SAM Audio: Segment Anything in Audio

Bowen Shi[*], **Andros Tjandra**[*], **John Hoffman**[*], **Helin Wang**[*], **Yi-Chiao Wu**[*], **Luya Gao**[*], **Julius Richter**[†], **Matt Le**[†], **Apoorv Vyas**[†], **Sanyuan Chen**[†], **Christoph Feichtenhofer**[‡], **Piotr Dollár**[‡], **Wei-Ning Hsu**[‡], **Ann Lee**[‡]

Meta Superintelligence Labs

[*]Core contributors (random order from second author onward), [†]Contributors (random order), [‡]Project leads (random order)

General audio source separation is a key capability for multimodal AI systems that can perceive and reason about sound. Despite substantial progress in recent years, existing separation models are either domain-specific, designed for fixed categories such as speech or music, or limited in controllability, supporting only a single prompting modality such as text. In this work, we present SAM Audio, a foundation model for general audio separation that unifies text, visual, and temporal span prompting within a single framework. Built on a diffusion transformer architecture, SAM Audio is trained with flow matching on large-scale audio data spanning speech, music, and general sounds, and can flexibly separate target sources described by language, visual masks, or temporal spans. The model achieves state-of-the-art performance across a diverse suite of benchmarks, including general sound, speech, music, and musical instrument separation in both in-the-wild and professionally produced audios, substantially outperforming prior general-purpose and specialized systems. Furthermore, we introduce a new real-world separation benchmark with human-labeled multimodal prompts and a reference-free evaluation model that correlates strongly with human judgment.

Date: December 15, 2025
Correspondence: Bowen Shi bshi@meta.com, Andros Tjandra androstj@meta.com
Demo: https://aidemos.meta.com/segment-anything/editor/segment-audio
Code: https://github.com/facebookresearch/sam-audio
Website: https://ai.meta.com/samaudio/

Meta

1 Introduction

Audio source separation aims to decompose a complex sound mixture into individual source tracks corresponding to distinct sound events. Separation systems play an essential role in a broad range of real-world applications. In the production and creative domains, they enable sound engineers to isolate and remix individual stems, restore archival recordings, or remove unwanted background noise. In education and accessibility, separation can highlight key sounds or voices for learners or assistive listening devices. From a research perspective, it also serves as a valuable testbed for studying audio understanding in multimodal AI systems, as effective separation requires identifying the individual sources, an essential element of AI models understanding audio.

Audio separation has been studied extensively, and existing approaches can be broadly grouped into *promptless* and *prompted* methods. Promptless systems aim to decompose an audio mixture into a fixed set of predefined sources and have shown strong performance in specialized tasks such as speech enhancement, speaker separation, and music demixing (Mitsufuji et al., 2021; Fabbro et al., 2024; Zhao et al., 2024; Kong et al., 2023). However, these methods assume a fixed output configuration and rely on predefined taxonomies of sound categories. As a result, they struggle to adapt to open-domain mixtures or user-defined sound types, where boundaries between sound classes are ambiguous and highly context dependent. Recent progress has shifted the field toward *prompted separation*, in which the target source is specified through an external signal. Text prompts (Liu et al., 2024c; Yuan et al., 2024; Ma et al., 2024; Wang et al., 2025; Hai et al., 2024) allow users to describe arbitrary sound events (e.g., *"dog barking"*, *"female speech"*) and thereby remove the need for fixed taxonomies. Visual prompts (Zhao et al., 2018a; Huang et al., 2024a; Li et al., 2024b; Ephrat et al., 2018; Dong et al., 2023) complement text by providing instance-level grounding, enabling disambiguation when multiple similar

sources appear in the same scene.

The prompted separation paradigm greatly expands the applicability of separation, but important challenges remain. First, current text-prompted systems are largely benchmarked on general sound effects and often struggle in specialized domains such as music or speech. For example, text-based instrument separation significantly lags behind domain-specific methods like Demucs (Rouard et al., 2023; Défossez, 2021). While these specialized models perform well within their own domain, they cannot generalize beyond fixed source types. Second, visual-prompted systems are comparatively underexplored and typically tested on small or synthetic datasets. Real-world videos contain a mix of on-screen and off-screen sources and ambiguous sound–vision correspondences, making it unclear whether simply selecting a visual region is sufficient for reliable separation. Finally, existing models struggle to distinguish subtle but perceptually important differences between similar sounds such as variants of movie sound effects, which are hard to express precisely with text alone. These challenges point to the need for a more flexible and universally high-performing prompting framework that integrates cues of various modalities to guide separation.

Meanwhile, progress in audio separation research has also been slowed by the *lack of* unified *benchmarks* and reliable evaluation *metrics*. Existing text-prompted models are typically evaluated on disparate test sets, often focusing narrowly on environmental sound separation, with limited coverage of speech and music. In contrast, traditional benchmarks such as MUSDB (Rafii et al., 2017a) for instrument separation are restricted to a few predefined stems and are incompatible with prompting. Furthermore, widely used objective metrics such as Signal-to-Distortion Ratio (SDR) rely on synthetic mixtures with clean reference stems, which are scarce or nonexistent in real recordings. Reference-free metrics like CLAP similarity (Wu et al., 2023b) offer some insight into audio–text alignment but correlate poorly with human judgment of separation quality (see Section 7.8). These limitations make it difficult to compare models fairly and to assess true performance in open-domain scenarios.

In this work, we introduce Segment Anything Model Audio (SAM Audio), a foundation model for general audio separation that unifies text, visual, and temporal prompting within a single framework. SAM Audio allows users to specify *what* to separate using a *text* description, where to separate using a *visual* mask via positive/negative clicks, and *when* to separate using a *temporal span* prompt. These modalities can be used independently or jointly, enabling flexible interaction — for example, a user can describe "piano playing" in a music video and highlight the corresponding region to isolate it precisely. At the core of SAM Audio is a diffusion transformer trained with flow matching on large-scale audio mixtures spanning speech, music, and general sound events. During inference, the model simultaneously produces the target stem along with a residual stem capturing all remaining audio content.

Contributions. Our main contributions are threefold.

- We propose SAM Audio, the first foundation model that supports *multimodal prompting* (text, visual, span) – used either individually or in combination, for open-domain audio separation, achieving state-of-the-art results for both in-the-wild and professional audios of general and specialized domains (e.g., speech, music).
- We introduce *span prompting*, a novel form of temporal conditioning that enables precise frame-level control for audio separation.
- We develop a unified and comprehensive separation benchmark, SAM Audio-Bench, that covers major audio domains (speech, music, and sound effects), includes human-labeled multimodal prompts, and provides a reference-free automatic evaluation model with strong correlation to human perception.

2 Related Work

2.1 Audio Separation Models

Speaker separation. Conventional source separation aims to decompose an audio mixture into individual *sources*. In speech, this typically involves isolating the utterances of multiple speakers from an audio mixture. A dominant line of work formulates this as a discriminative regression problem, where the model predicts a latent-domain mask for each source, and the clean signals are reconstructed by applying these masks.

Early deep learning models relied on spectrogram masking (Takahashi et al., 2018; Choi et al., 2021), while time-domain approaches such as Conv-TasNet (Luo and Mesgarani, 2019) demonstrated that directly operating on waveforms yields superior perceptual quality. Subsequent dual-path architectures such as DPRNN (Luo et al., 2020) and TF-Locoformer (Gusó and Thickstun, 2022) improved efficiency and long-range context modeling by combining intra- and inter-chunk processing. Transformer-based models further extended these ideas: SepFormer (Subakan et al., 2021) used dual-path self-attention to capture long-term dependencies, while MossFormer2 (Zhao et al., 2024) integrated multi-scale attention and mask refinement to improve robustness and latency in practical deployments.

Another line of research frames separation as a *generative* modeling problem. Instead of predicting deterministic masks, generative models learn the underlying distribution of source signals conditioned on the mixture. Early work includes GAN approaches (Subakan and Smaragdis, 2018; Kong and Ping, 2019), as well as normalizing-flow models (Jayaram and Thickstun, 2020). Recent methods (Mariani et al., 2024; Scheibler et al., 2024; Dong et al., 2025) leverage diffusion and flow matching (Lipman et al., 2023) to handle complex speech overlaps. SEP-Diff (Chen et al., 2023) employs a denoising diffusion process in the mel-spectrogram domain for speech separation, while DiffSep (Scheibler et al., 2023) formulates a mixture-conditioned score model in waveform space. Post-processing models such as Diffiner (Sawata et al., 2023) and Fast-GeCo (Wang et al., 2024) refine regression-based separators via diffusion.

Beyond blind separation, *target* or *promptable* speaker extraction leverages additional information to identify the desired speaker. For instance, SpeakerBeam (Delcroix and et al., 2020), VoiceFilter (Wang et al., 2019), and SpEx+ (Xu et al., 2020) condition on an enrollment utterance to extract speech from a specific speaker, achieving instance-level disambiguation but requiring clean reference samples. There has been little work on *text-prompted* speaker separation, where the target speaker is specified by semantic attributes (e.g., "female speaker" or "child voice"). Such an approach is practically valuable, as it enables flexible and interpretable control without requiring reference audio.

Audio enhancement. Audio enhancement is another domain where separation methods have been widely applied, aiming to remove additive noise or interference while preserving the underlying signal. Speech enhancement, in particular, has long served as a front-end for automatic speech recognition and hearing-aid systems (Loizou, 2013; Wang and Chen, 2018). Similar to speaker separation, generative modeling has been extensively explored for speech enhancement, improving perceptual quality across diverse acoustic conditions—from early GAN-based approaches (Pascual et al., 2017; Fu et al., 2019) to more recent diffusion-based models (Richardson and et al., 2023; Liu et al., 2024b; Li et al., 2024c). Music enhancement (Schaffer et al., 2022; Kandpal et al., 2022) has also been studied to restore noisy or degraded recordings, or as a post-processing step for instrument separation systems, with techniques often adapted from speech enhancement and source separation. While these enhancement models typically focus on *restoration*, our objective differs. In this work, we focus on *faithful separation*, where the separated track is a content-preserving extraction of the target source without altering intrinsic recording attributes such as echo.

Musical instrument separation. Separating a full-mix music audio into individual instrument tracks has been another widely studied problem in the separation literature, drawing many techniques from general source separation. For example, Demucs (Défossez et al., 2019) introduced a time-domain U-Net architecture with strided convolutions and learned synthesis, and was later extended with hybrid spectro-temporal paths and transformer modules to better capture long-range harmonic structure and transients (Défossez, 2021; Défossez, 2023). KUIELab-MDX-Net (Kim et al., 2021), features a two-stream architecture in the time-frequency domain blends their outputs to provide a strong accuracy-vs-computational-cost trade-off. More recently, Lu et al. (2023) proposes splitting the input complex spectrogram into sub-bands, then applying hierarchical Transformer layers to model both intra-band and inter-band dependencies.

Nevertheless, most music instrument separation methods remain restricted to a fixed ontology. A commonly used benchmark, MUSDB18 (Rafii et al., 2017a), defines output stems as vocals, drums, bass, and other. Accordingly high-performing models (Défossez, 2023; Lu et al., 2023) typically support only up to 5 stems. This limitation constrains open-domain usability and highlights the need for more flexible, prompt-able separation frameworks.

Target sound separation. Most prior work on source separation focuses on specialized domains such as speech or music, while relatively fewer efforts aim to generalize separation to open-domain sound events. Universal

Sound Separation (Kong et al., 2023) explored ontology-driven separation at scale, using weak labels and hierarchical conditioning to disentangle predefined sound classes derived from AudioSet (Gemmeke et al., 2017). However, its reliance on fixed class ontologies limits flexibility and control. To overcome this limitation, recent approaches incorporate natural language prompting, enabling open-vocabulary separation. AudioSep (Li et al., 2023) scales sound separation by leveraging CLAP-aligned audio–text embeddings and large-scale in-the-wild mixtures. CLIPSep (Dong et al., 2023) trains text-prompted separation models using unlabeled videos, conditioning on CLIP (Radford et al., 2021b) embeddings to align audio with textual descriptions. Parallel to these discriminative approaches, generative formulations have also gained traction. FlowSep (Yuan et al., 2024) employs rectified flow matching within a latent space conditioned on FLAN-T5 text embeddings (Raffel et al., 2020), while SoloAudio (Wang et al., 2025) adopts a diffusion-transformer architecture and augments training with synthetic audios. Although prompted separation greatly improves flexibility, it still struggles in specialized domains such as professional music demixing or multi-speaker scenes, where domain-specific systems (Défossez, 2023) remain superior. Moreover, in many practical cases, text prompts alone are insufficient to disambiguate target sounds—for example, differentiating between subtle sound effects or co-occurring events. While prior work (Wen et al., 2025; Xu et al., 2020) explored audio as an alternative conditioning modality, we instead introduce *span prompting*, a temporal conditioning mechanism that enables models to separate sound events without requiring external reference audio.

Visual-prompted separation. Beyond text prompting, several works have explored visual inputs as separation cues, allowing models to identify and extract sounds corresponding to specific visual objects or regions. Early work by Zhao et al. (2018a) showed that spatial location can guide separation by associating visual regions with corresponding sound components, effectively separating instruments in music videos. In the speech domain, Ephrat et al. (2018) demonstrated separating target speaker audio by conditioning on the speaker's facial regions. Recent work such as IIANet (Li et al., 2024b) integrates multi-level audio–visual cues through intra- and inter-modality attention mechanisms, improving cross-modal fusion for speaker separation. In the general sound and music domains, DAVIS (Huang et al., 2025) formulates visual-prompted separation as a conditional generative task, employing a diffusion-based framework to synthesize target sounds conditioned jointly on mixture audio and video inputs. Despite these advances, visual-prompted separation remains underexplored in real-world, open-domain scenarios. Most prior systems are domain-specific—focusing narrowly on speech or instrument separation—and are typically evaluated on synthetic or small-scale benchmarks.

2.2 Audio Separation Benchmark and Evaluation

Separation Benchmarks A large portion of audio source separation research relies on *synthetic mixtures* with available stems to enable reference-based metrics (SDR/SI-SDR). While these datasets have catalyzed progress (e.g., WSJ0-2mix and WHAM!/WHAMR! for speech) Hershey et al. (2016); Wichern et al. (2019a,b), they are limited in realism and domain diversity. Other suites broaden conditions (LibriMix) Cosentino et al. (2020) or emphasize *reference-free* conversational evaluation (AMI, CHiME-5/6, DIHARD, LibriCSS) via DER/WER Carletta (2005); Barker et al. (2018); Watanabe et al. (2020); Ryant et al. (2019); Chen et al. (2020), but lack prompt modality coverage. For music, MUSDB18/HQ and SiSEC/MDX focus on stem separation (vocals, drums, bass, other) Rafii et al. (2017b); Liutkus et al. (2021, 2023), with strong systems like HT-Demucs Défossez (2023), while Slakh2100/MedleyDB/URMP offer instrument diversity under synthetic or smaller-scale multitracks Manilow et al. (2019); Bittner et al. (2014); Li et al. (2018). General sound benchmarks (FUSS, AudioSep) aim at universal separation but are primarily synthetic (Wisdom et al., 2021; Li et al., 2023).

Multimodal AV approaches leverage video (AVSpeech, LRS2/LRS3, VoxCeleb; MUSIC/URMP) Ephrat et al. (2018); Afouras et al. (2018); Chung et al. (2018); Zhao et al. (2018b); Li et al. (2018), and AVS-style benchmarks (AVSBench, LU-AVS) add temporal sound on/off and spatial masks Wu et al. (2022); Liu et al. (2024a). Generative systems (diffusion/flow) further advance AV separation Huang et al. (2024b); Serrà et al. (2024). However, most benchmarks (i) focus on narrow taxonomies, (ii) rely on synthetic/studio stems, and (iii) evaluate a *single* prompt modality.

Separation Evaluation Traditional evaluation of audio separation systems relies on distortion-based metrics such as Signal-to-Distortion Ratio (SDR), Scale-Invariant SDR (SI-SDR), Signal-to-Interference Ratio (SIR), and Signal-to-Artifacts Ratio (SAR) (Vincent et al., 2006; Le Roux et al., 2019). These measures quantify energy

differences between separated outputs and reference signals, and have been widely adopted in benchmark datasets such as WSJ0-2mix (Hershey et al., 2016) and MUSDB18 (Stöter et al., 2019). However, they often fail to reflect perceptual quality: two outputs with similar SDR values can sound drastically different (Le Roux et al., 2019), and their correlation with human Mean Opinion Scores (MOS) is known to be weak (Cartwright et al., 2018; Cano et al., 2016).

Subjective listening tests remain the gold standard for evaluation (Series, 2014), but they are expensive, time-consuming, and difficult to scale. This creates a persistent gap between easily computed distortion errors and perceptually meaningful assessments. To narrow this gap, perceptual metrics originally designed for speech coding and transmission, such as POLQA (Beerends et al., 2013), attempt to model auditory mechanisms. While effective in their intended domains, they generalize poorly to the diverse artifacts produced by modern separation systems (Delgado and Herre, 2024).

More recently, data-driven quality predictors have gained traction, which have shown stronger alignment with human judgments (Huang et al., 2022; Chinen et al., 2020; Mittag et al., 2021; Manocha et al., 2021; Reddy et al., 2021). Nonetheless, most of these efforts focus on speech synthesis, enhancement, or audio generation. In source separation, evaluation still largely relies on SDR-like metrics, with listening studies (Jaffe and Burgoyne, 2025) confirming their poor correspondence to perceptual judgments. This gap underscores the need for evaluation frameworks that move beyond distortion errors and more faithfully reflect human listening experience.

3 Approach

SAM Audio is a generative separation model that extracts both target and residual stems from an audio mixture conditioned on text, visual and temporal span as prompts (see Figure 1). At its core, SAM Audio employs a flow-matching model built on a Diffusion Transformer (Peebles and Xie, 2023) and operates in a DAC-VAE (Polyak et al., 2024) latent space to generate target and residual audio jointly.

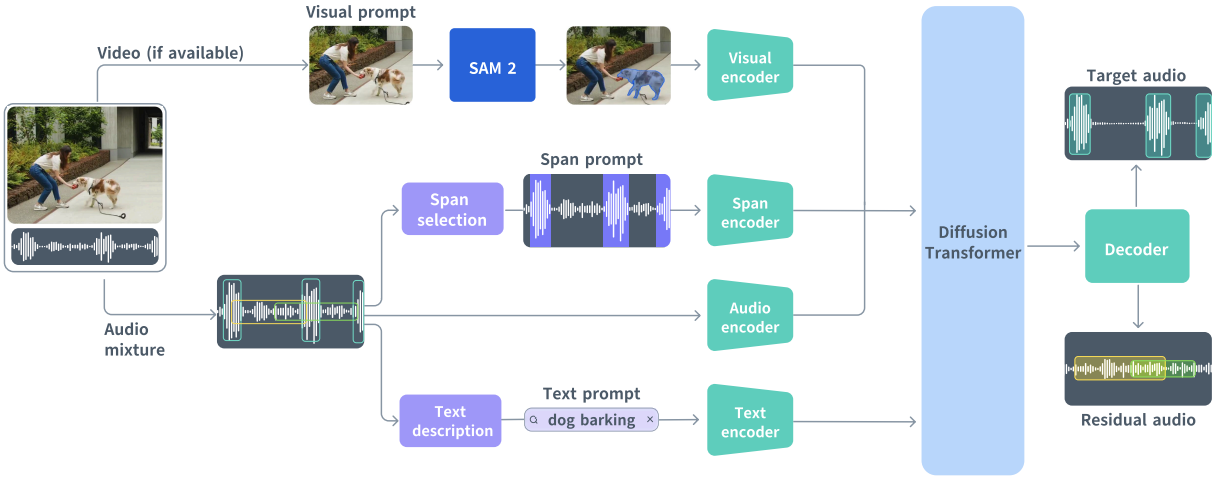

Figure 1 Overview of SAM Audio. Given an audio mixture, SAM Audio separates it into target and residual stems, conditioned on any combination of text descriptions (text prompts), visual masks (visual prompts), and temporal intervals (span prompts).

3.1 Model Architecture

3.1.1 Flow-Matching with Diffusion Transformer

Traditionally, audio separation has been formulated as a discriminative learning problem, where a model predicts a mask over the audio spectrogram to extract the target signal from a mixture (Liu et al., 2024c;

Dong et al., 2023). However, the task of isolating a target sound is *one-to-many* by nature–given an audio mixture, there often exist multiple plausible sounds, particularly in noisy or overlapping acoustic scenes.

In SAM Audio, we adopt a generative modeling approach. Instead of masked prediction, the model learns the target sound distribution conditioned on multimodal prompts, which better captures the underlying variability of sound sources. Specifically, SAM Audio follows the flow matching paradigm (Lipman et al., 2023), similar to recent works in text-to-audio generation (Le et al., 2023; Vyas et al., 2023) and video-to-audio generation (Polyak et al., 2024).

Our model learns a continuous vector field that transports a Gaussian prior sample $x_0 \sim \mathcal{N}(0, I)$ to the data distribution in latent space over a time variable $t \in [0, 1]$. At each step t, the model predicts the instantaneous velocity field $u(x_t, c, t; \theta)$, which is integrated to obtain the sample x_1 at $t = 1$, where c and θ are conditions and model parameters respectively. As is shown in recent works for audio generation, flow matching offers better efficiency and performance than diffusion models (Le et al., 2023; Mehta et al., 2024; Vyas et al., 2023; Prajwal et al., 2024; Polyak et al., 2024).

We use a Diffusion Transformer (DiT) based architecture (Peebles and Xie, 2023), where each transformer block is modulated by the flow time embedding via scale-and-shift operations applied to normalization layers and residual connections. A shared multi-layer perceptron (MLP) maps the scalar t to six modulation parameters (four scales, two biases) used across all transformer blocks, with layer-specific biases added to capture depth-dependent effects. This parameter sharing reduces model size without compromising performance.

We represent audio as a compact sequence of latent features using a separately trained DAC-VAE (Polyak et al., 2024), which adopts a DAC-like (Kumar et al., 2024) autoencoder architecture but replace the residual vector quantizater (RVQ) with a variational autoencoder (VAE) (Kingma, 2013) bottleneck layer. Each audio clip is encoded into a $T \times C$ sequence at 25 Hz with $C = 128$, which offers higher reconstruction quality and lower frame rate compared to commonly used Encodec features (Défossez et al., 2022). The residual vector quantizer is removed and the model is trained as a VAE (Kingma, 2013), yielding smooth Gaussian latents that are well-suited for continuous flow-matching in latent space.

For SAM Audio, we jointly model the target and residual sounds by concatenating their DAC-VAE features x_{tgt} and x_{res} along the channel dimension, forming $x = [x_{\text{tgt}}, x_{\text{res}}] \in \mathbb{R}^{T \times 2C}$. The model then gradually denoises this joint representation, allowing simultaneous prediction of both components in a single pass.

3.1.2 Prompt Types

SAM Audio supports three types of prompts: *text*, *span*, and *visual*, which can be used either individually or jointly to specify the target sound to extract.

Text prompt. The text prompt is a free-form natural language description of the target sound. For audio separation, we find that concise noun–verb phrase (NP/VP) descriptions (e.g., *"woman speaking"*) are more effective and natural for users than full sentences (e.g., *"a woman is delivering a speech"*). This format aligns better with common usage in audio editing.

Visual prompt. Video is a common source of multimodal audio data, and specifying a visual region offers an intuitive way to isolate a sound source. For example, to extract the sound of a barking dog in a video, the user can simply indicate the region of the dog in the visual frames (see Figure 1). We adopt SAM 2 (Ravi et al., 2024) to obtain the target visual mask. Given a video $V \in \mathbb{R}^{T \times H \times W \times 3}$, the user provides a binary mask $M \in \{0, 1\}^{T \times H \times W \times 3}$ by interacting with SAM2 through clicks or bounding boxes. SAM Audio takes (V, M) as input and, for simplicity, only processes the masked visual frames $V \odot M$.

Span prompt. Describing arbitrary audio events purely with text can be challenging, particularly for complex soundscapes such as movie soundtrack. To address this, we introduce *span prompting*, a form of temporal conditioning that specifies the time intervals during which the target sound occurs. The key intuition is that timestamps can disambiguate overlapping sound events. For example, in an audio mixture containing female speech from 0–6 s and dog barking from 1–2 s, the time interval alone suffices to isolate the barking sound. Formally, a span prompt is defined as a set of time intervals $S \in \mathbb{R}^{k \times 2}$ marking the start and end times of the target event. Note span prompting can be inherently ambiguous when multiple sound events occur

simultaneously. In practice, it performs particularly well for foreground sounds or when combined with text or visual prompts (see Section 7.3).

