Step-by-Step Video-to-Audio Synthesis via Negative Audio Guidance

Akio Hayakawa[1], Masato Ishii[1], Takashi Shibuya[1], and Yuki Mitsufuji[1,2]

[1] Sony AI, Tokyo, Japan
[2] Sony Group Corporation, Tokyo, Japan
{akio.hayakawa,masato.a.ishii,takashi.tak.shibuya,yuhki.mitsufuji}@sony.com

Abstract. We propose a step-by-step video-to-audio (V2A) generation method that provides finer control over the generation process and more realistic audio synthesis. Inspired by traditional Foley workflows, our approach enables incremental generation of *complementary* sounds, allowing users to author multiple sound events induced by a video. To avoid the need for costly multi-reference video-audio datasets, each generation step is formulated as a negatively guided V2A process that discourages duplication of sounds already present in previously generated tracks. The guidance model is trained by finetuning a pre-trained V2A model on audio pairs from non-overlapping segments of the same video, encouraging it to leverage acoustic context while remaining visually grounded, and enabling training with standard single-reference audiovisual datasets. Objective and subjective evaluations demonstrate that our method enhances the separability of generated sounds at each step and improves the overall quality of the final composite audio, outperforming existing baselines. Our project page is available at: `https://ahykw.github.io/sbsv2a/`.

Keywords: Video-to-audio generation · Guided diffusion · Flow matching · Controllable generation

1 Introduction

Generating realistic audio signals that align seamlessly with given visual content is a process referred to as Foley [1] in film or game production. In traditional workflows, Foley artists begin with field-recorded or library sounds and incrementally layer in missing elements (e.g., footsteps or fabric movements) to enhance audio realism. While essential for high-quality audiovisual content, this workflow is labor-intensive and time-consuming because even short clips often contain numerous audible events.

Recent video-to-audio (V2A) models [9, 29, 30, 35, 44, 45, 47] show promise for automating this workflow. These models produce high-quality audio that semantically and temporally aligns with input videos. However, most models generate an entire track in a single pass and do not offer a mechanism for incremental refinement (i.e., supplementing sounds missing in the generated results).

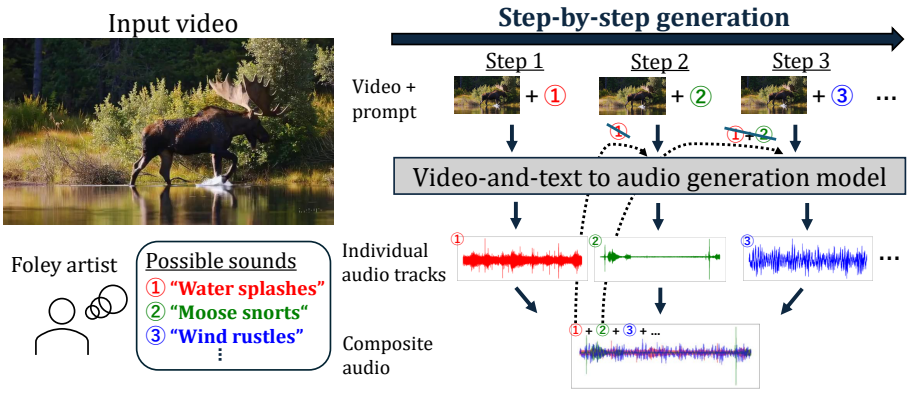

Fig. 1: Step-by-step video-to-audio generation for compositional sound effect creation. Video often contains numerous audible events, and Foley artists synthesize composite audio by adding complementary audio components step-by-step. Supporting this step-by-step mechanism with a video-to-audio generation model offers greater control and efficiency in the sound creation process.

This non-interactive design poses a significant challenge: if the output is missing specific events, creators must regenerate the entire track. Such inefficiencies limit the practical application of these models, particularly in collaborative workflows with human creators.

To resolve this issue, we argue that a step-by-step generation mechanism is crucial for practical V2A synthesis (Fig. 1). A model should generate not only a complete track aligned with the video, but also complementary audio that fills missing events without duplicating sounds already existing.[3] This offers greater control and efficiency in the sound creation process, as in the traditional Foley workflow.

A critical challenge to achieve this step-by-step generation is the scarcity of datasets. A straightforward approach (*i.e.*, training a conditional generation model that produces multiple plausible audio tracks per video) requires multi-reference video-audio pairs, which are difficult to obtain at scale. In this paper, we propose a guided generation method, Negative Audio Guidance (NAG), that enables step-by-step V2A synthesis without requiring specialized multi-reference datasets. Our key idea is to introduce an audio-conditioned branch trained on standard single-reference audiovisual datasets, and to use it *negatively* at inference time to discourage duplication of previously generated sounds. Concretely, we finetune a pre-trained V2A model using pairs of audio segments sampled from the same video, and during sampling, we apply NAG to push the current generation

[3] One might consider text-conditional V2A already provides sufficient control for the target audio event to be generated. However, existing text-conditional V2A models struggle to suppress the already generated sound, especially for prominent events in the video (e.g., in Fig. 4, the moose's footstep sounds are produced in all tracks regardless of the input text prompts).

away from the audio content already present in previously generated tracks. By iterating this negatively guided generation, the model produces a set of complementary tracks that can be mixed into a composite audio. Extensive experiments demonstrate that our method enables step-by-step completion of complementary sounds and enhances final audio quality while ensuring the separability of the generated audio at each step. In short, our main contributions are summarized as follows:

- We formulate *step-by-step* V2A synthesis as generating multiple *complementary* audio tracks for a single video, thereby enabling Foley-style interactive authoring.
- We propose Negative Audio Guidance (NAG), which uses an audio-conditioned guidance branch *negatively* during sampling to reduce duplication across sequentially generated tracks, while being trainable on standard single-reference datasets.
- We demonstrate improved track separability and composite audio quality compared to independent multi-prompt generation and text-based negative prompting.

2 Related work

2.1 Video-to-audio synthesis

The goal of video-to-audio synthesis is to generate an audio signal that aligns semantically and temporally with an input video. Early approaches used regression models [6] and GANs [18], while more recent ones have adopted autoregressive models [44] and diffusion models [8,9,29,30,35,45–47] due to their high capability in generation tasks. However, these models typically accept only videos (and optionally text prompts) as input conditions, making it impossible to specify sounds that users may want to combine with the generated audio.

Few studies have explored audio conditioning in video-to-audio synthesis to address their respective problem setting. MultiFoley [7] uses conditional audio as a reference for the generated audio. Sketch2Sound [13] takes a similar approach, but only uses a particular set of signal features extracted from the original conditional audio to accept sonic or vocal imitations as conditions. Action2Sound [4] focuses on disentangling foreground and ambient sound, using conditional audio to specify the appearance of ambient sound in the generated audio. ReWaS [20] introduces audio-energy conditioning predicted from video into a text-to-audio generator via ControlNet [53], enabling audio generation that is temporally aligned with the video. Concurrent to our work, SelVA [26] studies text-conditioned selective V2A via text-guided modulation of the video encoder. Unlike these studies, we utilize audio conditioning to specify what kind of audio *should not* appear in the generated audio, enabling step-by-step generation in video-to-audio synthesis.

