM (Wu et al., 2025) for query q at frame k, and let $F(a, q) = \max_k f(a, q, k)$. We use the OpenFLAM v1-base model at 48 kHz, zero-pad the 8-s audio to the 10-s model input, and take the maximum over frames inside the unpadded audio. We report

$$\text{FLAM-T} = F(a, T), \qquad \Delta\text{FLAM} = F(a, T) - F(a, S), \tag{28}$$

and the (+) Ratio, the fraction of triplets with $\Delta\text{FLAM} > 0$ (Lee et al., 2026). Higher values are better for all three measures. Peak pooling retains brief detections that clip-level averaging would dilute, and the source and target maxima may occur at different frames.

C.3.3 Temporal Alignment

DeSync. DeSync measures the audio–visual offset that Synchformer (Iashin et al., 2024) predicts between the generated audio and the original video, and lower is better. Synchformer classifies the offset of a 4.8-s window into 21 classes from −2 s to 2 s. For an 8-s output, we score two overlapping windows covering 0–4.8 s and 3.2–8.0 s and average their absolute predicted offsets. The windows are the first and last 14 of the 0.64-s audio segments taken every 0.32 s. For ReWaS, whose decoded output lasts 5.12 s, they therefore cover 0–4.8 s and 0.32–5.12 s. The model receives no prompt, so a synchronized reproduction of the source can obtain a favorable score. DeSync therefore measures synchronization without certifying the identity of the synchronized sound.

C.3.4 Audio Distribution and Quality

Fréchet Audio Distance. FAD (Kilgour et al., 2019) compares the distribution of generated audio with that of the original recordings in the 16-kHz PANNs Cnn14 embedding space (Kong et al., 2020). The reference set is the first 8 s of the original audio of the 1,189 source clips. For generated and reference means μ_g, μ_r and covariances Σ_g, Σ_r, it is

$$\mathrm{FAD} = \|\mu_g - \mu_r\|_2^2 + \mathrm{tr}\Big(\Sigma_g + \Sigma_r - 2(\Sigma_r^{1/2}\Sigma_g\Sigma_r^{1/2})^{1/2}\Big), \tag{29}$$

and lower is better. The reference set carries the source identities, so a successful replacement moves the generated distribution away from it. A higher FAD therefore cannot be attributed to lower audio quality alone, which is why we report FAD without ranking methods by it.

Inception Score. IS measures the divergence between the class probabilities of each output and their marginal over the set,

$$\mathrm{IS} = \exp\big(\mathbb{E}_a D_{\mathrm{KL}}\big(p(y \mid a) \,\|\, p(y)\big)\big). \tag{30}$$

We use the AudioSet logits of the same PANNs model with ten shuffled splits and shuffle seed 2020. Higher IS reflects both classifier confidence and class diversity, so it does not directly measure fidelity or replacement.

Production quality. PQ is the production-quality score of the released Audiobox Aesthetics model on the first 8 s, and higher is better. PQ also varies with the sample rate and capacity of each system, so we compare it primarily within a backbone.

C.3.5 Reading the Metrics Together

No single metric establishes counterfactual success, because each one misses a failure that another detects. A synchronized reproduction of the source scores well on DeSync while failing replacement. A source–target mixture scores well on target similarity, and a temporally misplaced target scores well on clip-level identity metrics. We therefore read source-sensitive identity scores, prompt-agnostic synchronization, and audio quality together. The target-filtered onset scores on Greatest Hits and the qualitative case studies of Appendix D.6 add frame-level target and source evidence from PE-A-Frame (Vyas et al., 2026). The human evaluation of Appendix D.1 checks whether the metric gains in identity replacement and temporal alignment hold perceptually.

D SUPPLEMENT TO THE RESULTS

D.1 HUMAN EVALUATION PROTOCOL

Stimuli. The listening test covers eight systems on 20 triplets from VGGSound-Counterfactual. The systems are Default CFG, CounterFlow, and **TAP-Foley** on each backbone, together with default CAFA and default ControlFoley, and each uses the same generated audio as the automatic evaluation. The triplets contain 20 distinct videos and 20 distinct target prompts. Five sound families, namely instruments, animals, vehicles, human voice, and sports, each appear four times among the sources and four times among the targets.

