FRIEREN: Efficient Video-to-Audio Generation Network with Rectified Flow Matching

Yongqi Wang[†], Wenxiang Guo[†], Rongjie Huang, Jiawei Huang, Zehan Wang, Fuming You, Ruiqi Li, Zhou Zhao[*]
Zhejiang University
cyanbox@zju.edu.cn

Abstract

Video-to-audio (V2A) generation aims to synthesize content-matching audio from silent video, and it remains challenging to build V2A models with high generation quality, efficiency, and visual-audio temporal synchrony. We propose FRIEREN, a V2A model based on rectified flow matching. FRIEREN regresses the conditional transport vector field from noise to spectrogram latent with straight paths and conducts sampling by solving ODE, outperforming autoregressive and score-based models in terms of audio quality. By employing a non-autoregressive vector field estimator based on a feed-forward transformer and channel-level cross-modal feature fusion with strong temporal alignment, our model generates audio that is highly synchronized with the input video. Furthermore, through reflow and one-step distillation with guided vector field, our model can generate decent audio in a few, or even only one sampling step. Experiments indicate that FRIEREN achieves state-of-the-art performance in both generation quality and temporal alignment on VGGSound, with alignment accuracy reaching 97.22%, and 6.2% improvement in inception score over the strong diffusion-based baseline. Audio samples and code are available at http://frieren-v2a.github.io.

1 Introduction

Recent advancements in deep generative models have significantly enhanced the quality and diversity of AI-generated content, including text [27], images [29, 30, 1], videos [32, 23] and audios [19, 20]. Among various content-generation tasks, video-to-audio (V2A) generation aims to synthesize semantically relevant and temporally aligned audio from video frames. Due to its immense potential for application in film dubbing, game development, YouTube content creation and other areas, the task of V2A has attracted widespread attention.

A widely applicable V2A solution is expected to have outstanding performance in the following aspects: **1) audio quality**: the generated audio should have good perceptual quality, which is the fundamental requirement of the audio generation task; **2) temporal alignment**: the generated audio should not only match the content but also align temporally with the video frames. This has a significant impact on user experience due to keen human perception of audio-visual information; and **3) generation efficiency**: the model should be efficient in terms of generation speed and resource utilization, which affects its practicality for large-scale and high-throughput applications.

Currently, considerable methods have been proposed for this task, including GAN-based models [3, 8], transformer-based autoregressive models [15, 31], and a recent latent-diffusion-based model, Diff-Foley [25]. However, these methods have not yet achieved a balanced and satisfactory performance

[†]Equal contribution.
[*]Corresponding author.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

across the above aspects. 1) For audio quality, early GAN-based models suffer from poor quality and lack practicality. Autoregressive and diffusion models make improvements in generation quality, but still leave room for further advancement. 2) For temporal alignment, autoregressive models lack the ability to align the generated audio with the video explicitly. And due to the difficulty of learning audio-visual alignment with the cross-attention-based conditional mechanism solely, Diff-Foley relies on additional classifier guidance to achieve good synchrony, which not only increases the model complexity but also leads to instability when reducing sampling steps. 3) For generation efficiency, autoregressive models suffer from high inference latency, while Diff-Foley requires considerable sampling steps to achieve good generation quality due to the curved sampling trajectories of diffusion models, increasing the temporal overhead in inference. In a nutshell, existing methods still leave significant room for improvement in performance.

In this paper, We introduce another generative modeling approach, namely rectified flow matching [21], into the V2A task. This method regresses the conditional transport vector field between noise and data distributions with as straight trajectories as possible, and conducts sampling by solving the corresponding ordinary differential equation (ODE). With simpler formulations, our rectified-flow-based model achieves higher audio quality and diversity. To improve temporal alignment, we adopt a non-autoregressive vector field estimator network with a feed-forward transformer with no temporal-dimension downsampling, thereby preserving temporal resolution. We also employ a channel-level cross-modal feature fusion mechanism for conditioning, leveraging the inherent alignment of audio-visual data and achieving strong alignment. These designs lead to high synchrony between generated audio and input video while upholding model simplicity. Moreover, through integrating reflow and one-step distillation techniques, our model can generate decent audio with a few, or even only one sampling step, significantly improving generation efficiency.

We name our model FRIEREN for e**ff**icient video-to-audio gene**r**ati**i**on n**e**twork with **re**ctfied flow matchi**n**g. Experiments indicate that FRIEREN outperforms strong baselines in terms of audio quality, generation efficiency, and temporal alignment on VGGSound [2], achieving a 6.2% improvement in inception score (IS) and a generation speed 7.3× that of Diff-Foley, as well as temporal alignment accuracy of up to 97.22% in 25 steps. Additionally, FRIEREN combining reflow and distillation achieves alignment accuracy of up to 97.85% with just one step, with a 9.3× acceleration compared to 25-step sampling, further boosting generation efficiency.

2 Related works

2.1 Video-to-audio generation

Video-to-audio (V2A) generation aims to synthesize audio of which content matches the visual information of a video clip. RegNet [3] designs a time-dependent visual encoder to extract appearance and motion features, which are then fed to a GAN for audio generation. FoleyGAN [8] also utilizes GAN for audio generation from visual features, together with a predicted action category as the conditional input. SpecVQGAN [15] takes RGB and optical flow of videos and uses a transformer to generate indices of a spectrogram VQVAE autoregressively. Im2Wav [31] adopts two transformers for different temporal resolutions and takes CLIP [28] features as the condition to generate VQVAE indices. Du et al. [5] mimics the real-world foley methodology and introduces an additional reference audio as the condition. Diff-Foley [25] designs an audio-visual contrastive feature and adopts a latent diffusion to predict spectrogram latents, achieving decent audio quality and inference speed.

In addition to training a whole model from scratch, some works integrate off-the-shelf audio generation models with modality mappers or multimodal encoders with joint embedding space for conditioning. V2A-Mapper [35] uses a lightweight mapper to transfer CLIP embeddings of videos to CLAP [40] embeddings as the condition for audio generation. Xing et al. [41] utilize an ImageBind[9]-based latent aligner for conditional guidance in audio generation. Despite the existence of plentiful works on V2A, there is still a large room left for improvement in quality, synchrony, and efficiency.

2.2 Flow matching generative models

Flow matching [18] models the vector field of transport probability path from noise to data samples. Compared to score-based models like DDPM [12], flow matching achieves more stable and robust training together with superior performance. Specifically, rectified flow matching [21] learns the

transport ODE to follow the straight paths connecting the noise and data points as much as possible, reducing the transport cost, and achieving fewer sampling steps with the reflow technique. This modeling paradigm has demonstrated excellent performance in accelerating image generation [22, 6].

In the area of audio generation, Voicebox [16] builds a large-scale multi-task speech generation model based on flow matching. Its successor, Audiobox [34], extends the flow-matching-based model to a unified audio generation model with natural language prompt guidance. Matcha-tts [26] trains an encoder-decoder TTS model with optimal-transport conditional flow matching. VoiceFlow [11] introduces rectified flow matching into TTS, achieving speech generation with fewer inference steps. However, for the task of V2A, there has been no exploration into utilizing flow matching models to enhance generation quality or inference efficiency.

