VAFlow: Video-to-Audio Generation with Cross-Modality Flow Matching

Xihua Wang[1] Xin Cheng[1] Yuyue Wang[1] Ruihua Song[1*] Yunfeng Wang[2]
[1]Gaoling School of Artificial Intelligence, Renmin University of China [2]ZHI-TECH GROUP
xihuaw@ruc.edu.cn, songruihua_bloon@outlook.com

Abstract

Video-to-audio (V2A) generation aims to synthesize temporally aligned, realistic sounds for silent videos, a critical capability for immersive multimedia applications. Current V2A methods, predominantly based on diffusion or flow models, rely on suboptimal noise-to-audio paradigms that entangle cross-modal mappings with stochastic priors, resulting in inefficient training and convoluted transport paths. We propose VAFlow, a novel flow-based framework that directly models the video-to-audio transformation, eliminating reliance on noise priors. To address modality discrepancies, we employ an alignment variational autoencoder that compresses heterogeneous video features into audio-aligned latent spaces while preserving spatiotemporal semantics. By retaining cross-attention mechanisms between video features and flow blocks, our architecture enables classifier-free guidance within video source-driven generation. Without external data or complex training tricks, VAFlow achieves state-of-the-art performance on VGGSound benchmark, surpassing even text-augmented models in audio fidelity, diversity, and distribution alignment. This work establishes a new paradigm for V2A generation with a direct and effective video-to-audio transformation via flow matching.

1. Introduction

Video-to-audio (V2A) generation plays a vital role in multimedia content generation, including Foley for films and AI-generated silent videos. In recent years, this task has attracted increasing attention within the generative community. Early V2A approaches [15, 24, 26, 29, 34] adopted language modeling strategies by discretizing audio into tokens, and used parallel mask-prediction models (BERT-like) or token-by-token autoregressive models (GPT-like) borrowed from NLP community as shown in Figure 1. However, these methods, reliant on discrete tokens, incur inherent information losses by the discretization process.

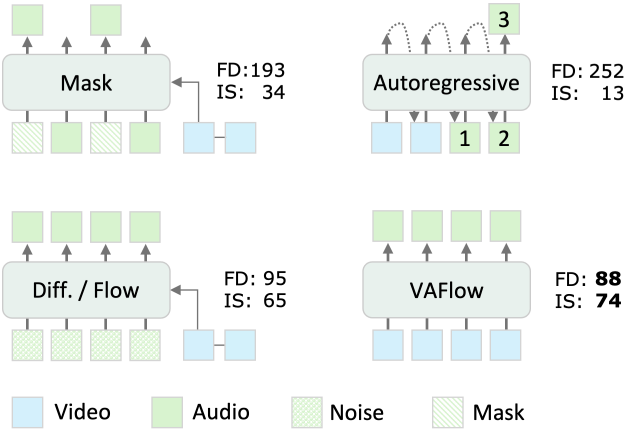

Figure 1. Four different V2A paradigms, along with state-of-the-art performance under each. The first row shows discrete token-based approaches (mask-prediction and autoregressive based generation), which are limited by information loss during tokenization. The second row presents methods generating continuous data. Previous diffusion and flow methods (bottom-left) model the noise-to-audio mapping conditioning on video. We find this approach suboptimal and demonstrate that directly modeling the video-to-audio transformation (bottom-right) yields better results.

Recent advancements in Diffusion and Flow Matching (FM) models in image [4, 10, 33] and video [3, 31] generation have spurred similar approaches for V2A [9, 25, 41, 43, 45]. Generally, these methods model V2A as a conditional generation task, transforming Gaussian noise conditioned on video into audio, as shown in Figure 1 (bottom-left). The generation process typically involves two steps: (1) sampling a random noise from Gaussian prior and (2) using this noise and video condition as input to iteratively denoise the noise into audio. This approach avoids discrete audio tokens, directly fitting the continuous distribution of audio data, and currently holds state-of-the-art results.

However, existing diffusion/FM-based V2A methods focus solely on the second step, i.e., denoising Gaussian noise into audio, assuming the first step is trivial. Even state-of-the-art methods [40, 43] begin with Gaussian noise, but we find that this approach is not optimal. Unlike text-to-

*Corresponding author.

image or text-to-video tasks, where textual conditions are abstract and highly defined by human, the video conditions in V2A are more complex and require both spatial and temporal understanding. The generation goal also demands a high degree of alignment between the generated audio and the video conditions, not just semantically, but also temporally. There are two inherent difficulties: (1) different conditions on the same noise require the model to generate different audio outputs, demanding advanced condition processing [8, 41, 46], and (2) noise variations under the same condition lead to unstable generation results, prompting research into methods for noise control [44], such as finding a "golden noise" for consistent results [47].

In contrast, we propose a new perspective by shifting focus to the first step of the generation process. We question: *If there are better priors than Gaussian noise?* Given that many diffusion and FM models dedicate significant effort to integrating video conditions into the denoising process for better generation performance, we further question: *If V2A can work better when directly transforming the video distribution to audio rather than from noise?* Fortunately, both answers are affirmative.

We introduce VAFlow, a novel framework for V2A generation, which directly denoises from the video space to the audio space. To implement this video-to-audio paradigm, we adopt the FM framework, as it theoretically supports any source distribution. There are three technical challenges for this implementation: (1) aligning source and target distributions: video and audio feature spaces typically have different time and spatial resolutions, yet FM requires the source and target distributions to be aligned (the same shape). (2) uncertainty of the feature type: identifying the optimal video representations/feature types as the source distribution. (3) conditioning mechanism: exploring additional video information integration ways beyond encoding it in the source distribution. In VAFlow, we first employ an alignment VAE to adjust video features to match the audio latent space. It then explores three different video feature types (semantic CLIP [32], spatiotemporal CAVP [25], reconstruction-focused VidTok [38]) to systematically investigate source video information. Experiments show that using video features both as a source distribution and as conditioning information yields the best results for VAFlow.

Moreover, the recent success of large language models (LLMs) [13, 28] has highlighted the effectiveness of scaling laws, and various generative works consequently explore similar scaling properties in each domain [30, 42]. However, existing V2A models have not demonstrated clear scaling trends, where increasing model size does not consistently achieve improvement [8, 43]. In contrast, our experiments show that VAFlow exhibits clear scaling property, with model size increases leading to consistent improvements in generation quality, further confirming its potential.

Without elaborate training tricks or external data (e.g., text-audio pairs), VAFlow achieves state-of-the-art results on standard benchmarks (VGGSound), outperforming even text-augmented models in generation fidelity, diversity, and cross-modality alignment with input video. Playable audios generated with VAFlow can be accessed at the demo page: https://vaflow.github.io/demo.