3.1.3 Audio Mixture and Prompt Encoding

To generate the target audio, SAM Audio conditions on both the input mixture and three prompts. The visual and span prompts are first encoded into frame-aligned feature sequences, which are concatenated with the noisy latents before being passed into the DiT backbone. In contrast, the text prompt is encoded into a global textual embedding that is integrated via cross-attention layers within the DiT backbone. The combination of frame-level and global semantic information help capture heterogeneous cues for separation.

Audio encoder. The input mixture is encoded with a separately trained DAC-VAE (Polyak et al., 2024), yielding a latent feature sequence $x_{\text{mix}} \in \mathbb{R}^{T \times C}$ at 25 Hz. We directly operate in the DAC-VAE space to minimize information loss in audios, which ensures that the separated output remains faithful to the original audio.

Text encoder. We encode the text using a pre-trained T5-base encoder (Raffel et al., 2020), producing a token sequence of features $c_{\text{text}} \in \mathbb{R}^{N_{\text{txt}} \times d_{\text{txt}}}$, with $d_{\text{txt}} = 768$. These features are injected into the DiT backbone through cross-attention layers, enabling the model to focus on content that matches the description. As the prompts are short and unambiguous, we do not need to fine-tune the text encoder.

Video encoder. To provide visual grounding, we condition on frame-level video features extracted by the Perception Encoder (PE) (Bolya et al., 2025), a state-of-the-art vision encoder trained with large-scale contrastive vision-language learning that achieves best-in-class zero-shot accuracy on image and video benchmarks. Compared to CLIP (Radford et al., 2021a) or MetaCLIP (Xu et al., 2023), which are widely used in prior works for video-conditioned audio generation (Polyak et al., 2024) or separation (Dong et al., 2023), PE produces more robust and semantically rich representations, particularly for actions and scene context (Bolya et al., 2025). We extract frame-level PE features and resample them to match the audio frame rate. A gated linear projection layer then maps these features to the DiT dimension, and the resulting sequence is concatenated with the audio representation on a frame-by-frame basis.

Span encoder. Analogous to the phoneme sequence used in text-to-speech (TTS), we convert the span prompt, originally expressed as a set of time intervals $S \in \mathbb{R}^{k \times 2}$, into a frame-synchronous token sequence $S'_{1:T}$. Each token $s'_t \in \{\texttt{<sil>}, +\}$ denotes whether the target event is silent or active at frame t. The token sequence is embedded using a learnable embedding table and concatenated channel-wise with the audio features. This provides an explicit temporal prior that guides the model to focus on the relevant regions for extraction.

Not all training examples have all three prompt modalities. In such cases, we provide dummy conditions: an empty string for text, an all-zero vector sequence for video, and a sequence of special null tokens (`<null>`) for span. To improve robustness, we also *randomly drop* each available prompt during training with probabilities $p_{\text{video}}, p_{\text{text}}, p_{\text{span}}$ and replace them with the corresponding dummy conditions. This encourages the model to robustly handle missing prompts at inference time.

3.2 Training Objective

Flow Matching Objective. At each flow step t, the model receives the noisy latent x_t and conditioning set

$$c = \{x_{\text{mix}}, c_{\text{text}}, c_{\text{vid}}, c_{\text{span}}\},$$

where c_{text} representing text features, c_{vid} representing masked video features, and c_{span} is the target audio span embedding. The model predicts the velocity field $u(x_t, c, t; \theta)$ used to update x_t toward the clean target-residual representation

$$\mathcal{L}_{FM} = \|u(x_t, t, c; \theta) - (x_1 - (1 - \sigma_{min})x_0)\|, \tag{1}$$

where x_1 and $x_0 \sim \mathcal{N}(0, 1)$ are respectively target audio and random noise.

Audio Representation Alignment. In order to separate correct sound events based on the user prompt, the model must first infer *what* to extract and *when* it occurs in the mixture. To encourage this behavior, we

introduce auxiliary MLP layers following Yu et al. (2025) that project intermediate joint representations, formed by conditioning the mixture audio with the input prompt, into the embedding space of an external audio event detection (AED) model. We then apply an auxiliary alignment loss that forces these projected representations to match the AED embedding of the ground-truth target audio. By explicitly guiding the hidden states to align with event-level semantics and temporal structure, this objective complements the flow-matching loss and encourages the model to internalize "what and where to extract" during training.

For the alignment objective, we first extract the target audio embedding $a_{tgt} = \text{AED}(x_{tgt}) \in \mathbb{R}^{T \times F}$, where F is the feature dimension of the AED module. Then, with $h_t \in \mathbb{R}^{T \times D}$ being the hidden DiT representation, where T is the input audio length and D is the transformer's channel dimensionality, we project it to $\hat{a}_t = \phi(h_t) \in \mathbb{R}^{T \times F}$ where $\phi(\cdot)$ is a small MLP. Then, we maximize the cosine similarity between a_{tgt} and $\hat{a}_t$ by minimizing

$$\mathcal{L}_{aux} = \mathbb{E}_t \left[1 - \text{sim}(\hat{a}_t, a_{tgt}) \right], \tag{2}$$

where $\text{sim}(A, B) = \frac{A \cdot B}{\|A\| \|B\|}$.

We combine both FM loss and aux loss into:

$$\mathcal{L} = \mathcal{L}_{FM} + \lambda * \mathcal{L}_{aux}$$

where λ is a hyperparameter to control the influence of the alignment loss. We use the AED model in (Kong et al., 2020) to extract the target audio representation a_{tgt}. For auxiliary MLP aux_ϕ, we deployed a 3-layer MLP with 2048 hidden dimensions, GeLU activation function, and layer normalization.

3.3 Boosting text prompting with span prediction

Text prompts provide the most accessible interface for specifying the target source, as users can simply describe the desired sound. Despite its simplicity, span annotations provide frame-level control when the target sound is active, resulting in more accurate separations. Empirically, joint text and span conditioning outperforms text-only conditioning for general sound event separation (see Section 7.3). However, obtaining ground-truth spans requires manual boundary labeling, which can be time-consuming. To bridge this gap, we propose a simple method for approximating span annotations at inference time. To this end, we use a helper model called $\text{PE}_{\text{A-Frame}}$ to predict the spans.

Specifically, $\text{PE}_{\text{A-Frame}}$ (Vyas et al., 2025) is a language-queried sound event detection model that learns to detect the precise time frames in an audio signal where a sound described by free-form text is active. Given an audio clip and a textual description (e.g., "a dog barking"), the model predicts frame-level probability scores indicating when that event occurs. It extends CLAP-style audio–text embeddings by introducing a frame-level loss function, enabling fine-grained temporal grounding of language in continuous audio.

Given an audio mixture x_{mix} and a text prompt c_{text}, we employ $\text{PE}_{\text{A-Frame}}$ (Vyas et al., 2025) to estimate the frame-wise activity of the sound event described by the text description. By thresholding the frame-wise activity, we obtain an approximate span sequence $\hat{c}_{\text{span}}$. Similar to Wang et al. (2022a), we then condition the separation model on both the predicted span $\hat{c}_{\text{span}}$ and the original text prompt c_{text}. Formally, given a SAM audio model $\mathcal{M}$ and text conditioning c_{text}, we perform text prompting through $\mathcal{M}(x_{\text{mix}}, c_{\text{text}}, \text{PE}_{\text{A-Frame}}(x_{\text{mix}}, c_{\text{text}}))$ instead of using it $\mathcal{M}(x_{\text{mix}}, c_{\text{text}})$.

3.4 Longform audio separation with Multi-diffusion

Processing arbitrarily long audio in a single forward pass is infeasible due to GPU memory limits. A naive chunk-wise approach: splitting the mixture into disjoint segments and applying SAM Audio independently, introduces boundary artifacts and discontinuities, especially for sustained or slowly varying sounds. To ensure temporal coherence, we adopt the *multi-diffusion* approach (Bar-Tal et al., 2023; Polyak et al., 2024) and adapt it for audio separation.

Specifically, we divide the mixture into overlapping windows. For window j, we construct window-specific conditioning $c^{(j)} = \{x^{(j)}_{\text{mix}}, c^{(j)}_{\text{vid}}, c^{(j)}_{\text{span}}, c_{\text{text}}\}$. At each flow-matching step t_i, we solve the ODE in parallel for all windows using the shared time schedule $\{t_i\}$: $\tilde{x}^{(j)}_{t_{i+1}} = x^{(j)}_{t_i} + \Delta t_i \, u\Big(x^{(j)}_{t_i}, t_i, c^{(j)}\Big)$.

We then merge the local predictions into the global latent using normalized soft masks:

$$x_{t_{i+1}} = \sum_j \mathrm{pad}\Big(m^{(j)} \odot \tilde{x}^{(j)}_{t_{i+1}},\, j\Big), \qquad \sum_j \mathrm{pad}\Big(m^{(j)},\, j\Big) = \mathbf{1},$$

where $m^{(j)}$ is a triangular window applied to segment j, and `pad` zero-pads it back to global length.

This iterative procedure allows information to propagate across overlapping regions at every diffusion step, producing smooth, globally coherent long-form separations without boundary artifacts.

4 Data

The training data for SAM Audio consists of pairs of the form $(x_{\text{mix}}, x_{\text{tgt}}, x_{\text{res}}, c)$, where x_{mix} is an audio mixture, x_{tgt} is the target stem, x_{res} is the residual stem, and $c \in \{\text{text}, \text{video}, \text{span}\}$ is one or more conditioning prompts. This section describes our strategy for constructing such training tuples. We first introduce the audio data used to form $(x_{\text{mix}}, x_{\text{tgt}}, x_{\text{res}})$, followed by how we generate prompts for c. As SAM Audio covers all audio sub-modalities including speech, music and general sound effects, we devise data strategy for each audio sub-modality.

4.1 Audios

Category	Input Audio Mixture x_{mix}	Target Audio x_{tgt}	Residual Audio x_{res}
Fully-real triplets	✓	✓	✓
Synthetic mixtures	✗	✓	✓
Pseudo-labeled stems	✓	✗	✗

Table 1 Overview of training data types used in SAM Audio. Each row indicates whether the mixture, target, and residual signals originate from real audio (✓) or are synthetic/pseudo-labeled (✗).

Broadly, our training data come from (1) a large-scale, medium-quality audio–video corpus (about ∼1M hours) and (2) a collection of small- to medium-scale high-quality audio datasets (see Table 2). Only a small subset includes ground-truth stems, which can be directly used for model training. The rest are used to construct synthetic mixtures or pseudo-labeled data depending on whether we synthesize the mixture (x_{mix}) or synthesize the target/residual stems ($x_{\text{tgt}}, x_{\text{res}}$).

Table 1 summarizes the three data construction regimes used in SAM Audio. Each regime differs in whether the mixture, target, and residual signals originate from real recordings or are obtained via synthesis or pseudo-labeling.

Data Source	Modality	w/ stem	Quality	Sound	Music	Speech	#Samples (M)	#Hours (K)
General Video	audio, video	✗	Medium	✓	✓	✓	$\mathcal{O}(100)$	$\mathcal{O}(1000)$
General Audio	audio	✗	Medium	✓	✓	✓	$\mathcal{O}(1)$	$\mathcal{O}(1)$
Speech Conversation	audio	✓	High	✗	✗	✓	$\mathcal{O}(10)$	$\mathcal{O}(10)$
HQ Music	audio	✗	High	✗	✓	✗	$\mathcal{O}(10)$	$\mathcal{O}(10)$
Multi-track Music	audio	✓	High	✗	✓	✗	$\mathcal{O}(0.1)$	$\mathcal{O}(1)$
HQ SFX	audio	✗	High	✓	✓	✓	$\mathcal{O}(10)$	$\mathcal{O}(10)$
HQ Video	audio, video	✗	High	✓	✓	✓	$\mathcal{O}(0.1)$	$\mathcal{O}(0.1)$

Table 2 Sources of training data for SAM Audio. "w/ stem" indicates whether ground-truth target/residual stems are provided.

4.1.1 Fully-real triplets

The ideal form of training data $(x_{\text{mix}}, x_{\text{tgt}}, x_{\text{res}})$ consists of real, perfectly isolated stems that sum to a natural mixture: $x_{\text{mix}} = x_{\text{tgt}} + x_{\text{res}}$. Such data provides the cleanest possible supervision, as the model can directly learn to map from a mixture to its constituent sources without synthetic artifacts.

Fully-real triplets are available in music and speech domains.

Fully-real music. We utilized internal high-quality music data with clear instrument stems (i.e., *Multi-track Music* in Table 2) to introduce instrument extraction capabilities. This datasets contains 10,610 unique music compositions over a total of 536 hours. Each music composition consists of multiple instruments (e.g. drums, bass, guitars) and vocal tracks. We create the triplet $x_{mix}, x_{tgt}, x_{res}$ by mixing those instruments within same composition. Assume $x_{\text{mix}} = \sum_{i=1}^{N} x_i$ where x_i denotes the audio of each instrument stem, then we create N triplets of different targets, where for each instrument audio x_i, we define $x_{\text{tgt}} = x_i$ and residual $x_{\text{res}} = \sum_{j=1, j\neq i}^{N} x_j$. To increase the robustness, instead of simply adding each stem, we rescale each instrument stem by SNR ± 5 respect to the target audio.

Fully-real speech. We use a conversational speech corpus (i.e., *Speech Conversation* in Table 2) containing 21,910 hours of audio. Each conversation provides two clean speaker stems, denoted x_{spk1} and x_{spk2}. We form the mixture as $x_{\text{mix}} = x_{\text{spk1}} + x_{\text{spk2}}$ and construct two training triplets: $(x_{\text{mix}}, x_{\text{spk1}}, x_{\text{spk2}})$ and $(x_{\text{mix}}, x_{\text{spk2}}, x_{\text{spk1}})$. To improve robustness, we rescale the residual speaker with a randomly sampled SNR offset of ± 15 dB relative to the target speaker.

4.1.2 Synthetic audio mixtures

While real stem data is highly valuable, it is relatively scarce and often domain-specific (e.g., multi-speaker speech data or instrument dataset). Synthetic mixtures consists of mixing two audios randomly, which have been a common training data strategy used in prior works (Li et al., 2023; Serrà et al., 2024; Wu et al., 2023a) for general sound effects. As SAM Audio covers both special and audio domains, we design noise mixing strategy tailored for each domain.

Synthetic noisy music. We utilized internal data that has $\sim 20K$ hours of clean music (i.e., *HQ Music* in Table 2). To create the synthetic mixture, we mix this dataset with non-music audio from the general sound data with SNR ± 15. To improve vocal and background music separation, we categorized our music datasets into vocal and non-vocal datasets with AED (Koutini et al., 2022) and set text prompts accordingly (see Section 4.2.1).

Synthetic noisy speech. For speech extraction, we use the same conversational speech corpus described in Section 4.1.1. To construct synthetic mixtures, we directly add the two speaker stems from each conversation into a single-channel mixture, producing $x_{\text{speech}} = x_{\text{spk1}} + x_{\text{spk2}}$, which naturally contains both single-speaker and multi-speaker segments. For the noise component x_{noise}, we randomly sample a non-speech audio clip from the general sound pool. The final mixture is then formed as $x_{\text{mix}} = x_{\text{speech}} + x_{\text{noise}}$.

Synthetic general sound. We categorize our sound effects data into two primary types: *in-the-wild* sound (i.e., *General Video* and *General Audio* in Table 2)and high-quality sound (i.e., *HQ-SFX* in Table 2). In-the-wild audio consists of recordings captured in uncontrolled environments and typically contains multiple overlapping sound events as well as ambient sound. In contrast, pro sound recordings are produced in controlled settings as source material for professional audio creation and they usually contain a single sound event with no background noise. For audio mixing, we generate training examples by mixing in-the-wild clips with other in-the-wild clips, and pro sound clips with either other pro sound clips or in-the-wild clips.

4.1.3 Pseudo-labeling Data Engine

A limitation of the random mixing strategy described above is that it often produces mixtures that are unrealistic and poorly reflect real-world audio mixture. For example, mixing crowd cheering in a stadium with bird chirps recorded in a forest yields an unnatural combination that rarely occurs in the wild. Training on such mixtures can circumvent the model learning meaningful separation cues.

To address this issue, we synthesize more realistic training tuples by using an intermediate checkpoint of SAM Audio, trained with the identical recipe but without pseudo-labeled data. We generate target and residual stems from natural audio recordings using this early verison of SAM Audio, essentially bootstrapping new training data from unlabeled mixtures.

The data synthesis pipeline consists of two stages: *Separation* with SAM Audio and *filtering* various characteristics of the separated audio using specialist models.

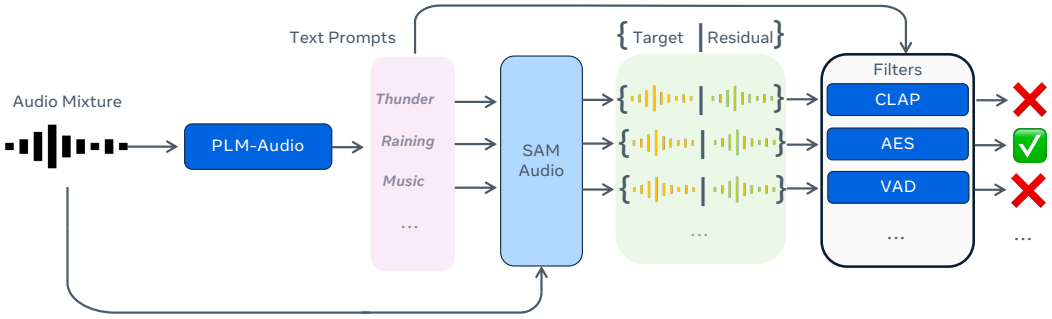

Figure 2 Illustration of our data synthesis pipeline. PLM-Audio generates text prompts from mixtures, which guide SAM Audio to produce target/residual stems. A filter stage retains only high-quality pseudo-labeled stems.

Separation. In the separation stage, we run the intermediate SAM Audio checkpoint as a data engine to generate target and residual stems from large-scale unlabeled audio recordings, forming a pool of pseudo-labeled training candidates.

We focus primarily on text-based synthesis, given the scalability of text-prompted separation. We first apply the audio captioning model PLM-Audio Vyas et al. (2025) (see Section 4.2.1) to extract textual descriptions for the 1M-hour *in-the-wild* general sound data. The output of PLM-Audio is a list of audio event descriptions and we use each one as the text prompt for SAM Audio pseudo-labeling.

Filtering. In practice, this preliminary checkpoint exhibits significant variability in separation quality. To avoid degrading the final model, we apply strong filtering to remove low-quality candidates using a set of criteria covering various quality aspects. We measure text–audio alignment using CLAP (Wu et al., 2023b), assess audio cleanliness via the production-complexity (PC) axis of the Audiobox-aesthetic model (Tjandra et al., 2025), and detect overly silent outputs using a voice-activity detector (pydub). To enhance visual-prompt training, we further curate a subset with strong audio–visual correspondence. Given the same text prompt used for separation, we apply an in-house text-prompted video segmentation model to obtain visual masks, and calculates the audio-visual alignment score and mask coverage. For the former metric, we leverage ImageBind (Girdhar et al., 2023), a large-scale audio–video–text contrastive model, to compute the cosine similarity between the embeddings of the audio track and masked visual frames. A pseudo-labeled sample is retained only if all of the following conditions in Table 3 hold.

Criterion	Threshold
Text–Audio Filtering (all must pass)	
CLAP(text, target audio)	> 0.35
CLAP(text, residual audio)	< 0.0
Aesthetic PC score (target audio)	< 2.5
Silence ratio (target audio)	$< 95\%$
Additional Visual-Audio Filtering	
Mask coverage ratio (masked region)	> 0.02
ImageBind(target audio, masked region)	> 0.2

Table 3 Filtering criteria for synthesized training data. Samples are kept only if all text–audio criteria are satisfied; visual-prompt samples require additional visual-related constraints.

Pseudo-labeling is applied to the general video portion of our training data (Table 2). We only run pseudo-labeling on mixtures that contain multiple sound events. After multi-stage filtering, the resulting pseudo-labeled set is substantially smaller than the high-quality audio used for synthetic mixtures. Table 4 summarizes the final pseudo-labeled datasets used for SAM Audio training.

Data Source	Modality	Synthetic Stems	Sound	Music	Speech	#Samples (M)	#Hours (K)
PL-Audio	audio	✓	✓	✓	✓	$\mathcal{O}(1)$	$\mathcal{O}(1)$
PL-Video	audio, video	✓	✓	✓	✓	$\mathcal{O}(0.1)$	$\mathcal{O}(0.1)$

Table 4 Pseudo-labeled training data used in SAM Audio. Both audio-only and audio–video inputs are processed by an intermediate SAM Audio checkpoint to produce synthetic stems.

4.2 Prompt Generation

Given an audio tuple $(x_{\text{mix}}, x_{\text{tgt}}, x_{\text{res}})$, we generate a corresponding prompt c that specifies the target audio x_{tgt}. This section describes how we generate textual, visual, and span-based prompts.

4.2.1 Text

Generating a text prompt for the target audio is equivalent to producing a concise and semantically faithful description of its content. In the context of separation, we find that prompts expressed in NP/VP form are more intuitive and effective than full natural-language sentences. For example, *"dog barking"* is preferred over *"a dog is barking"* because it more directly specifies the sound event to extract.

To obtain high-quality prompts at scale, we train an in-house audio–language model, *PLM-Audio*, a visually aware audio-LLM fine-tuned for audio captioning. Whenever raw metadata is available, we merge it with the PLM output, and we discard captions with low audio–text correspondence.

PLM-Audio. PLM-Audio is an audiovisual version of the Perception Language Model (PLM) (Cho et al., 2025) which instead of the visual Perception Encoder (PE) Bolya et al. (2025), uses an audiovisual encoder, PE-AV (Vyas et al., 2025). PE-AV extracts frame-level audio-visual representations which are decoded with an 8B-parameter LLaMA decoder (Dubey et al., 2024) for language output.

We train PLM-Audio using a three-stage process, closely mirroring the PLM (Cho et al., 2025) training recipe:

1. *Warm-up.* We freeze the backbone encoder and fine-tune only a lightweight projection MLP that maps PE-Audio embeddings into the LLM embedding space. This aligns the feature spaces while preserving the pre-trained representations.
2. *Mid-training.* We unfreeze the full model and fine-tune on a large corpus of synthetic captions generated by PE-Audio, including both audio-only and audio-visual samples. We prioritize utterances with high audio-video alignment scores (e.g., high ImageBind similarity), ensuring that training focuses on well-grounded examples.
3. *Post-training.* We fine-tune on a curated mixture of tasks that encourage prompt generation in NP/VP format and improve coverage of downstream separation-relevant attributes.

Incorporating metadata when available. A subset of our training data consists of professionally recorded ambience and sound-effects datasets. These datasets often include metadata such as titles, descriptions, and keyword tags. However, such metadata can be noisy, containing irrelevant, or overly broad information. Meanwhile, PLM-Audio typically produces relevant audio captions but may occasionally emit false-positive sound events. To obtain reliable NP/VP-style prompts, we apply a three-step rewriting pipeline:

1. Run PLM-Audio on each audio clip to obtain an *initial caption*.
2. Use an LLM (Dubey et al., 2024) to merge all available metadata (title, keywords, description) with the PLM-Audio caption and generate a cleaned, *detailed description* while discarding irrelevant content.
3. Instruct the same LLM to extract only the *NP/VP-style sound-event phrases* from the detailed description.

CLAP filtering. Because most of our captions are automatically generated rather than manually annotated, we perform a filtering step to remove low-quality captions. To this end, we compute text–audio similarity using CLAP (Wu et al., 2023b) and discard samples with similarity below 0.28, which corresponds to approximately the 25th percentile in preliminary experiments. We evaluated several percentile thresholds

($p = \{0.05, 0.10, 0.25, 0.50\}$) and found that filtering at $p = 0.25$ achieves the best balance between data scale and caption quality.

Note a small subset of our training data contains domains where quasi–ground-truth text descriptions can be obtained without the above pipeline. For the *fully-real music* subset (Section 4.1.1), we derive text prompts using simple templates applied to instrument labels (e.g., *piano* → *piano playing*). For the *fully-real speech* subset (Section 4.1.1), we apply a pretrained gender classifier (Huh, 2024) to identify the speaker's gender and then map it to a templated description (e.g., *female* → *woman speaking*).