3 Method

3.1 Preliminary: rectified flow matching

We first introduce the basic principles of rectified flow matching (RFM) [21] that we build our model upon. Conditional generation problems like V2A can be viewed as a conditional mapping from a noise distribution $\boldsymbol{x}_0 \sim p_0(\boldsymbol{x})$ to a data distribution $\boldsymbol{x}_1 \sim p_1(\boldsymbol{x})$. This mapping can be further taken as a time-dependent changing process of probability density (a.k.a. flow), determined by the ODE:

$$\mathrm{d}\boldsymbol{x} = \boldsymbol{u}(\boldsymbol{x}, t|\boldsymbol{c})\mathrm{d}t, t \in [0, 1], \tag{1}$$

where t represents the time position, $\boldsymbol{x}$ is a point in the probability density space at time t, $\boldsymbol{u}$ is the value of the transport vector field (i.e., the gradient of the probability w.r.t t) at $\boldsymbol{x}$, and $\boldsymbol{c}$ is the condition. In our case, the condition $\boldsymbol{c}$ is the visual features from the video frames, while the data $\boldsymbol{x}_1$ is the compressed mel-spectrogram latent of the corresponding audio from a pre-trained autoencoder. The fundamental principle of flow matching generative model is to use a neural network θ to regress the vector field $\boldsymbol{u}$ with the flow matching objective:

$$\mathcal{L}_{\mathrm{FM}}(\theta) = \mathbb{E}_{t, p_t(\boldsymbol{x})} \left\| \boldsymbol{v}(\boldsymbol{x}, t|\boldsymbol{c}; \theta) - \boldsymbol{u}(\boldsymbol{x}, t|c) \right\|^2, \tag{2}$$

where $p_t(\boldsymbol{x})$ is the distribution of $\boldsymbol{x}$ at timestep t. However, due to a lack of prior knowledge of target distribution $p_1(\boldsymbol{x})$ and the forms of p_t and $\boldsymbol{u}$, it is intractable to directly compute $\boldsymbol{u}(\boldsymbol{x}, t|c)$. As an alternative, conditional flow matching objective, which is proven in [18] to have identical gradient as eq. 2 w.r.t θ, is used for regression:

$$\mathcal{L}_{\mathrm{CFM}}(\theta) = \mathbb{E}_{t, p_1(\boldsymbol{x_1}), p_t(\boldsymbol{x}|\boldsymbol{x}_1)} \left\| \boldsymbol{v}(\boldsymbol{x}, t|\boldsymbol{c}; \theta) - \boldsymbol{u}(\boldsymbol{x}, t|\boldsymbol{x}_1, c) \right\|^2. \tag{3}$$

Through designing specific probabilistic paths that enable efficient sampling from $p_t(\boldsymbol{x}|\boldsymbol{x}_1)$ and computing of $\boldsymbol{u}(\boldsymbol{x}, t|\boldsymbol{x}_1, c)$, we achieve an unbiased estimation of $\boldsymbol{u}(\boldsymbol{x}, t|, c)$ with the CFM objective 3. Specifically, rectified flow matching attempts to establish straight paths between noise and data, aiming to facilitate sampling with larger step sizes and fewer steps. Given a noise-data pair $(\boldsymbol{x}_0, \boldsymbol{x}_1)$, $\boldsymbol{x}$ is located at $(1-t)\boldsymbol{x}_0 + t\boldsymbol{x}_1$ at timestep t, with the vector field being $\boldsymbol{u}(\boldsymbol{x}, t|\boldsymbol{x}_1, c) = \boldsymbol{x}_1 - \boldsymbol{x}_0$, pointing from the noise point to the data point. Hence, for each training step of the vector field estimator, we simply sample the data point $\boldsymbol{x}_1$ and noise point $\boldsymbol{x}_0$ from $p_1(\boldsymbol{x})$ and $p_0(\boldsymbol{x})$, respectively, and optimize the network with the rectified flow matching (RFM) loss

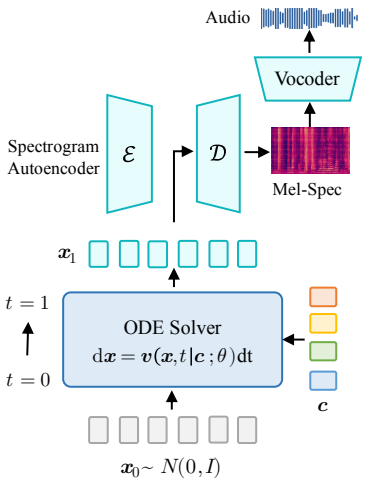
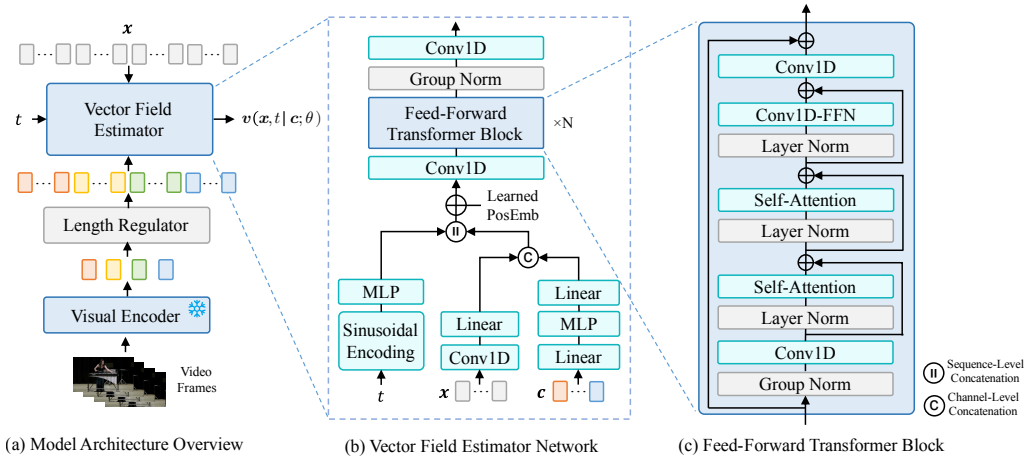


Figure 1: Illustration of the sampling process of our rectified-flow based V2A architecture.

$$\left\| \boldsymbol{v}(\boldsymbol{x}, t|\boldsymbol{c}; \theta) - (\boldsymbol{x}_1 - \boldsymbol{x}_0) \right\|^2. \tag{4}$$

Once the vector estimator network finishes training, we can adopt various solvers to approximate the solution of the ODE $\mathrm{d}\boldsymbol{x} = \boldsymbol{v}(\boldsymbol{x}, t|\boldsymbol{c}; \theta)$ at discretized time steps for sampling. A simple and commonly used ODE solver is the Euler method:

$$\boldsymbol{x}_{t+\epsilon} = \boldsymbol{x} + \epsilon \boldsymbol{v}(\boldsymbol{x}, t|\boldsymbol{c}; \theta) \tag{5}$$



(a) Model Architecture Overview (b) Vector Field Estimator Network (c) Feed-Forward Transformer Block

Figure 2: Illustration of model architecture of FRIEREN at different levels.

where ϵ is the step size. The sampled latent is fed to the decoder of the spectrogram autoencoder for spectrogram reconstruction, and the result is further used to reconstruct the audio waveform with a vocoder. Figure 1 provides a simple demonstration of the model's sampling process.