2. Related Work

2.1. Flow Matching and Diffusion Models

Diffusion models [17, 27, 35] reconstruct data distributions by iteratively denoising Gaussian-sampled latents, governmented by various designed schedulers achieving state-of-the-art performance in multimodal generation tasks. Unlike diffusions, flow-based models [1, 20, 22, 23] establish deterministic mappings between source (e.g., Gaussian) and target distributions through invertible neural transformations, with recent flow matching optimizing vector fields through optimal transport theory, enabling direct learning of straight transport paths between source and target distributions, thereby reducing training instability and inference steps. While both paradigms typically rely on Gaussian priors as their source distribution, existing video-to-audio systems on both paradigms model noise-to-audio generation conditioned on video features. In contrast, our approach directly aligns video latent spaces with target audio distributions via linear transformations under a flow matching framework, bypassing iterative noise modeling, preserving efficiency, and ensuring precise frame-to-frame alignment.

2.2. Video-to-Audio Generation

Framework. Recent advances in generative models have driven significant progress in video-to-audio (V2A) synthesis. Current V2A methods fall into three main paradigms: **(1) Autoregressive (AR)** models [15, 26, 34] discretize audio into token (codec) sequences and utilize video features as prefixes to model unified video-audio sequences. While straightforward, their token-by-token generation process results in quadratic inference time scaling, limiting efficiency for long audio sequences. **(2) Mask-based** generation paradigms [24, 29, 37] also employ discrete tokens but decode audio in parallel by predicting masked tokens. Video features serve as the condition for mask-to-audio prediction. However, both AR and mask-based methods suffer from an inherent upper bound of fidelity due to lossy compression from audio discretization. **(3) Flow/diffusion-based** methods [7–9, 25, 40, 41, 43, 45, 46] leverage continuous representations and currently achieve state-of-the-art V2A performance [8, 40, 43]. These techniques iteratively transform noise into target audio distributions while conditioning on video features to steer the denoising trajectory.

Video Conditional Modeling. Unlike typical conditioned

generation tasks like text-to-image [33], V2A involves fully aligned sequential correspondences between video and audio. Existing methods address this through various strategies: some AR-based methods investigate various attention mechanisms and masking schemes for temporally sequential alignment [26]; some mask-based approaches integrate hybrid ways to fuse video features into the mask-prediction process to synchronize audio generation [29]; and some flow/diffusion-based methods augment their pipelines with auxiliary modules (e.g., loudness or audio layout predictors [41, 46]) to enforce temporal consistency. In contrast, our approach builds on flow-based techniques by discarding the conventional noise-to-audio paradigm and complex video-conditioning designs. Instead, we propose a more intuitive method that directly transforms the video temporal space to the audio temporal space via a flow transport.

3. Method:

3.1. Preliminary: Flow Matching

Let $x_0 \sim p(x_0)$ and $x_1 \sim p(x_1)$ denote samples from source and target distributions, respectively, with c representing an optional conditioning signal. Flow Matching (FM) [20, 23] establishes an inter-distribution transport process governed by the ordinary differential equation (ODE):

$$dx(t) = v_\theta(x_t, t, c)dt, \tag{1}$$

where $t \in [0, 1]$ denotes the continuous time step, and v_θ represents a neural network-parameterized velocity field. The condition c is incorporated for conditional flow implementations. Optimal transport flows enforce velocity alignment with the direction $(x_1 - x_0)$ of the linear path pointing from x_0 to x_1 through regression:

$$\mathcal{L}(\theta) = \mathbb{E}_{t,x_0,x_1,c}\|(x_1 - x_0) - v_\theta(x_t, t, c)\|^2, \tag{2}$$

where $x_t = (1-t)x_0 + tx_1$ constitutes the linear interpolation between x_0 and x_1. The learned velocity field enables target generation by solving the ODE in Equation 1.

Previous video-to-audio flow approaches [8, 43] (and related diffusion methods [25, 41]) use a Gaussian prior $p(x_0) = \mathcal{N}(0, 1)$ for the source distribution, with audio as $p(x_1)$ and video as conditioning input c. During training, random noise x_0 is paired with ground-truth audio x_1, prompting v_θ to model the paths from noise to audio under visual conditions.

This work challenges the Gaussian prior by proposing a direct video-to-audio transport. Theoretically, FM permits arbitrary source distributions, motivating our formulation:

$$dx(t) = v_\theta(x_t, t)dt, \tag{3}$$

where $x_t = (1 - t)x_v + tx_a$ linearly interpolates between paired video and audio samples $x_v, x_a \sim p(v, a)$.

3.2. VAFlow Architecture

Model Overview. As illustrated in Figure 2, the input video features x_v are extracted via a visual encoder, while the target audio waveform can be obtained from latent representations x_a through WaveVAE. VAFlow attempts to establish FM between the video feature space x_v and audio latent space x_a, enabling direct audio generation from video features via ODE solving instead of Gaussian noise initialization. However, since FM requires consistent dimensional shapes between source and target distributions (see Equation 3), the inherent spatiotemporal resolution mismatch between x_v and x_a necessitates shape alignment. To address this, we employ a 1D VAE (namely alignment VAE) that compresses interpolated x_v into aligned features x_v^{AE} matching x_a's dimensions.

A diffusion transformer (DiT) subsequently works as a velocity field estimator to capture the transport dynamics between x_v^{AE} and x_a. This enables continuous trajectory sampling from x_v^{AE} to x_a through ODE integration. The sampled latent $\hat{x}_a$ is finally decoded by WaveVAE to produce the output audio. The following contents detail: the visual/audio representations (x_v, x_v^{AE}, x_a), and the velocity field estimator in VAFlow.

Video and Audio Representations. For an input video-audio pair (v, a), the audio waveform a is compressed by a pretrained 1D WaveVAE's [11] encoder into $x_a \in \mathbb{R}^{T_a \times D_a}$. This latent x_a serves as the target for flow generation and is later decoded to reconstruct the waveform.

For video representation, a pretrained visual encoder encodes v into $x_v \in \mathbb{R}^{T_v \times D_v}$ (e.g., CLIP [32], CAVP [25], or VidTok [38] as encoder). Given the discrepancies (e.g., $x_a \in \mathbb{R}^{215\times64}$ versus $x_v \in \mathbb{R}^{100\times768}$ for a 10-second clip), we propose an alignment VAE ϵ_ϕ to further compress and align video latents with audio latents' space while preserving essential visual information. Specifically, we utilize a 1D convolutional encoder-decoder as ϵ_ϕ to interpolate and project x_v into a variational space $x_v^{\text{AE}} \in \mathbb{R}^{T_a \times D_a}$, matching the shape of x_a. The encoder of ϵ_ϕ predicts the parameters $\mu_{x_v^{\text{AE}}}, \sigma_{x_v^{\text{AE}}} = E_{\epsilon_\phi}(x_v)$ for a Gaussian distribution, from which $x_v^{\text{AE}} \sim \mathcal{N}(\mu_{x_v^{\text{AE}}}, \sigma_{x_v^{\text{AE}}})$ is sampled. The decoder reconstructs x_v as $\hat{x}_v = D_{\epsilon_\phi}(x_v^{\text{AE}})$. This process ensures dimensional alignment with audio latents while retaining critical spatiotemporal features of the original video representation.