2.2 Generative *add* operation

Generative *add* operations in the audio domain are executed to generate audio that can be mixed with an input audio signal, often guided by a text prompt. These operations have been explored in text-to-audio [21,48] and text-to-music [16, 22, 31, 34, 36] synthesis and can be divided into two approaches: training-based and training-free.

In the training-based approach, the model is explicitly trained to perform the *add* operation given the input audio. This training requires a triplet comprising an input audio, a text prompt, and an audio to be added as training data [16, 34, 48]. Unfortunately, these methods are difficult to apply in our setting because such data is hard to obtain. Even within a single scene, a mixture of many sounds can be observed, and separating them into individual ones is challenging [33, 39, 55]. SonicVisionLM [51] introduces a timestamp-conditioned video-and-text-to-audio model that first converts video into text and then generates audio via a T2A backbone. While this design avoids audio contamination from visual features and enables additive video-to-audio generation, it loses the fine-grained audiovisual cues captured by multimodal V2A models [8, 9, 35, 46]. As a result, it struggles with subtle or weakly visible events and relies on specialized video–audio–text–timestamp datasets that are costly to build.

On the other hand, the training-free approach is more flexible, as it leverages a pre-trained text-to-audio/music model without requiring any specific training. The *add* operation is conducted as a partial generation of multi-track audio [22, 31, 36] or a re-generation with a target prompt from structured noise obtained through inverting the input audio [21]. Instead of specific training data, these methods require particular properties in the pre-trained model: multi-track joint generation [22, 31], data-space diffusion models [36], and specific types of model architectures [21], which limit their applicability to our video-to-audio setting.

Similar *add* operations have been explored in computer vision, where models generate an object image to be added to an input background image. They are also categorized into training-based [3, 38] and training-free approaches [42], but in either case, they rely on segmentation models to create training data or to guide the generation process. This means that a particular subset of pixels in the image is assumed to be entirely replaced by the generated pixels via the *add* operation. As *add* in the audio domain involves mixing rather than replacing, these approaches cannot be directly applied to our setting.

We adopted a training-based approach for this study, utilizing the *add* operation in the audio domain, and designed our framework to eliminate the need for specialized training data. This approach makes it more practical and adaptable for video-to-audio synthesis tasks.

3 Method

3.1 Preliminaries

Generative modeling with flow-matching. Let $p_1(x)$ be a data distribution where $x \in \mathbb{R}^d$. Flow matching [27] considers the probability flow ODE $\frac{\mathrm{d}}{\mathrm{d}t}\phi_t(x) =$

$u_t(\phi_t(x))$, where $t \in [0,1]$ is a timestep, u_t is the velocity field, and $\phi_t(x) = \phi(x,t) : \mathbb{R}^d \times \mathbb{R} \to \mathbb{R}^d$ is the flow that maps x to the intermediate data x_t. ϕ_t can be an arbitrary function that satisfies the terminal condition $\phi_1(x_1) = x_1$ and $\phi_0(x_1) \sim p_0$, where p_0 is a tractable distribution such as a standard normal distribution $\mathcal{N}(0, I)$. Following the most popular setting, we define $\phi_t(x_1) = tx_1 + (1-t)x_0$, where $x_0 \sim N(0, I)$, resulting in $u(\phi_t(x)) = x_1 - x_0$.

Solving the ODE from $t = 0$ to 1 with an initial sample $x_0 \sim p_0(x)$ enables sampling from the target data distribution $x_1 \sim p_1(x)$. To achieve this, a neural network is trained to predict $u_t(\phi_t(x))$, which corresponds to $x_1 - x_0$ in our case, by minimizing the squared error over both data and timesteps. In text-conditioned video-to-audio synthesis, the model takes additional conditional inputs, which are the input video V and text prompt C, to model a conditional flow $u_t(\phi_t(x)|V, C)$.

Guidance for flow-matching models with multiple conditions. Classifier-free guidance [17] is widely used to improve generation quality and fidelity to conditions. This guidance is typically conducted under a single condition, and it is nontrivial to extend it to two conditions, as in text-conditioned video-to-audio synthesis. To derive a proper guidance process, $p(x|V, C)$ is decomposed as follows:

$$p(x|V,C) = p(x)\left(\frac{p(x|V)}{p(x)}\right)\left(\frac{p(x|V,C)}{p(x|V)}\right). \tag{1}$$

Given this decomposition, Kushwaha and Tian [25] proposed the following guided flow:

$$\begin{aligned}\tilde{u}_\theta(x_t) = u_\theta(x_t, t, \varnothing, \varnothing) &+ w_1(u_\theta(x_t, t, V, \varnothing) - u_\theta(x_t, t, \varnothing, \varnothing)) \\ &+ w_2(u_\theta(x_t, t, V, C) - u_\theta(x_t, t, V, \varnothing)),\end{aligned} \tag{2}$$

where θ is a set of the model parameters, and $\varnothing$ denotes a null condition. The three terms of the right-hand side of Eq. (2) respectively correspond to the three factors on the right-hand side of Eq. (1). They empirically show that setting $w_1 = w_2$ achieves better results. In this case, $u_\theta(x_t, t, V, \varnothing)$ cancels out, which gives us the following simplified formulation:

$$\tilde{u}_\theta(x_t) = u_\theta(x_t, t, \varnothing, \varnothing) + w_1(u_\theta(x_t, t, V, C) - u_\theta(x_t, t, \varnothing, \varnothing)). \tag{3}$$

3.2 Problem setting: Step-by-step video-to-audio synthesis

We are interested in iteratively generating complementary audio that complements previously generated audio. Let $x^{(1)}$ be the audio generated at the previous step, $x^{(2)}$ be the target audio to be generated at this step, C_2 be the text prompt that specifies the sound event for $x^{(2)}$ missed by $x^{(1)}$, and V be the input video. Our goal is to sample $x^{(2)}$ from $p\big(x^{(2)}\big|V, C_2, x^{(1)}\big)$ so that $x^{(2)}$ corresponds to the concept described by C_2, semantically and temporally aligns with V, and does not contain duplicated audio present in $x^{(1)}$.