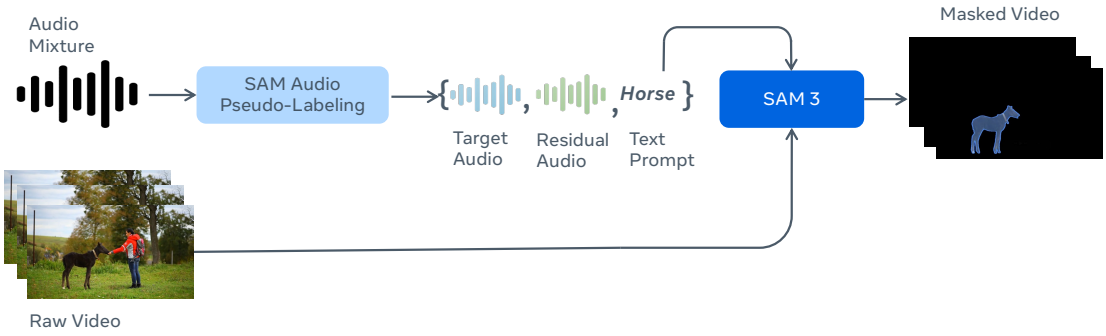

Figure 3 Illustration of pseudo-labeled visual data. The pseduo-labeling pipeline produces a text caption of the target audio, which is used to prompt SAM3 to obtain the visual mask.

4.2.2 Video

During inference, visual prompting allows users to specify a target source by providing a visual mask over the video, and the model is required to separate the corresponding sound. The visual mask is generated using SAM 2 (Ravi et al., 2024), and covers either a single object or a region of interest across all frames. The primary goal of model training is to encourage the model to associate visually grounded sources to the audio so that, at test time, users can perform instance-level separation by simply selecting an object in the video.

Synthesizing videos for arbitrary audio is challenging, as it requires generating both spatio-temporally coherent frames. We therefore rely exclusively on natural videos with paired audio tracks. Our visual training data construction recipe consists of two complementary components: (a) whole video, (b) visual masks.

Whole-video. For most of our training data, we directly use full video clips without relying on explicit segmentation masks, allowing the model to learn weakly supervised associations between visual content and audio events. However, many real-world videos contain sound that is not visually grounded—for example, background music added during post-production or off-screen narrations. Training on such samples can be detrimental, as it encourages spurious associations between visual frames and unrelated audio. To mitigate this, we compute the ImageBind score (Girdhar et al., 2023) for every video and retain only those exceeding a threshold, biasing the dataset toward diegetic, visually grounded sound sources. This filtering is applied to our ∼1M-hour general sound corpus.

Visual mask. For pseudo-labeled audio–visual data, we explicitly construct a visual mask corresponding to the target sound. Unlike whole-video conditioning, pseudo-labeled clips often contain multiple sounding objects, only a subset of which matches the target audio. We therefore generate masks by prompting SAM3 (Carion et al., 2025) with the target sound's text caption derived from PLM-Audio (see Figure 3). In practice, we observe substantial variance in mask quality due to factors such as audio–visual mismatch (e.g., the caption describes a sound that is not visually present) and SAM3 prediction errors. Accordingly, the ImageBind-based filtering pipeline described in Section 4.1.3 is critical to ensure that only samples with reliable audio–visual correspondence remain in the final training set.

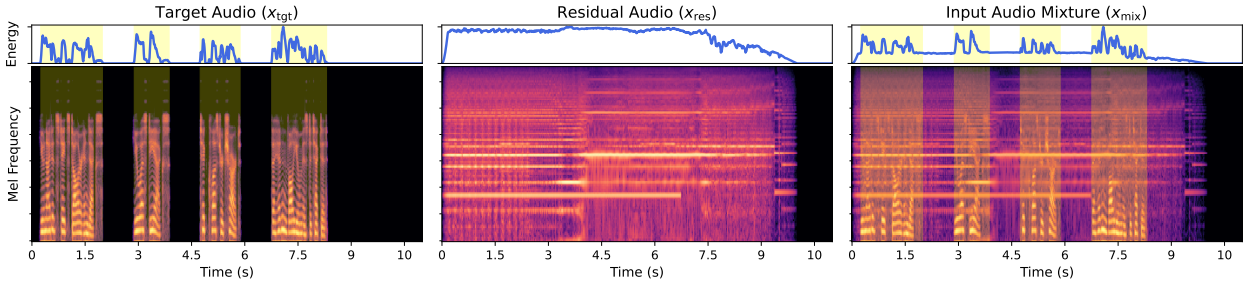

Figure 4 Illustration of span generation. RMS energy (top) and Mel-spectrograms (bottom) for the target x_{tgt}, residual x_{res}, and mixture x_{mix}. Yellow intervals denote detected spans corresponding to active sound events.

4.2.3 Span

Not all audio mixtures are suitable for span prompting. In particular, mixtures dominated by long, continuous ambience (e.g., rain, traffic) offer little temporal structure, making span information uninformative. Our key observation is that span prompting is most effective for *spiky*, discrete sound events—such as *door slam* or *dog bark*—whose temporal locations provide strong cues for separation.

To this end, we construct audio mixtures only from audios that satisfy this property. For general sound, the target audio is sampled from *HQ SFX* sound data (Section 4.1.2), which predominantly contains clean, isolated sound events, while noise audio is sampled from long-duration clips (> 10 s) that typically contain ambient or continuous backgrounds. For music and speech, we additionally incorporate the fully-real music and fully-real speech datasets to ensure broad domain coverage.

Given a target audio, we apply VAD (pydub) with a silence threshold of -40 dBFS and a minimum sounding duration of 250 ms to obtain a binary frame-level mask indicating sounding versus silent regions. Consecutive sounding segments are then converted into time intervals, which we treat as the span prompts supplied to SAM Audio. An overview of generating span-prompted audio data is shown in Figure 4.

5 Evaluation

5.1 SAM Audio-Bench

5.1.1 SAM Audio-Bench: Unifying Modalities, Domains, and Realism

We observe three persistent gaps in existing separation benchmarks: (i) realism vs. references (in-the-wild acoustics vs. availability of stems), (ii) limited *prompt modality* coverage across text/visual/time, and (iii) *cross-domain* breadth under a unified protocol (speech, music, instruments, general sounds).

We introduce a **real, in-the-wild, multi-modal separation benchmark** addressing these gaps:

1. **Ecological validity**: All items are sourced from in-the-wild audio/video or production-quality video: **AudioSet** (Gemmeke et al., 2017), **VGGSound** (Chen and Zisserman, 2020), **MUSIC** (Zhao et al., 2018b), **MUSIC-AVQA** (Li et al., 2022), **AVSpeech** (Ephrat et al., 2018) and **CondensedMovies** (Rong et al., 2020).
2. **Multi-modal prompting**: Each 10 s test instance includes human annotated *visual* **SAM masklets** (Kirillov et al., 2023) (when sounding object is on-screen), *temporal* **positive/negative spans**, and *language* **text descriptions**.
3. **Taxonomic coverage**: Stem taxonomy is seeded from **AudioSet** ontologies (Gemmeke et al., 2017), with annotator-extendable classes. Dedicated tasks: **speech cleaning**, **speaker separation**, **music cleaning/removal**, **instrument stems** (37 classes), and **general sounds**.

Annotation protocol. Annotators enumerate sound events with text descriptions (that we subsequently normalize to concise noun/verb phrases), draw visual masklets for visible sounding sources – these masklets

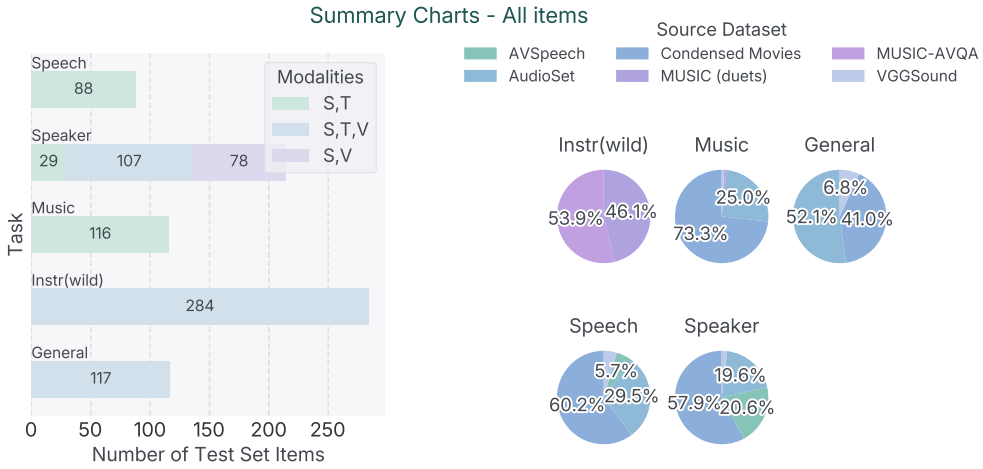

Figure 5 Summary of task, modality, and dataset coverage in SAM Audio-Bench. The modality abbreviations are as follows: "T" indicates the item can be used with a text-only prompt (e.g. for speaker separation this implies that the text description can be unambiguously associated with a single speaker), "V" indicates that the target sound is on-screen and that we have a SAM masklet provided and "S" denotes that there are event boundaries for the target sound.

are available for every frame in the video, and every video is standardized to 24fps – and mark *temporal* presence/absence spans. Each (target sound, video) pair yields interchangeable text/vision/time prompts, enabling controlled ablations across modality combinations.

5.1.2 Summary Comparison to Existing Evaluation Sets

Table 5 summarizes modality coverage, realism, domain scope, and reference availability across representative benchmarks versus our SAM Audio-Bench.

As Table 5 shows, SAM Audio-Bench uniquely combines (i) *real* in-the-wild audio/video, (ii) *multi-modal prompts* (text, visual masklets, temporal spans) on the *same* items, (iii) *cross-domain* coverage (speech, music, instruments, general sounds), and (iv) *reference-free human* evaluation.

Figure 5 shows summary statistics of the entire released SAM Audio-Bench dataset, including how many test set items are available for each task, and which prompt modalities are supported for each; we also show a breakdown of which datasets SAM Audio-Bench videos originate from.

Additional figures in Appendix B show additional breakdowns for each task, for both the entire set and also for a more balanced subset used for human evaluations.

5.2 SAM Audio Judge Model

To develop SAM Audio judge, we conduct a systematic investigation into perceptually aligned evaluation for audio separation. Specifically, we aim to design an evaluation framework that (i) operates without requiring reference signals, making it suitable for real-world applications, (ii) enables fine-grained perceptual assessment of separated audio, and (iii) exhibits stronger correlation with human listening judgments.

5.2.1 Data Collection

Existing evaluation guidelines for audio separation are typically simplistic, focusing only on coarse criteria such as the relevance between the separated audio and the prompt, or the overall audio quality of the output Liu et al. (2022). Such high-level and loosely defined objectives make the evaluation ambiguous. Scores collected under these settings often conflate multiple perceptual aspects and may be biased toward certain criteria depending on the raters' individual interpretations, resulting in outcomes that are difficult to interpret consistently.

Table 5 Comparison of representative audio separation benchmarks vs. SAM Audio-Bench. Real: Y/N/M (mixed real+synthetic). Prompts: T=Text, V=Visual, S=Temporal spans.

		Task coverage						
Benchmark	**Real**	**Sp. clean**	**Spkr sep.**	**General**	**Mus. clean**	**Instr. stems**	**Prompt(s)**	**Source / mixtures**
WSJ0-2mix / WHAM! / WHAMR![a]	N	–	✓	–	–	–	–	Synthetic 2-spk mixtures from WSJ0; WHAM/WHAMR add real noise
LibriMix[b]	N	–	✓	–	–	–	–	Synthetic speech mixtures from LibriSpeech + noise
DNS / VoiceBank+DEMAND[c]	M	✓	–	–	–	–	–	Noisy speech from VoiceBank, DNS challenge (synthetic + some real)
FUSS[d]	N	–	–	✓	–	–	–	Synthetic mixtures from diverse sound event stems
MUSDB18 / MDX[e]	M	–	–	–	–	✓	–	Studio multitrack music; vocals/drums/bass/other stems
Slakh2100[f]	N	–	–	–	–	✓	–	Fully synthetic MIDI-rendered multitrack music
MedleyDB / URMP[g]	M	–	–	–	–	✓	–	Real and studio multitrack ensembles, instrument stems
AudioSep (AudioSet / AudioCaps / Clotho mixtures)[h]	N	✓	–	✓	✓	✓	T	Synthetic mixtures from AudioSet, AudioCaps, Clotho, etc.
DCASE lang-queried source sep.[i]	M	–	–	✓	–	–	T	Synthetic mixtures from captioned / labeled sound events (e.g., FSD50K/AudioSet-style)
CLAP-based text-queried sep.[j]	N	–	–	✓	–	–	T	Synthetic mixtures from AudioSet / FSD50K-style datasets
AVSpeech-based AV speech sep.[k]	M	✓	✓	–	–	–	V	Synthetic 2-spk mixtures, sources from AVSpeech, LRS2/3, VoxCeleb
MUSIC[l]	M	–	–	–	–	✓	V	Real online performance videos; synthetic or curated multi-instrument mixtures
AVSBench / LU-AVS[m]	M	–	–	✓	–	✓	V,S	Real YouTube-like videos with object/instrument sounds, AV masks/spans
SAM Audio-Bench (ours)	**Y**	✓	✓	✓	✓	✓	**T,V,S**	**Real in-the-wild A/V from AudioSet, VGGSound, MUSIC, AVSpeech, CondensedMovies**

[a] WSJ0-2mix; WHAM!/WHAMR! (Hershey et al., 2016; Wichern et al., 2019a,b).
[b] LibriMix (Cosentino et al., 2020).
[c] VoiceBank+DEMAND; DNS challenge datasets (Valentini-Botinhao, 2016; Reddy et al., 2020).
[d] FUSS (Wisdom et al., 2021).
[e] MUSDB18/HQ; SiSEC/MDX (Rafii et al., 2017b; Liutkus et al., 2021, 2023).
[f] Slakh2100 (Manilow et al., 2019).
[g] MedleyDB; URMP (Bittner et al., 2014; Li et al., 2018).
[h] AudioSep benchmark from mixtures of AudioSet, AudioCaps, Clotho, etc. (Li et al., 2023; Kim et al., 2019; Drossos et al., 2020; Gemmeke et al., 2017).
[i] DCASE language-queried source separation benchmark (e.g., DCASE 2024 Task 8).
[j] CLAP-based text-queried separation benchmarks (Wu et al., 2023a).
[k] AV speech separation on AVSpeech, LRS2/LRS3, VoxCeleb (Ephrat et al., 2018; Afouras et al., 2018; Chung et al., 2018).
[l] MUSIC is a canonical AV instrument separation dataset, used by Zhao et al. (2018b).
[m] AVSBench; LU-AVS (Wu et al., 2022; Liu et al., 2024a).
Abbrev.: Real Y/N/M; Sp. clean = speech cleaning; Spkr sep. = speaker separation; General = open-domain / general sound separation; Mus. clean = music cleaning; Instr. stems = instrument stem separation; Prompts: T=text, V=visual, S=temporal spans.

In addition, existing studies rarely examine the difficulty of audio separation tasks themselves. Most prior work has focused solely on evaluating model outputs, without systematically analyzing how intrinsic factors such as the number of overlapping sources, loudness imbalance, or acoustic similarity between sounds affect human perception of task difficulty. Understanding separation difficulty is crucial, as it provides insights into the limitations and robustness of current models, facilitates the curriculum design for training and evaluation, and enables difficulty-aware benchmarking and adaptive model selection in real-world applications.

To better characterize the performance of audio separation models and the intrinsic difficulty of audio separation tasks, we introduce a new human annotation guideline, **SAM Audio Judge (SAJ)**, which defines nine perceptual dimensions.

The SAJ performance dimensions evaluate how well a model separates the target sounds:

- **Recall**: Does the extracted audio contain all of the target sounds specified in the prompt?
- **Precision**: How effectively does the model remove non-target sounds from the extracted audio?
- **Faithfulness**: For target sounds present in the extracted audio, how similar do they sound to their counterparts in the original mixture?
- **Overall quality**: What is the overall perceptual quality of the model's output?

In addition, the SAJ difficulty dimensions assess the complexity of the separation task itself:

- **Counting**: How many non-target sounds are present in the source audio?
- **Overlapping**: To what extent do the target sounds overlap with non-target sounds?
- **Loudness**: How loud are the target sounds relative to the non-target sounds?
- **Confusion**: How easily can the non-target sounds be mistaken for the target sounds?
- **Overall difficulty**: Considering all the above factors, how difficult is it to extract the target sounds from the mixture?

5.2.2 Data Annotation

Based on the above definitions, we design an annotation task to collect SAJ data, where human raters evaluate the nine axes using a five-point Likert scale (1–5). We focus on **text-prompted** SAJ, the most commonly used scenario. We provide a comprehensive annotation guideline, which offers detailed explanations of each axis and specifies fine-grained aspects to consider during evaluation. An example of the annotation interface is shown in Appendix C.1. To help raters calibrate their judgments, the guideline is supplemented with numerous audio examples and score references that illustrate what constitutes high or low scores along each dimension. We also design a rater qualification program to ensure the selection of high-quality annotators. Details appear in Appendix C.2.

Split	Modality	Duration	Samples
Training Set	Speech	59.31 hrs	13,149
	Music	133.64 hrs	26,101
	Sound	117.52 hrs	37,444
Test Set	Speech	6.38 hrs	2,311
	Music	9.32 hrs	3,367
	Sound	31.72 hrs	11,476

Table 6 Audio samples and duration in SAM Audio Judge DataSet

Paired data preparation. We collect a comprehensive set of datasets spanning music, speech, and sound effects. To mitigate the mismatch between real-world and simulated data, we use both real mixtures and synthetic mixtures as input audio. Table 6 shows the detailed data statistics. For each data source, we adopt either the original sound annotations (e.g., dog barking, man speaking, etc.) or the sound type predictions generated by PLM-Audio as text prompts, which is described in Section 4.2.1. We adopt the same text prompt settings

as the SAM Audio models, including speech extraction, speaker extraction, music extraction, instrument separation, and general sound event extraction. Following (Hai et al., 2024; Wang et al., 2025), the text prompts are designed as single-sound descriptions, which may refer to either a specific sound source (e.g., dog barking, guitar) or a general sound category encompassing multiple sources (e.g., dog, music playing). For each modality, we gather outputs from various publicly available audio separation models as well as from our intermediate SAM Audio checkpoints, which are listed in Table 7. To establish a unified SAJ score that is calibrated across different audio modalities, we shuffle audio samples from all modalities during annotation. We also apply loudness normalization to eliminate potential confounding effects introduced by variations in audio volume. Finally, we collect three independent ratings per audio sample to reduce variance and improve reliability.

Modality	Audio Separation Model
Speech	SAM Audio, MossFormer2, Tiger, FastGeCo
Music	SAM Audio, AudioSep, FlowSep, ClapSep, SoloAudio, Demucs, Spleeter
Sound	SAM Audio, AudioSep, FlowSep, ClapSep, SoloAudio

Table 7 Model outputs used in SAM Audio Judge DataSet

5.2.3 SAJ Model Training

Our main SAJ model is designed to predict separation model performance, thus we only use the annotations along the performance dimension. As shown in Figure 6, the SAJ model takes three inputs: the input audio, the output audio, and a text prompt. We use a pretrained audio encoder to extract audio features and a text encoder to extract text features. Both encoders are adopted from PE-AV (Vyas et al., 2025), which is trained through large-scale video-audio–text contrastive learning.

The text features are temporally aligned to match the length of the audio features and then fed into a Transformer to extract joint multimodal representations. Several linear layers are subsequently applied to predict the SAJ scores, including recall, precision, faithfulness, overall, and others.

In addition, we found that introducing a proxy task that predicts whether the output audio follows the text prompt (Wang et al., 2022b,a), significantly improves model performance. To this end, we pretrain the entire model on this text–audio alignment detection task using a large-scale simulated dataset that provides access to separated tracks within mixture audio. We alternate the output audio between the target sound and a random non-target sound from the same mixture to represent the presence or absence of the target sound. An additional linear layer is used to predict the presence or absence of the target sound described by the text prompt. After this pre-training stage, we finetune the SAJ model to predict the final SAJ scores.

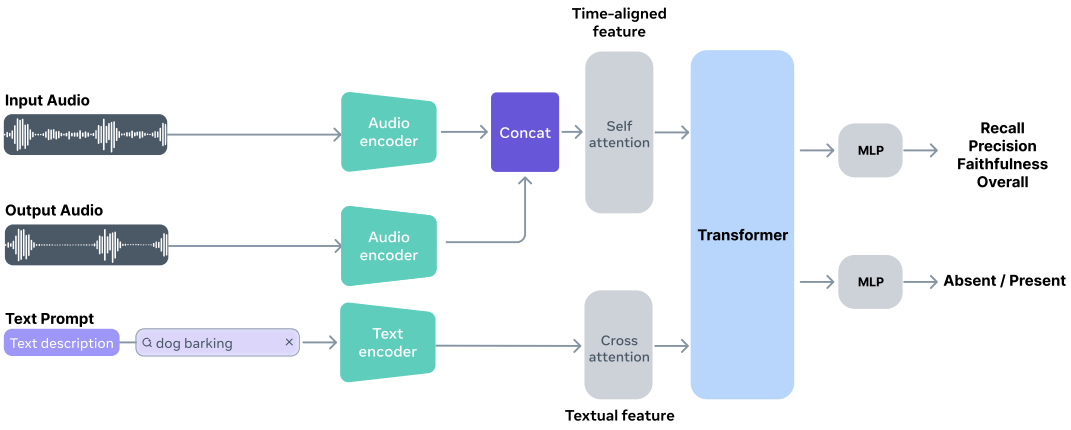

Figure 6 Diagram of SAM Audio Judge Model

5.3 Subjective Evaluation Protocol

Given the limitations of objective metrics on in-the-wild content and the absence of reliable, general-purpose reference-free measures for separation quality, our benchmark adopts human evaluation as the primary method. We employ a **side-by-side Absolute Category Rating (ACR)** protocol with an always-on preference tie-breaker, a hybrid that yields both absolute quality signals and robust relative comparisons, while mitigating common evaluator biases.

Briefly, annotators are shown the source audio (and video, when applicable), the user prompt, and the extracted outputs from two models, and are asked to judge how well each output reflects the requested target sounds. The protocol begins by verifying whether the target sounds actually occur in the source audio, then assesses how much of the target content is present in the extracted audio, whether any portions are missing, and how similar the extracted target sounds are to their originals. Annotators also evaluate the presence and degree of non-target sounds, including whether they originate from the source audio or are artifacts introduced by the model. After answering these structured questions, raters assign a 1–5 Overall score reflecting fidelity to the prompt and acoustic faithfulness, followed—when model scores tie—by a forced-choice preference between the two outputs. The procedure applies consistently across text-, visual-, and span-prompted conditions.

Empirically, we find that this side-by-side ACR framework with an always-on preference tie-breaker offers clear advantages over alternative protocols. The tie-breaker improves inter-annotator agreement and yields sharper, more discriminative ACR deltas that align with expressed preferences. Side-by-side presentation also reduces uncertainty in score differences—narrowing confidence intervals by up to 20% compared to single-stimulus rating, which translates to roughly 30% cost savings for equivalent A/B sensitivity. Although handling time per item increases relative to pairwise-only preference evaluations, the protocol produces both absolute and relative judgments in a single pass. Finally, consistent with anchoring effects reported in prior work, absolute ACR scores remain context-dependent.

For each system pair, the subjective evaluation yields both an absolute score (OVR) and a net win rate (NWR) for system A vs. B across four dimensions: overall quality, coverage, correctness, and faithfulness. For our main result, we primarily report the overall OVR and NWR.

Full details on the subjective evaluation protocol appear in Appendix A.1 and A.2.

5.4 Objective Metrics

Text prompting. For text-prompted separation, we evaluate model performance using the SAJ and CLAP (Wu et al., 2023b) scores[1]. SAJ measures separation quality across multiple dimensions and we report its overall score for simplicity. CLAP evaluates the semantic alignment between the separated audio and the input text prompt, serving as a proxy for perceptual correctness.

Span prompting. For span-prompted separation, we reuse the ground-truth text description associated with the target region, enabling the use of the same SAJ and CLAP socres. In addition, we introduce the **SpanIoU** metric, which quantifies temporal alignment by measuring the intersection-over-union (IoU) between the predicted and reference spans. The predicted span is obtained by applying a simple voice activity detection (VAD) on the separated waveform, using the silence detection module in the `pydub` library (pydub).