Presentation. Participants rate the outputs in a web interface without knowing which system produced each output. They first complete two unscored tutorial examples. Each page shows one muted video, its source and target prompts, and the eight outputs labeled A to H without system names. The order of the eight outputs is shuffled for each triplet and participant to spread position effects across systems, while the order of the 20 triplets is fixed. Each output lasts at most 8 s, and headphones in a quiet environment are recommended. Seventeen participants rate all 160 outputs, which gives 2,720 trials in total.

Items. Each output receives three ratings on a 1–5 scale in a fixed order. Two identity items ask how clearly the listener recognizes the output as the target prompt and as the source prompt. Each identity item is rated regardless of timing, from 1 (not recognized at all) to 5 (very clearly recognized). The temporal item asks whether the audio aligns with the visible motions and events, from 1 (strongly disagree) to 5 (strongly agree). Scores keep their original direction, so lower is better for the source item.

Aggregation. The mean opinion score (MOS) of each system and item is the mean of the 17 participant means, each over 20 triplets. Under the balanced design, this equals the pooled mean of the 340 ratings. The 95% confidence interval is the two-sided Student t interval over participant means, $\mathrm{MOS} \pm t_{0.975,16}\,\mathrm{SD}/\sqrt{17}$. The interval therefore treats participants as the random unit and the 20 triplets as fixed.

D.2 ONSET-LEVEL EVALUATION ON GREATEST HITS

D.2.1 PROTOCOL

Greatest Hits tests whether generated sounds begin at the reference onsets of visible strikes, which a clip-level offset such as DeSync cannot resolve. A clip can be well synchronized overall while individual events start early or late. The discrete stick strikes of Greatest Hits (Owens et al., 2016) provide reference event times without a learned synchronization model. MMAudio (Cheng et al., 2025) and ReWaS (Jeong et al., 2025) evaluate onset alignment on this dataset when reconstructing the original sound. We extend that evaluation to the counterfactual setting, in which the requested sound identity changes while the visible strike times stay fixed.

Clips and prompts. We build 1,280 counterfactual triplets from 160 ten-second clips. The clips cover eight struck materials with 20 clips each, namely cloth, dirt, glass, metal, paper, plastic, rock, and wood. We select them manually from videos annotated with hits on each material, choosing segments with clearly visible strikes. Six clips join two shorter segments to reach ten seconds, and the 160 clips come from 156 source videos. Each clip shows stick strikes on a single material and receives the source prompt `A stick hitting {material}`. Every clip is paired with the same eight target prompts. Four are instruments, *playing snare drum*, *playing bongo*, *playing glockenspiel*, and *playing tambourine*. The other four are *cap gun shooting*, *hammering nails*, *dog barking*, and *people finger snapping*. Each target is a short sound that can follow a discrete strike, so every triplet admits an output aligned with the visible strikes.

Compared systems. The evaluation follows the two comparisons of Section 4.1 with a smaller set of external systems. The cross-model comparison covers the two conflict-specific systems, CAFA and ControlFoley, as released and with negative-source CFG. The within-backbone comparison fixes

Table 7: Onset-level evaluation on 1,280 Greatest Hits counterfactual triplets. Onset scores count only detected onsets with PE-A-Frame evidence for the target identity, so they measure identity and timing jointly (Appendix D.2.2). Recall, precision, and F1 are the standard event-level scores over reference onsets, from a maximum one-to-one matching within a 0.3-s tolerance. Panel (a) covers the two conflict-specific systems. Panel (b) uses comparable NFE and the published transition for CounterFlow. Bold and underline mark the best and second-best values *across all rows in (a)* and *within each backbone in (b)*.