From a straightforward standpoint, learning to generate samples from this distribution would require tuples of $(x^{(2)}, V, C_2, x^{(1)})$ as training data. Unfortunately,

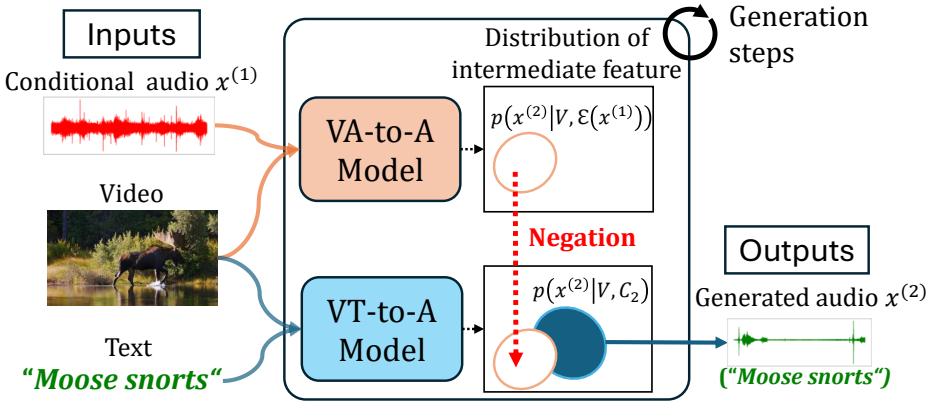

Fig. 2: Overview of the proposed method. Each audio track should represent a distinct audio event. Using previously generated audio tracks as conditioning signals for the negation concept, we explicitly push the current generation process away from the already generated audio tracks.

constructing such data from video is a challenging task known as visually-guided audio source separation [33, 39, 55], and thus, we cannot expect high-quality training datasets. Instead, we propose an alternative training framework that eliminates the need for such specialized data.

Note that this processing can be applied to the following generation step without loss of generality. In the k-th generation step, we can set the mix of the previously generated $(k-1)$ audios as $x^{(1)}$, and use it to sample $x^{(2)}$. Please refer to Section 5.1 for more details of this procedure.

3.3 Formulation of target distribution with concept negation

Recall that we want to generate $x^{(2)}$ to only cover a complementary sound event in $x^{(1)}$. In this sense, conditioning by $x^{(1)}$ corresponds to a concept negation [11, 28, 43] in generating $x^{(2)}$; the generated $x^{(2)}$ *should not* contain any concepts related to $x^{(1)}$. We denote this type of audio condition as $\bar{\mathcal{E}}(\cdot) = \neg\mathcal{E}(\cdot)$ to explicitly differentiate it from a standard type of audio condition denoted by $\mathcal{E}(\cdot)$. Based on the above-mentioned relationship between $x^{(1)}$ and $x^{(2)}$, we approximate the target distribution of $x^{(2)}$ using the concept negation as follows:

$$p\left(x^{(2)}\middle|V, C_2, x^{(1)}\right) \approx p\left(x^{(2)}\middle|V, C_2, \bar{\mathcal{E}}\left(x^{(1)}\right)\right). \tag{4}$$

Following the study by Du et al. [11], we assume that the concept negation holds

$$p(x, c_p, \neg c_n) \propto p(x)p(c_p|x)p(c_n|x)^{-1}, \tag{5}$$

where c_p and c_n denote conditional concepts to generate x.

To derive our guidance, we decompose the target distribution using Eq. (5) and Bayes' theorem as:

$$
\begin{aligned}
&p\Big(x^{(2)}\Big|V, C_2, \bar{\mathcal{E}}\Big(x^{(1)}\Big)\Big) \\
&\qquad \propto p\Big(x^{(2)}, V, C_2, \bar{\mathcal{E}}\Big(x^{(1)}\Big)\Big) \\
&\qquad \propto p\Big(x^{(2)}, V\Big) p\Big(C_2\Big|x^{(2)}, V\Big) p\Big(\mathcal{E}\Big(x^{(1)}\Big)\Big|x^{(2)}, V\Big)^{-1} \\
&\qquad \propto p\Big(x^{(2)}\Big)\left(\frac{p\big(x^{(2)}|V\big)}{p\big(x^{(2)}\big)}\right)\left(\frac{p\big(x^{(2)}\big|V, C_2\big)}{p\big(x^{(2)}\big|V\big)}\right)\left(\frac{p\big(x^{(2)}\big|V\big)}{p\big(x^{(2)}\big|V, \mathcal{E}\big(x^{(1)}\big)\big)}\right). \qquad (6)
\end{aligned}
$$

Similar to the derivation of Eq. (2), we can derive the guidance process based on this decomposition, as shown in the next section. This indicates that we can sample $x^{(2)}$ using the new guidance with flow-matching models. Iterating this process enables step-by-step generation in video-to-audio synthesis.

4 Implementation

4.1 Guided flow for step-by-step generation

The decomposition shown in Eq. (6) yields a new guidance formulation comprising four terms: one unconditional flow term and three guidance terms. However, adjusting the coefficients of the three guidance terms is cumbersome in practice. Given the empirical results in VinTAGe [25], where simplifying the guidance by removing $u_\theta(x_t, t, V, \varnothing)$ in Eq. (2) performs well, we also set the guidance coefficients so that $u_\theta(x_t, t, V, \varnothing)$ cancels out for simplification. Specifically, we set the sum of the coefficients of the first and third guidance terms to equal the coefficient of the second guidance term (see the Appendix for details). This leads to the following guided flow:

$$
\begin{aligned}
\tilde{u}_{\theta,\psi}(x_t) = u_\theta(x_t, t, \varnothing, \varnothing) &+ \alpha(u_\theta(x_t, t, V, C_2) - u_\theta(x_t, t, \varnothing, \varnothing)) \\
&+ \beta\Big(u_\theta(x_t, t, V, C_2) - u_{\theta,\psi}\Big(x_t, t, V, \varnothing, x^{(1)}\Big)\Big), \qquad (7)
\end{aligned}
$$

where α and β are the coefficients of the guidance terms. As we require an audio-conditioned flow in the last term, we introduce an additional set of trainable parameters ψ to adapt the text-conditioned video-to-audio model for this prediction, as detailed in the next subsection.

The second term on the right-hand side of Eq. (7) corresponds to a standard guidance term in text-conditioned video-to-audio models, which appeared in Eq. (3). It strengthens the fidelity of the generated audio to the conditional video and text prompt. The third term is a new guidance term introduced in our proposed method that pushes the generated audio away from the conditional audio $x^{(1)}$. This prevents the already-generated audio events from being re-generated during the current generation step, enabling step-by-step generation without overlapping audio events. Since $x^{(1)}$ is used similarly for a negative prompt, we refer to this new guidance as Negative Audio Guidance (NAG).