Visual prompting. For visual-prompted separation, text-based metrics such as CLAP are less effective because textual descriptions are often insufficiently discriminative for visually localized regions. Instead, we measure audio-visual consistency using the ImageBind (Girdhar et al., 2023) score, which captures the alignment between the separated audio and the corresponding visual mask region.

6 Experimental Setup

6.1 Training and inference configurations

We train three SAM Audio variants with parameter budgets of 500M, 1B, and 3B (Table 8). Parameter counts exclude the PE visual encoder, the T5 text encoder, and the DAC-VAE audio codec. Auxiliary

[1]We use the checkpoint: `https://huggingface.co/lukewys/laion_clap/blob/main/630k-best.pt`.

alignment losses are injected at the 2nd, 4th, and 6th Transformer blocks for 500M, 1B and 3B models, respectively.

All models are optimized following the two-stage recipe of Polyak et al. (2024): a large-scale *pre-training* stage, followed by a *fine-tuning* stage on curated high-quality data. The model is pre-trained on synthetic audio mixtures from general video data in Table 2. The remaining high-quality data, along with all pseudo-labeled data, is used exclusively during fine-tuning. Because many training audios exceed typical context lengths, all clips are capped at 30 seconds and randomly chunked when longer. Training uses fully-sharded data parallelism to fit the model size.

Model	Total params	Layers	Attn dim	FFN dim
SAM Audio-Small	500M	12	1,536	6,144
SAM Audio-Base	1B	16	2,048	8,192
SAM Audio-Large	3B	22	2,816	11,264

Table 8 SAM Audio model configurations.

Pre-training. We use an effective batch size of 1,024 sequences, each truncated or padded to 30 seconds (= 750 audio tokens). Models are trained for 500K updates with a constant learning rate of 1×10^{-4}, preceded by a 5K-step linear warmup. AdamW is used throughout with weight decay 0.1 and `bf16` precision. During pre-training, the auxiliary alignment loss is enabled with weight 1. Additionally, we apply conditioning dropout: the audio mixture, video, and text prompt are each independently dropped with probability 0.3.

Fine-tuning. Since fine-tuning data exhibit higher length variability, we adopt *variable-length batching* with a per-batch token budget of {96K, 120K, 144K} tokens for the {500M, 1B, 3B} models, respectively. Fine-tuning runs for 300K steps with a 5K warmup to a peak learning rate of 1×10^{-4}, kept constant thereafter. An exponential-moving-average (EMA) checkpoint (decay 0.999) is maintained and used for inference. The auxiliary loss is disabled during fine-tuning (weight 0), as we do not observe gains when training on clean outputs. During fine-tuning, we also disable conditioning dropout.

Inference. We use a 16-step midpoint ODE solver without classifier-free guidance, as CFG yielded no improvement in our setting. We additionally apply candidate re-ranking with beam size 8: for text prompting, a linear combination of SAM Audio Judge and CLAP scores (weights 1 and 5); for span prompting, span IoU; and for visual prompting, the ImageBind similarity between audio and masked region. By default, we enable span prediction for text-only separation, as it brings overall gains (see Section 7.4). Specifically, we employ the PE-A frame model (Vyas et al., 2025) to estimate the temporal spans corresponding to the target sound given the text prompt. The predicted spans are then combined with the original text prompt and used as conditioning for SAM Audio. We set the frame probability threshold to 0.3, following the threshold setting in Vyas et al. (2025).

As SAM Audio-Large shows the best performance among the three variants, we adopt SAM Audio-Large for all key comparisons and analyses, denoting it simply as SAM Audio in the remainder of the paper unless otherwise specified.

6.2 Tasks

We benchmark SAM Audio across a diverse set of separation tasks that reflect common real-world use cases, as mentioned in Section 5. Each task is to evaluate the model's ability to leverage text, video, and span prompts individually or jointly.

Text-prompting tasks. We evaluate SAM Audio on a broad range of text-prompted source separation tasks:

1. **General sound event extraction:** Extract any target sound event described by a text query (e.g., "dog barking" or "glass shattering"). This task tests the model's ability to handle extraction of general audio events.

2. **Speech extraction:** Separate all speech from a noisy audio mixture. The number of speakers is unconstrained, and the goal is to recover the complete speech track regardless of speaker count or background noise.
3. **Speaker extraction:** Isolate speech from a specific speaker given a text prompt describing speaker attributes such as gender or age (e.g., "female speaking"). This task evaluates fine-grained speaker conditioning. We primarily focus on gender-based separation.
4. **Music extraction:** Separate music from mixed audio, including cases where music is accompanied by speech or sound effects.
5. **Instrument separation in the wild:** Extract a single instrument stem (e.g., "piano," "drums") from full music audio. This benchmark includes both clean studio recordings and noisy in-the-wild music mixtures.
6. **Professional instrument separation:** Separate stems from professionally recorded music (e.g., studio tracks). Unlike the general instrument separation benchmark, this focuses on high-quality, multi-track recordings and uses MUSDB18 (Rafii et al., 2017a) as the standard benchmark.

Visual-prompting tasks. Visual prompts consist of a pair of video masks and the raw video clip. These tasks evaluate the model's ability to perform instance-level source separation conditioned on visual information:

1. **General sound event separation:** Extract the sound associated with the highlighted region of interest (e.g., isolating the sound of car honking).
2. **Instrument separation:** In a music video, extract the audio of the instrument corresponding to the highlighted mask, regardless of whether the mixture is noisy.
3. **Speaker separation:** In a multi-speaker video, separate the speech of the highlighted speaker.

Span-prompting tasks. Span-prompting tasks mirror the text-prompting tasks. Instead of text descriptions, text model is conditioned on timespans that specify when the target source is active.

Within each modality, we filter out samples with ambiguous prompts that could map to multiple plausible sources (e.g., videos with multiple visually indistinguishable instruments or overlapping events that are not temporally resolvable). Each sample in our evaluation set are of $\sim 10s$, which is to facilitate subjective evaluation. As the original music in MUSDB span several minutes, we randomly extract 10s chunks of the original full-mix audio for professional instrument separation.

6.3 Baselines

We compare SAM Audio against a broad set of baselines across all prompting modalities. For a fair comparison, we evaluate each model on tasks it is designed to handle, using its native input format where available.

6.3.1 Text-prompting baselines

Most existing research on text-guided audio separation focuses on general audio events, where broad categories such as *speech* and *music* are treated as single classes. We select four representative and recent open-domain baselines for comparison: AudioSep (Li et al., 2023), FlowSep (Yuan et al., 2024), SoloAudio (Wang et al., 2025), and CLAPSep (Wu et al., 2023a). These models are designed to handle general audios and are trained on large-scale audio-text datasets. Although they are not specialized for individual domains, their pretraining data often include samples from speech and music, allowing a comparison to SAM Audio across general and specialized tasks.

Beyond these open-domain systems, we also include a range of specialized baselines targeting specific audio domains. Unlike general-purpose models, these systems do not support free-form text prompting; instead, they decompose an audio mixture into a fixed ontology of stems (e.g., vocals, drums, bass). For a fair comparison, we extract the separated stem that corresponds to the target event type and evaluate it against SAM Audio. The specialized baselines are outlined below.

Instrument separation We evaluate against Demucs (Défossez, 2023), Spleeter (Hennequin et al., 2020), AudioShake (Audioshake, 2025), MoisesAI (Moises.AI, 2025), LalalAI (Lalal.AI, 2025), and FADR (FADR, 2025). Demucs and Spleeter are publicly available models, while the others are proprietary systems accessed via public APIs. All baseline models are limited to a small set of supported stems (see Table 9). For the professional instrument separation benchmark, we adopt MUSDB (Rafii et al., 2017a), which only include vocals, drums and bass separation. In contrast, for instrument separation *in the wild*, we exclude Demucs and Spleeter since their supported stem vocabulary covers only a small fraction of required instruments. For the remaining baselines, we compare only on examples where the target instrument is within the supported list.

Model	Supported Instruments
Demucs (Défossez, 2023)	vocals, drums, bass
Spleeter (Hennequin et al., 2020)	vocals, drums, bass, piano
AudioShake (Audioshake, 2025)	vocals, drums, guitar, bass, wind, piano
MoisesAI (Moises.AI, 2025)	vocals, guitar, bass, drums, piano, wind, strings
LalalAI (Lalal.AI, 2025)	vocals, drums, bass, guitar, piano, synthesizer, strings, wind
FADR (FADR, 2025)	vocals, drums, piano, guitars, strings, wind
SAM Audio	open-vocabulary

Table 9 Supported instrument types for each baseline.

Speech separation Speech separation aims at removing background noise of a speech recording. We compare SAM Audio against LalaAI (Lalal.AI, 2025), ElevenLabs (ElevenLabs, 2025), Auphonic (Auphonic, 2025), and AudioShake (Audioshake, 2025). To ensure fair comparison, we disable post-processing or enhancement modules in the baselines, as our evaluation focuses on *separation fidelity*—the model's ability to isolate speech content—rather than perceptual enhancement or reverb suppression.

Music separation For music separation, we benchmark against MoisesAI (Moises.AI, 2025) and AudioShake (Audioshake, 2025), both of which can isolate or remove background music. Similar to speech separation, we disable any additional enhancement modules. We report results both for *music extraction* (isolating music) and *music removal* (removing music while preserving other sounds).

Speaker separation Speaker separation aims to isolate speech from specific individuals. Most specialized systems rely on fixed-stem decomposition and do not natively support prompting for arbitrary speakers. To enable comparison under the prompted setting, we select the separated stem with the highest CLAP similarity score (Wu et al., 2023b). We evaluate public models such as Mossformer2 (Zhao et al., 2024), Tiger (Xu et al., 2024), and FastGeCo (Wang et al., 2024), as well as the proprietary AudioShake model (Audioshake, 2025).

6.3.2 Visual-prompting baselines

Compared to text-guided separation, visual-prompted audio separation is much less explored. No commercial models are available to our knowledge, thus we focus on public research models. We evaluate SAM Audio against two general-purpose visual separation models, DAVIS-Flow (Huang et al., 2025) and CLIPSep (Dong et al., 2023), across general sound, speaker, and instrument separation tasks. In addition, we include the instrument-specific checkpoint[2] of DAVIS-Flow (denoted as *DAVIS-Flow(Music)*) for the visual-prompted instrument separation benchmark.

Among visual separation tasks, speaker separation has been studied more extensively (Wu et al., 2019; Pan et al., 2025; Li et al., 2024a,b). We therefore compare to two strong baselines, IIANet (Li et al., 2024b) and AV-Mossformer2 (Zhao et al., 2025), which achieve state-of-the-art results on a recent visual-prompted speech separation benchmark (ClearVoice, 2025). These models rely on a preprocessing pipeline involving face detection and lip-motion extraction; we follow their official preprocessing setup and feed masked videos accordingly. In practice, about 20% of our evaluation video fail in AV-Mossformer2 preprocessing stage, and these samples are excluded from comparison with SAM Audio.

[2]We use the DAVIS-Flow checkpoints trained on AVE and MUSIC respectively for general and instrument-specific separation.

6.3.3 Span-prompting baselines

To our knowledge, there are no public separation models supporting span prompting. Therefore we use our text-prompting model as a baseline, comparing span-only and joint text+span conditioning against text-only separation to quantify the benefits of span prompting.

7 SAM Audio Results

In this section, we present the main results of SAM Audio, covering text, visual, and span prompting, sound removal, model latency, and long-form separation. Detailed ablation studies are deferred to Appendix D.

7.1 Text-prompted separation

			General SFX			Speech			Speaker			Music			Instr(wild)			Instr(pro)		
Model	**OSS**	**Promptable**	SAJ	CLAP	OVR	SAJ	CLAP	OVR	SAJ	CLAP	OVR	SAJ	CLAP	OVR	SAJ	CLAP	OVR	SAJ	CLAP	OVR
MossFormer2 (Zhao et al., 2024)	✓	✗	-	-	-	-	-	-	2.43	0.14	2.54	-	-	-	-	-	-	-	-	-
Tiger (Xu et al., 2024)	✓	✗	-	-	-	-	-	-	2.47	0.15	2.50	-	-	-	-	-	-	-	-	-
Fast-GeCo (Wang et al., 2024)	✓	✗	-	-	-	-	-	-	2.66	0.16	2.71	-	-	-	-	-	-	-	-	-
Demucs (Défossez, 2023)	✓	✗	-	-	-	-	-	-	-	-	-	-	-	-	-	-		4.48	0.15	4.26
Spleeter (Hennequin et al., 2020)	✓	✗	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	4.26	0.11	3.90
FlowSep (Serrà et al., 2024)	✓	✓	2.36	0.21	2.65	2.18	0.20	2.14	1.85	0.09	2.13	2.73	0.18	2.90	2.37	0.10	2.69	2.13	-0.01	2.02
AudioSep (Li et al., 2023)	✓	✓	2.63	0.25	2.88	2.93	0.28	2.85	2.50	0.17	2.79	3.47	0.27	3.51	2.16	0.13	2.59	2.34	0.04	2.45
CLAPSep (Ma et al., 2024)	✓	✓	2.68	0.23	2.92	2.30	0.22	2.47	2.80	0.17	2.79	2.48	0.04	2.97	2.47	0.14	2.81	2.48	0.04	2.56
SoloAudio (Wang et al., 2025)	✓	✓	3.29	0.25	2.97	3.45	0.30	3.32	2.26	0.19	2.45	2.68	0.21	2.47	2.92	0.13	2.71	2.65	0.01	2.30
AudioShake (Audioshake, 2025)	✗	✗	-	-	-	3.90	0.28	3.95	3.28	0.14	3.51	3.22	0.29	3.37	3.37	0.29	3.43	3.87	0.29	4.28
MoisesAI (Moises.AI, 2025)	✗	✗	-	-		-	-		-	-	-	3.79	0.27	3.90	3.03	0.29	3.12	3.78	0.28	4.22
FADR (FADR, 2025)	✗	✗	-	-	-	-	-	-	-	-	-	-	-	-	2.44	0.19	2.45	3.63	0.25	3.92
LalalAI (Lalal.AI, 2025)	✗	✗	-	-	-	3.77	0.33	3.92	-	-	-	-	-	-	3.07	0.25	3.03	3.83	0.27	4.18
Auphonic (Auphonic, 2025)	✗	✗	-	-	-	4.32	0.27	4.08	-	-	-	-	-	-	-	-	-	-	-	-
ElevenLabs (ElevenLabs, 2025)	✗	✗	-	-	-	3.79	0.25	3.72	-	-	-	-	-	-	-	-	-	-	-	-
SAM Audio	✓	✓	**4.35**	**0.31**	**3.59**	**4.67**	**0.35**	**4.29**	**4.51**	**0.18**	**4.15**	**4.45**	**0.26**	**4.05**	**4.32**	**0.31**	**4.00**	**4.82**	**0.28**	**4.45**

Table 10 Comparison against text-prompted baselines. –: not applicable. OVR: overall subjective score.

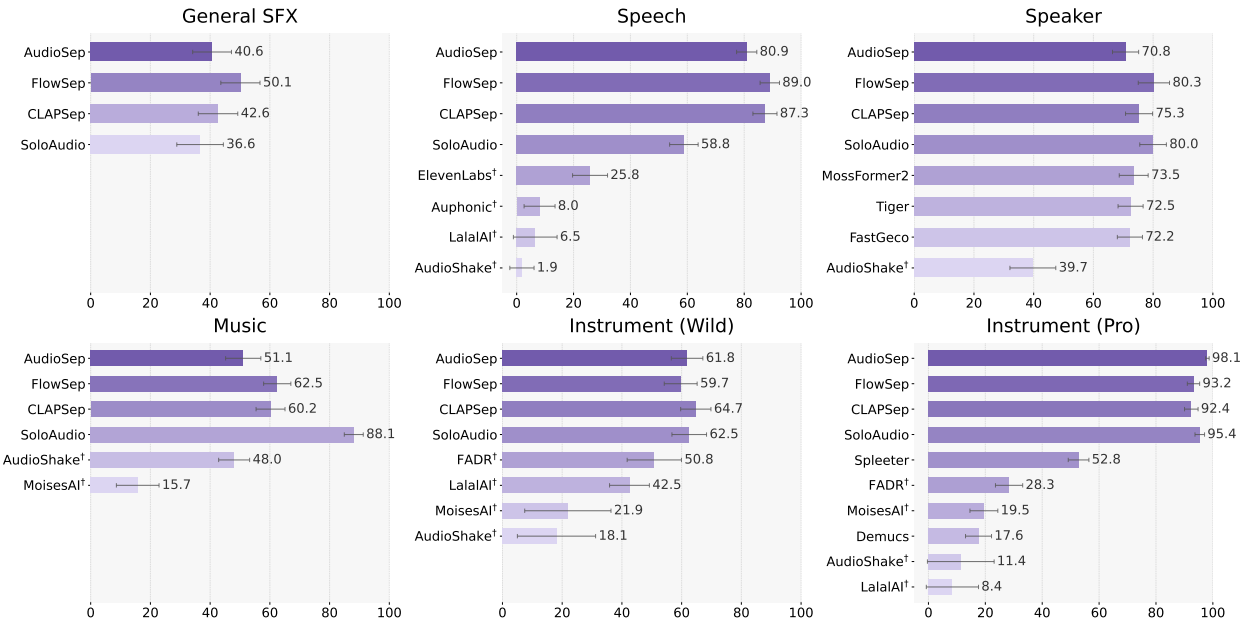

Figure 7 Net Win Rate (%) of SAM Audio against SoTA separation models in text-prompted tasks. †: proprietary models

Table 10 presents quantitative comparisons between SAM Audio and a range of public text-prompted separation models. The baselines fall into two broad categories: general models that aim to handle a wide variety of separation tasks (such as AudioSep, FlowSep, SoloAudio, and CLAPSep), and specialized models

optimized for specific domains like speech or music (e.g., MossFormer2, Demucs, Spleeter). We further show the net win rate against the baselines in Figure 7. In Table 10, we additionally show the overall subjective score (i.e., OVR) aside from the objective metrics. As our evaluation protocol follows a pairwise comparison setup, the final OVR score is obtained by averaging the overall preference scores across all pairwise comparisons involving SAM Audio.

Overall, SAM Audio consistently outperforms all prior models by a substantial margin across nearly all categories. In general sound event separation, SAM Audio achieves roughly a $\sim 36\%$ net win rate over one of the best public general sound separation model (SoloAudio (Wang et al., 2025)). In specialized domains such as instrument or speaker separation, we observe that general-purpose models like FlowSep (Serrà et al., 2024) or AudioSep (Li et al., 2023) fail to reach competitive quality. Their mean judge scores remain below 3, significantly trailing specialized systems such as Demucs (Défossez, 2023). Yet even in such domains, SAM Audio surpasses specialized models (e.g., NWR of SAM Audio vs. Demucs = 17.6%). While proprietary systems generally outperform the OSS counterparts, SAM Audio remains superior in most settings. On instrument separation for professional audios (MUSDB (Rafii et al., 2017a)), SAM Audio achieves an overall subjective score of 4.45 compared to 4.28 from AudioShake (Audioshake, 2025) (a relative gain of $\sim 4\%$), and in speaker separation, it achieves 4.15 versus 3.51, corresponding to a net win rate improvement of $\sim 39\%$. Through unified training, SAM Audio generalize across domains and achieves SoTA performance.

7.2 Visual-prompted separation

		General SFX		Speaker		Instr (wild)	
Model	**Generic**	IB	OVR	IB	OVR	IB	OVR
AV-MossFormer2 (Zhao et al., 2025)	✗	-	-	0.20	2.62	-	-
IIANet (Li et al., 2024b)	✗	-	-	0.16	2.41	-	-
ClipSep (Dong et al., 2023)	✓	0.16	1.53	0.14	1.47	0.15	1.12
DAVIS-Flow (Huang et al., 2025)	✓	0.14	1.96	0.13	1.97	0.13	2.08
DAVIS-Flow (Music) (Huang et al., 2025)	✗	-	-	-	-	0.13	2.40
SAM Audio	✓	**0.25**	**2.61**	**0.24**	**3.07**	**0.24**	**2.56**

Table 11 Comparison against visual-prompted baselines. –: Not applicable. OVR: overall subjective score.

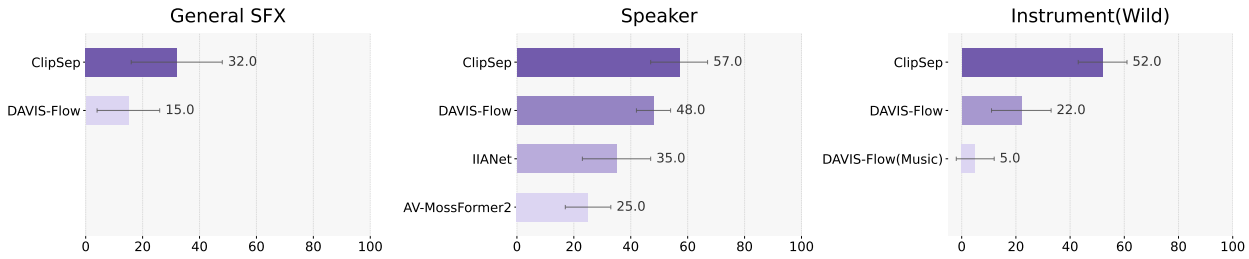

Figure 8 Net Win Rate (%) of SAM Audio against SoTA separation models in visual-prompted tasks

We show the comparison between SAM Audio and existing visual-prompted separation models in Table 11. Compared to text-prompted separation, there are substantially fewer general-purpose visual separation systems available publicly.

Across all settings, SAM Audio achieves stronger improvements over prior work. On average, SAM Audio outperforms DAVIS by a large margin, achieving net win rates ranging from 5% to 48% depending on the separation task. Similar to the trends observed in text-prompted separation, we find that general visual models struggle on specialized domains such as *instrument* and *speaker* separation. In contrast, SAM Audio maintains and surpasses the best specialized baselines by approximately 25% on speaker separation and 5% on instrument separation.

Additionally, we notice that visual prompting yields notably lower overall subjective scores compared to text prompting. The advantage of text prompt stems primarily from the availability of a much larger pool of high-quality text–based training data, whereas video-based supervision is generally noisier due to errors in

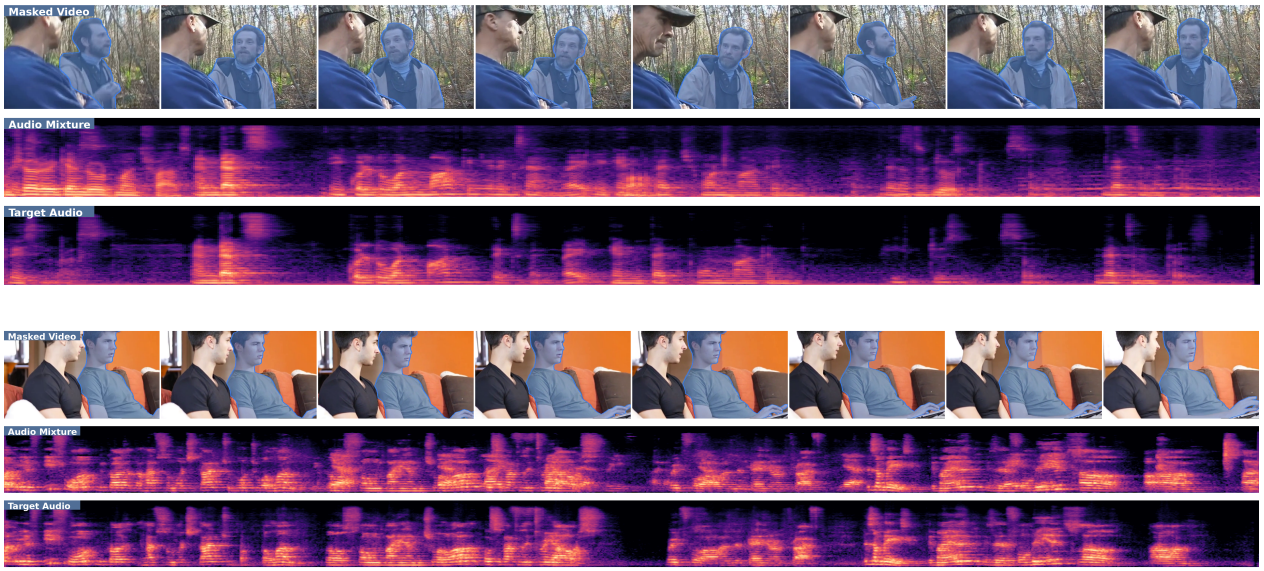

Figure 9 Visual-prompted samples where text descriptions are ambiguous. Each example shows (**top**) masked video, (**middle**) input mixture spectrogram, and (**bottom**) separated target spectrogram. The target speaker is highlighted in each video.

visual masks and the frequent presence of off-screen sounds in training. Beyond scale, text also tends to provide more specific cues about the target source. For instance, a prompt such as *"man shuffling"* directly points to a unique sound event, whereas a visual mask of a person is inherently ambiguous since it can be associated with multiple sounds.