Method		Sound Identity Replacement		Onset Analysis		
		CLAP-T ↑	Clf. acc. ↑	Onset recall ↑	Onset precision ↑	Onset F1 ↑
(a) Comparison with existing VT2A systems						
Ours						
TAP-Foley	MMAudio	**0.343**	**0.991**	**0.851**	<u>0.818</u>	**0.801**
	Omni2Sound	0.297	0.948	<u>0.803</u>	0.780	0.758
Baseline						
ControlFoley		0.301	0.881	0.771	**0.833**	<u>0.769</u>
CAFA		<u>0.305</u>	0.956	0.775	0.802	0.756
Baseline (with negative-source CFG)						
ControlFoley		<u>0.305</u>	<u>0.981</u>	0.754	0.632	0.616
CAFA		0.276	0.976	0.767	0.722	0.704
(b) Comparison of inference-time guidance methods						
MMAudio						
TAP-Foley		**0.343**	**0.991**	**0.851**	<u>0.818</u>	**0.801**
CounterFlow		0.305	0.981	0.705	0.673	0.631
Default CFG		0.295	0.877	0.746	**0.830**	<u>0.751</u>
Negative-source CFG		<u>0.324</u>	<u>0.984</u>	<u>0.798</u>	0.724	0.716
Omni2Sound						
TAP-Foley		**0.297**	**0.948**	**0.803**	**0.780**	**0.758**
CounterFlow		<u>0.249</u>	<u>0.913</u>	<u>0.713</u>	0.632	<u>0.612</u>
Default CFG		0.174	0.541	0.495	0.574	0.506
Negative-source CFG		0.184	0.823	0.603	<u>0.639</u>	0.581

backbone and seed and uses a comparable number of function evaluations across guidance methods, as in Table 2(b).

D.2.2 Target-Filtered Onset Metrics

A target-conditioned gate between onset detection and temporal matching credits an onset only when it carries target evidence. Without the gate, an onset detector credits a correctly timed source sound even when the requested target is absent. The gate admits an onset only when PE-A-Frame finds local evidence for the target prompt around it. The onset detector and candidate suppression follow the Greatest Hits evaluation of MMAudio (Cheng et al., 2025) without change.

Target-conditioned gate. The gate thresholds the PE-A-Frame target evidence in a short neighborhood of each candidate onset. Let $p_T[k]$ denote the sigmoid output of the frozen PE-A-Frame-Large model (Vyas et al., 2026) for the generated audio and target prompt T at native frame k. The scorer uses its checkpoint-defined audio preprocessing and keeps the frames that cover the unpadded audio, with native spacing $\Delta = 0.04\,\text{s}$. For a candidate onset at time τ, we define

$$j(\tau) = \left\lfloor \frac{\tau}{\Delta} \right\rfloor, \qquad q_T(\tau) = \max_{\substack{0 \le k < K \\ |k - j(\tau)| \le 2}} p_T[k], \qquad \mathcal{O}_T = \{\tau \in \mathcal{O} : q_T(\tau) > 0.3\}, \tag{31}$$

where $\mathcal{O}$ contains the onset candidates after suppression and K is the number of valid frames. An empty neighborhood has score zero. The threshold 0.3 follows the event-span operating point of SAM Audio (Shi et al., 2026, Sec. 6.1), which attributes it to the PE-A-Frame authors. The five-frame neighborhood spans at most 200 ms and is clipped at the audio boundaries.