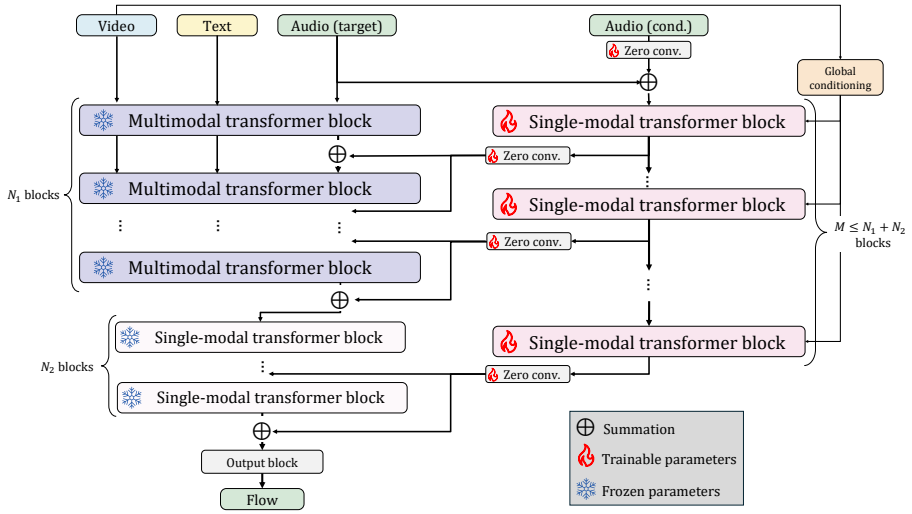

Fig. 3: Overview of network architecture for the audio-conditional flow estimator. We adopt ControlNet for the multi-modal diffusion transformer (MM-DiT) to incorporate audio conditioning into the pre-trained MMAudio.

4.2 Training Flow estimator for Negative Audio Guidance

All the flows appearing on the right-hand side of Eq. (7), except for the last one, can be estimated using standard text-conditional video-to-audio models. The remaining term is a conditional flow corresponding to the distribution $p(x^{(2)}|V, \mathcal{E}(x^{(1)}))$. As $\mathcal{E}(\cdot)$ is a standard type of audio conditioning, this flow estimator can be viewed as an extension of the video-to-audio model, incorporating conditional audio as an additional input. Therefore, we train the flow estimator using ControlNet [53] parameterized by ψ. This audio-conditioned flow estimator is trained to remain visually grounded while leveraging acoustic context from the reference audio, and is applied negatively at inference as a repulsive prior to avoid re-instantiating components already present in previous tracks. As a base video-to-audio model, we used MMAudio, parameterized by θ, for its high capability in video-to-audio synthesis.

Model architecture. Figure 3 shows an overview of the ControlNet architecture of the flow estimator. Since MMAudio uses a sophisticated architecture that extends MM-DiT [12], we adapt the ControlNet architecture accordingly. Inspired by Stable Diffusion 3.5 [40], we stack several single-modal transformer blocks to extract features from the conditional audio, and the features extracted at each block are added to the intermediate features of the main branch in MMAudio. During training, we freeze the pre-trained parameters of MMAudio and only update the parameters of the additional modules. Please refer to the Appendix for more details on model architecture, training, and inference.

Training dataset. We follow the training strategy of MMAudio, jointly using text-video-audio and text-audio paired datasets. Specifically, we used VGGSound [5] as a text-video-audio dataset, while Clotho [10], AudioCaps [23], and WavCaps [32] were used as text-audio datasets. From each audio clip, we sampled a four-second audio segment as x^{tgt} and another one as x^{cond} so that the two segments do not overlap. With this adjacent split, the two non-overlapping segments are more likely to share acoustic context (*e.g.*, environment and recording conditions). We also extracted a video segment corresponding to x^{tgt} as a conditional video V when the data came from VGGSound; otherwise, we set the pretrained empty token of MMAudio as V. Then, the sampled clips were used to compute the flow $u_{\theta,\psi}\big(x_t^{\mathrm{tgt}}, t, V, \varnothing, x^{\mathrm{cond}}\big)$, and ψ is optimized by minimizing the flow-matching loss. Importantly, we use non-overlapping segments to encourage the audio-conditioned branch to extract higher-level acoustic cues (*e.g.*, environment, timbre, recording conditions) that are shared across the video.

Design rationale for non-overlapping training pairs. Our goal at inference is to suppress re-generation of the *mixture of previously generated tracks for the same time window*. To instantiate NAG, we introduce an audio-conditioned branch to predict $u_{\theta,\psi}(x_t, t, V, \varnothing, x^{\mathrm{cond}})$, which provides the flow direction induced by the conditioning audio. In Eq. (7), this audio-conditioned estimator is applied with a negative sign. This yields a repulsive guidance that steers the sampling trajectory away from making the sample more consistent with the conditioning audio.

During training, we optimize the audio-conditioned branch to follow the audio condition (positive guidance), *i.e.*, to generate audio consistent with the acoustic cues in x^{cond} while remaining grounded in V. We sample $(x^{\mathrm{tgt}}, x^{\mathrm{cond}})$ from non-overlapping segments of the same clip to encourage the branch to capture shared *acoustic context* (e.g., environment, timbre, and recording conditions) that is more likely to be shared within a video. In practice, this clip-level context provides a useful signal for repelling components already present in previous tracks at inference, while avoiding a degenerate solution that copies the conditioning waveform. This is a good match for NAG because the branch is used *negatively*: it only needs to identify directions that would increase consistency with x^{cond}, which we then subtract during sampling.

5 Experiments

5.1 Evaluation setup

Multi-Caps VGGSound: multi-captioned audio-video dataset for evaluation. We constructed a new audiovisual dataset, Multi-Caps VGGSound, to evaluate step-by-step video-to-audio generation. We generated five captions using Qwen2.5-VL [2] for each video in the test split of the VGGSound dataset, totaling 15,221 video clips. We instructed the model to produce captions that describe different sound events that could plausibly occur in the scene, including both foreground and background audio. Since Qwen2.5-VL does not accept audio as input, captions are generated solely from visual inputs and may therefore deviate from the original

audio (e.g., off-screen sounds). However, this deviation is not an issue for our benchmark goal, which is to evaluate visually plausible and complementary sound authoring from video rather than reproducing the original recording. Please refer to the Appendix for more details.

Task setup: Step-by-step audio generation. We generated five audio tracks $\left\{x^{(k)}|k \in \{1, 2, \dots 5\}\right\}$ corresponding to audio captions $\{C_k|k \in \{1, 2, \dots 5\}\}$ for each video V in the Multi-Caps VGGSound dataset. The generation order is determined based on the semantic similarity between the video and the caption, so that the model generates core events first (See the Appendix for more details). We extracted the first 8-second segment from the video and generated sounds for this segment using different captions. Given the multiple generated audio tracks, we synthesized a composite audio $\tilde{x}$ by $\tilde{x} = \text{normalize}(\Sigma_k x^{(k)})$. We employed loudness normalization [41] as a simple mixing strategy for the composition, ensuring that the total loudness remained consistent with that of natural audio. The target loudness was set to -20 LUFS, which corresponds to the mean loudness of the VGGSound test set.