Nevertheless, visual prompting plays a complementary role in scenarios where text alone is insufficient. A common example arises in conversational scenes where multiple people of the same gender are speaking. A text prompt such as *"male speech"* cannot disambiguate between the two male speakers, but visual masks can localize the target speaker and enables separation effectively. In such cases, visual cues provide instance-level grounding that is difficult to achieve with text alone (see Figure 9).

Prompt Modality	General			Speech			Speaker			Music			Instr(wild)		
	SAJ	CLAP	OVR	SAJ	CLAP	OVR	SAJ	CLAP	OVR	SAJ	CLAP	OVR	SAJ	CLAP	OVR
text	4.11	0.31	3.32	4.59	0.33	4.18	4.08	0.17	3.63	4.30	0.28	4.09	**4.45**	0.30	3.78
span	3.27	0.30	3.54	3.37	0.28	3.65	3.26	0.26	4.08	2.18	-0.04	2.57	1.81	0.01	2.20
text + span	**4.25**	**0.31**	**4.04**	**4.66**	**0.35**	**4.33**	**4.51**	**0.18**	**4.22**	**4.38**	**0.27**	**4.19**	4.42	**0.32**	**3.88**

Table 12 Comparison between span prompting and text prompting for SAM Audio. OVR: overall subjective score. For all the metrics below, higher is better.

7.3 Span-prompted separation

Table 12 and Figure 11 compare models conditioned on text, span, and text+span inputs. To better ablate the effect of ground-truth span, no predicted span is used in text prompting. Using span prompts alone does not consistently improve performance, as the target and distractor sounds may co-occur throughout the same temporal regions. This effect is particularly evident for long-duration or ambient sounds such as speech and music, where span-only models exhibit large performance degradations (NWR of -16% to -49.6% relative to text-only baselines). In contrast, for short-duration and well-localized sounds, where temporal cues are more discriminative, span-only conditioning yields noticeable gains (NWR $+26\%$ over text-only in sound).

Despite these fluctuations, combining text and span inputs consistently improves performance across all domains, achieving NWR between $+12.9\%$ and $+39.0\%$. These results show that temporal localization from span prompts complements the semantic information in text, using both would enable more precise separation.

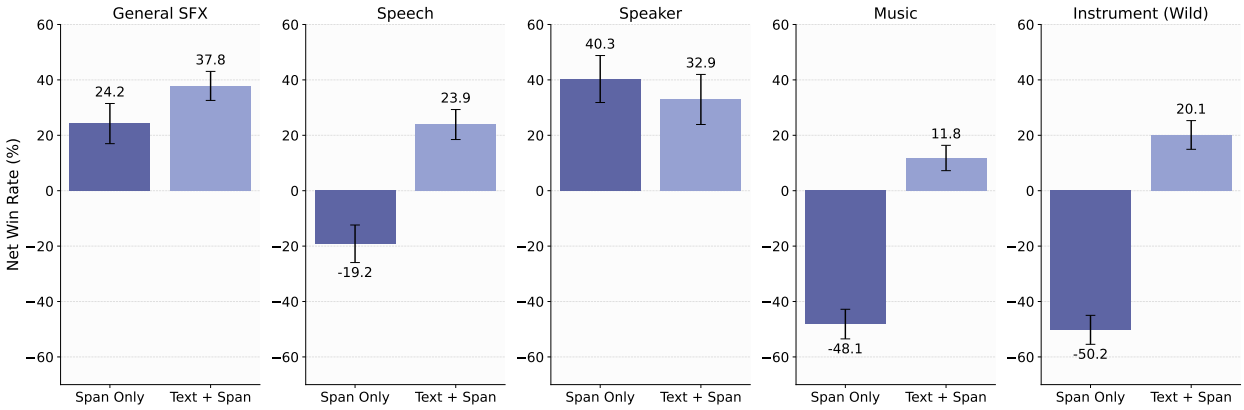

Figure 10 Net Win Rate (%) of SAM Audio with text & span / span as input against a text-only model

7.4 Span prediction boosts text-prompted separation

Leveraging predicted spans as additional input is our default choice for text prompting. Here, we compare using vs. not using span prediction across all separation tasks. The comparison is shown in Table 13. Note that the OVR scores for *text + predicted span* differ from those in Table 10, as the absolute ratings depend on the specific baseline used in each pairwise evaluation under our human evaluation protocol.

Incorporating predicted spans boosts performance in majority of the domains, including general SFX, speech, speaker, and music, where temporal cues play a critical role in disambiguating target sounds. We noticed a minor degradation in professional instrument separation, likely because many MUSDB instrument stems span the entire segment, leaving limited room for temporal cues to provide additional benefit. There is only a small gap between using predicted spans and ground-truth spans (see Table 12), which shows the robustness of SAM Audio to temporal inaccuracies in span estimation. Importantly, span prediction improves separation quality without costly human annotations, allowing SAM Audio to separate precisely at scale.

Task	w/ Pred Span	SAJ	CLAP	OVR
General SFX	✗	4.11	**0.31**	3.36
	✓	**4.35**	**0.31**	**3.89**
Speech	✗	4.59	0.33	4.17
	✓	**4.67**	**0.35**	**4.22**
Speaker	✗	4.08	0.17	3.62
	✓	**4.51**	**0.18**	**4.01**
Music	✗	4.30	**0.28**	**4.16**
	✓	**4.45**	0.26	4.12
Instr(wild)	✗	**4.45**	0.30	3.70
	✓	4.32	**0.31**	**3.88**
Instr(pro)	✗	**4.83**	**0.28**	**4.16**
	✓	4.82	**0.28**	4.12

Table 13 Using vs. not using predicted span for text-prompting. OVR: overall subjective score. For all the metrics below, higher is better.

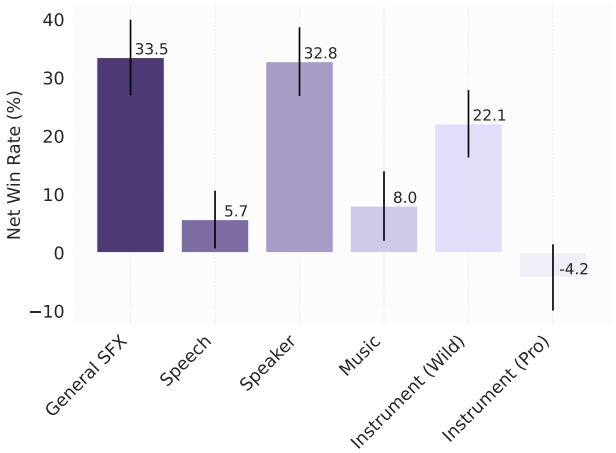

Figure 11 Net Win Rate (%) of SAM Audio using predicted span against not using predicted span for text prompting.

7.5 Sound Removal

SAM Audio outputs both a *target* and a *residual* audio. While earlier sections focus on evaluating target quality, we now evaluate the residual, corresponding to the *removal* task—removing the prompt-specified sound from the mixture. We use text-prompted music removal as a representative removal task.

Among existing baselines, only MoisesAI (Moises.AI, 2025) and AudioShake (Audioshake, 2025) support explicit sound removal. Therefore, we compare SAM Audio against these two systems. As shown in Table 14 and Figure 12, SAM Audio outperforms both systems. The trends mirror the music extraction results in Figure 7, suggesting the high correlation of extraction and removal modes. The high OVR score by SAM Audio further shows the model's ability to cleanly suppress target sources.

Model	OVR
AudioShake (Audioshake, 2025)	3.75
MoisesAI (Moises.AI, 2025)	4.00
SAM Audio	**4.05**

Table 14 Comparison against baselines for music removal

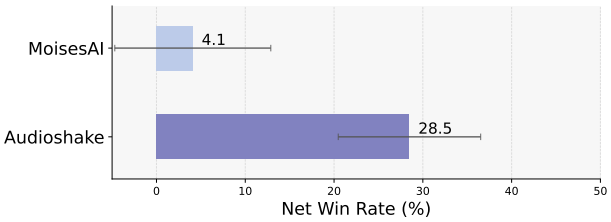

Figure 12 Net Win Rate (%) of SAM Audio over baselines in music removal

7.6 Latency

Our default inference configuration uses 16 ODE steps with the midpoint solver. For text prompting, each separation on SAM Audio-Large takes approximately 7.3 s for a 10-second input on one A100 GPU, including 6.5 s for model forward computation, 0.1 s for span prompting, and 0.5 s for judge reranking.

We further study the trade-off between inference cost and output quality by varying the number of ODE steps. As shown in Figure 13, increasing ODE steps generally improves performance across tasks as expected. Nonetheless, the model achieves surprisingly competitive results even with as few as two ODE steps (e.g., speech separation). Qualitatively, we find a larger performance gap between 16 and 2 ODE steps for speaker or instrument separation. In contrast, for general sound effects that are often short and sparse, lower NFEs still yield acceptable perceptual quality. We hypothesize that the input audio mixture provides strong conditioning signal, enabling separation with fewer refinement steps compared to fully generative tasks such as TTS. Overall, fewer NFEs offer a favorable trade-off between speed and quality for many practical separation scenarios for SAM Audio.

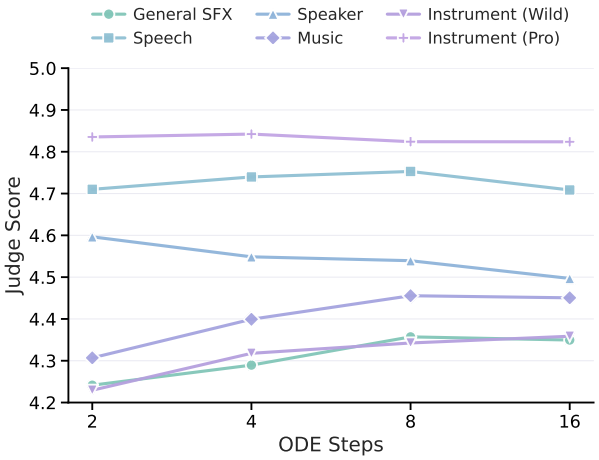

Figure 13 Effect of varying ODE steps under the midpoint solver. Fewer steps reduce computation at a modest cost in quality.

7.7 Long-form Audio Separation

Most samples in our evaluation set are around 10 seconds. To further assess the performance of SAM Audio on longer inputs, we evaluate its performance on long-form separation using the multi-diffusion approach described in Section 3. Specifically, we adopt a 20-second window with a 5-second context overlap. For comparison, we consider two baselines: (a) **chunk-wise separation**, where the audio is divided into 20-second segments that are processed independently and stitched

Method	SAJ	CLAP
One-shot	3.48	0.26
Chunk-wise	3.57	0.24
Multi-diffusion	**3.67**	**0.27**

Table 15 Comparison of methods for long-form separation.

together, and (b) **one-shot separation**, where the entire audio is processed in a single forward pass.

We curate an internal test set of 50 1-minute audio samples, to evaluate long-horizon separation quality. As shown in Table 15, the one-shot model exhibits a noticeable degradation on long recordings, which aligns with expectations given that most training samples are shorter than 30 seconds. The chunk-wise baseline mitigates this issue but introduces audible discontinuities at segment boundaries. In contrast, the proposed multi-diffusion strategy maintains high perceptual quality across the full sequence and achieves the best judge scores.

7.8 SAM Audio Judge Results

7.8.1 Subjective score correlation

We compare the proposed SAM Audio Judge model with several representative baseline systems covering diverse evaluation paradigms:

- CLAP (Wu et al., 2023c): A large-scale contrastive audio–text model trained on millions of audio–caption pairs. We use its cosine similarity between the text prompt and output audio embeddings as a proxy for perceptual alignment. This represents the current standard for audio–language correspondence evaluation.
- Audiobox Aesthetics Production Complexity (PC) (Tjandra et al., 2025): A model originally trained to predict subjective aesthetic preference scores for audio samples. It focuses on the complexity of audio scene, measured by the number of audio components.
- SDR Estimator (Dang et al., 2023): A regression model trained to estimate SDR without access to ground-truth references. It reflects the performance of traditional distortion-based metrics that focus on signal fidelity rather than perceptual judgment. We trained the SDR estimator using the same architecture as the SAJ model, except that its training target was the SDR value. We used a balanced training dataset covering target audio levels from -25 dB to 25 dB, totaling 496 hours across speech, music, and sound effects. The SDR estimator achieved PPC scores of 0.923, 0.681, and 0.665 on speech, music, and sound effects, respectively.
- Gemini-2.5-pro (Comanici et al., 2025): A large multimodal LLM capable of reasoning over both text and audio inputs. We prompt it to rate the separated audio quality according to the same evaluation axes used in SAJ, representing a language-model-based perceptual evaluation baseline. We also provide several examples with different scores to enable few-shot learning. Our prompts could be found in Appendix C.3.

These baselines cover the main paradigms of signal processing-based separation measures and general-purpose multimodal LLM-based metrics, allowing for a comprehensive comparison with our task-specific SAJ model. We report Pearson (PCC) and Spearman (SRCC) correlation between automatic metrics and human ratings.

	Speech				Music				Sound			
Model	Overall	Recall	Precision	Faithfulness	Overall	Recall	Precision	Faithfulness	Overall	Recall	Precision	Faithfulness
Pearson Correlation Coefficient (PCC)												
CLAP	0.490	0.431	0.283	0.477	0.487	0.416	0.385	0.432	0.367	0.431	0.283	0.418
Audiobox Aesthetics PC	-0.410	0.080	-0.543	0.013	-0.092	0.227	-0.325	0.182	-0.278	0.072	-0.493	0.030
SDR Estimator	0.336	0.004	0.403	0.055	0.369	0.157	0.388	0.182	0.181	0.040	0.222	0.055
Gemini-2.5-pro	0.487	0.498	0.169	0.430	0.351	0.287	0.115	0.303	0.462	0.493	0.192	0.369
SAM Audio Judge	**0.883**	**0.943**	**0.841**	**0.891**	**0.815**	**0.858**	**0.766**	**0.791**	**0.815**	**0.837**	**0.775**	**0.818**
Spearman Rank Correlation Coefficient (SRCC)												
CLAP	0.380	0.291	0.325	0.273	0.285	0.293	0.199	0.296	0.493	0.376	0.388	0.406
Audiobox Aesthetics PC	-0.461	0.042	-0.566	-0.076	-0.045	0.218	-0.267	0.155	-0.260	0.097	-0.516	0.022
SDR Estimator	0.338	0.000	0.395	0.079	0.390	0.203	0.375	0.210	0.173	0.053	0.220	0.073
Gemini-2.5-pro	0.495	0.361	-0.015	0.117	0.338	0.232	0.010	0.008	0.390	0.324	-0.006	0.180
SAM Audio Judge	**0.817**	**0.573**	**0.774**	**0.573**	**0.714**	**0.569**	**0.658**	**0.476**	**0.781**	**0.660**	**0.734**	**0.607**

Table 16 Comparison Between SAM Audio Judge Model and Baselines

As shown in Table 16, the proposed SAM Audio Judge model consistently outperforms all baselines across the

three modalities, *i.e.* speech, music, and sound effects, under both PCC and SRCC. SAJ achieves markedly higher correlations with human ratings, reaching PCCs of 0.883, 0.815, and 0.815 for speech, music, and sound, respectively, and SRCCs of 0.817, 0.714, and 0.781. In contrast, baseline models such as CLAP and Gemini-2.5-pro show moderate correlations, while distortion- or aesthetic-based metrics like SDR Estimator and Audiobox Aesthetics PC fail to capture perceptual quality, often yielding low or even negative correlations. The consistent performance of SAJ across modalities highlights its ability to generalize beyond speech to more complex acoustic domains such as music and environmental sounds. These results confirm that the proposed SAJ model effectively captures human perceptual judgment by leveraging joint audio–text representations and text-conditioned pretraining, offering a more reliable and fine-grained evaluation framework than existing baselines. Appendix C.4 presents the automatic fine-grained performance analysis enabled by SAJ.

Reranker A	Reranker B	Speech	Speaker	Music	Instr(wild)	General
SAJ	CLAP	0.17	0.19	0.09	0.01	0.15
CLAP w/ SAJ	CLAP	0.18	0.18	0.14	0.03	0.06
CLAP w/ SAJ	SAJ	0.00	0.02	0.10	0.15	0.07

Table 17 Net Win Rate Comparison Between Rerankers (Reranker A vs. Reranker B)

7.8.2 Judge as a reranker

To evaluate the impact of different rerankers, we apply them to the SAM Audio model, which generates eight candidate separation outputs for each input mixture. Each reranker is responsible for selecting the best candidate according to its scoring mechanism, and the selected outputs are then evaluated on the test set (see Table 6). We compare three configurations: Judge, CLAP, and their combination (CLAP w/ Judge). The combined reranker computes a linear combination of the two scores with a weighting ratio of 5:1 (CLAP : Judge).

Table 17 presents a comparison of NWR between different reranking configurations across five separation tasks: speech, speaker, music, instrument, and general sound separation. Overall, we observe that integrating the SAM Audio Judge model as a reranker consistently improves or stabilizes performance relative to using CLAP alone.

When comparing Judge vs. CLAP, the Judge reranker achieves higher NWR in most categories (e.g., 0.17 vs. 0.19 in speech and 0.15 in sound separation), indicating its stronger ability to align selection decisions with perceptual quality. After combining both signals (CLAP w/ Judge vs. CLAP), the hybrid reranker further improves NWR on music (0.14) and instrument (0.03) separation, suggesting that CLAP score can be complementary to SAJ. Finally, when CLAP w/ Judge is evaluated against Judge directly, the NWR drops generally compared to its evaluation against CLAP (e.g., 0.00 in speech, 0.02 in speaker, and 0.10–0.15 in music/instrument), showing that SAJ score already captures most of the reranking signal.

In summary, the results suggest that the Judge-based reranker is highly effective across audio domains. Combining judge and CLAP produces largest gains, particularly in speech and general sound separation, where reranking with CLAP alone tends to underperform.

8 Conclusion

We presented SAM Audio, a general-purpose audio separation model that supports multimodal prompting and achieves state-of-the-art performance across various audio separation tasks. SAM Audio advances universal audio separation by scaling both data and model capacity with flow matching. To mitigate the scarcity of ground-truth stems, we developed scalable data construction pipelines, including domain-aware synthetic mixing and a large pseudo-labeling process that bootstraps stems from natural recordings using intermediate SAM Audio checkpoints. These strategies provide broad coverage without requiring manual stem annotation. We further introduced visual and span prompting as complementary modalities to text prompting. Notably, span prompting substantially improves text-based separation and enables practical iterative refinement.

To support future research, we additionally release SAM Audio-Bench and SAM Audio-Judge. SAM Audio-Bench offers a carefully balanced benchmark with human-annotated text, visual, and span prompts. SAM Audio-Judge provides a reference-free metric with significantly higher correlation to human perception than existing alternatives, and can also be used as a post-processing module to improve separation quality.

Despite its strong performance, our results also reveal several limitations. In particular, visual prompting is noticeably less effective than text prompting, and general sound effects remains more challenging than specialized domains such as speech. Addressing these gaps will likely require stronger audio–visual grounding and better modeling of complex, multi-source acoustic scenes, which we leave for future work.

Acknowledgment

The authors would like to thank Peng-Jen Chen, Dangna Li, Robin San Roman, Leying Zhang, Carleigh Wood, Andrew Westbury, George Orlin, Anushka Sagar, Vivian Lee, Cynthia Gao, Ida Cheng, Roman Rädle, Victor Loh, Alex He, Dex Honsa, Eric Gan, Kei Koyama, Kevin Ngo, Meng Wang, Michelle Chan, Phillip Thomas, Andrew Huang, Robbie Adkins, Jason Holland, Josh Terry, Ben Samples, Karla Martucci, Bruno Nakano, Yoko Kristiansen, Ashley Gabriel, Athyu Eleti, Andy Bass, Helen Klein, Emma Leibman Baker, Britt Montalvo, Nikhila Ravi, and Manohar Paluri for their inspiring discussions and timely support throughout this work.

References

Triantafyllos Afouras, Joon Son Chung, and Andrew Zisserman. Lrs3-ted: A large-scale dataset for visual speech recognition. In *INTERSPEECH*, 2018.

Audioshake. Audioshake, 2025. https://www.audioshake.ai/.

Auphonic. Auphonic, 2025. https://auphonic.com.

Omer Bar-Tal, Lior Yariv, Yaron Lipman, and Tali Dekel. Multidiffusion: Fusing diffusion paths for controlled image generation. *arXiv preprint arXiv:2302.08113*, 2023.

Jon Barker, Shinji Watanabe, Emmanuel Vincent, and Jan Trmal. The fifth CHiME speech recognition challenge: Dataset, task and baselines. In *INTERSPEECH*, 2018.

John G Beerends, Christian Schmidmer, Jens Berger, Matthias Obermann, Raphael Ullmann, Joachim Pomy, and Michael Keyhl. Perceptual objective listening quality assessment (polqa), the third generation itu-t standard for end-to-end speech quality measurement part i—temporal alignment. *journal of the audio engineering society*, 61(6): 366–384, 2013.

Rachel Bittner, Justin Salamon, Mike Tierney, Matthias Mauch, Chris Cannam, and Juan P. Bello. Medleydb: A dataset for annotated musical audio. In *International Society for Music Information Retrieval Conference (ISMIR)*, 2014.

Daniel Bolya, Po-Yao Huang, Peize Sun, Jang Hyun Cho, Andrea Madotto, Chen Wei, Tengyu Ma, Jiale Zhi, Jathushan Rajasegaran, Hanoona Rasheed, Junke Wang, Marco Monteiro, Hu Xu, Shiyu Dong, Nikhila Ravi, Daniel Li, Piotr Dollár, and Christoph Feichtenhofer. Perception encoder: The best visual embeddings are not at the output of the network. *arXiv:2504.13181*, 2025.

Estefanía Cano, Derry FitzGerald, and Karlheinz Brandenburg. Evaluation of quality of sound source separation algorithms: Human perception vs quantitative metrics. In *2016 24th European Signal Processing Conference (EUSIPCO)*, pages 1758–1762. IEEE, 2016.

Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, Jie Lei, Tengyu Ma, Baishan Guo, Arpit Kalla, Markus Marks, Joseph Greer, Meng Wang, Peize Sun, Roman Rädle, Triantafyllos Afouras, Effrosyni Mavroudi, Katherine Xu, Tsung-Han Wu, Yu Zhou, Liliane Momeni, Rishi Hazra, Shuangrui Ding, Sagar Vaze, Francois Porcher, Feng Li, Siyuan Li, Aishwarya Kamath, Ho Kei Cheng, Piotr Dollár, Nikhila Ravi, Kate Saenko, Pengchuan Zhang, and Christoph Feichtenhofer. Sam 3: Segment anything with concepts, 2025. https://arxiv.org/abs/2511.16719.

Jean Carletta. The ami meeting corpus: A pre-announcement. In *Proceedings of the International Workshop on Machine Learning for Multimodal Interaction (MLMI 2005)*, 2005.

Mark Cartwright, Bryan Pardo, and Gautham J Mysore. Crowdsourced pairwise-comparison for source separation evaluation. In *2018 ieee international conference on acoustics, speech and signal processing (icassp)*, pages 606–610. IEEE, 2018.

B. Chen, C. Wu, and W. Zhao. Sepdiff: Speech separation based on denoising diffusion model. In *ICASSP*, 2023.

Relja Chen and Andrew Zisserman. VGGSound: A large-scale audio-visual dataset. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020.

Zili Chen, Shuyuan Chen, Shinji Watanabe, and Sanjeev Khudanpur. Libricss: A corpus and evaluation framework for conversational speech recognition. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020.

Michael Chinen, Felicia SC Lim, Jan Skoglund, Nikita Gureev, Feargus O'Gorman, and Andrew Hines. Visqol v3: An open source production ready objective speech and audio metric. In *2020 twelfth international conference on quality of multimedia experience (QoMEX)*, pages 1–6. IEEE, 2020.

Jang Hyun Cho, Andrea Madotto, Effrosyni Mavroudi, Triantafyllos Afouras, Tushar Nagarajan, Muhammad Maaz, Yale Song, Tengyu Ma, Shuming Hu, Hanoona Rasheed, Peize Sun, Po-Yao Huang, Daniel Bolya, Suyog Jain, Miguel Martin, Huiyu Wang, Nikhila Ravi, Shashank Jain, Temmy Stark, Shane Moon, Babak Damavandi, Vivian Lee, Andrew Westbury, Salman Khan, Philipp Krähenbühl, Piotr Dollár, Lorenzo Torresani, Kristen Grauman, and Christoph Feichtenhofer. Perceptionlm: Open-access data and models for detailed visual understanding. *arXiv:2504.13180*, 2025.