Velocity Estimator. Built upon the diffusion transformer (DiT) architecture [11], the velocity estimator processes two key inputs: time embeddings of t and the interpolated state $x_t = (1 - t)x_v^{\text{AE}} + tx_a$, following the Equation 3. The time embedding is concatenated with the audio sequence to form an input tensor $\in \mathbb{R}^{(T_a+1)\times D_a}$, which is processed through stacked transformer blocks. The first T_a dimensions of the output sequence constitute the pre-

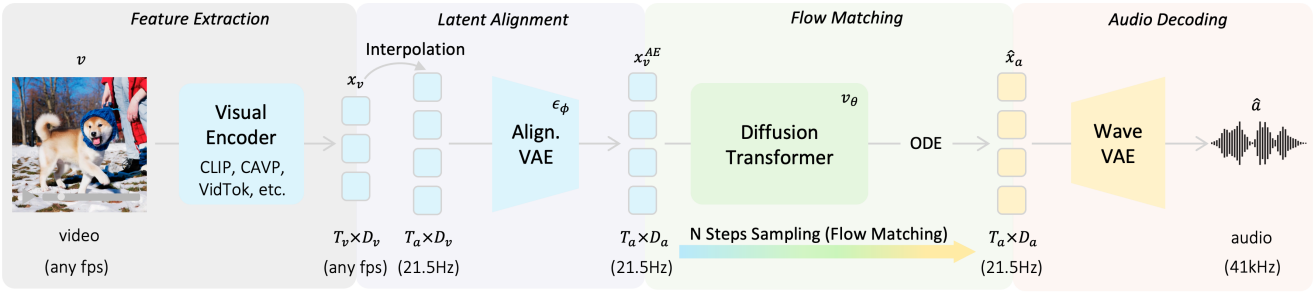

Figure 2. VAFlow Architecture. The figure illustrates VAFlow in the inference manner. A visual encoder initially extracts temporal features $x_v \in \mathbb{R}^{T_v \times D_v}$ from variable-frame-rate video inputs. Then, a 1D alignment VAE (Variational Autoencoder) ϵ_ϕ compresses interpolated x_v into audio-aligned latent features x_v^{AE} with matched dimensions ($T_v^{\text{AE}} \times D_v^{\text{AE}} = T_a \times D_a$). Next, a diffusion transformer (v_θ) predicts the velocity field for the transport $(x_a - x_v^{\text{AE}})$, and by solving the corresponding ODE (see Equation 3), an estimated $\hat{x}_a$ is obtained. Finally, the WaveVAE decoder converts $\hat{x}_a$ into the waveform representing the predicted audio. Note that during diffusion transformer prediction, x_v can optionally be incorporated via cross-attention, detail discussion is conducted in the experiment Section 4.5.

dicted velocity $v_\theta(x_t, t)$, with 1D Rotary Position Embeddings (RoPE [36]) to preserve positional awareness in attention layers. Notably, while the interpolated state x_t implicitly encodes visual information through x_v^{AE}, this signal diminishes as $t \to 1$. To address this limitation, we retain cross-attention layers in the transformer blocks that persistently integrate the original video features x_v. Further studies (Section 4.5) demonstrate that this architectural choice is critical to enhance visual perception throughout the flow trajectory and improve audio generation quality.

3.3. VAFlow Training and Inference

We conduct three-stage training: (1) pretrain the alignment VAE, (2) train the VAFlow estimator with frozen alignment VAE, and (3) jointly fine-tune both alignment VAE and estimator. We detail training and inference as follows:

Alignment VAE Training. Given encoded visual features $x_v \in \mathbb{R}^{T_v \times D_v}$, the alignment VAE encoder predicts distribution parameters $(\mu_{x_v^{\text{AE}}}, \sigma_{x_v^{\text{AE}}})$, while the decoder reconstructs $\hat{x}_v$ from a sampled $x_v^{\text{AE}} \sim \mathcal{N}(\mu_{x_v^{\text{AE}}}, \sigma_{x_v^{\text{AE}}})$. The model optimizes three objectives to preserve visual semantics while regularizing x_v^{AE} within $\mathbb{R}^{T_a \times D_a}$ Gaussian space:

- *KL Regularization*: enforces Gaussian latent priors via:

$$\mathcal{L}_{\text{KL}} = D_{\text{KL}}\left(\mathcal{N}(\mu_{x_v^{\text{AE}}}, \sigma_{x_v^{\text{AE}}}) \parallel \mathcal{N}(0, I)\right). \quad (4)$$

- *Reconstruction Loss*: ensures feature-level consistency:

$$\mathcal{L}_{\text{rec}} = \|x_v - \hat{x}_v\|_2^2. \quad (5)$$

- *Contrastive Alignment*: strengthens feature preservation:

$$\mathcal{L}_{\text{con}} = -\sum_{s \in \mathcal{S}} \log \frac{\exp\left(f_s(\hat{x}_v) \cdot f_s(x_v)/\tau\right)}{\sum_{x' \in \mathcal{B}} \exp\left(f_s(\hat{x}_v) \cdot f_s(x')/\tau\right)}, \quad (6)$$

where $\mathcal{S} = \{\text{frame, clip, global}\}$ defines temporal granularities and $\tau = 0.7$ serves as a temperature parameter [6]. For scale s, $f_s(\cdot)$ extracts features via three ways: frame-level: single-frame sampling + mean pooling; clip-level: $K = 8$-adjacent-frame sampling + pooling; global-level: full temporal mean pooling at T_v dimension of feature x. Negative samples x' are drawn from training batch $\mathcal{B}$.

The composite objective integrates these terms:

$$\mathcal{L}_{\text{VAE}} = \lambda_{\text{KL}}\mathcal{L}_{\text{KL}} + \lambda_{\text{rec}}\mathcal{L}_{\text{rec}} + \lambda_{\text{con}}\mathcal{L}_{\text{con}}, \quad (7)$$

with $\lambda_{\text{KL}} = 8 \times 10^{-4}$, $\lambda_{\text{rec}} = 1.0$, and $\lambda_{\text{con}} = 1.0$.