The gate decides which onsets are eligible but leaves their timing unchanged. It removes candidates without local target evidence, and it neither moves the retained timestamps to a PE-A-Frame peak

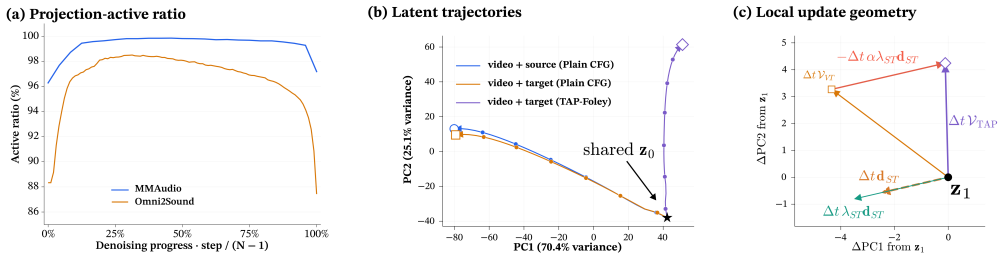

Figure 5: **Projection dynamics of TAP-Foley. (a)** Projection-active ratio, the fraction of the 11,890 triplets with $\lambda_{ST} > 0$, at each step of denoising for MMAudio and Omni2Sound at $\alpha = 1.2$. **(b)** Latent trajectories of the video+source and video+target joint predictions under Plain CFG and of **TAP-Foley**, from a shared initial latent $\boldsymbol{z}_0$, projected onto their top two principal components, for one MMAudio triplet (source "machine gun shooting," target "chopping wood," sample `OnRh-5IRG3A_000000`, seed 42, 25-step Euler solver, $\alpha = 1.2$). **(c)** Local update geometry at step 1 of the same triplet, in the same principal-component coordinates as (b), decomposing $\boldsymbol{\mathcal{V}}_{\mathrm{TAP}} = \boldsymbol{\mathcal{V}}_{VT} - \alpha\lambda_{ST}\boldsymbol{d}_{ST}$ into the uncorrected joint velocity $\boldsymbol{\mathcal{V}}_{VT}$, the contrast axis $\boldsymbol{d}_{ST}$, its projection $\lambda_{ST}\boldsymbol{d}_{ST}$, and the deployed correction $\alpha\lambda_{ST}\boldsymbol{d}_{ST}$.

nor filters the reference events. Sound identity thus determines eligibility, while the detected onset time alone determines temporal alignment.

Matching and reported scores. Retained onsets are matched to reference onsets within a 0.3 s tolerance over the first eight seconds. Reference onsets are detected automatically from the original ten-second audio and restricted to $[0, 8)$ s, and all targets and methods of a video share them. Generated audio is cropped before detection, and matching is the maximum-cardinality one-to-one assignment between retained predictions and reference events with $|\tau_{\mathrm{pred}} - \tau_{\mathrm{ref}}| < 0.3$ s. For H matches, G reference events, and P retained predictions, recall is H/G, precision is H/P, and F1 is $2H/(P+G)$. The MMAudio evaluation refers to recall as onset accuracy. Every reference event enters G and every retained prediction enters P, so a missed reference lowers recall and an extra prediction lowers precision. We average each per-triplet metric equally over the 1,280 triplets.

D.2.3 Results

TAP-Foley on MMAudio attains the highest onset recall in the cross-model comparison of Table 7(a), together with the highest identity scores. Its onset recall is 0.851, with a CLAP-T of 0.343 and a classification accuracy of 0.991. The Omni2Sound instantiation reaches the second-highest onset recall of 0.803.

Negative-source CFG changes onset recall in opposite directions for the external systems and within each backbone. It raises classification accuracy for both external systems and lowers their onset recall only slightly, from 0.771 to 0.754 for ControlFoley and from 0.775 to 0.767 for CAFA. Within each backbone, negative-source CFG instead raises onset recall over Default CFG, so its effect on onset timing is not consistent across systems.

Within each backbone, **TAP-Foley** improves onset recall over CounterFlow, from 0.705 to 0.851 on MMAudio and from 0.713 to 0.803 on Omni2Sound. Standard F1 moves in the same direction, from 0.631 to 0.801 on MMAudio and from 0.612 to 0.758 on Omni2Sound. This gap mirrors the DeSync gap in Table 2(b), consistent with CounterFlow dropping video conditioning at its transition step. Among guidance methods, **TAP-Foley** attains the highest precision on Omni2Sound, and Default CFG leads on MMAudio by emitting fewer onsets and so committing fewer false positives.