Hee-Soo Choi, Hyeong-Seok Kim, and Kyogu Lee. Tasnet-stft: Improving time-domain speech separation via stft magnitude and phase estimation. In *ICASSP*, 2021.

Joon Son Chung, Arsha Nagrani, and Andrew Zisserman. Voxceleb2: Deep speaker recognition. In *INTERSPEECH*, 2018.

ClearVoice. Clearvoice, 2025. https://github.com/modelscope/ClearerVoice-Studio/tree/main/train/target_speaker_extraction#4-audio-visual-speaker-extraction-conditioned-on-face-or-lip-recording.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

Ewen Cooper and Junichi Yamagishi. Range equalization bias in single-stimulus mos for synthetic speech. In *Proc. Interspeech*, Dublin, Ireland, 2023. ISCA. ISCA Archive: interspeech_2023/cooper23_interspeech.

Joseph Cosentino, Manuel Pariente, Samuele Cornell, Joris Thienpondt, Antoine Deleforge, and Emmanuel Vincent. Librimix: An open-source dataset for speech separation. arXiv:2005.11262, 2020.

Shaoxiang Dang, Tetsuya Matsumoto, Yoshinori Takeuchi, and Hiroaki Kudo. Using semi-supervised learning for monaural time-domain speech separation with a self-supervised learning-based si-snr estimator. In *Proc. INTERSPEECH*, volume 2023, pages 3759–3763, 2023.

Alexandre Défossez. Hybrid spectrogram and waveform source separation. In *Proceedings of the ISMIR 2021 Workshop on Music Source Separation*, 2021.

Marc Delcroix and et al. Improving speaker discrimination of target speech extraction with time-domain speakerbeam. In *Interspeech*, 2020.

Pablo M Delgado and Jürgen Herre. Towards improved objective perceptual audio quality assessment-part 1: A novel data-driven cognitive model. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2024.

Hao-Wen Dong, Naoya Takahashi, Yuki Mitsufuji, Julian McAuley, and Taylor Berg-Kirkpatrick. Clipsep: Learning text-queried sound separation with noisy unlabeled videos. In *Proceedings of International Conference on Learning Representations (ICLR)*, 2023.

Jiarui Dong, Xiang Wang, and Qingyu Mao. Edsep: An efficient diffusion-based framework for speech source separation. In *ICASSP*, 2025.

Konstantinos Drossos, Timo Lipping, and Tuomas Virtanen. Clotho: An audio captioning dataset. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurélien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Rozière, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Graeme Nail, Grégoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel M. Kloumann, Ishan Misra, Ivan Evtimov, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, and et al. The llama 3 herd of models. *CoRR*, abs/2407.21783, 2024. doi: 10.48550/ARXIV.2407.21783. https://doi.org/10.48550/arXiv.2407.21783.

Alexandre Défossez. Hybrid transformer Demucs for music source separation. arXiv:2301.04604, 2023.

Alexandre Défossez, Nicolas Usunier, Léon Bottou, and François Raposo. Music source separation in the waveform domain. In *International Society for Music Information Retrieval Conference (ISMIR)*, 2019.

Alexandre Défossez, Jade Copet, Gabriel Synnaeve, and Yossi Adi. High fidelity neural audio compression. *arXiv preprint arXiv:2210.13438*, 2022.

ElevenLabs. Elevenlabs, 2025. https://elevenlabs.io/voice-isolator.

Ariel Ephrat, Inbar Mosseri Zeiters, Tavi Halperin, A Krishnan, Kevin Wilson, William T. Freeman, and Michael Rubinstein. Looking to listen at the cocktail party: A speaker-independent audio-visual model for speech separation. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018.

Matteo Fabbro, Stefan Uhlich, Minseok Kim, Antoine Liutkus, and Yoshiaki Mitsufuji. The sound demixing challenge 2023 – music demixing track. *Transactions of the International Society for Music Information Retrieval*, 7(1):66–77, 2024. doi: 10.5334/tismir.171. https://transactions.ismir.net/articles/10.5334/tismir.171.

FADR. Fadr, 2025. https://fadr.com/.

Szu-Wei Fu et al. Metricgan: Generative adversarial networks based black-box metric scores optimization for speech enhancement. In *ICML*, 2019.

Jort F. Gemmeke, Daniel P. W. Ellis, Anja Freedman, Aren Jansen, Wade Lawrence, R. Channing Moore, Manoj Plakal, and Ethan Ritter. Audioset: An ontology and human-labeled dataset for audio events. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017.

Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. Imagebind: One embedding space to bind them all. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15180–15190, 2023.

Andrés Gusó and John Thickstun. Tf-locoformer: Efficient time-frequency transformer for audio source separation. In *Interspeech*, 2022.

Kilem L. Gwet. Handbook of inter-rater reliability: The definitive guide to measuring the extent of agreement among multiple raters. *Advanced Analytics, LLC*, 2014. https://www.agreestat.com/book4.pdf.

Jiarui Hai, Helin Wang, Dongchao Yang, Karan Thakkar, Najim Dehak, and Mounya Elhilali. Dpm-tse: A diffusion probabilistic model for target sound extraction. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1196–1200, 2024.

Romain Hennequin, Anis Khlif, Felix Voituret, and Manuel Moussallam. Spleeter: a fast and efficient music source separation tool with pre-trained models. *Journal of Open Source Software*, 5(50):2154, 2020.

John R. Hershey, Zhuo Chen, Jonathan Le Roux, Shinji Watanabe, Michael Wilson, and Pablo Sprechmann. Deep clustering: Discriminative embeddings for segmentation and separation. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016.

Chao Huang, Susan Liang, Yapeng Tian, Anurag Kumar, and Chenliang Xu. High-quality visually-guided sound separation from diverse categories. In *Proceedings of the Asian Conference on Computer Vision (ACCV)*, pages 35–49, December 2024a.

Chao Huang, Susan Liang, Yapeng Tian, Anurag Kumar, and Chenliang Xu. High-quality sound separation across diverse categories via visually-guided generative modeling. *arXiv preprint arXiv:2509.22063*, 2025.

Wen-Chin Huang, Erica Cooper, Yu Tsao, Hsin-Min Wang, Tomoki Toda, and Junichi Yamagishi. The voicemos challenge 2022. *arXiv preprint arXiv:2203.11389*, 2022.

Zhichao Huang, Yizhuo Wang, Shuai Yang, Xiaohong Xu, and Yizhou Wang. DAVIS: Diffusion audio-visual source separation. arXiv:2403.12345, 2024b.

Jaesung Huh. Voice gender classifier. https://huggingface.co/JaesungHuh/voice-gender-classifier, 2024. HuggingFace Model Hub.

ITU-R. Method for the subjective assessment of intermediate quality level of audio systems (mushra). Recommendation BS.1534-3, International Telecommunication Union, Radiocommunication Sector, Geneva, Switzerland, 2015. Latest revision; earlier versions date back to 2001.

ITU-T. Methods for subjective determination of transmission quality. Recommendation P.800, International Telecommunication Union, Telecommunication Standardization Sector, Geneva, Switzerland, 1996.

Noah Jaffe and John Ashley Burgoyne. Musical source separation bake-off: Comparing objective metrics with human perception. *arXiv preprint arXiv:2507.06917*, 2025.

Vinay Jayaram and John Thickstun. Source separation with normalizing flows. *arXiv preprint arXiv:2011.10317*, 2020.

Nikhil Kandpal, Oriol Nieto, and Zeyu Jin. Music enhancement via image translation and vocoding. *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3124–3128, 2022. https://api.semanticscholar.org/CorpusID:248427207.

Keunwoo Kim, Doyeon Kim, Jongpil Lee, and Juhan Nam. Audiocaps: Generating captions for audios in the wild. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019.

Minseok Kim, Woosung Choi, Jaehwa Chung, Daewon Lee, and Soonyoung Jung. Kuielab-mdx-net: A two-stream neural network for music demixing. In *ISMIR*, 2021.

Diederik P Kingma. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanqi Mao, Chloe Rolland, Tete Fu, Varun Pósz, Stephanie Rohan, Damien Zhan, Laura Collins, , et al. Segment anything. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

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.

Qiuqiang Kong, Ke Chen, Haohe Liu, Xingjian Du, Taylor Berg-Kirkpatrick, Shlomo Dubnov, and Mark D Plumbley. Universal source separation with weakly labelled data. *arXiv preprint arXiv:2305.07447*, 2023.

Zhifeng Kong and Wei Ping. Speech enhancement with deep feature losses. In *Interspeech*, 2019.

Khaled Koutini, Jan Schlüter, Hamid Eghbal-zadeh, and Gerhard Widmer. Efficient training of audio transformers with patchout. In *Interspeech 2022, 23rd Annual Conference of the International Speech Communication Association, Incheon, Korea, 18-22 September 2022*, pages 2753–2757. ISCA, 2022. doi: 10.21437/Interspeech.2022-227. https://doi.org/10.21437/Interspeech.2022-227.

Rithesh Kumar, Prem Seetharaman, Alejandro Luebs, Ishaan Kumar, and Kundan Kumar. High-fidelity audio compression with improved RVQGAN. 2024.

Lalal.AI. lalalai, 2025. https://www.lalal.ai.

Matt Le, Apoorv Vyas, Bowen Shi, Brian Karrer, Leda Sari, Rashel Moritz, Mary Williamson, Vimal Manohar, Yossi Adi, Jay Mahadeokar, and Wei-Ning Hsu. Voicebox: Text-guided multilingual universal speech generation at scale. *ArXiv*, abs/2306.15687, 2023.

Jonathan Le Roux, Scott Wisdom, Hakan Erdogan, and John R Hershey. Sdr–half-baked or well done? In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 626–630. IEEE, 2019.

Haoxin Li, Rui Zhang, Jing Xu, Jingjing Zhou, and Yizhou Wang. MUSIC-AVQA: A large-scale dataset for music audio-visual question answering. arXiv:2203.14072, 2022.

Jiarui Li, Xinhao Liu, Mark D. Plumbley, and Qiuqiang Kong. Audiosep: Language-based audio source separation. arXiv:2308.01883, 2023.

Kai Li, Fenghua Xie, Hang Chen, Kexin Yuan, and Xiaolin Hu. An audio-visual speech separation model inspired by cortico-thalamo-cortical circuits. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024a.

Kai Li, Runxuan Yang, Fuchun Sun, and Xiaolin Hu. IIANet: An intra- and inter-modality attention network for audio-visual speech separation. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp, editors, *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 29181–29200. PMLR, 21–27 Jul 2024b. https://proceedings.mlr.press/v235/li24cf.html.

Meng Li, Xin Zhou, et al. Audionet-diff: Conditional diffusion for end-to-end speech enhancement. *IEEE Signal Processing Letters*, 2024c.

Z. Li, Y. Wang, and Zhiyao Duan. URMP: University of rochester multimodal music performance dataset. http://www2.ece.rochester.edu/projects/air/projects/urmp.html, 2018.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. 2023.

Chen Liu, Peike Li, Qingtao Yu, Hongwei Sheng, Dadong Wang, Lincheng Li, and Xin Yu. Benchmarking audio visual segmentation for long-untrimmed videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22712–22722, 2024a.

Jiayi Liu, Wen Zhang, et al. Resflow-se: Residual flow matching for speech enhancement. *arXiv preprint arXiv:2405.04561*, 2024b.

Xubo Liu, Haohe Liu, Qiuqiang Kong, Xinhao Mei, Jinzheng Zhao, Qiushi Huang, Mark D Plumbley, and Wenwu Wang. Separate what you describe: Language-queried audio source separation. In *Proc. INTERSPEECH 2022*, 2022.

Xubo Liu, Qiuqiang Kong, Yan Zhao, Haohe Liu, Yi Yuan, Yuzhuo Liu, Rui Xia, Yuxuan Wang, Mark D Plumbley, and Wenwu Wang. Separate anything you describe. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2024c.

Antoine Liutkus, Fabian-Robert Stöter, Nobutaka Ito, and Daiki Kitamura. Music demixing challenge 2021 (MDX). https://www.aicrowd.com/challenges/music-demixing-challenge-ismir-2021, 2021.

Antoine Liutkus, Fabian-Robert Stöter, et al. Music demixing challenge 2023 (MDX). https://www.aicrowd.com/challenges/music-demixing-challenge-ismir-2023, 2023.

Philipos C Loizou. *Speech Enhancement: Theory and Practice*. CRC Press, 2013.

Wei-Tsung Lu, Ju-Chiang Wang, Qiuqiang Kong, and Yun-Ning Hung. Music source separation with band-split rope transformer. 2023. https://api.semanticscholar.org/CorpusID:261556702.

Yi Luo and Nima Mesgarani. Conv-tasnet: Surpassing ideal time–frequency magnitude masking for speech separation. In *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2019.

Yi Luo, Zhuo Chen, and Takuya Yoshioka. Dual-path rnn: Efficient long sequence modeling for time-domain single-channel speech separation. In *ICASSP*, 2020.

Hao Ma, Zhiyuan Peng, Xu Li, Mingjie Shao, Xixin Wu, and Ju Liu. Clapsep: Leveraging contrastive pre-trained model for multi-modal query-conditioned target sound extraction. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 32:4945–4960, 2024.

Ethan Manilow, Prem Seetharaman, and Bryan Pardo. Slakh2100: A dataset for music source separation. arXiv:1909.08466, 2019.

Pranay Manocha, Zeyu Jin, Richard Zhang, and Adam Finkelstein. Cdpam: Contrastive learning for perceptual audio similarity. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 196–200. IEEE, 2021.

Gianluca Mariani, Ilaria Tallini, Elena Postolache, Matteo Mancusi, and Emanuele Rodolà. Multi-source diffusion models for simultaneous music generation and separation. In *ICLR*, 2024.

Shivam Mehta, Ruibo Tu, Jonas Beskow, Éva Székely, and Gustav Eje Henter. Matcha-TTS: A fast TTS architecture with conditional flow matching. In *Proc. ICASSP*, 2024.

Yuki Mitsufuji, Giorgio Fabbro, Stefan Uhlich, and Fabian-Robert Stöter. Music demixing challenge 2021, 2021.

Gabriel Mittag, Babak Naderi, Assmaa Chehadi, and Sebastian Möller. Nisqa: A deep cnn-self-attention model for multidimensional speech quality prediction with crowdsourced datasets. In *Proc. Interspeech 2021*, pages 2127–2131, 2021.

Moises.AI. Moises.ai, 2025. https://moises.ai/.

Viet-An Nguyen, Henry C. Lin, Peibei Shi, Jagdish Ramakrishnan, Udi Weinsberg, Steve Metz, Neil Chandra, Jane Jing, and Dimitris Kalimeris. Clara: Confidence of labels and raters. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. ACM, 2020.

Zexu Pan, Shengkui Zhao, Tingting Wang, Kun Zhou, Yukun Ma, Chong Zhang, and Binchao Ma. Plug-and-play co-occurring face attention for robust audio-visual speaker extraction. *ArXiv*, abs/2505.20635, 2025.

Santiago Pascual, Antonio Bonafonte, and Joan Serrà. Segan: Speech enhancement generative adversarial network. In *INTERSPEECH*, 2017.

William Peebles and Saining Xie. Scalable diffusion models with transformers. 2023.

Adam Polyak et al. Movie gen: A cast of media foundation models. *ArXiv*, abs/2410.13720, 2024.

KR Prajwal, Bowen Shi, Matthew Le, Apoorv Vyas, Andros Tjandra, Mahi Luthra, Baishan Guo, Huiyu Wang, Triantafyllos Afouras, David Kant, et al. MusicFlow: Cascaded flow matching for text guided music generation. 2024.

pydub. pydub. https://github.com/jiaaro/pydub.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, 2021a.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, 2021b.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020. http://jmlr.org/papers/v21/20-074.html.

Zafar Rafii, Antoine Liutkus, Fabian-Robert Stöter, Stylianos Ioannis Mimilakis, and Rachel Bittner. The MUSDB18 corpus for music separation, December 2017a. https://doi.org/10.5281/zenodo.1117372.

Zafar Rafii, Antoine Liutkus, Fabian-Robert Stöter, Stylianos Ioannis Mimilakis, and Derry FitzGerald. MUSDB18: A dataset for music source separation. Zenodo, 2017b.

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. https://arxiv.org/abs/2408.00714.

Chandan K. A. Reddy, Ebrahim Beyrami, Ross Cutler, Sefik Srinivasan, Sagie Ivry, Yosef Attias, Sylvain Dubreuil, and Orith Aharon. The Deep Noise Suppression challenge: A competitive framework for speech enhancement. In *INTERSPEECH*, 2020.

Chandan KA Reddy, Vishak Gopal, and Ross Cutler. Dnsmos: A non-intrusive perceptual objective speech quality metric to evaluate noise suppressors. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6493–6497. IEEE, 2021.

Erez Richardson and et al. Speech enhancement using diffusion probabilistic models. In *ICASSP*, 2023.

Yuan Rong, Chen Sun, Ishan Misra, Cordelia Schmid, Josef Sivic, and Andrew Zisserman. Condensed movies dataset. https://www.robots.ox.ac.uk/~vgg/data/condensed-movies/, 2020.

Simon Rouard, Francisco Massa, and Alexandre Défossez. Hybrid transformers for music source separation. In *ICASSP 23*, 2023.

Neville Ryant, Mark Liberman, Jesus Villalba Macias, Umar Sheikh Pardo, Yeliz Isik, Apurv Sangwan, and Sanjeev Khudanpur. The second DIHARD diarization challenge: Dataset, task, and baselines. In *INTERSPEECH*, 2019.

Ryoichi Sawata et al. Diffiner: Diffusion-based refinement for speech separation and enhancement. In *Interspeech*, 2023.

Noah Schaffer, Boaz Cogan, Ethan Manilow, Max Morrison, Prem Seetharaman, and Bryan Pardo. Music separation enhancement with generative modeling. In *ISMIR*, 2022.

Robin Scheibler, Yoonmi Ji, Sang-Won Chung, and Min-Seok Choi. Diffusion-based generative speech source separation. In *ICASSP*, 2023.

Robin Scheibler, John R. Hershey, Arnaud Doucet, and Henry Li. Source separation by flow matching. *arXiv preprint arXiv:2402.11543*, 2024.

B Series. Method for the subjective assessment of intermediate quality level of audio systems. *International Telecommunication Union Radiocommunication Assembly*, 2, 2014.

Joan Serrà, Dmitry Bogdanov, Johan Deja, and Others. Flow matching for audio(-visual) source separation. arXiv:2405.00000, 2024.

Fabian-Robert Stöter, Stefan Uhlich, Antoine Liutkus, and Yuki Mitsufuji. Open-unmix-a reference implementation for music source separation. *Journal of Open Source Software*, 4(41):1667, 2019.

Fabian-Robert Stöter, Antoine Liutkus, et al. A large-scale human evaluation of audio source separation and metrics. In *Proceedings of [venue]*, 2024. Preprint/venue TBD; update DOI and proceedings details.

Cem Subakan and Paris Smaragdis. Generative adversarial source separation. In *ICASSP*, 2018.

Cem Subakan, Mirco Ravanelli, Samuele Cornell, Marco Bronzi, and Jianyuan Zhong. Attention is all you need in speech separation. In *ICASSP*, 2021.

Naoya Takahashi, Nabarun Goswami, and Yasuhiko Mitsufuji. Multi-scale multi-band densenets for audio source separation. In *WASPAA*, 2018.

Andros Tjandra, Yi-Chiao Wu, Baishan Guo, John Hoffman, Brian Ellis, Apoorv Vyas, Bowen Shi, Sanyuan Chen, Matt Le, Nick Zacharov, et al. Meta audiobox aesthetics: Unified automatic quality assessment for speech, music, and sound. *arXiv preprint arXiv:2502.05139*, 2025.

Amos Tversky and Daniel Kahneman. Judgment under uncertainty: Heuristics and biases. *Science*, 185(4157): 1124–1131, 1974. doi: 10.1126/science.185.4157.1124.

Cassia Valentini-Botinhao. Noisy speech database for training speech enhancement algorithms and tuning parameters (VoiceBank-DEMAND). https://datashare.ed.ac.uk/handle/10283/2791, 2016.

V. Varadhan et al. Anchoring and reference-matching bias in mushra listening tests. *arXiv preprint arXiv:2411.12719*, 2024. Preprint; update authors/title if needed.

Emmanuel Vincent, Rémi Gribonval, and Cédric Fevotte. Performance measurement in blind audio source separation. In *IEEE Transactions on Audio, Speech, and Language Processing*, volume 14, pages 1462–1469, 2006.

Apoorv Vyas, Bowen Shi, Matthew Le, Andros Tjandra, Yi-Chiao Wu, Baishan Guo, Jiemin Zhang, Xinyue Zhang, Robert Adkins, William Ngan, et al. Audiobox: Unified audio generation with natural language prompts. *arXiv preprint arXiv:2312.15821*, 2023.

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. 2025.

DeLiang Wang and Jitong Chen. Supervised speech separation based on deep learning: An overview. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 26(10):1702–1726, 2018.

Helin Wang, Dongchao Yang, Chao Weng, Jianwei Yu, and Yuexian Zou. Improving target sound extraction with timestamp information. In *Proc. Interspeech*, pages 1526–1530, 2022a.

Helin Wang, Dongchao Yang, Yuexian Zou, Fan Cui, and Yujun Wang. Detect what you want: Target sound detection. In *DCASE*, 2022b.

Helin Wang, Jesús Villalba, Laureano Moro-Velazquez, Jiarui Hai, Thomas Thebaud, and Najim Dehak. Noise-robust speech separation with fast generative correction. In *Proc. Interspeech 2024*, pages 2165–2169, 2024.

Helin Wang, Jiarui Hai, Yen-Ju Lu, Karan Thakkar, Mounya Elhilali, and Najim Dehak. Soloaudio: Target sound extraction with language-oriented audio diffusion transformer. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2025.

Quan Wang, RJ Skerry-Ryan, Ye Jia, and et al. Voicefilter: Targeted voice separation by speaker-conditioned spectrogram masking. In *Interspeech*, 2019.

Shinji Watanabe, Michael I. Mandel, Jon Barker, Emmanuel Vincent, Amin Karbasi, Gregory Sell, and Najim Dehak. CHiME-6 challenge: Automatic speech recognition for multi-speaker meetings. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020.

Yutong Wen, Ke Chen, Prem Seetharaman, Oriol Nieto, Jiaqi Su, Rithesh Kumar, Minje Kim, Paris Smaragdis, Zeyu Jin, and Justin Salamon. Promptsep: Generative audio separation via multimodal prompting. 11 2025. doi: 10.48550/arXiv.2511.04623.

Gordon Wichern, Joseph Antognini, Emmanuel Vincent, Jonathan McQuinn, Ashutosh Pandey, Vighnesh Manohar, and Jonathan Le Roux. Wham!: Extending speech separation to noisy environments. arXiv:1907.01160, 2019a.

Gordon Wichern, Jonathan Le Roux, Joseph Antognini, Emmanuel Vincent, Jonathan McQuinn, Vighnesh Manohar, and Ashutosh Pandey. Whamr!: Noisy and reverberant speech separation. arXiv:1904.09715, 2019b.

Scott Wisdom, Hakan Erdogan, Jacob R. White, Daniel P. W. Chinen, Tamar Remez, Kevin Wilson, and John R. Hershey. FUSS: Free universal sound separation dataset. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021.

Cheng Wu, Keunwoo Choi, Keunwoo Kim, and Barnabás Póczos. Language-queried audio source separation with CLAP embeddings. arXiv:2305.10973, 2023a.

Jian Wu, Yong Xu, Shi-Xiong Zhang, Lianwu Chen, Meng Yu, Lei Xie, and Dong Yu. Time domain audio visual speech separation. *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 667–673, 2019.

Y Wu et al. Large-scale contrastive language-audio pretraining with feature fusion and keyword-to-caption augmentation. In *ICASSP 2023*, 2023b.

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 *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2023c.

Yuwei Wu, Chenyu Zhang, Limin Wang, and Bo Li. AVSBench: A benchmark for audio-visual segmentation. arXiv:2204.11574, 2022.

Hu Xu, Saining Xie, Xiaoqing Ellen Tan, Po-Yao Huang, Russell Howes, Vasu Sharma, Shang-Wen Li, Gargi Ghosh, Luke Zettlemoyer, and Christoph Feichtenhofer. Demystifying clip data. 2023.