Velocity Estimator Training Given encoded features x_v, x_v^{AE}, and x_a, we train the velocity estimator v_θ to predict the linear transport direction $x_a - x_v^{\text{AE}}$ through:

$$\mathcal{L}(\theta) = \mathbb{E}_{t \sim \mathcal{U}(0,1)} \left\| (x_a - x_v^{\text{AE}}) - v_\theta(x_t, t, x_v) \right\|_2^2, \quad (8)$$

where $x_t = (1-t)x_v^{\text{AE}} + t x_a$. For the conditioned version, visual features x_v are incorporated via cross-attention (with a 10% probability of zeroing out x_v during training). The vanilla version omits this conditioning, simplifying the input of estimator in Equation 8 to $v_\theta(x_t, t)$.

Joint Turning After separate training of the alignment VAE and the velocity estimator, a joint optimization stage is performed. In this stage, the VAE encoder and v_θ are co-trained (discarding the VAE decoder), while the WaveVAE and visual feature extractor remain frozen. For a pair (v, a), the visual extractor and WaveVAE produce x_v and x_a. The VAE encoder compresses x_v to obtain $x_v^{\text{AE}} \sim q_\phi(x_v^{\text{AE}}|x_v)$, which together with x_a and a random $t \sim \mathcal{U}(0,1)$ forms x_t. The joint objective integrates KL regularization in Equation 4 with velocity prediction in Equation 8:

$$\mathcal{L}(\theta, \phi_{\text{Enc}}) = \lambda_{\text{KL}}\mathcal{L}_{\text{KL}} + \mathbb{E}\|(x_a - x_v^{\text{AE}}) - v_\theta(x_t, t, x_v)\|_2^2. \quad (9)$$

Inference Upon training, VAFlow synthesizes audio a from an input video v through the following pipeline: The visual extractor first encodes v into a feature x_v, which is

compressed by the alignment VAE encoder into distribution parameters $(\mu_{x_v^{\text{AE}}}, \sigma_{x_v^{\text{AE}}})$. A latent sample $x_v^{\text{AE}} \sim \mathcal{N}(\mu_{x_v^{\text{AE}}}, \sigma^2_{x_v^{\text{AE}}})$ is drawn as the initial state (x_0) for solving the ODE in Equation 1 with learned estimator. The predicted $\hat{x}_a$ (x_1) is then decoded into waveform via the frozen WaveVAE. During inference, the variational sampling of x_v^{AE} introduces stochasticity – even with fixed ODE integration paths, distinct samples yield diverse outputs.

For conditional velocity estimators, classifier-free guidance (CFG) [14] enhances synthesis quality by interpolating conditioned and unconditioned velocity predictions:

$$v_\theta^{\text{CFG}}(x_t, t, x_v) = (1+\gamma)v_\theta(x_t, t, x_v) - \gamma v_\theta(x_t, t, \emptyset), \tag{10}$$

where γ controls guidance strength and $\emptyset$ denotes null (zero) conditioning.

4. Experiments

4.1. Experimental Setup

Datasets. We employ two datasets: VGGSound [5] and AudioSet-V2A [25]. For VGGSound, we adhere to the original split, which comprises 186k training samples and 15k testing samples. AudioSet-V2A contains 390k videos that are exclusively used for training. During training, the alignment VAE is trained on a combined set of VGGSound training and AudioSet-V2A to yield more generalized visual feature compression. The subsequent velocity estimator training and joint tuning are conducted solely on the VGGSound training set. At inference, we evaluate on the VGGSound test set as in prior work.

Implementation Details. For the estimator's DiT architecture, we adapted the DiT implementation from Stable-Audio-Open-1.0 [11] (SDA1.0) by eliminating duration encoding, applying global time encoding solely to time steps, and restricting the cross-attention layer's key/value inputs to video features. For the visual extractor, we use three variants: CLIP [32] (a widely adopted visual semantic extractor in V2A tasks), CAVP [25] (which captures both visual semantic and temporal features), and VidTok [38] (a visual tokenizer for reconstruction). We train three alignment VAEs with comparable parameter scales—one for each visual extractor. In the velocity estimator training stage, the alignment VAE remains frozen; in the joint training stage, both the estimator and the alignment VAE are trained. In all stages, the audio WaveVAE and the visual extractor are kept frozen. Notably, we do not perform any further rectified iterations on the estimator; all VAFlow variants are trained for a single rectified iteration (Rectified-1, RF-1). During sampling, we employ ODE solvers of various orders—first-order Euler, second-order Midpoint, and fifth-order dopri5—with differing sampling steps and classifier-free guidance scales.

Metrics. We employ both objective and subjective evaluations across three dimensions: generation quality, audiovisual relevance, and audiovisual synchronization. For objective evaluation, generation quality is measured using FD score [21], KL divergence [15], and IS score [21]. In previous work, each metric may be computed with different feature extraction models (e.g., Melception, PasST [1]). Audiovisual relevance is quantified by computing the cosine similarity between the embeddings of the input video and the generated audio using the ImageBind model [12]. For synchronization, we use the synchronization accuracy (Acc.) [25] reflects the alignment between the input video and generated audio.

Baselines. Our method is comprehensively compared against all major generative paradigms, including autoregressive (AR), masked prediction (Mask), diffusion-based, and flow-based approaches, as detailed in Table 1. All baseline models were evaluated using either officially released code and models or directly downloaded audio outputs by their authors, with consistent experimental settings maintained across all evaluations on the same hardware platform. Implementation specifics include: (1) Im2Wav generates audio in 4s clips, truncated to 10s after three clips. (2) Diff-Foley here employs its double-guidance variant (demonstrating superior performance); (3) MMAudio-L here is the largest variant of MMAudio; and (4) Frieren here employs its final no-reflow configuration (achieving optimal Fréchet distance). For comprehensive evaluation, we also include text-augmented V2A models (denoted in gray rows) that leverage external textual data beyond benchmark audiovisual corpora during training or inference.

4.2. Diffusion vs. Flow vs. VAFlow Models

We examine framework-specific performance by training same VAFlow-CLIP-142M models under three paradigms, maintaining identical configurations: CLIP visual features, architecture parameters, initialization weights, and other hyperparameters. The frameworks include: ***Diffusion***: Standard DDPM [27, 33] with cross-attention video conditioning. ***Flow***: Standard continuous-time flow (Equation 2) using Gaussian prior. ***VAFlow***: Our framework employing aligned VAE-compressed video features as source distribution. All frameworks use their respective optimal inference settings (DDPM: DPMSolver; Flows: dopri5). Joint tuning in VAFlow begins at 100K steps.