3.2 Model architecture

Model overview We illustrate the model architecture of FRIEREN at different levels in Figure 2. As shown in Figure 2(a), we first utilize a pre-trained visual encoder with frozen parameters to extract a frame-level feature sequence from the video. Usually, the video frame rate is lower than the temporal length per second of the spectrogram latent. To align the visual feature sequence with the mel latent at the temporal dimension for the cross-modal feature fusion mentioned below, we adopt a length regulator, which simply duplicates each item in the feature sequence by the ratio of the latent length per second and the video frame rate for regulation. The regulated feature sequence is then fed to the vector field estimator as the condition, together with $\boldsymbol{x}$ and t, to get the vector field prediction $\boldsymbol{v}$.

Visual and audio representations Various audio-aligned visual representations [9, 25, 14, 38, 37, 36, 39] can potentially be applied to video-to-audio generation, and we conduct experiments with two types of visual representations. For a fair comparison with Diff-Foley [25], we mainly utilize the CAVP feature proposed in [25], which is a visual-audio contrastive feature considering both content and temporal alignment. Meanwhile, to investigate the impact of visual feature characteristics on model performance, we also attempt the visual feature from MAViL [3] [14], which is an advanced self-supervised visual-audio representation learner that employs both masked-reconstruction and contrastive learning, and exhibits formidable performance in audio-visual understanding (See section 4.3.2 for comparison). For audio representation, we follow a previous text-to-audio work [13] to train a mel-spectrogram VAE with 1D convolution over the temporal dimension. Details of the VAE are provided in appendix A.

Vector field estimator Figure 2(b) demonstrates the structure of the vector field estimator, which is composed of a feed-forward transformer and some auxiliary layers. The regularized visual feature $\boldsymbol{c}$ and the point $\boldsymbol{x}$ on the transport path are first processed by stacks of shallow layers separately, with output dimensions being both half of the transformer hidden dimension, and are then concatenated along the channel dimension to realize cross-modal feature fusion. This simple mechanism leverages the inherent alignment within the video and audio, achieving enforced alignment without relying on learning-based mechanisms such as attention. As a result, the generated audio and input video sequences exhibit excellent temporal alignment. After appending the time step embedding to the beginning, the sequence is added with a learnable positional embedding and is then fed into the feed-forward transformer. The structure of the transformer block is illustrated in Figure 2 (c), the design of which is derived from the spatial transformer in latent diffusion [29], with the 2D convolution layers replaced by 1D ones. The feed-forward transformer does not involve temporal downsampling,

[3] Implementation of MAViL is from av-superb [33]: `https://github.com/roger-tseng/av-superb`

thus preserving the resolution of the temporal dimension and further ensuring the preservation of alignment. The output of the stacked transformer blocks is then passed through a normalization layer and a 1D convolution layer to finally obtain the prediction of the vector field.

3.3 Re-weighting RFM objective with logit-normal coefficient

The original RFM objective samples uniformly over time span $[0, 1]$. However, for modeling the vector field, positions in the middle of the transport path (equivalent to time steps in the middle of [0, 1]) present greater difficulty, as these positions are distant from both noise and data distributions. On the other hand, positions near the boundaries of the time span typically lie close to corresponding noise or data points, and their vector field direction tends to align with the lines connecting these points and the centroid of the distribution on the opposite side, and therefore relatively easy to regress. Upon this insight, we introduce time-based re-weighting to the original RFM objective, allocating more weight to intermediate time steps to achieve better modeling effectiveness. This is equivalent to increasing the sampling frequency of intermediate time steps. In practice, logit-normal weighting coefficients have been proven [6] to yield promising results, with the formula being

$$w(t) = \frac{1}{\sqrt{2\pi}} \frac{1}{t(1-t)} \exp\left(-\frac{(\ln t - \ln(1-t))^2}{2}\right). \tag{6}$$

We re-weight the RFM objective with this weighting function to replace the original objective and observe in our experiment that this re-weighting helps to slightly improve audio quality and temporal alignment at the cost of a marginal decrease in audio diversity.

3.4 Classifier-free guidance

Similar to diffusion-based models, we observe that classifier-free guidance (CFG) is highly important for generating audio that semantically matches and temporally aligns with the video. During training, we randomly replace the condition sequence $\boldsymbol{c}$ with a zero tensor with a probability of 0.2, and during sampling, we modify the vector field using the formula

$$\boldsymbol{v}_{\mathrm{CFG}}(\boldsymbol{x}, t|\boldsymbol{c}; \theta) = \gamma \boldsymbol{v}(\boldsymbol{x}, t|\boldsymbol{c}; \theta) + (1-\gamma)\boldsymbol{v}(\boldsymbol{x}, t|\varnothing; \theta), \tag{7}$$

where γ is the guidance scale trading off the sample diversity and generation quality, and $\boldsymbol{v}_{\mathrm{CFG}}$ degenerates into the original vector field $\boldsymbol{v}$ when $\gamma = 1$. We set γ to 4.5 in our major experiments.

3.5 Reflow and one-step distillation with guided vector field

In this section, we introduce two techniques we adopt for reducing sampling steps. The first one is reflow, which is a crucial component of the rectified flow paradigm [21, 22]. Training the estimator network with objective 4 for once is insufficient to construct straight enough transport paths, and an extra reflow procedure is needed to strengthen the transport trajectories without altering the marginal distribution learned by the model, enabling sampling with larger step sizes and fewer steps. Given a model θ trained with RFM objective, the reflow procedure applies θ to conduct sampling over the entire training dataset to obtain sampled data $\hat{\boldsymbol{x}}_1$ and save the corresponding input noise $\boldsymbol{x}'_0$, finally obtaining triplets $(\boldsymbol{x}'_0, \hat{\boldsymbol{x}}_1, \boldsymbol{c})$. The noise-data pair $(\boldsymbol{x}_0, \boldsymbol{x}_1)$ in the RFM objective 4 is replaced by $(\boldsymbol{x}'_0, \hat{\boldsymbol{x}}_1)$ for a secondary training of θ. This process can be repeated multiple times to obtain straighter trajectories with diminishing marginal effects. We conduct reflow for once as it is sufficient for achieving straight enough trajectories.

While many rectified-flow-based models regress the same velocity field $\boldsymbol{v}$ during both the initial training and the reflow process, we observe that when incorporating CFG, conducting sampling and reflow with the original vector field $\boldsymbol{v}$ is ineffective in straightening the sampling trajectories with the guided vector field $\boldsymbol{v}_{\mathrm{CFG}}$. Therefore, we use $\boldsymbol{v}_{\mathrm{CFG}}$ for generating $\hat{\boldsymbol{x}}_1$ and as the target of regression in reflow. The reflow objective can be written as:

$$\mathcal{L}_{\mathrm{reflow}}(\theta') = \mathbb{E}_{t, p(\boldsymbol{x}'_0, \hat{\boldsymbol{x}}_1|\boldsymbol{c}), p_t(\boldsymbol{x}|\boldsymbol{x}'_0, \hat{\boldsymbol{x}}_1)} \left\| \boldsymbol{v}_{\mathrm{CFG}}(\boldsymbol{x}, t|\boldsymbol{c}; \theta') - (\hat{\boldsymbol{x}}_1 - \boldsymbol{x}'_0) \right\|^2 \tag{8}$$

with same weighting function as eq. 6.

Upon the model θ' obtained from reflow, we further conduct one-step distillation [21, 22] to enhance the single-step generation performance of the model. As a type of self-distillation, this procedure tries

Table 1: Results of V2A models on VGGSound dataset. R+F and RN50 denote the RGB+Flow and ResNet50 versions of SpecVQGAN, and CG denotes classifier guidance in Diff-Foley.