Mohan Xu, Kai Li, Guo Chen, and Xiaolin Hu. Tiger: Time-frequency interleaved gain extraction and reconstruction for efficient speech separation. *arXiv preprint arXiv:2410.01469*, 2024.

Y. Xu, J. Shi, et al. Spex+: A complete time-domain speaker extraction network. In *Interspeech*, 2020.

Sihyun Yu, Sangkyung Kwak, Huiwon Jang, Jongheon Jeong, Jonathan Huang, Jinwoo Shin, and Saining Xie. Representation alignment for generation: Training diffusion transformers is easier than you think. In *International Conference on Learning Representations*, 2025.

Yi Yuan, Xubo Liu, Haohe Liu, Mark D Plumbley, and Wenwu Wang. Flowsep: Language-queried sound separation with rectified flow matching. In *IEEE International Conference on Acoustic, Speech and Signal Procssing (ICASSP)*, 2024.

Hang Zhao, Chuang Gan, Andrew Rouditchenko, Carl Vondrick, Josh McDermott, and Antonio Torralba. The sound of pixels. In *The European Conference on Computer Vision (ECCV)*, September 2018a.

Hang Zhao, Chuang Gao, Liwei Wu, Raphaël Féraud, Antonio Torralba, and Ting Dai. The sound of pixels. In *European Conference on Computer Vision (ECCV)*, 2018b.

Shengkui Zhao, Yukun Ma, Chongjia Ni, Chong Zhang, Hao Wang, Trung Hieu Nguyen, Kun Zhou, Jia Qi Yip, Dianwen Ng, and Bin Ma. Mossformer2: Combining transformer and rnn-free recurrent network for enhanced time-domain monaural speech separation. In *ICASSP*, pages 10356–10360. IEEE, 2024. ISBN 979-8-3503-4485-1. http://dblp.uni-trier.de/db/conf/icassp/icassp2024.html#ZhaoMNZ000YN024.

Shengkui Zhao, Zexu Pan, and Bin Ma. Clearervoice-studio: Bridging advanced speech processing research and practical deployment. In *Proc. Interspeech*. ISCA, 2025.

Appendix

A Subjective Evaluation

A.1 Design

Our protocol is designed to meet four requirements: (i) *ecological validity* in complex, real audio/video; (ii) *sensitivity* to small deltas for ablations; (iii) *comparability* across prompt modalities (text, visual masklets, temporal spans) and domains (speech, music, instruments, general sounds); and (iv) *operational feasibility* given the scale and diversity of test items. These criteria led us away from classic single-stimulus MOS/CMOS and MUSHRA-style multi-stimulus tests for this task, and toward a side-by-side ACR hybrid.

A.1.1 Why not MUSHRA, MOS, or CMOS?

MUSHRA (ITU-R BS.1534) excels when a high-quality *reference* and *anchors* are available for each item, enabling multi-stimulus ratings with hidden references and low anchors ITU-R (2015). In our setting, however, (a) we often lack any trustworthy reference extract from real mixtures; (b) anchor selection is non-trivial for heterogeneous targets (speech, instruments, events) and can induce *anchor and reference-matching biases*—higher-quality references can depress scores for otherwise good outputs; scores can drift toward "similarity to reference" rather than perceived quality Varadhan et al. (2024); and (c) multi-stimulus sessions amplify *range equalization* and *session drift* effects Cooper and Yamagishi (2023). Combined with significant *cost and time* overhead, these make MUSHRA mismatched to our in-the-wild, multi-domain separation benchmark.

MOS/ACR (ITU-T P.800) offers simple single-stimulus absolute ratings on a 5-point scale ITU-T (1996). While operationally easy, single-stimulus MOS has well-known shortcomings for our use case: (i) limited sensitivity for small model deltas in ablation studies; (ii) susceptibility to *anchoring* and *range* biases—absolute scores drift with context over a session Tversky and Kahneman (1974); Cooper and Yamagishi (2023); and (iii) no direct handle on *relative* model ranking without substantially increasing sample sizes.

CMOS/CCR (paired comparisons) addresses relative sensitivity by asking evaluators to express preference (and sometimes strength) between two stimuli ITU-T (1996). Pure preference designs, however, provide *no* insight into absolute performance thresholds (e.g., "good enough" for deployment), and can be brittle when different error modes produce similar overall impressions.

A.1.2 Our Hybrid: Side-by-Side ACR with Preference Tie-Breaker

We therefore adopt a **side-by-side ACR** protocol with an **always-on preference** question:

- **Presentation**: Two model outputs (A, B) for the *same* item are presented *side-by-side*, with the target specification (text prompt, SAM masklets, and/or temporal spans) and the original mixture for reference context.
- **Absolute ratings**: Evaluators assign *independent* 5-point ACR scores to A and B for three dimensions aligned to separation goals: *Recall* (how much target was extracted), *Precision* (how much non-target leaked in), and *Faithfulness* (similarity to how the target sounded in-context). We also collect an *Overall* score for both model outputs.
- **Preference tie-breaker**: Evaluators *always* answer a direct preference question ("Which extracted audio do you prefer, A or B?"), even when ACR scores differ. This "forced" tie-breaker increases inter-annotator agreement (IAA) and sharpens delta estimates. Conditionally asking this question only when evaluators provide identical ACR scores for A and B may induce evaluators to provide different scores more frequently to avoid the extra conditional question (though this hypothesis was not directly tested).
- **Modality coverage**: The same protocol applies across prompt modalities—text, visual (SAM masklets), and temporal spans—allowing controlled ablations (e.g., text-only vs. text+visual vs. visual-only) on the *same* in-the-wild items.

This hybrid provides *the best of both worlds*: *absolute* quality signals for deployment-readiness and *relative* signals for ablations and model selection.

A.1.3 Empirical Observations

Across multiple studies, we observe:

- **Inter-annotator agreement improves** with the preference tie-breaker: preference agreement and the consistency of absolute deltas increase relative to ACR-without-tie-breaker and pairwise-only preference protocols, measured in terms of Gwet's AC2 (Gwet, 2014).
- **Sharper deltas**: The presence of the tie-breaker leads evaluators to produce *larger* ACR score deltas on average (approximately +0.15 points per item), aligning absolute scores with expressed preference and increasing sensitivity for near-tied comparisons.
- **Lower uncertainty on deltas**: Side-by-side judgment reduces confidence interval (CI) width for differences between models by up to ∼20% versus single-stimulus rating, implying ∼30% cost savings to achieve the same sensitivity in A/B testing, where CI width is measured by bootstrap.
- **Handling time trade-off**: Side-by-side ACR roughly doubles per-item handling time relative to pairwise-only (more questions per item, 2 minute handling vs 1 minute handling for CMOS), but yields both absolute and relative signals in a single assignment.
- **Anchoring effects persist**: Absolute scores depend on comparison context—presenting an output alongside a higher-quality counterpart depresses its ACR scores, consistent with broader anchoring and reference-matching literature Tversky and Kahneman (1974); Varadhan et al. (2024); Cooper and Yamagishi (2023). We therefore emphasize comparative statistics (win rate, ACR deltas) over raw absolute means.

A.1.4 Operational Details

Instructioning and training. Appendix A.2 contains the full evaluator instructions and examples. Evaluators are trained to: (i) attend to the provided prompts (text/masklets/spans), (ii) separate "precision" (non-target leakage) from "recall" (missed target), (iii) judge "faithfulness" relative to the target's timbral/temporal character in the mixture (not similarity to any external reference).

Randomization and blinding. Model identities are blinded; A/B order is randomized per item; items are batched to avoid long runs from one domain/modality.

Quality controls. We leverage several tools for quality controls; (i) vendor-side quality assurance protocols, where a core group of trusted and trained evaluators check evaluations from a sample of annotators, (ii) Bayesian modeling of rater quality: we leverage CLARA (Nguyen et al., 2020) to obtain posteriors over rater confusion matrices and remove raters with anomalously high aggregate measures of deviation from ground truth.

Given anchoring risks, we emphasize *comparative* statistics over raw absolute means, and caution against over-interpreting cross-experiment shifts in absolute levels.

A.1.5 Positioning vs. Standards and Literature

Our protocol inherits *absolute* rating interpretability from ACR/MOS ITU-T (1996) while retaining *relative* sensitivity akin to CMOS/CCR ITU-T (1996), but avoids the *reference/anchor* dependencies and multi-stimulus biases of MUSHRA ITU-R (2015); Varadhan et al. (2024). It is aligned with recent large-scale audio separation evaluations that highlight the *disconnect* between reference-based metrics (SDR/SI-SDR/SAR) and human perception across stems Stöter et al. (2024), and with broader findings on evaluator behavior (*anchoring*, *range equalization*, *session drift*) Tversky and Kahneman (1974); Cooper and Yamagishi (2023).

Limitations. Side-by-side ACR remains subject to context effects; absolute scores should be interpreted cautiously. Developing better reference-free objective metrics that correlate with human judgments—especially for vocals and complex mixtures—remains an open area Stöter et al. (2024).

A.2 Subjective evaluation protocol

Below we present the content of the instructions presented to annotators when evaluating audio separation models.

Audio Separation

You will be evaluating models that perform audio separation.

What is audio separation? When using an audio separation model, users provide the model with either an audio track or a video with sound, and descriptions of which parts of the sound they want the model to extract (for example "guitar" or "dog barking"). The model then extracts the part of the audio that the user requests without any other sounds.

Key Terms

Prompt The user description of the target audio (e.g., "dog barking" or the visually highlighted dog in the video).

Source audio The original video or audio provided to the model.

Target sounds The portions of the source audio that the user is requesting.

Non-target sounds All other sounds in the source audio not requested by the user.

Extracted audio The audio that the model produces from the source audio and the prompt.

What is "distortion"?

When we say a sound is "distorted", we are referring to how a particular sound may have changed from the way it sounds in the source audio.

Distortions Distortions are modifications of sounds that originate from a discernable source in the source audio. These modifications are not the result of mixing with other sounds present in the source audio. If you can't tell if a sound is a distortion or simply the result of mixing with other sounds in the source audio, you may consider it to be a distortion.

Examples of distortions

- Different levels of "bass" or "treble": the target sound may sound like it is more or less "bass"-y, or there are more high pitched sounds.
- Pitch differences: if the sound is higher or lower pitched than in the source audio.
- Static / artifacts / other noises with no discernable origin in the source audio: crackling, static, popping noises, etc. that are not present in the source audio, or are hard to tell.
- Garbling: if, for instance, speech sounds garbled when it wasn't in the source audio.

Overview

Model inputs and outputs that you will be presented with In this evaluation you will be presented with:

- The source audio (and possibly video).
- The prompt, which will be in one of three forms described later in this document.

- The extracted audio from two separate models.

First you will be asked several questions about how well the extracted audio of each model matches the target sounds that the user requested in their prompt. After answering these questions, you will then give the extracted audio an overall score based on its:

1. Faithfulness to the prompt: how well did the model follow the user's instructions about which sounds to extract?
2. Faithfulness to the sound of the target audio: does the extracted target sound actually sound like the target sound in the source audio?

The types of prompts you may see

Prompt type	Example	Description
Text-based	—	A text description of the target sound(s) that the user wants the model to extract from the source audio.
Video-based	—	A highlighted portion of the source video that is creating the target sounds that the user wants the model to extract from the source audio.
Span-based	—	Highlighted yellow segments of the audio where the target sound that the user wants the model to extract **is** present; highlighted red segments of the audio where the target sound is **not** present; regions of the audio without spans may or may not contain the target sound.

What you will be annotating

Survey questions for each model's extracted audio We will ask you to provide specific information about different aspects of each model's performance on a particular set of inputs. First, we will ask a question to understand if the target sounds that are requested in the prompt are actually present in the source audio. *This first question does not have anything to do with model outputs!*

Q1. Does the source audio contain all of the target sounds requested in the prompt?

- Yes
- No
- Possibly, I cannot tell

If the answer to the above question is "Yes," you will then be asked the following question:

Q2. What portion of the target sound(s) is/are present in the extracted audio, regardless if other non-target sounds or distortions are present? Consider portion in terms of (1) duration and (2) number of target sounds.

- All
- Most (a majority of the target sound is present in extracted audio but some is missing)
- Some (a minority of the target sound is present in extracted audio but most is missing)
- None

If the answer to Q2 is not "All", you will be asked about what aspects of the target sound(s) are missing:

Q2a. Choose one or more options to clarify what aspects of the target sound(s) may be missing from the extracted audio:

- One or more of the target sounds requested in the prompt is completely absent.
- One or more target sounds are missing for part of their duration.

- When present in the extracted audio, one or more complex target sounds are only partially extracted (ignoring distortions) [e.g., if "music" is the target sound and extracted audio is missing an instrument].
- Other (please specify).

If your answer to Q2 is not "None", we will also ask about similarity:

Q3. For target sounds that are present in the extracted audio, how similar do they sound to how they sounded in the source audio?

- Exactly the same
- Some distortions / artifacts
- Moderate distortions / artifacts
- Sounds completely different, barely recognizeable

Afterwards, you will be asked:

Q4. Are there distinguishable non-target sounds present in the extracted audio?

- Yes
- No

Q5. How many non-target sounds present in the extracted audio are also present in the source audio?

- All
- Some
- None

Q6. How well do you feel the model removed non-target sounds from the extracted audio?

- Almost perfectly: the vast majority of non-target sounds present in the source audio are filtered out of the extracted audio, no additional non-target sounds are added.
- Adequately: Most non-target sounds present in the source audio are filtered out of the extracted audio.
- Poorly: Some filtering but most non-target sounds present in the source audio are also mostly present in the extracted audio.
- Very poorly: no non-target sounds are filtered out, and additional non-target sounds may be present.

Q7. Are there any non-target sounds from the source audio that appear completely unfiltered in the extracted audio?

- Yes, more than one
- Yes, one
- No [Any non-target audios in the extracted audio have been at least partially filtered from the source audio]

Overall score for each model's extracted audio

After you provide responses to the above questions, we will ask you to provide an overall score for how well each model performed, on a scale from 1 to 5. This should ideally incorporate and be consistent with all of the information you provided above but just as importantly we would like you to use your own sense of judgement.

Score	Criteria
5: Perfect	All target sounds are present in their entirety, and sound identical to the way they sound in the source audio. No non-target sounds present in extracted audio.
4: Good	Only minor issues with extracted audio (e.g., minor portions of target audio may be missing or slightly distorted, a very small amount of non-target sounds may be present).
3: OK	Some serious issues with extracted audio: target audio is maybe half present and/or non-target sounds are about half present.
2: Poor	Many serious issues with extracted audio: some target audio in the extracted audio but heavily distorted, missing, and several non-target sounds present.
1: Terrible	Extracted audio is completely incorrect; none of the target audio is in the extracted audio, or, if present, none of the non-target audio is filtered.

Which model's extracted audio do you prefer? If your scores for each model output are the same, we will also ask you to pick which model performed better in your opinion. Use your best judgement about which output you prefer, even if neither model performed well.

- Group 1
- Group 2
- Can't decide

Frequently asked questions

Distinct types of sounds: you may be asked to provide input on the number of distinct target or non-target sounds. In some cases this may be simple (the audio has a dog barking and a cat meowing — there are two distinct sounds here), however in many cases this task may be more ambiguous. For instance, in an audio where two people are talking but there is a rock song playing in the background — is that three sounds: one for each person speaking and one for the music? Or do we need to count each instrument that makes up the rock song as a distinct sound?

- If the prompt asks for a specific instrument or song component you may consider each instrument of the music in the source audio to be an individual target or non-target sound.
- Background noise: for sounds of an environment (street sounds, honking horns, cars passing etc.; or sounds of a cafe) that are not easily discernible, these can be considered a single non-target "background" sound.

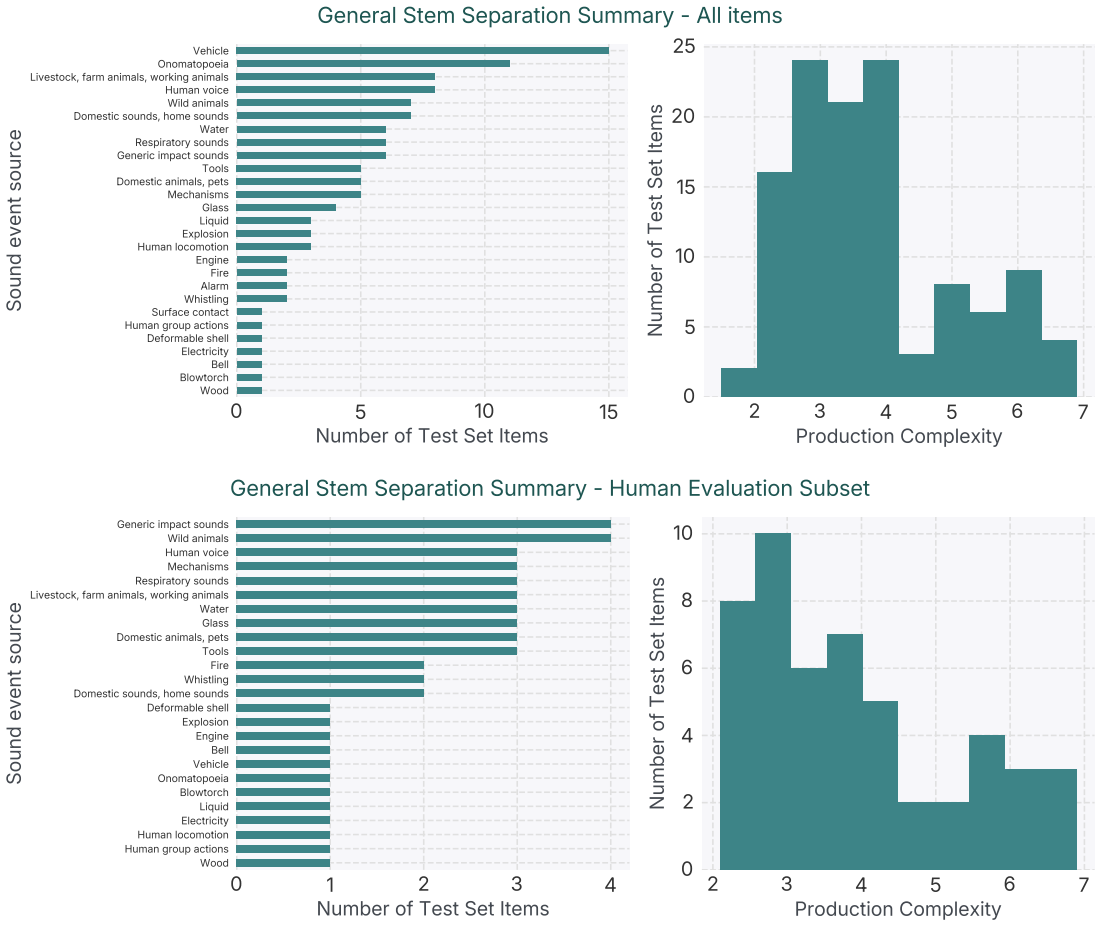

Figure 14 Statistics on **General** sound separation task. **Left**: distribution of target sounds using the second level of the Audioset hierarchy. **Right**: Distribution of the audio production complexity for each benchmark item. **Top**: Summary for all samples in SAM Audio-Bench, **bottom**: summary for samples in the subset used for human evaluations.

B SAM Audio-Bench

B.1 Characterization and Statistics

Figures 14 and 16 provide an overview of the data distribution for the instrument and speaker separation tasks in SAM Audio-Bench. For instruments, the benchmark is dominated by a handful of common sources, with a long tail of less frequent instruments, and most audios contain only two or three active instruments. For speakers, the dataset spans multiple speaker types and prompting modalities, and exhibits substantial variation in overlap between the target and interfering speakers. Non-speech distractors are diverse, ranging from environmental sounds to background music, which reflects the complexity of real-world audio mixtures.

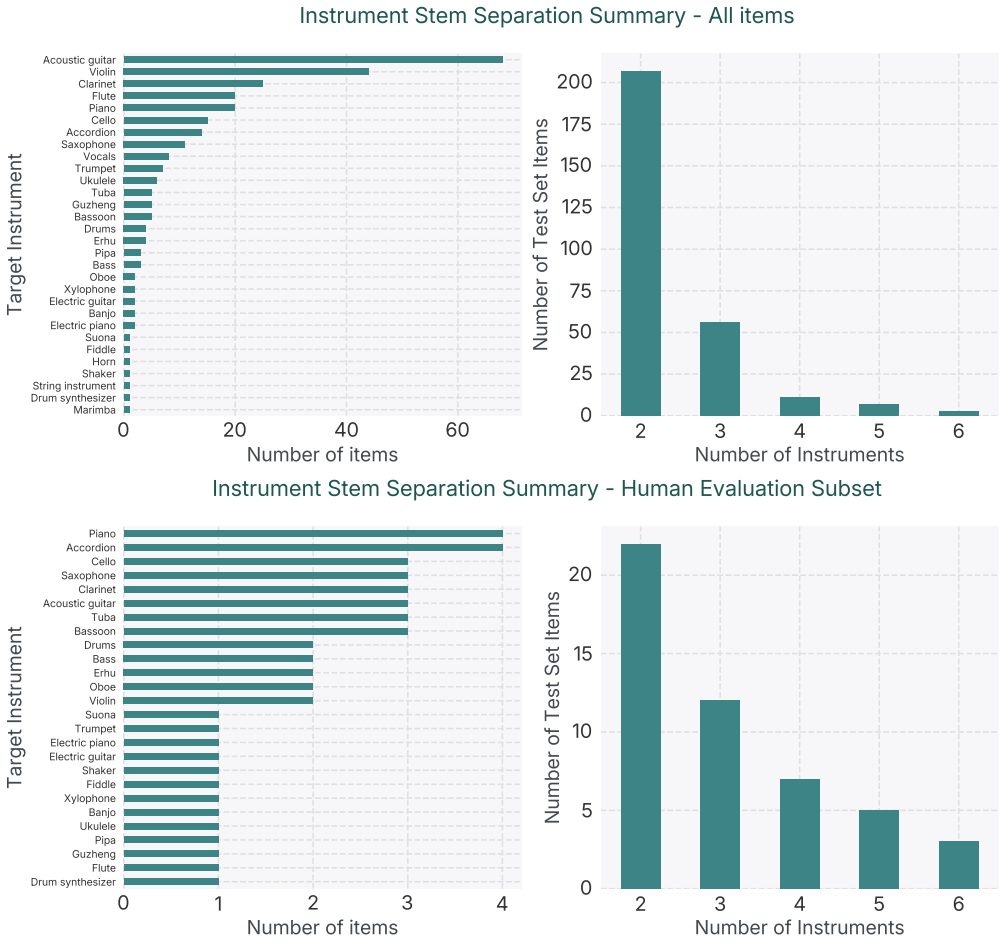

Figure 15 Statistics on the instrument stem separation task in SAM Audio-Bench. **Left**: histogram of target instruments. **Right**: histogram of the number of instruments playing in the video for each item in the test set. **Top**: Summary for all samples in SAM Audio-Bench, **bottom**: summary for samples in the subset used for human evaluations.

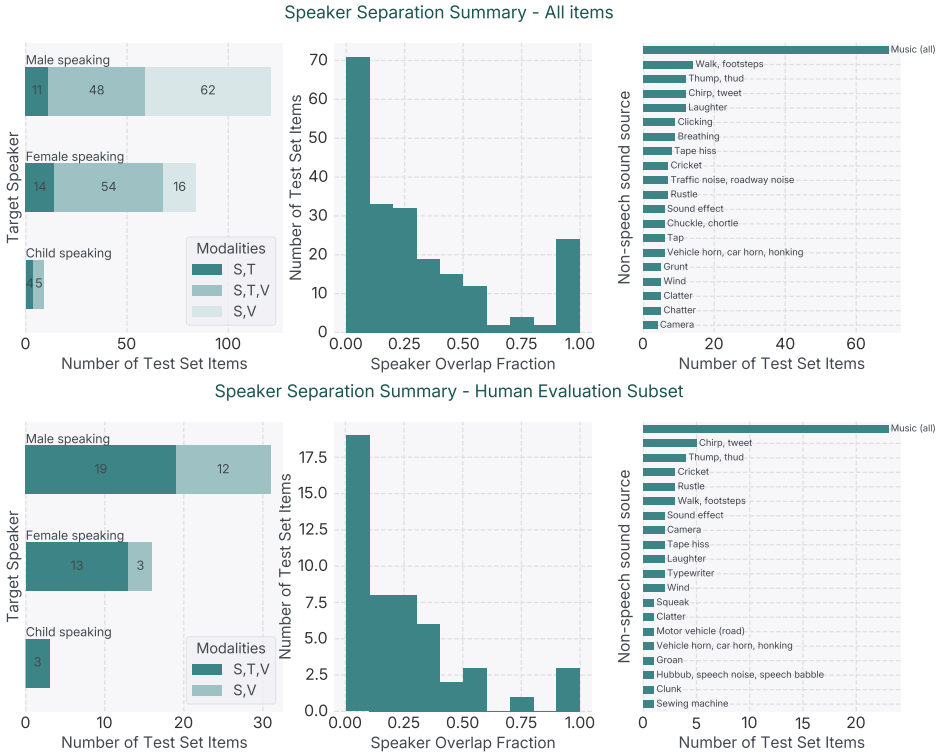

Figure 16 Statistics on the speaker separation task for SAM Audio-Bench. **Left**: is the distribution of target speaker type and modalities available for prompting (some items may not have visual or text prompts available). **Middle**: distribution of "speaker overlap" – the fraction of target speaker duration during which there is other speech occurring. **Top**: Summary for all samples in SAM Audio-Bench, **bottom**: summary for samples in the subset used for human evaluations.