D.3 Projection Activity and Correction Geometry

The target-anchored constraint is violated at nearly every step, even though each corrected velocity lies inside $\mathbb{H}_T$, so the video moves the joint velocity back across the anchor at the next forward pass. We evaluate the uncorrected joint velocity $\boldsymbol{\mathcal{V}}_{VT}$ at each state that the **TAP-Foley** solver visits and

Table 8: Full metrics of the **TAP-Foley** correction-scale sweep and the CounterFlow transition sweep in Figure 2(b). The CounterFlow percentage is the transition step relative to the total steps, and 100% keeps video conditioning at every step. ‡ marks the setting whose ΔFLAM matches that of CounterFlow at its published transition. All rows share the initial noise on the 11,890 triplets, and †FAD is not ranked.

Setting	CLAP-T ↑	Clf. acc. ↑	FLAM-T ↑	ΔFLAM ↑	(+) Ratio ↑	DeSync ↓	IS ↑	PQ ↑	FAD†
MMAudio									
$\alpha = 1.0$	0.290	0.854	0.953	0.334	0.871	0.394	19.50	6.20	19.9
$\alpha = 1.2$ (main)	0.296	0.885	0.959	0.384	0.900	0.427	20.64	6.21	21.6
$\alpha = 1.4$	0.299	0.900	0.960	0.418	0.914	0.452	21.04	6.21	22.9
$\alpha = 1.6$	0.300	0.912	0.961	0.445	0.924	0.477	21.22	6.22	23.7
$\alpha = 1.8$	0.300	0.917	0.962	0.462	0.932	0.507	21.54	6.22	24.3
$\alpha = 2.0^{\ddagger}$	0.298	0.925	0.960	0.474	0.934	0.527	21.39	6.20	24.6
CounterFlow, 52%	0.293	0.923	0.949	0.495	0.936	0.730	19.98	6.19	24.5
CounterFlow, 68% (published)	0.292	0.916	0.949	0.475	0.927	0.649	19.75	6.16	23.7
CounterFlow, 100% (no drop)	0.290	0.907	0.947	0.450	0.919	0.579	19.26	6.13	22.9
Omni2Sound									
$\alpha = 1.0^{\ddagger}$	0.232	0.770	0.878	0.224	0.761	0.462	17.75	6.25	23.2
$\alpha = 1.2$ (main)	0.241	0.829	0.887	0.299	0.821	0.510	19.83	6.31	24.5
$\alpha = 1.4$	0.242	0.861	0.886	0.346	0.849	0.550	21.01	6.36	25.6
$\alpha = 1.6$	0.241	0.876	0.882	0.366	0.868	0.582	21.49	6.40	26.4
$\alpha = 1.8$	0.240	0.887	0.879	0.380	0.875	0.619	21.85	6.44	26.9
$\alpha = 2.0$	0.238	0.890	0.875	0.391	0.881	0.646	22.07	6.46	27.4
CounterFlow, 52%	0.218	0.794	0.866	0.263	0.772	0.627	15.76	6.08	24.0
CounterFlow, 68% (published)	0.213	0.765	0.862	0.226	0.740	0.576	15.00	6.06	23.7
CounterFlow, 100% (no drop)	0.208	0.738	0.860	0.198	0.713	0.539	14.61	6.04	23.4

call a triplet-step projection-active when $\lambda_{ST} > 0$. On MMAudio, the projection-active ratio starts at 96.3%, peaks at 99.9% at step 11, and ends at 97.2% (Figure 5(a)). On Omni2Sound, it starts at 88.3%, peaks at 98.5% at step 31, and ends at 87.4%. These violations therefore arise from the next forward pass with the video rather than from the previous correction.