Model	FD↓	IS↑	KL↓	FAD↓	KID(10^{-3}) ↓	Acc(%) ↑	MOS-Q↑	MOS-A↑
SpecVQGAN (R+F)	31.69	5.23	3.37	5.42	8.53	61.83	3.30 ± 0.06	2.35 ± 0.05
SpecVQGAN (RN50)	32.52	5.21	3.41	5.39	9.00	56.92	3.25 ± 0.07	2.17 ± 0.05
Im2Wav	14.98	7.20	**2.57**	5.49	3.35	56.70	3.39 ± 0.06	2.29 ± 0.06
Diff-Foley (CG ✓)	23.94	11.11	3.38	4.72	9.58	95.03	3.57 ± 0.08	3.74 ± 0.07
Diff-Foley (CG ✗)	24.97	11.69	3.23	7.10	10.32	92.53	3.64 ± 0.07	3.59 ± 0.06
LDM	11.79	10.09	2.86	1.77	**2.26**	95.33	3.72 ± 0.05	3.79 ± 0.07
FRIEREN	12.26	12.42	2.73	**1.32**	2.49	**97.22**	3.78 ± 0.06	$\mathbf{3.90 \pm 0.05}$
FRIEREN (Dopri5)	**11.64**	**12.76**	2.75	1.37	2.39	96.87	$\mathbf{3.81 \pm 0.06}$	3.85 ± 0.06

to reduce the error between the single-step sampling result $\boldsymbol{x}_0' + \boldsymbol{v}_{\mathrm{CFG}}(\boldsymbol{x}_0', t|\boldsymbol{c}; \theta)$ and the multi-step sampling result $\hat{\boldsymbol{x}}_1$. The objective function can be written as:

$$\mathcal{L}_{\mathrm{distill}}(\theta'') = \mathbb{E}_{t, p(\boldsymbol{x}_0', \hat{\boldsymbol{x}}_1|\boldsymbol{c}), p_t(\boldsymbol{x}|\boldsymbol{x}_0', \hat{\boldsymbol{x}}_1)} \left\| \boldsymbol{x}_0' + \boldsymbol{v}_{\mathrm{CFG}}(\boldsymbol{x}_0', t|\boldsymbol{c}; \theta'') - \hat{\boldsymbol{x}}_1 \right\|^2 \tag{9}$$

Formally, the distillation objective 9 can be viewed as a reflow objective with the sampling timestep fixed at $t = 0$. We observe in the experiment that due to a limited number of sampling steps in reflow data generation, the model may experience a decrease in sampling quality after the reflow process. Therefore, we opt to use the same training data used in reflow for distillation, rather than re-sampling the training data with the reflow model, which is based on the theoretical basis that reflow does not alter the marginal distribution modeled by the estimator.

4 Experiments

4.1 Experiment setup

Dataset and pre-processing Following most previous works, we take VGGSound [2] as the benchmark, which consists of 200k+ 10-second video clips from YouTube spanning 309 categories. Excluding videos already removed from YouTube, we follow the original train and test splits of VGGSound, the sizes of which are about 182.6k and 15.3k. We downsample the audios to 16kHz and transform them to mel-spectrogram with 80 bins and a hop size of 256. We follow [25] to downsample the videos to 4 FPS. Data samples are truncated to 8-second clips for training and inference.

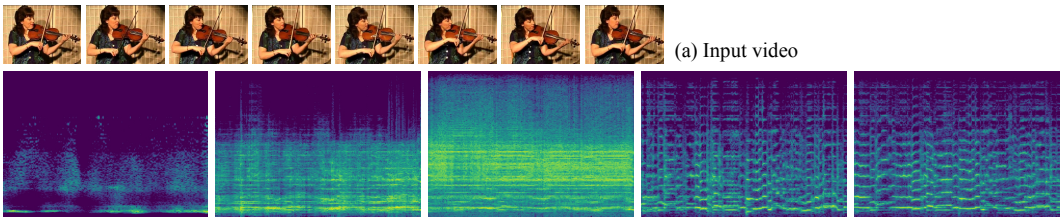
Model configuration The transformer of the vector field estimator mainly used in the experiments has 4 layers and a hidden dimension of 576. Each model is trained with 2 NVIDIA RTX-4090 GPUs. We train the estimator for 1.3M steps for the first training, and 600k and 500k steps for reflow and distillation, with the learning rate being 5e-5 for all stages. For waveform generation, we train a BigVGAN [17] vocoder on AudioSet [7]. Details of model parameters are provided in appendix A.

Metrics We combine objective and subjective metrics to evaluate model performance over audio quality, diversity, and temporal alignment. For objective evaluation, we calculate Frechet distance (FD), inception score (IS), Kullback–Leibler divergence (KL), Frechet audio distance (FAD), kernel inception distance (KID), and alignment accuracy (Acc). We utilize audio evaluation tools provided by AudioLDM [19], which are widely used in audio generation tasks, as well as the alignment classifier provided in [25]. For metrics with reference like FAD, we duplicate the reference audio samples in the test set for 10 times as we generate 10 samples for each data item. For subjective evaluation, we conduct crowd-sourced human evaluations with 1-5 Likert scales and report mean-opinion-scores (MOS) over audio quality (MOS-Q) and content alignment (MOS-A) with 95% confidence intervals (CI). We sample 10 audios for each test video for evaluation. Details of subjective evaluation are provided in appendix B.

Baseline models We adopt three advanced V2A models as baselines, including: 1) SpecVQGAN [15], a transformer-based autoregressive model generating spectrogram VQVAE indices from visual features; 2) Im2Wav [31], a hierarchical autoregressive V2A model predicting audio VQVAE indices conditioned on CLIP features; and 3) Diff-Foley [25], a strong latent-diffusion-based V2A model. For SpecVQGAN, we evaluate two versions using RGB+Flow and ResNet features as input visual

Table 2: Results of FRIEREN and Diff-Foley under different sampling steps. CG denotes classifier guidance, R denotes reflow and D denotes one-step distillation.

Model	Steps	FD↓	IS↑	KL↓	FAD↓	KID(10^{-3}) ↓	Acc(%) ↑	MOS-Q↑	MOS-A↑
Diff-Foley (CG ✓)	1	82.61	2.31	4.44	13.64	43.96	31.60	1.28 ± 0.04	1.35 ± 0.03
Diff-Foley (CG ✗)		86.97	1.86	4.17	14.66	39.73	37.02	1.17 ± 0.03	1.63 ± 0.04
FRIEREN (R ✗, D ✗)		70.48	2.95	4.21	13.07	26.99	43.18	2.12 ± 0.04	1.71 ± 0.04
FRIEREN (R ✓, D ✗)		18.61	6.63	2.60	3.13	3.49	94.96	3.32 ± 0.07	3.74 ± 0.06
FRIEREN (R ✓, D ✓)		**17.58**	**8.66**	**2.56**	**1.85**	**2.91**	**97.85**	**3.48 ± 0.06**	**3.93 ± 0.05**
Diff-Foley (CG ✓)	5	60.99	3.42	3.62	9.61	3.60	73.30	2.66 ± 0.07	2.98 ± 0.07
Diff-Foley (CG ✗)		51.52	5.14	3.45	10.96	2.66	91.30	3.03 ± 0.08	3.56 ± 0.07
FRIEREN (R ✗, D ✗)		28.78	6.69	3.02	4.34	8.56	87.69	3.30 ± 0.07	3.37 ± 0.08
FRIEREN (R ✓, D ✗)		**14.65**	**8.28**	**2.60**	**2.11**	**2.28**	**96.82**	**3.43 ± 0.06**	**3.83 ± 0.06**
Diff-Foley (CG ✓)	25	23.94	11.11	3.28	4.72	9.58	95.03	3.57 ± 0.08	3.74 ± 0.07
Diff-Foley (CG ✗)		24.97	11.69	3.23	7.10	10.32	92.53	3.64 ± 0.07	3.59 ± 0.06
FRIEREN (R ✗, D ✗)		12.26	**12.42**	2.73	**1.32**	2.49	97.22	**3.78 ± 0.06**	**3.90 ± 0.05**
FRIEREN (R ✓, D ✗)		13.39	9.79	**2.64**	1.66	2.01	**97.36**	3.61 ± 0.07	3.88 ± 0.05