Fréchet Distance (FD) results in Figure 3 reveals: 1) Flow paradigms converge faster with lower FD than diffusion, demonstrating inherent optimization benefits. 2) VAFlow attains lower final FD than standard flow despite delayed early-stage convergence (<100K steps). This per-

[1]We adopt PasST [19] model over PANN [18] for its improved robustness following the practice introduced in AudioLDM [21] evaluation repository: https://github.com/haoheliu/audioldm_eval.

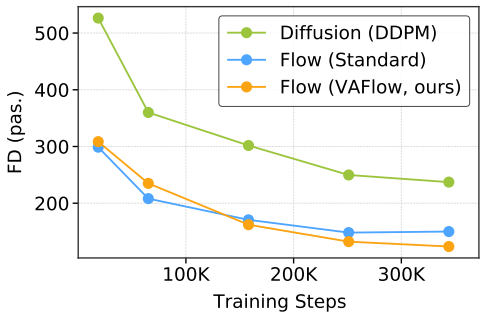

Figure 3. Comparison of different training frameworks. We examine the same model under different training paradigms by controlling training data, steps, and other hyperparameters. Results demonstrate that VAFlow converges to superior optima, validating the effectiveness of its direct video-to-audio transformation design and highlighting this paradigm advantages in V2A generation .

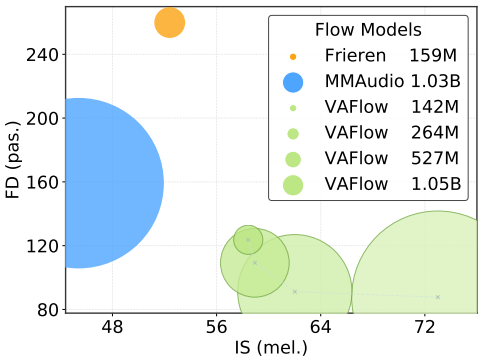

Figure 4. Performance of flow-based baselines and VAFlow variants in scale. VAFlow models (142M, 1.05B) consistently outperform comparable-scale baselines (Frieren-159M, MMAudio-1.03B), with stable scaling property indicated by progressive IS gains (diversity) and saturating FD improvements (quality).

formance attributes to VAFlow's explicit modeling of deterministic video-audio transport paths, avoiding the ambiguous noise-to-audio mappings in Gaussian-prior approaches. VAFlow learns optimal transport trajectories between video-audio distribution, bypassing error-prone intermediate transformations in Gaussian-noise-to-audio paths.

4.3. VAFlow Scaling Analysis

We investigate VAFlow's scaling capabilities by developing four model variants (142M, 264M, 527M, 1.05B parameters) through width/depth scaling of DiT blocks while maintaining fixed training protocols (data, steps, etc.). Evaluation metrics (FD, IS) versus parameter counts are visualized in Figure 4 (bubble sizes denote model scales).

There are two key findings here: 1) *Parameter Efficiency*: VAFlow consistently outperforms flow-based baselines (Frieren, MMAudio) at comparable sizes (e.g., 142M vs. 159M, 1.05B vs. 1.03B) across both metrics. 2) *Scaling Capability*: Previous flow-based studies have struggled to show consistent scalability [8, 43], where larger models fail to achieve better metrics consistently. To the best of our knowledge, VAFlow is the first flow-based V2A framework to demonstrate stable scaling properties. As the parameter size increases, the models achieve better FD and IS scores. Interestingly, larger models tend to improve IS scores more significantly than FD scores, suggesting that as the VAFlow model scales, the quality of generated audio gradually saturates, and the model becomes more capable of deriving diverse audio from different video-sampled latents.

4.4. Benchmark Results

Quantitative evaluations are presented in Table 1, where VAFlow demonstrates state-of-the-art performance across key audio generation metrics: $\text{FD}_{pas.}$, $\text{FD}_{mel.}$, $\text{IS}_{pas.}$, $\text{KL}_{pas.}$, and $\text{KL}_{mel.}$. For audio-visual synchronization, VAFlow achieves 96.3% accuracy (Acc.), comparable to Frieren (97.0%). In semantic relevance, it attains an IB-VA score of 28.6, second only to TiVA's 30.3. Notably, VAFlow maintains a simple and straightforward model design without relying on complex video conditioning modules [8, 41, 46], carefully designed noise schedules [9], or additional guidance modules [25, 45]. Its performance across various metrics, achieved with a standard Flow Matching training approach and basic ODE solvers, underscores the simplicity and effectiveness of the direct video-to-audio mapping design. When compared to text-enhanced baselines (gray rows), VAFlow maintains advantages in $\text{FD}_{pas.}$, $\text{FD}_{mel.}$, $\text{IS}_{pas.}$, and synchronization Acc., though text-augmented models exhibit superior audio semantic metrics (KL distances and IB-VA). This aligns with expectations since text conditioning explicitly constrains semantic space. We note that Seeing&Hearing's elevated IB-VA score (36.7) stems from classifier guidance using the same ImageBind model, which however compromises generation quality and synchronization performance.

We further conduct human evaluation with sampled 50 videos from the test set to compare our method with the flow-based baseline, Frieren. The generated results from both models were randomized, and 10 experts were invited to score the models based on three criteria: sound quality (Quality), audio-visual semantic consistency (Semantic) and audio-visual synchronization (Sync.). The scoring was done by selecting a winner or declaring a tie for each comparison. The results, summarized in Table 2, show that our model outperforms the baseline in all three dimensions.

4.5. Ablative Analysis

Visual Features. Three visual feature types were experimented with: *semantic features* (CLIP embeddings extracted per frame at 10 fps), *temporally-enhanced semantic*