Step-by-step audio generation with NAG. To generate the audio tracks step-by-step using NAG, we used the composite audio of all audio tracks generated in the previous generation steps as the conditioning signal for NAG. Specifically, at the generation step for $x^{(k)}$, we synthesized a composite audio $\tilde{x}_{:k} = \text{normalize}\left(\Sigma_l^{k-1} x^{(l)}\right)$ for the condition. We generated the first audio only using the standard classifier-free guidance in Eq. (3), since no audio track had been generated at the first generation step. For the guidance coefficients, we empirically set $\alpha = 4.5$ and $\beta = 1.5$ in Eq. (7) (see the Appendix for a sensitivity analysis).

Baseline models. We compared our proposed method with open-source text-and-video conditional audio (TV2A) generation models. We chose each State-of-the-Art TV2A model among various training approaches: Seeing-and-Hearing [52] as a TV2A model adapted from the T2A model without training, FoleyCrafter [54] as a TV2A model adapted from the T2A model through fine-tuning, and MMAudio [9] as a TV2A model trained from scratch. Note that our model is built upon MMAudio with additional audio conditions introduced by the proposed ControlNet architecture. We compared the proposed NAG to the original MMAudio-S-16k model with the classifier-free guidance (CFG) or negative prompting.

We tested two generation processes to generate multiple audio tracks and obtain a composite audio using the baseline models: independent generation and step-by-step generation based on negative prompts. In the independent generation, we generated five sounds per video using different text conditions. In this case, each generation process does not access the other audio tracks or their corresponding captions. In the step-by-step generation based on negative prompts, we generated each audio track using negative prompting [49] to ensure it was distinct from all the captions used in the previous generation steps. Specifically, for the video V with the k-th audio caption C_k, we computed the guided flow

by $\tilde{u}_\theta(x_t) = u_\theta(x_t, t, \varnothing, \varnothing) + w_1(u_\theta(x_t, t, V, C_k) - u_\theta(x_t, t, \varnothing, C_{k,\text{neg}}))$ at each timestep, where $C_{k,\text{neg}}$ is the concatenation of the other captions $\{C_l | l < k\}$.

Evaluation metrics. We assessed the quality of both the composite audio and the individual audio tracks to evaluate the step-by-step audio generation.

Following MMAudio [9], we evaluated the composite audio in terms of audio quality, semantic alignment, and temporal alignment. We assessed the audio quality of the generated audio using Fréchet Distance (FD), Kullback–Leibler (KL) distance, and Inception Score (IS) [37]. We used PANNs [24] (FD_{PANNs}) and VGGish [14] (FD_{VGG}) for computing FD, and PANNs (KL_{PANNs}) and PaSST (KL_{PaSST}) for computing KL, and PANNs for computing IS, respectively. We assessed the semantic alignment between the input video and the composite audio by the cosine similarity between their embeddings extracted by ImageBind [15] (IB-score). We assessed temporal alignment between the input video and the generated audio using Synchformer [19] (DeSync), where we took two 4.8-second segments at the beginning and end and averaged their scores.

For each audio track evaluation, we assessed its quality across four aspects: audio separability between audio tracks generated for the same video, audio quality, audio-text alignment, and audio-video alignment. Since distinct audio components should be represented in separate audio tracks, each generated audio track should differ from the other tracks. To evaluate audio separability, we computed the similarity between the CLAP [50] audio embeddings for each pair of audio tracks (10 pairs per video). For Audio-Text alignment, we computed the similarity between CLAP text embeddings from the input prompts (used to generate each audio track) and the corresponding CLAP audio embeddings. For audio quality and audio-video alignment, we adopted IS and IB-score, respectively, as in the composite audio evaluation protocol.

5.2 Main results

Objective evaluation on the composite audio. Table 1 shows the quantitative evaluation of the composite audio. Our proposed method achieves the best results for all metrics except IS among all the methods. We also evaluated the baseline models' one-step generation with the caption created by fusing the five captions for each video. Though this generation process does not provide each audio track and differs from our goal of step-by-step generation, these values indicate the best possible scores of each baseline model.

Objective evaluation on each audio track. Table 2 shows the quantitative evaluation of the individual audio tracks. The vanilla MMAudio-S-16k with CFG struggles to generate well-separated sounds for each audio track, as reflected in a lower audio separability score, although it achieves high audio quality and A-V Alignment. Using negative prompting improves the audio separability but drastically degrades all the other scores. Using NAG also successfully improves the audio separability while maintaining high audio quality and A-V alignment. The A-T alignment score marginally improves from that of the vanilla MMAudio.

Table 1: Quantitative evaluation of the composite audio synthesized from the generated multiple audio tracks. The results of one-step generation using a fused caption are shown as a reference.

	Audio Quality					A-V Align.	
Method	FD_{PANNs}↓	FD_{VGG}↓	KL_{PANNs}↓	KL_{PaSST}↓	IS↑	IB-score↑	DeSync↓
One-Step Generation with Fused Caption (no separated audio track)							
Seeing-and-Hearing	25.42	5.68	2.81	2.76	6.45	36.88	1.22
FoleyCrafter	16.93	2.29	2.60	2.52	11.93	27.78	1.23
MMAudio-S-16k	6.75	1.03	2.09	2.04	13.66	29.45	0.46
Independent Generation							
Seeing-and-Hearing	31.81	7.68	3.10	2.65	4.12	20.16	1.19
FoleyCrafter	20.04	3.23	2.70	2.36	9.21	25.26	1.18
MMAudio-S-16k	7.76	1.35	2.02	1.84	10.42	28.13	**0.42**
Step-by-Step Generation with Negative Prompting							
FoleyCrafter	22.34	4.64	2.94	2.47	6.02	18.83	1.19
MMAudio-S-16k	9.21	1.77	2.15	1.89	9.08	25.89	0.45
Step-by-Step Generation with Negative Audio Guidance							
Ours	**6.47**	**0.98**	**2.01**	**1.76**	**10.58**	**28.65**	**0.42**

Table 2: Quantitative evaluation of individual audio tracks. Our proposed method improves audio separability among multiple tracks while maintaining other scores.

	Separability	Quality	A-T Align.	A-V Align.
Method	CLAP A-A↓	IS↑	CLAP T-A↑	IB-score↑
MMAudio-S-16k	79.75	**12.47**	<u>28.36</u>	**27.76**
MMAudio-S-16k + Neg. Prompting	<u>75.57</u>	11.19	27.14	24.53
MMAudio-S-16k + NAG (Ours)	**71.38**	<u>12.01</u>	**28.91**	<u>26.67</u>

We hypothesize that lower contamination by other audio concepts yields a higher A-T alignment score.