Raising α from 1.0 to 1.2 strengthens replacement while leaving the projection-active ratio nearly unchanged. With $\alpha = 1.0$, the ratio stays between 96.3% and 99.9% on MMAudio and between 86.9% and 98.3% on Omni2Sound, close to the values at $\alpha = 1.2$. Over the same change, ΔFLAM increases from 0.334 to 0.384 on MMAudio and from 0.224 to 0.299 on Omni2Sound. DeSync rises from 0.394 to 0.427 and from 0.462 to 0.510, consistent with the larger departure along $\boldsymbol{d}_{ST}$ anticipated in Section 2.2. Figure 2 and Appendix D.4 trace this trade-off over the full range of α.

Figure 5(b) visualizes this pull toward the source identity directly. On one MMAudio triplet (source "machine gun shooting," target "chopping wood," sample `OnRh-5IRG3A_000000`), we track the latent $\boldsymbol{z}_s$ of three settings from a shared initial latent $\boldsymbol{z}_0$ and project each trajectory onto its top two principal components. Plain CFG denotes Default CFG with either the source or the target prompt. The video+source and video+target joint predictions under Plain CFG follow nearly the same path throughout denoising, so conditioning on the target prompt alone does not steer the trajectory away from the source identity. **TAP-Foley** departs from this shared path from an early step and reaches a distinct final latent, so the projection changes the denoising path rather than only its endpoint.

Figure 5(c) decomposes the correction at step 1 of the same triplet into its geometric components, in the same principal-component coordinates as panel (b). From the uncorrected joint velocity $\boldsymbol{\mathcal{V}}_{VT}$, subtracting the projection $\lambda_{ST}\boldsymbol{d}_{ST}$ of Equation 4 reaches the boundary of $\mathbb{H}_T$, and subtracting the deployed correction $\alpha\lambda_{ST}\boldsymbol{d}_{ST}$ yields $\boldsymbol{\mathcal{V}}_{\mathrm{TAP}} = \boldsymbol{\mathcal{V}}_{VT} - \alpha\lambda_{ST}\boldsymbol{d}_{ST}$ of Equation 5.

D.4 Correction-Scale Sweep

Raising α increases ΔFLAM and DeSync monotonically on both backbones, so α acts as a single knob along the replacement–alignment trade-off. Table 8 lists all metrics for the two sweeps plotted in Figure 2(b). Clf. acc. and (+) Ratio also rise with α, and IS rises up to $\alpha = 1.8$ on MMAudio and through $\alpha = 2.0$ on Omni2Sound. CLAP-T and FLAM-T change by at most 0.012 across the sweep.

Where the two sweeps overlap in ΔFLAM, **TAP-Foley** reaches a lower DeSync than CounterFlow. A later CounterFlow transition lowers DeSync and ΔFLAM together, and Clf. acc., (+) Ratio, and IS

Table 9: Paired comparison of **TAP-Foley** against the mean-matched constant weight of Table 4 on the 11,890 triplets. Differences are **TAP-Foley** minus the constant, with 95% confidence intervals from 100,000 video-level bootstrap resamples and p-values from a two-sided paired permutation test over the 1,189 video-level mean differences, Holm-corrected over the six metrics of each backbone.

	MMAudio			Omni2Sound		
Metric	Diff.	95% CI	p	Diff.	95% CI	p
CLAP-T ↑	$+0.012$	$[+0.010, +0.013]$	$< 10^{-3}$	$+0.002$	$[+0.001, +0.004]$	0.019
Clf. acc. ↑	$+0.039$	$[+0.032, +0.045]$	$< 10^{-3}$	$+0.011$	$[+0.004, +0.019]$	0.013
FLAM-T ↑	$+0.012$	$[+0.009, +0.015]$	$< 10^{-3}$	$+0.003$	$[-0.001, +0.007]$	0.224
ΔFLAM ↑	$+0.044$	$[+0.036, +0.051]$	$< 10^{-3}$	$+0.020$	$[+0.011, +0.028]$	$< 10^{-3}$
(+) Ratio ↑	$+0.041$	$[+0.034, +0.047]$	$< 10^{-3}$	$+0.024$	$[+0.016, +0.032]$	$< 10^{-3}$
DeSync ↓	$+0.018$	$[+0.011, +0.026]$	$< 10^{-3}$	$+0.007$	$[-0.002, +0.016]$	0.224