Paradigm	Method	Vis. Feat.	Quality					Sync.	Semantic
			$FD_{pas.}$ ↓	$FD_{mel.}$ ↓	$IS_{mel.}$ ↑	$KL_{pas.}$ ↓	$KL_{mel.}$ ↓	Acc. ↑	IB-VA ↑
AR	SpecVQGAN [15]	RGB,Opti.	342.4	18.15	20.5	3.10	6.84	55.9	14.1
	Im2Wav [34]	CLIP	252.2	13.03	35.8	2.26	5.41	78.9	19.6
Mask	VATT [24]	eva-CLIP+	154.1	7.07	64.3	1.34	3.46	84.8	25.0
	VAB [37]	eva-CLIP	193.1	12.21	34.4	2.31	5.09	82.4	25.9
Diffusion	MultiFoley [7]	CAVP	241.6	13.00	67.7	1.59	3.34	85.2	27.0
	Seeing&Hearing [45]	ImageBind+	251.5	12.6	30.1	2.65	6.37	54.2	36.7
	Diff-Foley [25]	CAVP	512.1	11.45	50.4	2.91	6.18	88.0	20.7
	LoVA [9]	CLIP	149.1	6.90	58.7	2.10	5.12	87.0	26.3
	FoleyCrafter [46]	CLIP	134.2	7.40	56.6	2.29	5.65	83.5	27.8
	TiVA [41]	CLIP	111.9	4.31	67.0	2.02	5.21	87.5	**30.3**
	V2A-Mapper [40]	CLIP	94.9	4.21	64.9	2.46	6.06	78.6	22.2
Flow	MMAudio-L [8]	CLIP+	159.2	9.38	45.4	1.95	3.84	77.5	33.9
	Frieren [43]	CAVP	259.9	6.70	52.4	2.92	5.64	**97.0**	23.0
	VAFlow (ours)	CLIP	**87.7**	**3.86**	**73.6**	**1.91**	**4.81**	88.1	28.6
	VAFlow (ours)	CAVP	91.8	4.73	64.8	2.41	5.96	96.3	25.1
	VAFlow (ours)	VidTok	128.5	6.17	48.1	3.64	8.94	49.9	12.9

Table 1. Quantitative video-to-audio results on the VGGSound test set. "Vis. Feat." indicates the employed visual features, with a "+" denoting the inclusion of auxiliary visual information (e.g., visual captions via LLama-2 [39] or Qwen-VL [2] in VATT and Seeing&Hearing, or SyncFormer [16] features in MMAudio). All results are obtained from official code or released audio outputs. The MultiFoley results here is an 8k subset filtered by ImageBind from the authors. Rows in gray denote methods that incorporate extra text data (e.g., video captions); for fairness, these results are provided for reference only and are not directly compared with standard video-to-audio models. The best score is **bolded**, and the second-best is underlined. Our three VAFlow variants , differing in visual features, achieve definitive gains in audio quality (FD, IS, KL) while delivering near state-of-the-art performance for synchronization (Acc.) and correlation (IB-VA).

Comparison	Win rate(%)		
	Quality	Sync.	Semantic
Ours vs. Frieren	**66.00**$_{(\pm 4.18)}$	**60.44**$_{(\pm 5.72)}$	**60.89**$_{(\pm 4.48)}$

Table 2. Human evaluation results, with comparison between our model and the flow-based baseline Frieren across three criteria: sound quality, audio-visual synchronization, and audio-visual semantic consistency. Ours outperforms the baseline in all criteria.

features (CAVP features via video-audio contrastive learning at 4 fps), *reconstruction features* (VidTok tokenized latents at 15 fps). Utilizing the same model size and training process for these features, we developed three variants: VAFlow-CLIP, VAFlow-CAVP, and VAFlow-VidTok, with results presented in the last three rows of Table 1. The results indicate that the widely-used CLIP features overall perform the best, yielding superior audio quality and semantic relevance. The temporally enhanced CAVP features, on the other hand, demonstrate the best video-audio synchronization but slightly lag in other metrics compared to VAFlow-CLIP. Models based on reconstruction features exhibit the poorest performance across all evaluated aspects. These outcomes suggest two key insights: 1) The type of video features directly influences VAFlow's performance, with different feature types leading to models with distinct performance preferences (either better semantic performance or synchronization); 2) Modality alignment matters more than information preservation. Although reconstruction features retain the most comprehensive video information among the three, their lack of modality alignment hampers the V2A model's ability to learn effective cross-modal generative capabilities.

Flow Matching Source Distributions. We evaluate VAFlow under two source distributions: video latent priors from alignment VAE and standard Gaussian noise. For video prior, we further test variants with/without joint training of the alignment VAE encoder (Method 3.3). All models share same training budgets (300K steps for non-joint training variants; 170K+130K for joint training).

Results in Table 3 demonstrate that video latent priors trained solely under alignment VAE objective in Equation 7 fail to surpass Gaussian noise baselines. Only when coupled with joint training—adapting the prior to flow-matching tasks—do video latent priors outperform Gaussian prior. This highlights the necessity of co-optimizing prior distributions for flow-based generation rather than relying on standalone pretraining on reconstruction task.

Conditioning Mechanisms and Guidance. Prior work highlights the critical role of condition and guidance mech-

V-Prior	Tuned	$FD_{pas.}$ ↓	$IS_{mel.}$ ↑	Acc. ↑	IB-VA ↑
✗	-	102.6	67.6	87.9	27.9
✓	✗	111.8	57.4	82.3	25.4
✓	✓	**87.7**	**73.6**	**88.1**	**28.6**

Table 3. Ablation study on flow-matching source distributions. 'V-Prior' denotes sampling starts from video latent priors (vs. Gaussian priors). 'Tuned' indicates joint optimization of the alignment VAE encoder. Results show that jointly tuned video priors outperform Gaussian priors as source distributions for V2A flow models.

Cond.	CA	CFG	$FD_{pas.}$ ↓	$IS_{mel.}$ ↑	Acc. ↑	IB-VA ↑
✗	✗	-	135.9	41.4	31.7	8.5
✗	✓	✗	160.5	36.5	18.9	5.4
✓	✓	✗	133.8	42.3	78.0	21.0
✓	✓	✓	**87.7**	**73.6**	**88.1**	**28.6**

Table 4. Ablation study on conditioning mechanisms and guidance. 'Cond.' denotes explicit video conditioning during inference; 'CA' indicates the presence of cross-attention layers of different variants; 'CFG' refers to classifier-free guidance. Results demonstrate that optimal performance requires both explicit conditioning and CFG when solving VAFlow ODE.

anisms (e.g., classifier-free guidance, CFG) in enhancing model performance. We address two questions:

(1) *Can VAFlow generate high-quality audio using only video priors, omitting explicit video conditioning during sampling?* We experimented with two variants without explicit condition: *No cross-attention (CA)*: Remove CA layers in DiT blocks and retrain the model (Table 4, Row 1). *Zero-input CA*: Retain CA layers but zero out video inputs during inference (Table 4, Row 2). 'No CA' variant exhibit comparable audio quality metrics (FD, IS) to the video-condition-visible variant (Table 4, Row 3: explicit video conditioning without CFG), yet suffer significant degradation in semantic relevance (IB-VA) and synchronization (Acc.). This indicates that while video priors partially preserve visual information, explicit video conditioning during sampling is essential for temporal and semantic alignment. We think that as the latent trajectory approaches the target audio distribution ($t \rightarrow 1$), initial visual cues in the prior degrade, necessitating direct video information.

(2) *Does explicit video conditioning during sampling still require CFG for optimal performance?* Adding CFG to the video-condition-visible variant (Row 4 in Table 4) further improves all metrics, demonstrating VAFlow's compatibility with established guidance techniques. This underscores the dual necessity of explicit video conditioning and CFG for optimal V2A synthesis with VAFlow framework.