(a) Input video

(b) Diff-Foley (c) LDM (d) FRIEREN (no reflow) (e) FRIEREN (reflow) (f) FRIEREN (reflow + distillation)

Figure 3: One-step generation results of different models. (a): The content of the input video is a woman playing the violin. (b): Diff-Foley generates meaningless audio with one step. (c, d): LDM and FRIEREN without reflow generate highly noisy audio. (e, f): reflow enables FRIEREN to generate meaningful audio in one step, and distillation further improves the one-step generation quality.

conditions. For Diff-Foley, we evaluate its performance with and without classifier guidance to examine the impact of its complex external alignment mechanism. To better validate the superiority of our rectified flow model, we also train a diffusion model sharing the same architecture as FRIEREN but has a different prediction target, labeled as LDM in the following tables. For diffusion models, we use DPM-Solver [24] for sampling. For our rectified flow model, we use the Euler method 5 in most cases without further specification. We also explore the more advanced Dormand–Prince method (Dopri5) [4] method for higher generation quality.

4.2 Results and analysis

Video-to-audio generation results The results of different models are illustrated in table 1. We sample with diffusion models and FRIEREN with 25 steps, and report the result of FRIEREN without reflow and distillation, which shows the best overall performance with a high number of sampling steps. It can be seen that FRIEREN significantly outperforms other models in IS, FAD, and alignment accuracy, with the values reaching up to 12.42, 1.32, and 97.22%, together with high subjective scores of 3.78 and 3.90 on quality and alignment. For FD, KL, and KID, the scores of FRIEREN are also very close to the best values among other models. When we employ the higher-order Dopri5 ODE solver, FRIEREN achieves further improvements in FD and IS, attaining best values of 11.64 and 12.76, respectively, while maintaining stable performance in other objective metrics, at the cost of slower sampling speed. This indicates the effectiveness of our approach. Generally, the performance of FRIEREN surpasses that of the LDM, demonstrating the superiority of rectified flow matching over the score-based paradigm of diffusion. Additionally, both FRIEREN and LDM outperform Diff-Foley in temporal alignment, proving that our architecture design achieves strong temporal alignment without the need for complex mechanisms, and can produce audio that is highly synchronized with visual input. Additionally, our model also has an advantage in sampling time, with details provided in appendix C.

Few and single step generation results We further demonstrate the results of Diff-Foley and FRIEREN on reduced sampling steps in table 2 to illustrate the impact of reflow and one-step

Table 3: Ablation results on different model size of vector field estimator network.

Model Size	FD↓	IS↑	KL↓	FAD↓	KID(10^{-3}) ↓	Acc(%) ↑	MOS-Q↑	MOS-A↑
Small (70.90 M)	13.02	12.16	2.78	1.50	2.79	96.04	3.71 ± 0.07	3.83 ± 0.06
Base (158.88 M)	12.26	12.42	2.73	1.32	2.49	97.22	3.78 ± 0.06	3.90 ± 0.05
Large (421.12 M)	12.20	12.29	2.76	1.36	2.97	95.16	3.78 ± 0.07	3.80 ± 0.06

distillation, together with trend graphs of IS and FAD in figure 4 for intuitive presentation. The data for reflow are generated with the Euler method for 25 steps. We observe an obvious drop in performance of Diff-Foley as well as FRIEREN without reflow when sampling with as few as 5 steps, and their scores become extremely poor when we further reduce the step number to 1. Figure 3 (b) (c) and (d) illustrate that the audio generated by these models as well as LDM degrades into unacceptably noisy or meaningless audio within one step. This is due to the convoluted nature of the sampling trajectories of these models, which disables them from sampling with large step sizes and few steps. We also notice that when sampling with 5 steps, using additional classifier guidance deteriorates the audio quality and synchrony of Diff-Foley, where alignment accuracy and IS drop by 18.0% and 1.72 respectively, while FD, KL, and KID increase by 9.47, 0.17, and 0.94 $\times 10^{-3}$. This indicates the lack of robustness of the complex alignment mechanism that Diff-Foley relies on.

In contrast, FRIEREN with reflow achieves an alignment accuracy of up to 96.82% in just 5 steps, with significant advantages in quality, diversity, and subjective metrics. Additionally, it maintains an accuracy of 94.96% in single-step generation, as well as decent quality and diversity. This proves that reflow functions significantly in straightening the sampling trajectories, enabling the rectified flow model to generate decent audio with a small number of sampling steps. Furthermore, single-step distillation following reflow further improves the model performance with one step, with alignment accuracy reaching up to 97.85%, and KL, FAD, and KID being close to the 25-step results of FRIEREN trained once, with differences of 0.17, 0.53 and 0.42$\times 10^{-3}$. It also achieves high MOS-Q and MOS-A of 3.48 and 3.93.

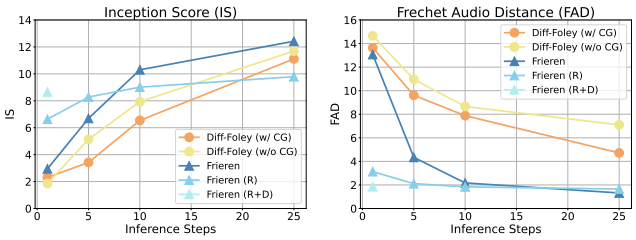


Figure 4: IS and FAD of the models with different steps.

Figure 3 (e) and (f) show that results from FRIEREN with reflow and reflow+distillation have distinguishable spectrograms, with the latter showing higher quality and sharper edges. This fully demonstrates that the combination of reflow and one-step distillation endows our model with strong single-step generation capabilities, significantly enhancing the efficiency on the V2A task. Notice that reflow brings in some quality degradation in sampling with 25 steps. We speculate that this is because the limited number of sampling steps restricts the data quality when generating data for reflow, resulting in a shift in the marginal distribution learned by the model. This cumulative error might be mitigated by increasing the number of sampling steps during reflow data generation.

4.3 Ablation study

4.3.1 Model size of vector field estimator

We adjust the number of parameters of the vector field estimator and evaluate the model performance at different scales. We label the major model as "base", and obtain "small" and "large" models by decreasing and increasing the hidden dimension and / or the number of transformer layers, respectively. The parameter counts of the estimator and results are presented in table 3.