Visual comparison with baselines. Figure 4 visually compares these three methods. The first audio track is the same across all methods because it is generated using only CFG. The vanilla MMAudio-S-16k tends to generate similar audio across multiple audio tracks. All generated audio tracks from the vanilla MMAudio-S-16k contain the sound of water as the moose walks (visually shown as vertical segments that appear at regular intervals). Since the moose and its movement are prominent in the input video, MMAudio tends to include the sound related to them regardless of the input text prompt. This is also reflected in the higher IB-score, indicating that all audio tracks are semantically aligned with the input video, regardless of whether the input prompt represents background audio (as in the case of the second audio track). Using negative prompting suppressed

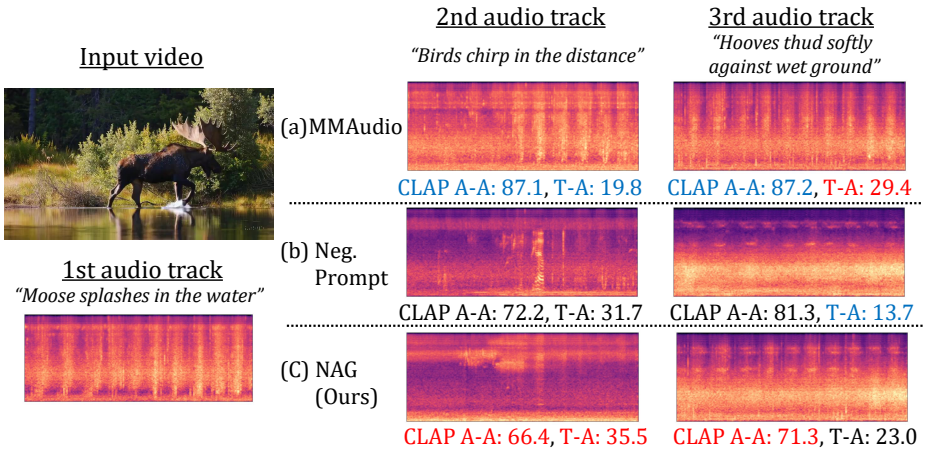

Fig. 4: Spectrogram visualizations of step-by-step audio generation using (a) vanilla MMAudio, (b) MMAudio with negative prompting, and (c) MMAudio with Negative Audio Guidance (NAG). The best and worst CLAP A-A and T-A scores are highlighted in red and blue, respectively. The first audio track is generated using (a) in all settings, resulting in identical outputs. Our proposed method effectively suppresses previously generated sounds in subsequent steps (second and third tracks) while maintaining high alignment with the target text prompts.

Table 3: The win rates of our proposed method (MMAudio-S-16k + NAG) against MMAudio-S-16k for composite audio. 95% confidence intervals are reported as $\pm X$.

	Audio quality↑	Semantic alignment↑	Temporal alignment ↑
Win rate	**71.36** ±7.71	**76.00** ±7.62	**61.14** ±7.85

this contamination of the audio content, but it tends to suffer from worse text alignment. In contrast, the proposed NAG successfully suppressed the audio components already generated in the early generation steps while achieving better text alignment. It generates a complementary sound specified by the text prompt by explicitly steering the generation process away from previously generated sounds. Please refer to the Appendix for additional generated samples.

Subjective evaluation. We also conducted a user study for subjective evaluation. Table 3 presents the user study results for the final composite audio, showing that our method is preferred for audio quality, semantic alignment, and temporal alignment compared to the baseline. Table 4 shows the results for the individual audio tracks. Our method received significantly higher ratings for separability and marginally higher ratings for both audio quality and text fidelity. Please refer to the Appendix for a more detailed statistical analysis and details of the user study setup.

Table 4: Average ratings for individual audio tracks generated by the baseline and our method. 95% confidence intervals are reported as $\pm X$.

Method	Separability↑	Audio quality↑	Text fidelity ↑
MMAudio-S-16k	2.24±0.15	2.89±0.14	2.42±0.18
MMAudio-S-16k + NAG (Ours)	**3.35**±0.15	**3.30**±0.14	**3.12**±0.18

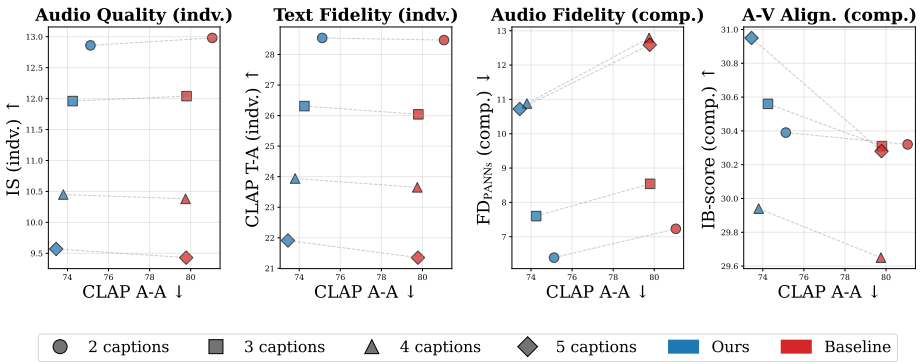

Fig. 5: Analysis of the number of captions using VGGSounder. We compare MMAudio-S-16k (Baseline) and MMAudio-S-16k + NAG (Ours) on individual-track quality and text fidelity (left), and on composite-audio fidelity and audiovisual alignment (right). Audio separability is shown on the x-axis. Our method consistently improves separability without degrading quality, with larger gains as the number of captions increases.

5.3 Analysis on number of captions

We used a fixed number of captions for Multi-Caps VGGSound to ensure consistent evaluation. In real use cases, however, the number of sequential generation steps is chosen by the user based on the visual content and the desired number of audio events. To examine whether our method remains effective across a broader range of plausible audio events, we conducted additional experiments on the VGGSounder [56] dataset. VGGSounder provides a flexible number of human-annotated captions per video, reflecting both the visual content and the ground-truth audio. We selected videos with 2–5 captions and evaluated the baseline model (MMAudio-S-16k) and our method (MMAudio-S-16k + NAG) on each subset separately.

Figure 5 summarizes the results. The two plots on the left report audio quality and text fidelity for the individual tracks, while the two plots on the right report audio fidelity and audiovisual alignment for the composite audio. All plots use the audio separability of each track on the x-axis. Across all caption counts, the baseline model exhibits consistently low separability, whereas our method reliably improves separability without sacrificing audio quality or text alignment. The improvement becomes more pronounced as the number of captions increases. For composite audio evaluation, our method consistently enhances both audio

fidelity and audiovisual alignment. Overall, these results demonstrate that the proposed step-by-step generation is beneficial even when only a few sound events are present, and its advantage increases as the number of audio events grows.