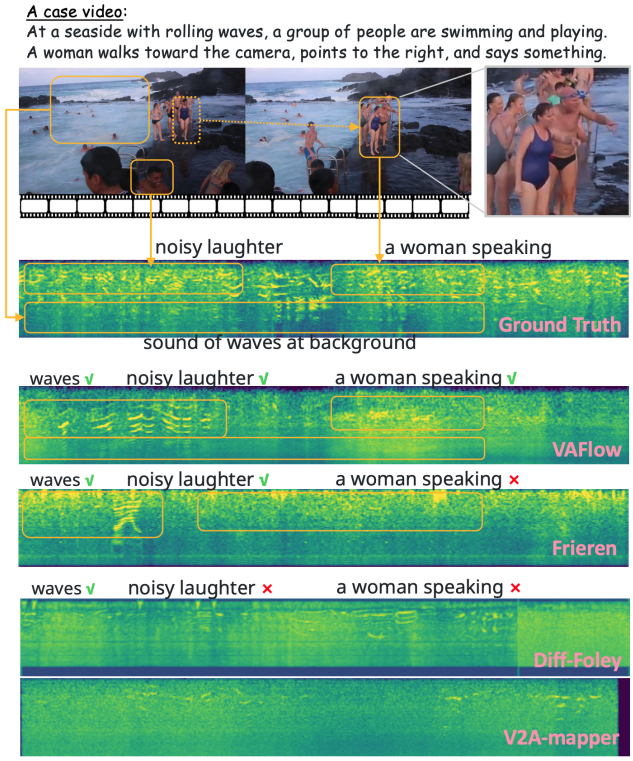

Figure 5. Mel-spectrograms of audio generated by VAFlow and the baseline for a complex seaside scene, including background waves and varying voices, demonstrating VAFlow's superior content understanding and synchronization with the video.

4.6. Case Studies

Figure 5 illustrates the performance of VAFlow and the baselines on the complex scene video. The video depicts a seaside scene with various sounds include background waves and different voices associated with shifting characters (noisy laughter, woman speaking). We present the ground truth and the mel-spectrograms of the audio generated by different methods. Our results show that VAFlow not only accurately understands the video content and generates all the necessary sounds but also maintains synchronization with the visual timing.

5. Conclusion and Future Work

In this paper, we introduce VAFlow, a novel framework for V2A task, which directly denoises from the video space to the audio space via flow matching. demonstrating superior performance compared to previous methods. Future work will explore VAFlow's potential in more diverse audio domains, such as speech and music. Expanding the range of tasks and datasets will enable VAFlow's scaling properties to shine even further, laying a more general-purpose the foundation model for content generation community.

Acknowledgement

This work is supported by the National Natural Science Foundation of China (No. 62276268), Public Computing Cloud, Renmin University of China and ZHI-TECH GROUP.

References

[1] Michael S. Albergo and Eric Vanden-Eijnden. Building normalizing flows with stochastic interpolants. In *ICLR*, 2023. 2

[2] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond. *arXiv preprint arXiv:2308.12966*, 2023. 7

[3] Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, Clarence Ng, Ricky Wang, and Aditya Ramesh. Video generation models as world simulators, 2024. 1

[4] Li-Wen Chang, Wenlei Bao, Qi Hou, Chengquan Jiang, Ningxin Zheng, Yinmin Zhong, Xuanrun Zhang, Zuquan Song, Chengji Yao, Ziheng Jiang, et al. Flux: fast software-based communication overlap on gpus through kernel fusion. *arXiv preprint arXiv:2406.06858*, 2024. 1

[5] Honglie Chen, Weidi Xie, Andrea Vedaldi, and Andrew Zisserman. Vggsound: A large-scale audio-visual dataset. In *ICASSP*, pages 721–725, 2020. 5

[6] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning*, pages 1597–1607, 2020. 4

[7] Ziyang Chen, Prem Seetharaman, Bryan Russell, Oriol Nieto, David Bourgin, Andrew Owens, and Justin Salamon. Video-guided foley sound generation with multimodal controls. *arXiv preprint arXiv:2411.17698*, 2024. 2, 7

[8] Ho Kei Cheng, Masato Ishii, Akio Hayakawa, Takashi Shibuya, Alexander Schwing, and Yuki Mitsufuji. Taming multimodal joint training for high-quality video-to-audio synthesis. *arXiv preprint arXiv:2412.15322*, 2024. 2, 3, 6, 7

[9] Xin Cheng, Xihua Wang, Yihan Wu, Yuyue Wang, and Ruihua Song. Lova: Long-form video-to-audio generation. *arXiv preprint arXiv:2409.15157*, 2024. 1, 2, 6, 7

[10] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *International Conference on Machine Learning*, pages 12606–12633, 2024. 1

[11] Zach Evans, Julian D Parker, CJ Carr, Zack Zukowski, Josiah Taylor, and Jordi Pons. Stable audio open. *arXiv preprint arXiv:2407.14358*, 2024. 3, 5

[12] 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 *CVPR*, pages 15180–15190, 2023. 5

[13] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024. 2

[14] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 5

[15] Vladimir iashin and Esa Rahtu. Taming visually guided sound generation. In *BMVC*, 2021. 1, 2, 5, 7

[16] Vladimir Iashin, Weidi Xie, Esa Rahtu, and Andrew Zisserman. Synchformer: Efficient synchronization from sparse cues. In *ICASSP*, pages 5325–5329, 2024. 7

[17] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. In *Advances in Neural Information Processing Systems*, pages 26565–26577, 2022. 2

[18] 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. 5

[19] Khaled Koutini, Jan Schlüter, Hamid Eghbal-Zadeh, and Gerhard Widmer. Efficient training of audio transformers with patchout. *arXiv preprint arXiv:2110.05069*, 2021. 5

[20] Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *ICLR*, 2023. 2, 3

[21] Haohe Liu, Zehua Chen, Yi Yuan, Xinhao Mei, Xubo Liu, Danilo Mandic, Wenwu Wang, and Mark D Plumbley. Audioldm: Text-to-audio generation with latent diffusion models. In *International Conference on Machine Learning*, pages 21450–21474, 2023. 5

[22] Qihao Liu, Xi Yin, Alan Yuille, Andrew Brown, and Mannat Singh. Flowing from words to pixels: A noise-free framework for cross-modality evolution. In *CVPR*, pages 2755–2765, 2025. 2

[23] Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *ICLR*, 2023. 2, 3

[24] Xiulong Liu, Kun Su, and Eli Shlizerman. Tell what you hear from what you see - video to audio generation through text. In *Advances in Neural Information Processing Systems*, pages 101337–101366, 2024. 1, 2, 7