We observe that when the model parameters are reduced to 71M, performance declines across all metrics, where FD, KL, FAD, and KID increase by 0.76, 0.05, 0.18, and 0.3$\times 10^{-3}$, and IS, alignment accuracy, MOS-Q and MOS-A drop by 0.26, 1.18%, 0.07 and 0.07, respectively. However, when the parameter number increases to 421M, there is a performance degradation across multiple metrics, with KL, FAD, and KID increasing by 0.03, 0.04, and 0.48$\times 10^{-3}$, and IS, alignment acc declining by 0.13 and 2.06%. We speculate that this anomalous phenomenon may be due to the convergence difficulty

Table 4: Results on different types visual features.

Type	Feat. FPS	FD↓	IS↑	KL↓	FAD↓	KID(10^{-3}) ↓	Acc(%) ↑	MOS-Q↑	MOS-A↑
CAVP [25]	4	12.26	12.42	2.73	1.32	2.49	97.22	3.78 ± 0.06	3.90 ± 0.05
MAViL [14]	2	12.08	12.17	2.49	1.26	2.52	90.17	3.75 ± 0.06	3.46 ± 0.07

Table 5: Ablation results on RFM objective re-weighting.

Re-weighting	FD↓	IS↑	KL↓	FAD↓	KID(10^{-3}) ↓	Acc(%) ↑	MOS-Q↑	MOS-A↑
✗	11.95	12.20	2.73	1.25	2.12	97.04	3.74 ± 0.07	3.82 ± 0.06
✓	12.26	12.42	2.73	1.32	2.49	97.22	3.78 ± 0.06	3.90 ± 0.05

for the larger model under similar training steps, or the redundant model capacity tends to cause overfitting on a relatively small dataset like VGGSound, deteriorating the model's generalization performance. In summary, we achieve relatively balanced model performance with the parameter of the estimator being around 160M. Details of model parameters are provided in appendix A.

4.3.2 Visual feature characteristics

In table 4, we compare the results of FRIEREN using two different types of visual features from CAVP and MAViL. Intuitively, the MAViL feature should be more robust and contain richer audio-related semantic information, as it utilizes masked-reconstruction together with inter-modal and intra-modal contrastive learning, in contrast to CAVP trained solely with inter-modal contrastive learning. On the other hand, however, due to MAViL's convolutional downsampling in the temporal dimension, its feature sequence has a lower effective FPS of 2 with the same 4 FPS video input as CAVP. The results in the table indicate that the model with MAViL feature excels in audio diversity, with differences of FD, KL, and FAD being 0.18, 0.24, and 0.06. Meanwhile, it exhibits a 7.05% decrease in alignment accuracy and a 0.25 decrease in IS. This result yields two insights for V2A tasks: 1) at relatively low frame rates, the frame rate of features, rather than content, is more likely to become the bottleneck for audio quality and visual-audio synchrony; 2) compared to high video frame rates, the semantic information and robustness of visual features are more crucial for the diversity of generated audio.

4.3.3 Classifier-free guidance scale

In figure 5, we illustrate the impact of various CFG scales on the performance of FRIEREN. In terms of audio diversity (FD, KL, KID, FAD), the metrics initially increase with the CFG scale, reaching an optimal value at around 2 and 3. After that, the metrics go down as the increasing CFG scale suppresses the diversity. For audio quality (IS) and temporal alignment, as larger scales make the content of the generated audio closer to the visual information, the metrics initially increase with the scale, reaching an optimal value between 4 and 4.5, and decrease after that due to audio distortion. We prioritize audio quality and synchrony and adopt a CFG scale of 4.5.

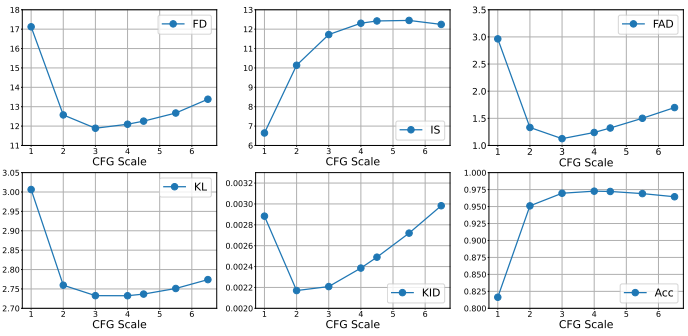


Figure 5: Model performance of FRIEREN under different CFG scales.

4.3.4 Re-weighting RFM objective

We conduct ablation on RFM objective re-weighting and report the results in table 5. We can see that compared to the vanilla objective, introducing re-weighting results in improvements of 0.22 and 0.18% for IS and alignment accuracy. This validates the positive impact of objective re-weighting on audio quality and temporal alignment. On the other hand, objective re-weighting causes a decrease in audio diversity, with differences in FD, FAD, and KID being 0.31, 0.07, and 0.37×10^{-3}, respectively.

5 Conclusion

In this paper, we propose FRIEREN, an efficient video-to-audio generation model based on rectified flow matching. We use a neural network to regress the conditional transport vector field with straight paths from noise to spectrogram latents, and conduct sampling by solving ODE, achieving better performance than diffusion-based and other V2A models. We adopt a vector field estimator based on a feed-forward transformer as well as channel-level cross-modal feature fusion to realize strong audio-video synchrony. Through a combination of reflow and one-step distillation, our model can generate high-quality audio with a few or even one sampling step, boosting the generation efficiency significantly. Experiments show that our model achieves state-of-the-art V2A performance on VGGSound. For future work, we will explore extending the model to larger scales and larger datasets to achieve V2A generation on a broader data domain. Besides, we will attempt audio generation from longer video sequences with variable lengths, rather than being limited to fixed-length short clips. These efforts aim to build a more versatile and widely applicable V2A model.

Acknowledgment

This work is supported by the National Natural Science Foundation of China under Grant No. 62222211 and No.62072397.

References

[1] James Betker, Gabriel Goh, Li Jing, Tim Brooks, Jianfeng Wang, Linjie Li, Long Ouyang, Juntang Zhuang, Joyce Lee, Yufei Guo, et al. Improving image generation with better captions. *Computer Science. https://cdn. openai. com/papers/dall-e-3. pdf*, 2(3):8, 2023.

[2] Honglie Chen, Weidi Xie, Andrea Vedaldi, and Andrew Zisserman. Vggsound: A large-scale audio-visual dataset. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 721–725. IEEE, 2020.

[3] Peihao Chen, Yang Zhang, Mingkui Tan, Hongdong Xiao, Deng Huang, and Chuang Gan. Generating visually aligned sound from videos. *IEEE Transactions on Image Processing*, 29: 8292–8302, 2020.

[4] John R Dormand and Peter J Prince. A family of embedded runge-kutta formulae. *Journal of computational and applied mathematics*, 6(1):19–26, 1980.

[5] Yuexi Du, Ziyang Chen, Justin Salamon, Bryan Russell, and Andrew Owens. Conditional generation of audio from video via foley analogies. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2426–2436, 2023.

[6] 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. *arXiv preprint arXiv:2403.03206*, 2024.

[7] Jort F Gemmeke, Daniel PW Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R Channing Moore, Manoj Plakal, and Marvin Ritter. Audio set: An ontology and human-labeled dataset for audio events. In *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pp. 776–780. IEEE, 2017.

[8] Sanchita Ghose and John J Prevost. Foleygan: Visually guided generative adversarial network-based synchronous sound generation in silent videos. *IEEE Transactions on Multimedia*, 2022.

[9] 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*, pp. 15180–15190, 2023.

[10] Daniel Griffin and Jae Lim. Signal estimation from modified short-time fourier transform. *IEEE Transactions on acoustics, speech, and signal processing*, 32(2):236–243, 1984.