C SAM Audio Judge

C.1 Data Annotation protocol

Below we present the content of the instructions presented to annotators when annotating the SAM Audio Judge scores.

Audio Separation

You will be evaluating models that perform audio separation.

What is audio separation? When using an audio separation model, users provide the model with either an audio track or a video with sound, and descriptions of which parts of the sound they want the model to extract (for example "guitar" or "dog barking"). The model then extracts the part of the audio that the user requests without any other sounds.

Key Terms

Prompt The user description of the target audio (e.g., "dog barking").

Source audio The original video or audio provided to the model.

Target sounds The portions of the source audio that the user is requesting.

Non-target sounds All other sounds in the source audio not requested by the user.

Extracted audio The audio that the model produces from the source audio and the prompt.

What is "distortion"?

When we say a sound is "distorted", we are referring to how a particular sound may have changed from the way it sounds in the source audio.

Distortions Distortions are modifications of sounds that originate from a discernable source in the source audio. These modifications are not the result of mixing with other sounds present in the source audio. If you can't tell if a sound is a distortion or simply the result of mixing with other sounds in the source audio, you may consider it to be a distortion.

Examples of distortions

- Different levels of "bass" or "treble": the target sound may sound like it is more or less "bass"-y, or there are more high pitched sounds.
- Pitch differences: if the sound is higher or lower pitched than in the source audio.
- Static / artifacts / other noises with no discernable origin in the source audio: crackling, static, popping noises, etc. that are not present in the source audio, or are hard to tell.
- Garbling: if, for instance, speech sounds garbled when it wasn't in the source audio.

Overview

Model inputs and outputs that you will be presented with In this evaluation you will be presented with:

- The source audio (and possibly video).
- The prompt, which is a text description of the target sound(s) that the user wants the model to extract from the source audio.
- The extracted audio from two separate models.

First you will be asked several questions about how well the extracted audio of each model matches the target sounds that the user requested in their prompt. After answering these questions, you will then give the extracted audio an overall score based on its:

1. Faithfulness to the prompt: how well did the model follow the user's instructions about which sounds to extract?
2. Faithfulness to the sound of the target audio: does the extracted target sound actually sound like the target sound in the source audio?

What you will be annotating

First, we will ask three questions (Q1-Q3) to assess whether the target sounds specified in the prompt are indeed present in the source audio, and whether any non-target sounds are also included.

If the answers to Q1 and Q3 are both "Yes", you will then be asked the four questions (Q4-Q8) to assess the difficulty of the audio separation task. *Note: Q1-Q8 have nothing to do with model outputs!*

After that, we will ask you questions (Q9-Q12) to assess the quality of the model's extracted audio. *Note: Q9-Q12 need model outputs!*

Q1. Is it clear to you which sounds in the source audio the prompt is referring to?

- Yes
- No

Q2. (Follow-up to Q1 if answer is No)

- There are no sounds in the source audio that match the prompt description.
- I'm not sure what the prompt is supposed to sound like.
- The prompt is ambiguous: there are several sounds present in the source audio that the prompt could be referring to. It is not clear which one is the target sound.

Q3. Does the source audio contain any non-target sounds not requested in the prompt?

- Yes
- No
- Possibly, I cannot tell

If the answer to Q3 is "Yes", you will then be asked the following questions:

Q4. How many non-target sounds are there in the source audio?
Guidance: A non-target sound is any clearly audible sound that is not requested in the prompt. Multiple instances of the same type of sound (e.g., repeated coughs) count as one. Sounds that are not identifiable but clearly come from different sources (e.g., speech vs. machinery) should be counted separately. Completely unidentifiable or ambiguous background sounds should be grouped and counted as a single non-target sound.

- 1
- 2
- 3
- 4
- 5+

Q5. How much do the target sound(s) overlap with the non-target sounds?

- When the target sounds are audible, no non-target sounds are audible at the same time.

- There is some slight overlap ($< 25\%$) of target sounds with non-target sounds.
- For about half of their duration the target sounds overlap with one or more non-target sounds.
- Target sounds are overlapped by one or more target sounds for most of their duration ($\sim$75%)
- All target sounds overlap completely with one or more non-target sounds.
- The target sounds are ambiguous, so I cannot tell.

Q6. How loud are the target sound(s) compared to the non-target sounds?
Guidance: Rate the overall perceived loudness of the target sound(s) relative to the non-target sounds throughout the audio. If the relative loudness varies over time, focus on the dominant trend or the most prominent portion of the target sound(s).

- Much louder than non-target sounds.
- Slightly louder than non-target sounds.
- About the same loudness.
- Slightly quieter than non-target sounds.
- Much quieter than non-target sounds.
- The target sounds are ambiguous, so I cannot tell.

Q7. How confusing non-target sounds are with the target sound(s)?
Guidance: Confusion may arise when non-target sounds share similar acoustic characteristics (e.g., pitch, timbre, timing) with the target, or occur at similar moments. For example, If the prompt is "a baby crying" and there is a cat meowing in the background that briefly resembles a baby cry, that would be slightly or moderately confusing depending on how similar it sounds.

- Not confusing (Non-target sounds are completely distinct from the target sounds.)
- Slightly confusing (Non-target sounds are mostly distinguishable but may briefly resemble the target sounds.)
- Moderately confusing (Some parts of the non-target sounds resemble the target sounds, causing occasional confusion.)
- Very confusing (Non-target sounds are often hard to distinguish from the target sounds.)
- Extremely confusing (Non-target sounds are nearly indistinguishable from the target sounds.)

Q8. Considering all the factors listed in Q4-Q7, how difficult do you think it is to extract the target sound(s) from the source audio?
Guidance: Confusion may arise when non-target sounds share similar acoustic characteristics (e.g., pitch, timbre, timing) with the target, or occur at similar moments. For example, If the prompt is "a baby crying" and there is a cat meowing in the background that briefly resembles a baby cry, that would be slightly or moderately confusing depending on how similar it sounds.

- 1: Very easy (Target sounds are loud, distinct from and barely overlap with the non-target sounds. There are very few non-target sounds)
- 2: Easy
- 3: Medium
- 4: Hard
- 5: Very hard (Target sounds are quiet, similar to and significantly overlap with the non-target sounds. There are many non-target sounds)

The following questions need model outputs! Next, you will then be asked the following question:

Q9. What portion of the target sound(s) is/are present in the extracted audio, regardless if other non-target sounds or distortions are present? Consider portion in terms of (1) duration and (2) number of target sounds.

- All
- Most (a majority of the target sound is present in extracted audio but some is missing)
- Some (half of the target sound is present in extracted audio but half is missing)
- Few (a minority of the target sound is present in extracted audio but most is missing)
- None

If your answer to "what portion of the target sounds is/are present in the extracted audio" is not "None", we will also ask about how similar the target sounds in the extracted audio sound to the way they sound in the source audio.

Q10. For target sounds that are present in the extracted audio, how similar do they sound to how they sounded in the source audio?

- Exactly the same
- Minor distortions / artifacts
- Moderate distortions / artifacts
- Serious distortions / artifacts
- Sounds completely different, barely recognizeable

Afterwards, you will be asked

Q11. How well do you feel the model removed non-target sounds from the extracted audio?

- Perfectly: No non-target sounds are present in the extracted audio.
- Almost perfectly: Most non-target sounds in the source audio are filtered out, with only minimal non-target sounds remaining.
- Reasonably: Some non-target sounds are removed from the source audio, but a noticeable portion remains in the extracted audio.
- Poorly: Only limited filtering occurs; most non-target sounds in the source audio are still present in the extracted audio.
- Very poorly: No non-target sounds are removed, and additional non-target sounds may be introduced during extraction.

Q12. Overall score

After you provide responses to the above questions, we will ask you to provide an overall score for how well the model performed, on a scale from 1 to 5. This should ideally incorporate and be consistent with all of the information you provided above but just as importantly we would like you to use your own sense of judgement.

Frequently asked questions

Distinct types of sounds: you may be asked to provide input on the number of distinct target or non-target sounds. In some cases this may be simple (the audio has a dog barking and a cat meowing — there are two distinct sounds here), however in many cases this task may be more ambiguous. For instance, in an audio where two people are talking but there is a rock song playing in the background — is that three sounds: one for each person speaking and one for the music? Or do we need to count each instrument that makes up the rock song as a distinct sound?

Score	Criteria
5: Perfect	All target sounds are present in their entirety, and sound identical to the way they sound in the source audio. No non-target sounds present in extracted audio.
4: Good	Only minor issues with extracted audio (e.g., minor portions of target audio may be missing or slightly distorted, a very small amount of non-target sounds may be present).
3: OK	Some serious issues with extracted audio: target audio is maybe half present and/or non-target sounds are about half present.
2: Poor	Many serious issues with extracted audio: some target audio in the extracted audio but heavily distorted, missing, and several non-target sounds present.
1: Terrible	Extracted audio is completely incorrect; none of the target audio is in the extracted audio, or, if present, none of the non-target audio is filtered.

- If the prompt asks for a specific instrument or song component you may consider each instrument of the music in the source audio to be an individual target or non-target sound.
- Background noise: for sounds of an environment (street sounds, honking horns, cars passing etc.; or sounds of a cafe) that are not easily discernible, these can be considered a single non-target "background" sound.

Speech prompts: you may see some prompts like "one of the men speaking", "one of the women speaking", "one of the people talking". In these cases, please focus on a single, consistent speaker throughout the audio.

- For example, if the prompt is "one of the men speaking" and the extracted audio contains two men speaking, the overall score should be rated low due to inconsistency with the prompt.
- As there may be multiple candidates, you could choose "The target sounds are ambiguous, so I cannot tell." for Q5 and Q6.

Volume: is also an important factor that influences the scores.

- For Q10, changing the volume of the target sound(s) should be considered as a distortion.
- For Q11, reducing the volume of non-target sounds should be considered as a valid form of removal.

Distortion: If the quality of the extracted audio sounds better than that in the source audio, it also should be considered as a distortion when answering Q10.

C.2 High quality raters selection in SAM Audio Judge

We design a rater qualification program to ensure the selection of high-quality annotators. Specifically, we curate a 40-sample golden set, with scores labeled by experts and treated as ground truth. We then invite outsourced vendors to annotate this golden set and qualify their annotators using a Bayesian rater modeling approach. This model estimates a posterior distribution over each annotator's confusion matrix, capturing their reliability and bias across rating values. From this posterior, we derive the expected Mean Absolute Deviation (MAD) of each annotator's scores, which naturally accounts for partially completed tasks, item-score uncertainty, and imbalances in score distributions. Annotators with the lowest expected MAD on the overall score are selected as qualified raters. While we do not directly compare against expert annotations, we use them as a spot check to ensure that the selected annotators exhibit reasonable consensus and consistent rating behavior. In the end, we successfully recruit 128 qualified raters, representing diverse subjective perspectives from the general public.

C.3 Gemini-2.5-pro Prompts for Judge Evaluation

You will be evaluating models that perform audio separation.
When using an audio separation model, users provide the model with either an audio track, and descriptions of which parts of the sound they want the model to extract (for example "guitar" or "dog barking"). The model then extracts the part of the audio that the user requests without any other sounds.
The user description of the target audio (e.g. "dog barking" or the visually highlighted dog in the video) is referred to as the prompt. The original video or audio provided to the model is the source video/audio. The portions of the source audio that the user is requesting are known as the target sounds. All other sounds in the source audio not requested by the user are known as non-target sounds. The audio that the model produces from the source audio and the prompt is referred to as the extracted audio.
Task:

- The user provides:
  1. A prompt, which is a text description of the target sound(s) that the user wants the model to extract from the source audio
  2. Two audio files:
     - The source audio (contains the target and other sounds)
     - The extracted audio from the separation model

Please evaluate the extracted audio compared with the mixture and the target description, along the following four dimensions:

- **Recall**: What portion of the target sound(s) is/are present in the extracted audio, regardless if other non-target sounds or distortions are present? Consider portion in terms of (1) duration and (2) number of target sounds.
- **Precision**: For target sounds that are present in the extracted audio, how similar do they sound to how they sounded in the source audio?
- **Faithfulness**: How well do you feel the model removed non-target sounds from the extracted audio?
- **Overall**: Please provide an overall score for the model's performance.

For each dimension, provide a score from **1 (very poor) to 5 (excellent)**:

- **1: Terrible**: Extracted audio is completely incorrect; none of the target audio is in the extracted audio, or, if present, none of the non-target audio is filtered.
- **2: Poor**: Many serious issues with extracted audio: some target audio in the extracted audio but heavily distorted, missing, and many non-target sounds present.
- **3: OK**: Some serious issues with extracted audio: target audio is maybe half present and/or non-target sounds are about half present.
- **4: Good**: Only minor issues with extracted audio (e.g. minor portions of target audio may be missing or slightly distorted, a very small amount of non-target sounds may be present).
- **5: Perfect**: All target sounds are present in their entirety, and sound identical to the way they sound in the source audio. No non-target sounds present in extracted audio.

We also provide 5 reference examples that illustrate what score 1,2,3,4,5 means.

C.4 SAJ for Automatic and Fine-grained Performance analysis

We could use SAJ to automatically sample evaluation cases at different difficulty levels for any target audio concept list, enabling a fully automatic and fine-grained performance analysis pipeline. To support this, we train a text-prompted SAJ model to predict the intrinsic difficulty of a separation task. Because human annotations show that levels 4 and 5 are rare, we merge them into a four-level scale (1–4) and balance the dataset across these levels.

Unlike the standard SAJ model, which predicts separation performance using both mixture and output audio, the difficulty model only takes the mixture audio and text prompt as input, allowing it to estimate task difficulty before running any separation system.

We apply the predicted difficulty levels to the SAJ test set and compute human ratings for the corresponding SAM Audio outputs. As shown in Table 18, human-annotated performance monotonically decreases as difficulty increases: Level 1 cases are easiest, while Level 4 shows clear degradation. This enables automatic curation of evaluation subsets at different difficulty levels, supporting fine-grained, concept-specific robustness analysis.

Difficulty Level	Overall	Recall	Precision	Faithfulness
1	3.716	4.084	3.984	3.963
2	3.327	3.995	3.537	3.798
3	3.272	4.022	3.326	3.807
4	2.894	3.791	3.024	3.528

Table 18 Human-rated separation quality across different task difficulty levels.

D SAM Audio Ablation Study

D.1 Effect of Model Scale

Tables 19 and 20, together with Figure 17, compare models of three sizes—500M, 1B, and 3B parameters. Across the scale comparison, we disable span prediction. Overall, the 3B model achieves the strongest performance across most tasks, though in certain cases such as general SFX separation it performs on par with or slightly below the smaller models.

Scaling model capacity provides the greatest benefit in specialized domains. For instrument separation, for example, SAM Audio-Large achieves substantial gains over the smaller variants: on instrument-in-the-wild separation, it outperforms SAM Audio-Base by 23% NWR and SAM Audio-Small by 20%. These results suggest that larger models better capture the fine-grained acoustic cues required for high-fidelity separation in structured domains such as musical instruments, while smaller models remain competitive for broader sound categories.

	General SFX			Speech			Speaker			Music			Instr(wild)			Instr(pro)		
Model	SAJ	CLAP	OVR	SAJ	CLAP	OVR	SAJ	CLAP	OVR	SAJ	CLAP	OVR	SAJ	CLAP	OVR	SAJ	CLAP	OVR
SAM Audio-Small	**4.25**	0.30	**3.62**	4.55	**0.35**	3.99	3.89	**0.17**	3.12	4.32	**0.28**	4.11	4.27	0.27	3.56	4.78	**0.30**	4.24
SAM Audio-Base	4.23	0.28	3.28	**4.61**	0.33	**4.25**	3.94	0.15	3.57	4.26	0.27	3.87	4.33	0.29	**3.66**	4.78	**0.30**	4.27
SAM Audio-Large	4.11	**0.31**	3.50	4.59	0.33	4.03	**4.08**	**0.17**	**3.60**	**4.30**	**0.28**	**4.22**	**4.45**	**0.30**	**3.66**	**4.83**	0.28	**4.49**

Table 19 Comparison of SAM Audio of different scales in text prompting. –: not applicable. OVR: overall subjective score.

	General SFX		Speaker		Instr (wild)	
Model	IB	OVR	IB	OVR	IB	OVR
SAM Audio-Small	0.24	2.62	0.23	2.79	0.21	2.25
SAM Audio-Base	**0.25**	**2.63**	0.24	**3.25**	0.22	**2.76**
SAM Audio-Large	**0.25**	2.61	**0.24**	2.95	**0.24**	2.58

Table 20 Comparison of SAM Audio of different scales in visual prompting. –: not applicable. OVR: overall subjective score.

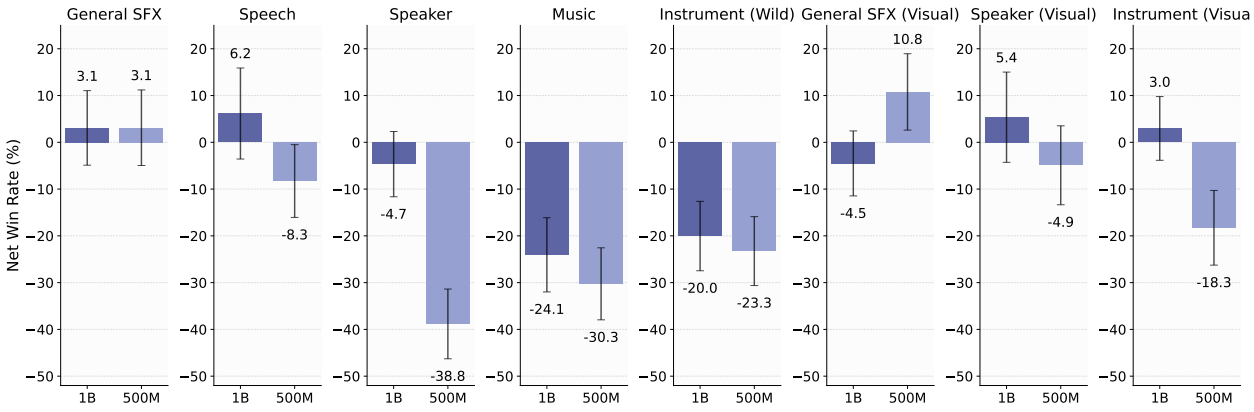

Figure 17 Net Win Rate (%) of SAM Audio-Base and SAM Audio-Small against SAM Audio-Large

D.2 Effect of Auxiliary Loss

Table 21 reports the effect of incorporating the representation alignment loss during SAM Audio pre-training. We compare two 3B pre-trained checkpoints: (1) a baseline model without the auxiliary loss, and (2) a model trained with the auxiliary loss using $\lambda = 1.0$. We evaluate both checkpoints on the general SFX test set under

	General SFX (Text)		General SFX (Visual)
Auxiliary target	SAJ	CLAP	IB
None	2.48	0.24	0.17
Target AED	**3.18**	**0.29**	**0.18**

Table 21 Effect of AED-based representation alignment loss in pre-training

		Text						Visual		
Separation Task	**Stage**	General SFX	Speech	Speaker	Music	Instr(wild)	Instr(pro)	General SFX	Speaker	Instr(wild)
SAJ (↑)	PT	3.93	2.90	3.28	4.14	3.17	3.36	-	-	-
	FT	**4.14**	**4.55**	**4.07**	**4.38**	**4.48**	**4.82**	-	-	-
CLAP (↑)	PT	**0.31**	0.28	**0.23**	**0.31**	0.21	0.15	-	-	-
	FT	0.30	**0.36**	0.21	**0.31**	**0.30**	**0.29**	-	-	-
IB (↑)	PT	-	-	-	-	-	-	**0.24**	0.22	0.20
	FT	-	-	-	-	-	-	**0.24**	**0.24**	**0.22**
AES-PC (↓)	PT	2.57	2.91	2.98	4.53	3.54	4.72	3.09	3.49	3.69
	FT	**2.08**	**1.89**	**1.94**	**4.44**	**3.16**	**3.24**	**2.22**	**2.20**	**3.26**

Table 22 Comparison of pre-trained vs fine-tuned results across audio separation tasks. –: not applicable.

text- and visual-prompted separation, as the pre-trained model alone underperforms on specialized domains such as speech and music.

As shown in Table 21, adding the AED-based alignment objective yields consistent improvements in both settings, with a larger gain in the text-prompted case (over 20% relative improvement in text alignment) compared to visual prompting (∼5% relative improvement). This ablation is performed only during pre-training, as we do not apply the auxiliary loss in fine-tuning. We hypothesize that the benefit arises from the noisier audio target in pre-training corpus, where the alignment loss helps the model learn intermediate semantic representations beneficial for separation.

D.3 Effect of fine-tuning

Table 22 compares the 3B model after pre-training only with the same model further fine-tuned on curated separation data. In addition to standard metrics, we also report the PC score from Audiobox-Aesthetics (Tjandra et al., 2025), which serves as a proxy for audio cleanness.

Overall, fine-tuning leads to consistent gains, though the magnitude varies across tasks. For text-prompted separation, the largest improvements appear in instrument extraction, speech extraction, and speaker separation. These tasks benefit from the availability of high-quality datasets containing professionally recorded stems, providing clean and reliable supervision. By contrast, improvements in general sound event separation and music separation are smaller. Large-scale pre-training already covers a wide distribution of sound events and mixtures, including many with background music, which leaves less room for fine-tuning to provide additional gains.

For visual-prompted separation, the visual alignment score (IB) remain relatively stable for general SFX, mirroring the trend seen in text-prompted separation. Fine-tuning data provide broader coverage for music and speech videos than for general sound events. Large-scale audio–visual pre-training already establishes strong correspondences between visual regions and audio, explaining the smaller incremental benefit for SFX.

Finally, we observe a substantial improvement in audio cleanness across tasks, according to the PC metric. Fine-tuned models consistently produce cleaner separated audios with fewer artifacts, owing to the clean audio targets used during fine-tuning.

D.4 Effect of using pseudo-labeled audio stem data

Table 23 shows a comparison of using or not using pseudo-labeled audio data in the fine-tuning stage. Details of the pseudo-labeled data are summarized in Table 4. The baseline model is trained using only fully real triplets and synthetic mixtures. As shown in Table 23, incorporating pseudo-labeled data yields consistent gains for both text and visual prompting. Notably, the largest improvements are observed in AES-PC for general sound, indicating that pseudo-labeled audio helps the model learn to produce cleaner separation stems.

		Text						Visual		
Separation Task	**Setting**	General SFX	Speech	Speaker	Music	Instr(wild)	Instr(pro)	General SFX	Speaker	Instr(wild)
SAJ (↑)	w/o PL	4.02	4.50	4.06	4.34	4.34	4.80	-	-	-
	PL	**4.14**	**4.55**	**4.07**	**4.38**	**4.48**	**4.82**	-	-	-
CLAP (↑)	w/o PL	**0.30**	0.34	**0.21**	**0.31**	0.29	0.28	-	-	-
	PL	**0.30**	**0.36**	**0.21**	**0.31**	**0.30**	**0.29**	-	-	-
IB (↑)	w/o PL	-	-	-	-	-	-	0.23	0.21	**0.22**
	PL	-	-	-	-	-	-	**0.24**	**0.24**	**0.22**
AES-PC (↓)	w/o PL	2.36	1.95	2.02	**4.38**	3.19	**3.24**	2.28	2.29	**3.25**
	PL	**2.08**	**1.89**	**1.94**	4.44	**3.16**	**3.24**	**2.22**	**2.20**	3.26

Table 23 Effect of using pseudo-labeled audio stem data