6 Conclusion

We introduced a novel video-to-audio generation method, guided by text, video, and audio conditions, that enables step-by-step synthesis. By applying negative audio guidance alongside a text prompt, our approach generates multiple complementary sounds for the same video input, facilitating high-quality composite audio synthesis. Importantly, our method does not require specialized training datasets. We built it on a pre-trained video-to-audio model by adapting ControlNet for audio conditioning, enabling training on accessible single-reference datasets. Quantitative and subjective evaluations show that our method improves the separability and text fidelity of generated audio at each step and the quality of the final composite audio.

Acknowledgment

We sincerely thank Koichi Saito, Khaled Koutini, and Naoki Murata for their valuable comments and suggestions on an earlier version of this manuscript.

References

1. Ament, V.T.: The Foley Grail: The Art of Performing Sound for Film, Games, and Animation. Routledge (2021)
2. Bai, S., Chen, K., Liu, X., Wang, J., Ge, W., Song, S., Dang, K., Wang, P., Wang, S., Tang, J., Zhong, H., Zhu, Y., Yang, M., Li, Z., Wan, J., Wang, P., Ding, W., Fu, Z., Xu, Y., Ye, J., Zhang, X., Xie, T., Cheng, Z., Zhang, H., Yang, Z., Xu, H., Lin, J.: Qwen2.5-vl technical report. arXiv preprint arXiv:2502.13923 (2025)
3. Canberk, A., Bondarenko, M., Ozguroglu, E., Liu, R., Vondrick, C.: Erasedraw: Learning to insert objects by erasing them from images. In: Proceedings of the European Conference on Computer Vision (2024)
4. Chen, C., Peng, P., Baid, A., Xue, Z., Hsu, W.N., Harwath, D., Grauman, K.: Action2sound: Ambient-aware generation of action sounds from egocentric videos. In: Proceedings of the European Conference on Computer Vision (2024)
5. Chen, H., Xie, W., Vedaldi, A., Zisserman, A.: Vggsound: A large-scale audio-visual dataset. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (2020)
6. Chen, P., Zhang, Y., Tan, M., Xiao, H., Huang, D., Gan, C.: Generating visually aligned sound from videos. IEEE Transactions on Image Processing (2020)
7. Chen, Z., Seetharaman, P., Russell, B., Nieto, O., Bourgin, D., Owens, A., Salamon, J.: Video-guided foley sound generation with multimodal controls. arXiv preprint arXiv:2411.17698 (2024)
8. Chen, Z., Seetharaman, P., Russell, B., Nieto, O., Bourgin, D., Owens, A., Salamon, J.: Video-guided foley sound generation with multimodal controls. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2025)

9. Cheng, H.K., Ishii, M., Hayakawa, A., Shibuya, T., Schwing, A., Mitsufuji, Y.: 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 (2025)
10. Drossos, K., Lipping, S., Virtanen, T.: Clotho: An audio captioning dataset. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE (2020)
11. Du, Y., Li, S., Mordatch, I.: Compositional visual generation with energy based models. In: Proceedings of the Advances in Neural Information Processing Systems (2020)
12. Esser, P., Kulal, S., Blattmann, A., Entezari, R., Müller, J., Saini, H., Levi, Y., Lorenz, D., Sauer, A., Boesel, F., Podell, D., Dockhorn, T., English, Z., Rombach, R.: Scaling rectified flow transformers for high-resolution image synthesis. In: Proceedings of the International Conference on Machine Learning (2024)
13. García, H.F., Nieto, O., Salamon, J., Pardo, B., Seetharaman, P.: Sketch2sound: Controllable audio generation via time-varying signals and sonic imitations. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (2025)
14. Gemmeke, J.F., Ellis, D.P., Freedman, D., Jansen, A., Lawrence, W., Moore, R.C., Plakal, M., Ritter, M.: Audio set: An ontology and human-labeled dataset for audio events. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (2017)
15. Girdhar, R., El-Nouby, A., Liu, Z., Singh, M., Alwala, K.V., Joulin, A., Misra, I.: Imagebind: One embedding space to bind them all. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2023)
16. Han, B., Dai, J., Hao, W., He, X., Guo, D., Chen, J., Wang, Y., Qian, Y., Song, X.: Instructme: An instruction guided music edit and remix framework with latent diffusion models. In: Proceedings of the International Joint Conference on Artificial Intelligence (2024)
17. Ho, J., Salimans, T.: Classifier-free diffusion guidance. In: Proceedings of the NeurIPS Workshop on Deep Generative Models and Downstream Applications (2021)
18. Iashin, V., Rahtu, E.: Taming visually guided sound generation. In: Proceedings of the British Machine Vision Conference (2021)
19. Iashin, V., Xie, W., Rahtu, E., Zisserman, A.: Synchformer: Efficient synchronization from sparse cues. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (2024)
20. Jeong, Y., Kim, Y., Chun, S., Lee, J.: Read, watch and scream! sound generation from text and video. In: Proceedings of the AAAI Conference on Artificial Intelligence (2025)
21. Jia, Y., Chen, Y., Zhao, J., Zhao, S., Zeng, W., Chen, Y., Qin, Y.: Audioeditor: A training-free diffusion-based audio editing framework. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (2025)
22. Karchkhadze, T., Izadi, M.R., Dubnov, S.: Simultaneous music separation and generation using multi-track latent diffusion models. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (2025)
23. Kim, C.D., Kim, B., Lee, H., Kim, G.: Audiocaps: Generating captions for audios in the wild. In: Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (2019)