[11] Yiwei Guo, Chenpeng Du, Ziyang Ma, Xie Chen, and Kai Yu. Voiceflow: Efficient text-to-speech with rectified flow matching. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 11121–11125. IEEE, 2024.

[12] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[13] Jiawei Huang, Yi Ren, Rongjie Huang, Dongchao Yang, Zhenhui Ye, Chen Zhang, Jinglin Liu, Xiang Yin, Zejun Ma, and Zhou Zhao. Make-an-audio 2: Temporal-enhanced text-to-audio generation. *arXiv preprint arXiv:2305.18474*, 2023.

[14] Po-Yao Huang, Vasu Sharma, Hu Xu, Chaitanya Ryali, Yanghao Li, Shang-Wen Li, Gargi Ghosh, Jitendra Malik, Christoph Feichtenhofer, et al. Mavil: Masked audio-video learners. *Advances in Neural Information Processing Systems*, 36, 2024.

[15] Vladimir Iashin and Esa Rahtu. Taming visually guided sound generation. In *The 32st British Machine Vision Virtual Conference*. BMVA Press, 2021.

[16] Matthew Le, Apoorv Vyas, Bowen Shi, Brian Karrer, Leda Sari, Rashel Moritz, Mary Williamson, Vimal Manohar, Yossi Adi, Jay Mahadeokar, et al. Voicebox: Text-guided multilingual universal speech generation at scale. *Advances in neural information processing systems*, 36, 2024.

[17] Sang-gil Lee, Wei Ping, Boris Ginsburg, Bryan Catanzaro, and Sungroh Yoon. Bigvgan: A universal neural vocoder with large-scale training. In *The Eleventh International Conference on Learning Representations*, 2022.

[18] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations*, 2022.

[19] 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. *arXiv preprint arXiv:2301.12503*, 2023.

[20] Haohe Liu, Qiao Tian, Yi Yuan, Xubo Liu, Xinhao Mei, Qiuqiang Kong, Yuping Wang, Wenwu Wang, Yuxuan Wang, and Mark D Plumbley. Audioldm 2: Learning holistic audio generation with self-supervised pretraining. *arXiv preprint arXiv:2308.05734*, 2023.

[21] Xingchao Liu, Chengyue Gong, et al. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *The Eleventh International Conference on Learning Representations*, 2022.

[22] Xingchao Liu, Xiwen Zhang, Jianzhu Ma, Jian Peng, et al. Instaflow: One step is enough for high-quality diffusion-based text-to-image generation. In *The Twelfth International Conference on Learning Representations*, 2023.

[23] Yixin Liu, Kai Zhang, Yuan Li, Zhiling Yan, Chujie Gao, Ruoxi Chen, Zhengqing Yuan, Yue Huang, Hanchi Sun, Jianfeng Gao, et al. Sora: A review on background, technology, limitations, and opportunities of large vision models. *arXiv preprint arXiv:2402.17177*, 2024.

[24] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. *Advances in Neural Information Processing Systems*, 35:5775–5787, 2022.

[25] Simian Luo, Chuanhao Yan, Chenxu Hu, and Hang Zhao. Diff-foley: Synchronized video-to-audio synthesis with latent diffusion models. *Advances in Neural Information Processing Systems*, 36, 2024.

[26] Shivam Mehta, Ruibo Tu, Jonas Beskow, Éva Székely, and Gustav Eje Henter. Matcha-tts: A fast tts architecture with conditional flow matching. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 11341–11345. IEEE, 2024.

[27] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. 2018.

[28] 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*, pp. 8748–8763. PMLR, 2021.

[29] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10684–10695, 2022.

[30] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022.

[31] Roy Sheffer and Yossi Adi. I hear your true colors: Image guided audio generation. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5. IEEE, 2023.

[32] Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, et al. Make-a-video: Text-to-video generation without text-video data. *arXiv preprint arXiv:2209.14792*, 2022.

[33] Yuan Tseng, Layne Berry, Yi-Ting Chen, I-Hsiang Chiu, Hsuan-Hao Lin, Max Liu, Puyuan Peng, Yi-Jen Shih, Hung-Yu Wang, Haibin Wu, et al. Av-superb: A multi-task evaluation benchmark for audio-visual representation models. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6890–6894. IEEE, 2024.

[34] 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.

[35] Heng Wang, Jianbo Ma, Santiago Pascual, Richard Cartwright, and Weidong Cai. V2a-mapper: A lightweight solution for vision-to-audio generation by connecting foundation models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 15492–15501, 2024.

[36] Zehan Wang, Ziang Zhang, Xize Cheng, Rongjie Huang, Luping Liu, Zhenhui Ye, Haifeng Huang, Yang Zhao, Tao Jin, Peng Gao, et al. Freebind: Free lunch in unified multimodal space via knowledge fusion. In *Forty-first International Conference on Machine Learning*.

[37] Zehan Wang, Ziang Zhang, Luping Liu, Yang Zhao, Haifeng Huang, Tao Jin, and Zhou Zhao. Extending multi-modal contrastive representations. *arXiv preprint arXiv:2310.08884*, 2023.

[38] Zehan Wang, Yang Zhao, Haifeng Huang, Jiageng Liu, Aoxiong Yin, Li Tang, Linjun Li, Yongqi Wang, Ziang Zhang, and Zhou Zhao. Connecting multi-modal contrastive representations. *Advances in Neural Information Processing Systems*, 36:22099–22114, 2023.

[39] Zehan Wang, Ziang Zhang, Hang Zhang, Luping Liu, Rongjie Huang, Xize Cheng, Hengshuang Zhao, and Zhou Zhao. Omnibind: Large-scale omni multimodal representation via binding spaces. *arXiv preprint arXiv:2407.11895*, 2024.

[40] 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)*, pp. 1–5. IEEE, 2023.

[41] Yazhou Xing, Yingqing He, Zeyue Tian, Xintao Wang, and Qifeng Chen. Seeing and hearing: Open-domain visual-audio generation with diffusion latent aligners. *arXiv preprint arXiv:2402.17723*, 2024.

A Implementation details

Table 6: Architecture details of 1D VAE for spectrogram compression.

Hyperparameter	1D VAE
Input tensor shape for 10-sec audio	(80,624)
Embedding dimension	20
Channels	224
Channel multiplier	1, 2, 4
Downsample layer position	after block 1
Attention layer position	after block 3
Output tensor shape for 10-sec audio	(20,312)

Table 7: Hyperparameters of the vector field estimator of FRIEREN with different sizes.

Hyperparameter	Small	Base	Large
Layers	4	4	6
Hidden dimension	384	576	768
Attention heads	8	8	8
Conv1D-FFN dimension	1,536	2,304	3,072
Number of parameters	70.90M	158.88M	421.12M

In table 6, we provide the architecture details of the mel-spectrogram VAE. Different from the commonly used 2D VAE for spectrogram, the 1D VAE we adopt does not involve an extra channel dimension, but takes the frequency axis of the spectrogram as the channel dimension, and conducts convolution along the temporal axis. This design is derived from the insight that the spectrogram is not translation invariant along the frequency axis, and it can better synergize with the feed-forward transformer. In table 7, we present the hyperparameters of the vector field estimator networks with different sizes.

Additionally, we observe that although there is no significant difference in objective metrics, initializing the vector field estimator with the weights of a diffusion model for text-to-audio (T2A) generation [13] helps improve the subjective perceptual quality of the generated audio marginally. This improvement derives from the knowledge of audio generation on a broader data domain learned by the T2A model. We adopt this trick in our model training.