fall with them. The settings marked ‡ match the ΔFLAM of CounterFlow at its published transition within 0.002. At these settings, **TAP-Foley** matches or exceeds CounterFlow on CLAP-T, Clf. acc., FLAM-T, (+) Ratio, IS, and PQ, and it lowers DeSync by 19% and 20%. Against the no-drop setting, which uses the same NFE as **TAP-Foley**, $\alpha = 1.8$ on MMAudio and $\alpha = 1.0$ on Omni2Sound reach a higher ΔFLAM with 12% and 14% lower DeSync.

D.5 Mean-Matched Constant Weight

The constant weight in Table 4 is $\alpha\bar{\lambda}_{ST}$, where $\bar{\lambda}_{ST}$ is the mean of λ_{ST} over the 11,890 triplets and all steps of the main run at $\alpha = 1.2$, including steps with $\lambda_{ST} = 0$, which gives 0.77 on MMAudio and 1.82 on Omni2Sound. The control applies this constant at every step without the gate of Equation 3, so it also moves velocities that already lie inside $\mathbb{H}_T$. The scale that each backbone needs also differs by a factor of 2.4, so a constant must be set per backbone, whereas both backbones use the same α.

Table 9 reports the paired differences. Target evidence is similar under both weights, with CLAP-T and FLAM-T within 0.012 and the FLAM-T difference on Omni2Sound not significant. The metrics that reflect source suppression, namely Clf. acc., ΔFLAM, and (+) Ratio, favor the per-step weight on both backbones after correction. The constant applies the same correction at every step regardless of how far the joint velocity lies outside $\mathbb{H}_T$, so it removes less of the source where the video steers strongly toward it, which is consistent with the weaker suppression. The constant lowers DeSync by 0.018 on MMAudio and by 0.007 on Omni2Sound, where the difference is not significant.

D.6 Qualitative Case Studies

Two clips illustrate how each method distributes source and target evidence over time (Figure 6). We score each output against the source and target prompts with PE-A-Frame-Large (Vyas et al., 2026) at 40-ms frames over the first 8 s. Each curve is smoothed with a three-frame moving average for display. In the first clip, chopping wood replaces typing on a typewriter, and in the second, playing table tennis replaces typing on a computer keyboard.

TAP-Foley keeps source evidence low on both backbones while its target evidence follows the activity of the original audio. In both clips, the target curve rises within each gray interval and falls in the gaps between intervals. The target sound therefore inherits the timing of the visible typing without the typing sound itself.

Default CAFA and ControlFoley retain the source identity. On the typewriter clip, their source evidence follows the activity intervals while target evidence stays low, so they reproduce the typewriter sound. On the keyboard clip, their target evidence rises together with substantial source evidence, and the two ControlFoley curves nearly coincide, so the outputs mix both identities.

Negative-source CFG and CounterFlow lower source evidence but lose target evidence in part of a clip. CAFA with negative-source CFG leaves both identities below the threshold on the typewriter clip, and ControlFoley with negative-source CFG weakens the target after about 3 s on the keyboard clip. CounterFlow on Omni2Sound keeps target evidence near the source level during the first 5 s of the typewriter clip. CounterFlow on MMAudio loses the target after about 4.7 s on the keyboard clip,