24. Kong, Q., Cao, Y., Iqbal, T., Wang, Y., Wang, W., Plumbley, M.D.: Panns: Large-scale pretrained audio neural networks for audio pattern recognition. IEEE/ACM Transactions on Audio, Speech, and Language Processing (2020)
25. Kushwaha, S.S., Tian, Y.: Vintage: Joint video and text conditioning for holistic audio generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2025)
26. Lee, J., Nam, J., Lee, J.: Hear what matters! text-conditioned selective video-to-audio generation. arXiv preprint arXiv:2512.02650 (2025)
27. Lipman, Y., Chen, R.T., Ben-Hamu, H., Nickel, M., Le, M.: Flow matching for generative modeling. In: Proceedings of The International Conference on Learning Representations (2023)
28. Liu, N., Li, S., Du, Y., Torralba, A., Tenenbaum, J.B.: Compositional visual generation with composable diffusion models. In: Proceedings of the European Conference on Computer Vision (2022)
29. Liu, X., Su, K., Shlizerman, E.: Tell what you hear from what you see-video to audio generation through text. In: Proceedings of the Advances in Neural Information Processing Systems (2024)
30. Luo, S., Yan, C., Hu, C., Zhao, H.: Diff-foley: Synchronized video-to-audio synthesis with latent diffusion models. In: Proceedings of the Advances in Neural Information Processing Systems (2023)
31. Mariani, G., Tallini, I., Postolache, E., Mancusi, M., Cosmo, L., Rodolà, E.: Multi-source diffusion models for simultaneous music generation and separation. In: Proceedings of the International Conference on Learning Representations (2024)
32. Mei, X., Meng, C., Liu, H., Kong, Q., Ko, T., Zhao, C., Plumbley, M.D., Zou, Y., Wang, W.: Wavcaps: A chatgpt-assisted weakly-labelled audio captioning dataset for audio-language multimodal research. IEEE/ACM Transactions on Audio, Speech, and Language Processing (2024)
33. Owens, A., Efros, A.A.: Audio-visual scene analysis with self-supervised multisensory features. In: Proceedings of the European Conference on Computer Vision (2018)
34. Parker, J.D., Spijkervet, J., Kosta, K., Yesiler, F., Kuznetsov, B., Wang, J.C., Avent, M., Chen, J., Le, D.: Stemgen: A music generation model that listens. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (2024)
35. Polyak, A., Zohar, A., Brown, A., Tjandra, A., Sinha, A., Lee, A., Vyas, A., Shi, B., Ma, C.Y., Chuang, C.Y., Yan, D., Choudhary, D., Wang, D., Sethi, G., Pang, G., Ma, H., Misra, I., Hou, J., Wang, J., Jagadeesh, K., Li, K., Zhang, L., Singh, M., Williamson, M., Le, M., Yu, M., Singh, M.K., Zhang, P., Vajda, P., Duval, Q., Girdhar, R., Sumbaly, R., Rambhatla, S.S., Tsai, S., Azadi, S., Datta, S., Chen, S., Bell, S., Ramaswamy, S., Sheynin, S., Bhattacharya, S., Motwani, S., Xu, T., Li, T., Hou, T., Hsu, W.N., Yin, X., Dai, X., Taigman, Y., Luo, Y., Liu, Y.C., Wu, Y.C., Zhao, Y., Kirstain, Y., He, Z., He, Z., Pumarola, A., Thabet, A., Sanakoyeu, A., Mallya, A., Guo, B., Araya, B., Kerr, B., Wood, C., Liu, C., Peng, C., Vengertsev, D., Schonfeld, E., Blanchard, E., Juefei-Xu, F., Nord, F., Liang, J., Hoffman, J., Kohler, J., Fire, K., Sivakumar, K., Chen, L., Yu, L., Gao, L., Georgopoulos, M., Moritz, R., Sampson, S.K., Li, S., Parmeggiani, S., Fine, S., Fowler, T., Petrovic, V., Du, Y.: Movie gen: A cast of media foundation models. arXiv preprint arXiv:2410.13720 (2024)
36. Postolache, E., Mariani, G., Cosmo, L., Benetos, E., Rodolà, E.: Generalized multi-source inference for text conditioned music diffusion models. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (2024)

37. Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., Chen, X.: Improved techniques for training gans. In: Proceedings of the Advances in Neural Information Processing Systems (2016)
38. Singh, J., Zhang, J., Liu, Q., Smith, C., Lin, Z., Zheng, L.: Smartmask: context aware high-fidelity mask generation for fine-grained object insertion and layout control. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2024)
39. Song, Z., Zhang, Z.: Visually guided sound source separation with audio-visual predictive coding. IEEE Transactions on Neural Networks and Learning Systems (2023)
40. Stability-AI: sd3.5 (2024), https://github.com/Stability-AI/sd3.5
41. Steinmetz, C.J., Reiss, J.D.: pyloudnorm: A simple yet flexible loudness meter in python. In: Proceedings of the Audio Engineering Society Convention (2021)
42. Tewel, Y., Gal, R., Samuel, D., Atzmon, Y., Wolf, L., Chechik, G.: Add-it: Training-free object insertion in images with pretrained diffusion models. In: Proceedings of the International Conference on Learning Representations (2025)
43. Valle, R., Badlani, R., Kong, Z., gil Lee, S., Goel, A., Kim, S., Santos, J.F., Dai, S., Gururani, S., Aljafari, A., Liu, A.H., Shih, K.J., Prenger, R., Ping, W., Yang, C.H.H., Catanzaro, B.: Fugatto 1: Foundational generative audio transformer opus 1. In: Proceedings of the International Conference on Learning Representations (2025)
44. Viertola, I., Iashin, V., Rahtu, E.: Temporally aligned audio for video with autoregression. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (2025)
45. Wang, H., Ma, J., Pascual, S., Cartwright, R., Cai, W.: V2a-mapper: A lightweight solution for vision-to-audio generation by connecting foundation models. In: Proceedings of the AAAI Conference on Artificial Intelligence (2024)
46. Wang, J., Zeng, X., Qiang, C., Chen, R., Wang, S., Wang, L., Zhou, W., Cai, P., Zhao, J., Li, N., Li, Z., Liang, Y., Wang, X., Zheng, H., Wen, M., Yin, K., Wang, Y., Li, N., Deng, F., Dong, L., Zhang, C., Zhang, D., Gai, K.: Kling-foley: Multimodal diffusion transformer for high-quality video-to-audio generation. arXiv (2025)
47. Wang, Y., Guo, W., Huang, R., Huang, J., Wang, Z., You, F., Li, R., Zhao, Z.: Frieren: Efficient video-to-audio generation network with rectified flow matching. In: Proceedings of the Advances in Neural Information Processing Systems (2024)
48. Wang, Y., Ju, Z., Tan, X., He, L., Wu, Z., Bian, J., et al.: Audit: Audio editing by following instructions with latent diffusion models. In: Proceedings of the Advances in Neural Information Processing Systems (2023)
49. Woolf, M.: Stable diffusion 2.0 and the importance of negative prompts for good results (2022), https://minimaxir.com/2022/11/stable-diffusion-negative-prompt/
50. Wu, Y., Chen, K., Zhang, T., Hui, Y., Berg-Kirkpatrick, T., Dubnov, S.: 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 (2023)
51. Xie, Z., Yu, S., Li, M., He, Q., Chen, C., Jiang, Y.G.: Sonicvisionlm: Playing sound with vision language model. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2024)
52. Xing, Y., He, Y., Tian, Z., Wang, X., Chen, Q.: Seeing and hearing: Open-domain visual-audio generation with diffusion latent aligners. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2024)

53. Zhang, L., Rao, A., Agrawala, M.: Adding conditional control to text-to-image diffusion models. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (2023)
54. Zhang, Y., Gu, Y., Zeng, Y., Xing, Z., Wang, Y., Wu, Z., Chen, K.: Foleycrafter: Bring silent videos to life with lifelike and synchronized sounds. arXiv preprint arXiv:2407.01494 (2024)
55. Zhu, L., Rahtu, E.: Visually guided sound source separation using cascaded opponent filter network. In: Proceedings of the Asian Conference on Computer Vision (2020)
56. Zverev, D., Wiedemer, T., Prabhu, A., Bethge, M., Brendel, W., Koepke, A.S.: Vggsounder: Audio-visual evaluations for foundation models. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (2025)