B Subjective evaluation

Figure 6: Screenshot of subjective evaluation on audio quality.

For each evaluated model, we select 150 items for subjective evaluation, accounting for about 1% of the entire test split.

Our subjective evaluation tests are crowd-sourced and conducted via Amazon Mechanical Turk. For audio quality evaluation, we ask the testers to examine the audio quality and ignore the content. And

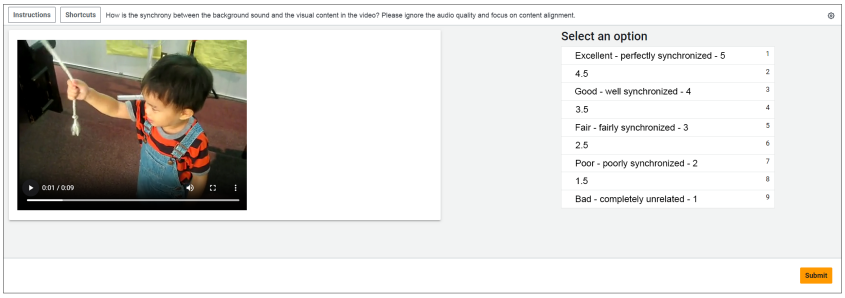


Figure 7: Screenshot of subjective evaluation on temporal alignment.

for temporal alignment, we instruct the testers to evaluate the synchrony between the background audio and the video content, while ignoring the audio quality. The testers rate scores on 1-5 Likert scales. We provide screenshots of the testing interfaces in figure 6 and 7. Each data item is rated by 6 testers, and the testers are paid $8 hourly.

C Time efficiency

Table 8: Inference time per sample of different models with batch size = 1.

Model	Inference Time (sec)
SpecVQGAN	3.936
Im2Wav	333.246
Diff-Foley (step=25)	2.104
FRIEREN (Dopri5, step=25)	1.510
FRIEREN (Euler, step=25)	0.288
FRIEREN (Euler, step=5)	0.064
FRIEREN (Euler, step=1)	0.031

In table 8, we compare the inference time per sample of different models. The inference is conducted on a single RTX-4090 GPU with a batch size of 1. We can see that the inference procedure of transformer-based autoregressive models, including SpecVQGAN and Im2Wav, is more time-consuming, especially for Im2Wav, which takes several minutes to generate a single sample. This is because Im2Wav conducts a cascaded generation with 2 transformers. Moreover, its use of high-bitrate audio VQVAE results in very long sequences of audio representation, significantly increasing the inference time required for the transformers, which has quadratic time complexity concerning sequence length. In contrast, Diff-Foley and FRIEREN require less inference time, and FRIEREN with Euler solver enjoys a higher speed, achieving 7.3 times faster than Diff-Foley with 25 sampling steps. This is the result of a combination of multiple factors, including model architecture, model parameters, sampling methods, and so on. Furthermore, when using FRIEREN model with reflow and one-step distillation, we can generate 5-step sampled audio in 0.064 seconds and 1-step sampled audio in just 0.031 seconds, achieving 4.5$\times$ and 9.3$\times$ acceleration compared to 25-step sampling. This demonstrates the extremely high generation efficiency of our model on the task of V2A.

D Impact of the vocoder on model performance

Different selections of vocoders can significantly impact the performance of various audio generation models. Diff-Foley uses the simple Griffin-Lim method [10] to map spectrograms to waveforms, while FRIEREN employs the more efficient BigVGAN. To compare the performance of the spectrogram generation models while minimizing the influence of the vocoder, we apply BigVGAN and Griffin-Lim separately to each model. The output from Diff-Foley is converted into an 80-bin mel-spectrogram and then fed into BigVGAN. The number of Griffin-Lim iterations for FRIEREN is the same as Diff-Foley. The results are shown in table 9.

Table 9: Comparison of the performance of Diff-Foley and FRIEREN using the same vocoder.

Model	FD↓	IS↑	KL↓	FAD↓	KID(10^{-3}) ↓
BigVGAN					
Diff-Foley (CG ✓)	18.02	10.89	2.88	6.32	5.32
FRIEREN	**12.26**	**12.42**	**2.73**	**1.32**	**2.49**
Griffin-Lim					
Diff-Foley (CG ✓)	**23.94**	**11.11**	3.38	4.72	**9.58**
FRIEREN	28.29	10.67	**3.17**	**3.70**	12.30

It can be seen that using BigVGAN for Diff-Foley improves its FD, KL, and KID, indicating its effectiveness. On this basis, FRIEREN outperforms Diff-Foley across all metrics, with a greater difference than when using Griffin-Lim for both. This further demonstrates that our model is superior to Diff-Foley.

On the other hand, when using Griffin-Lim for both models, despite the performance drop, FRIEREN still surpasses Diff-Foley in KL and FAD, with FAD showing a significant advantage while maintaining competitive FD and IS values. We speculate that the Griffin-Lim algorithm is so weak that it forms a performance bottleneck, narrowing the performance gap between FRIEREN and Diff-Foley. Additionally, differences in spectrogram hyperparameters may also lead to a performance gap. Diff-Foley uses 128 frequency bins, more than the 80 bins used by FRIEREN, allowing it to carry finer-grained information and may give Diff-Foley an advantage when using Griffin-Lim.

E Limitations and boarder impacts

Limitations Despite that FRIEREN achieves outstanding performance on audio quality, temporal alignment, and generation efficiency, it still has two major limitations: 1) Currently, experiments have only been conducted on a small-scale dataset, VGGSound, and we have not yet scaled the model to large-scale datasets. Therefore, it is still difficult to apply our model to a wide range of real-world scenarios for now; 2) our current model design only targets audio generation for fixed-length short video clips, and it lacks the ability of audio generation for long videos with various lengths. We will explore the solutions to these issues in future work.

Potential positive impacts The achievements of our model on the V2A task may reduce the cost of sound effect synthesis, and could potentially drive advancements in the film, gaming, and social media industries.

Potential negative social impacts The automatic sound effect generation technology may lead to job losses for related personnel. Additionally, there is a risk of the model being used to generate harmful content or fake media. Constraints are needed to guarantee that people will not use the model in illegal cases.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Please refer to abstract and section 1.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Please refer to appendix E.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [No]

Justification: As an application-oriented paper, the theory of the rectified flow model we rely upon has been thoroughly demonstrated and proven in the related original papers. We cite these original papers and encourage readers to refer to them for the complete proofs.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Please refer to section 4.1 and appendix A.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: Due to time limiations, we have not yet organized a publicly shareable version of the code. We will open-source our code after the paper is accepted.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: Please refer to section 4.1, appendix A and appendix B.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We report 95% confidence interval of subjective evaluation results (MOS-Q and MOS-A). No statistical significance is applicable for objective metrics we use.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Please refer to *Model configuration* in section 4.1.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We faithfully adhered to the NeurIPS Code of Ethics throughout our research.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Please refer to appendix E.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [No]

    Justification: At this stage, we have not made our code and models open-source. We will consider relevant protection measures during the open-sourcing stage.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: For code and data we use in our paper, we cite the original papers and comply with their licensing.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.

- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: We do not release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [Yes]

    Justification: Please refer to appendix B.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The crowdsourced evaluations we conduct do not involve potential risks or activities requiring approval.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.