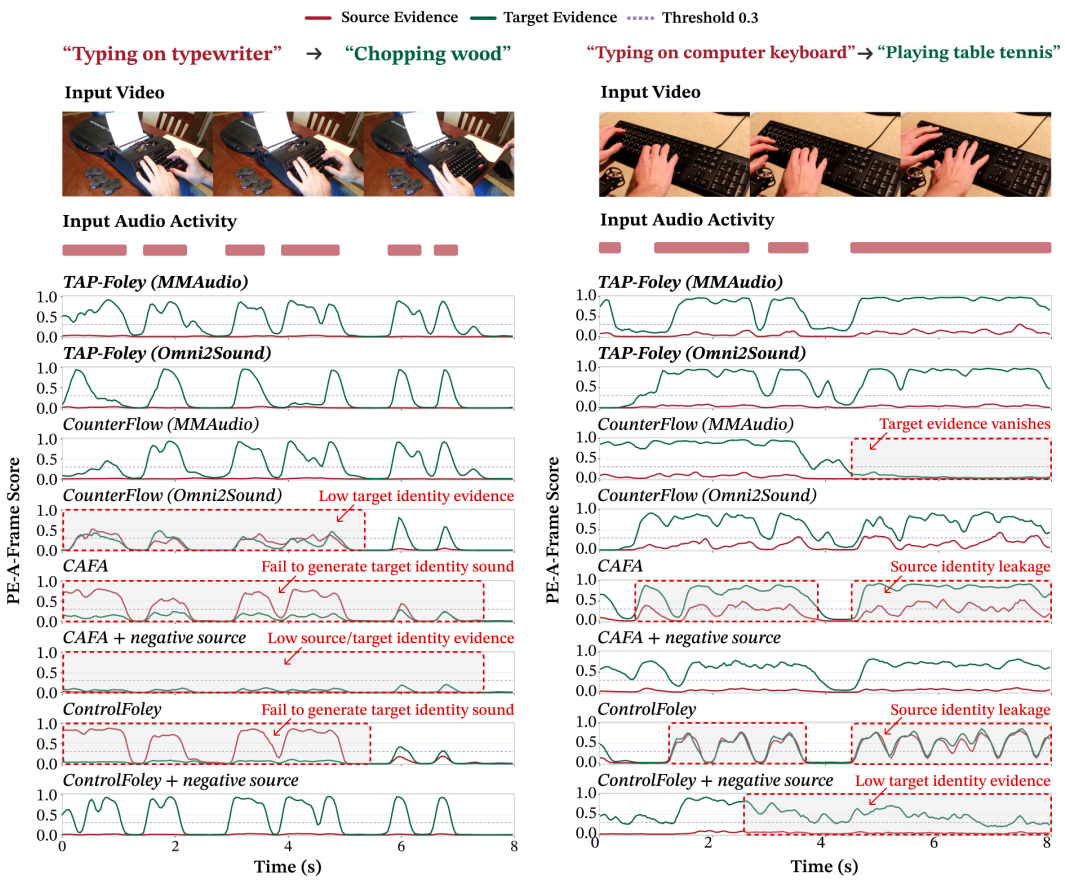

Figure 6: **Frame-level source and target identity evidence.** PE-A-Frame evidence for the source (red) and target (green) prompts for two clips, with the prompts above each column and gray bars marking activity in the original audio. The dotted line marks the 0.3 threshold, and red boxes mark failure intervals.

although the typing continues. These failures vary across clips, since each negative-source variant tracks the target on the other clip, whereas **TAP-Foley** avoids both failure modes on both clips.

E SUPPLEMENT TO THE CONCLUSION

E.1 LIMITATIONS AND FUTURE WORK

The evaluation data bound the scope of our evidence. Neither VGGSound-Counterfactual nor Greatest Hits covers multi-source scenes, out-of-domain footage, or arbitrary user prompts. Temporal descriptors select target prompts by category, without timestamp annotations, so they do not guarantee that every source–target pair is perceptually compatible.

The method itself also has limits in what it guarantees and what it requires. The projection leaves every component orthogonal to $\boldsymbol{d}_{ST}$ unchanged, but it does not certify that these components carry the timing. The correction scale $\alpha = 1.2$ is an empirical choice tested on two backbones, and the method requires the source prompt S at inference.

These limits suggest several directions for future work. A larger listening test would strengthen the perceptual evidence. A family-stratified analysis would further quantify the validity of the benchmark. On the method side, extending the constraint to multi-source scenes and reducing the number of backbone queries per step remain open.