[25] Simian Luo, Chuanhao Yan, Chenxu Hu, and Hang Zhao. Diff-foley: Synchronized video-to-audio synthesis with latent diffusion models. In *Advances in Neural Information Processing Systems*, pages 48855–48876, 2023. 1, 2, 3, 5, 6, 7

[26] Xinhao Mei, Varun Nagaraja, Gael Le Lan, Zhaoheng Ni, Ernie Chang, Yangyang Shi, and Vikas Chandra. Foleygen: Visually-guided audio generation. In *2024 IEEE 34th International Workshop on Machine Learning for Signal Processing*, pages 1–6, 2024. 1, 2, 3

[27] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, pages 8162–8171, 2021. 2, 5

[28] Shen Nie, Fengqi Zhu, Chao Du, Tianyu Pang, Qian Liu, Guangtao Zeng, Min Lin, and Chongxuan Li. Scaling up masked diffusion models on text. *arXiv preprint arXiv:2410.18514*, 2024. 2

[29] Santiago Pascual, Chunghsin Yeh, Ioannis Tsiamas, and Joan Serrà. Masked generative video-to-audio transformers with enhanced synchronicity. In *ECCV*, pages 247–264, 2024. 1, 2, 3

[30] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *ICCV*, pages 4195–4205, 2023. 2

[31] Adam Polyak, Amit Zohar, Andrew Brown, Andros Tjandra, Animesh Sinha, Ann Lee, Apoorv Vyas, Bowen Shi, Chih-Yao Ma, Ching-Yao Chuang, David Yan, Dhruv Choudhary, Dingkang Wang, Geet Sethi, Guan Pang, Haoyu Ma, Ishan Misra, Ji Hou, Jialiang Wang, Kiran Jagadeesh, Kunpeng Li, Luxin Zhang, Mannat Singh, Mary Williamson, Matt Le, Matthew Yu, Mitesh Kumar Singh, Peizhao Zhang, Peter Vajda, Quentin Duval, Rohit Girdhar, Roshan Sumbaly, Sai Saketh Rambhatla, Sam Tsai, Samaneh Azadi, Samyak Datta, Sanyuan Chen, Sean Bell, Sharadh Ramaswamy, Shelly Sheynin, Siddharth Bhattacharya, Simran Motwani, Tao Xu, Tianhe Li, Tingbo Hou, Wei-Ning Hsu, Xi Yin, Xiaoliang Dai, Yaniv Taigman, Yaqiao Luo, Yen-Cheng Liu, Yi-Chiao Wu, Yue Zhao, Yuval Kirstain, Zecheng He, Zijian He, Albert Pumarola, Ali Thabet, Artsiom Sanakoyeu, Arun Mallya, Baishan Guo, Boris Araya, Breena Kerr, Carleigh Wood, Ce Liu, Cen Peng, Dimitry Vengertsev, Edgar Schonfeld, Elliot Blanchard, Felix Juefei-Xu, Fraylie Nord, Jeff Liang, John Hoffman, Jonas Kohler, Kaolin Fire, Karthik Sivakumar, Lawrence Chen, Licheng Yu, Luya Gao, Markos Georgopoulos, Rashel Moritz, Sara K. Sampson, Shikai Li, Simone Parmeggiani, Steve Fine, Tara Fowler, Vladan Petrovic, and Yuming Du. Movie gen: A cast of media foundation models, 2024. 1

[32] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763, 2021. 2, 3, 5

[33] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, pages 10684–10695, 2022. 1, 3, 5

[34] Roy Sheffer and Yossi Adi. I hear your true colors: Image guided audio generation. In *ICASSP*, pages 1–5, 2023. 1, 2, 7

[35] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *ICLR*, 2021. 2

[36] Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024. 4

[37] Kun Su, Xiulong Liu, and Eli Shlizerman. From vision to audio and beyond: A unified model for audio-visual representation and generation. In *International Conference on Machine Learning*, pages 46804–46822, 2024. 2, 7

[38] Anni Tang, Tianyu He, Junliang Guo, Xinle Cheng, Li Song, and Jiang Bian. Vidtok: A versatile and open-source video tokenizer. *arXiv preprint arXiv:2412.13061*, 2024. 2, 3, 5

[39] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023. 7

[40] Heng Wang, Jianbo Ma, Santiago Pascual, Richard Cartwright, and Weidong Cai. V2a-mapper: A lightweight solution for vision-to-audio generation by connecting foundation models. In *AAAI*, pages 15492–15501, 2024. 1, 2, 7

[41] Xihua Wang, Yuyue Wang, Yihan Wu, Ruihua Song, Xu Tan, Zehua Chen, Hongteng Xu, and Guodong Sui. Tiva: Time-aligned video-to-audio generation. In *ACM MM*, pages 573–582, 2024. 1, 2, 3, 6, 7

[42] Xiang Wang, Shiwei Zhang, Hangjie Yuan, Zhiwu Qing, Biao Gong, Yingya Zhang, Yujun Shen, Changxin Gao, and Nong Sang. A recipe for scaling up text-to-video generation with text-free videos. In *CVPR*, pages 6572–6582, 2024. 2

[43] Yongqi Wang, Wenxiang Guo, Rongjie Huang, Jiawei Huang, Zehan Wang, Fuming You, Ruiqi Li, and Zhou Zhao. Frieren: Efficient video-to-audio generation network with rectified flow matching. In *Advances in Neural Information Processing Systems*, pages 128118–128138, 2024. 1, 2, 3, 6, 7

[44] Tianxing Wu, Chenyang Si, Yuming Jiang, Ziqi Huang, and Ziwei Liu. Freeinit: Bridging initialization gap in video diffusion models. In *ECCV*, pages 378–394, 2024. 2

[45] Yazhou Xing, Yingqing He, Zeyue Tian, Xintao Wang, and Qifeng Chen. Seeing and hearing: Open-domain visual-audio generation with diffusion latent aligners. In *CVPR*, pages 7151–7161, 2024. 1, 2, 6, 7

[46] Yiming Zhang, Yicheng Gu, Yanhong Zeng, Zhening Xing, Yuancheng Wang, Zhizheng Wu, and Kai Chen. Foleycrafter: Bring silent videos to life with lifelike and synchronized sounds. *arXiv preprint arXiv:2407.01494*, 2024. 2, 3, 6, 7

[47] Zikai Zhou, Shitong Shao, Lichen Bai, Zhiqiang Xu, Bo Han, and Zeke Xie. Golden noise for diffusion models: A learning framework. *arXiv preprint arXiv:2411.09502*, 2024. 2