Detailed Experimental Setup

Table 5: Statistics for datasets used for training and fine-tuning.

Usage	Dataset	Hours
Train	Audioset (Gemmeke et al. 2017)	$\sim$ 2000
Train	VGGSound (Chen et al. 2020)	72
Adaptation	AIST++ (Li et al. 2021)	5.2
Adaptation	Landscape (Lee et al. 2022b)	2.7
Adaptation	MV100K (Su et al. 2024b)	100

As shown in Table 5, we present the statistics for datasets used for training and fine-tuning. Specifically,

- VGGSound dataset consists of videos uploaded to YouTube with audio-visual correspondence, containing $\sim$200k 10-second videos. We follow the original VGGSound train/test splits.
- AudioSet comprises 2.1M videos with 527 sound classes, with most of the videos labeled as music and speech.
- Landscape dataset is a high-fidelity audio-video dataset with nature scenes. The total duration is about 2.7 hours of 300K frames.
- AIST++ is a subset of AIST dataset (Tsuchida et al. 2019), which contains street dance videos with 60 copyright-cleared dancing songs. The dataset includes 1020 video clips with a total duration of 5.2 hours of about 560K frames.
- MV100K is filtered from a publicly available video dataset (Abu-El-Haija et al. 2016) to videos with the label music videos, and we use 100 hours of data for fine-tuning.

Contrastive Audio-Video Pre-training (CAVP)

Denote the training data as $D = \{(X_i^a, X_i^v)\}_{i=1}^{i=N}$. Let f_{audio} be the audio encoder and f_{video} be the video encoder, which is a learnable embedding function. The model is trained with the contrastive learning paradigm between the audio embeddings E_i^a and video embeddings E_i^v in pair:

$$E_i^a = MLP_{\text{audio}}\left(f_{\text{audio}}\left(X_i^a\right)\right), E_i^v = MLP_{\text{video}}\left(f_{\text{video}}\left(X_i^v\right)\right) \tag{5}$$

Recent multi-modal generative models (Podell et al. 2023; Girdhar et al. 2023) have collected large paired data and trained models to embed multi-modal in a joint space using contrastive learning, exhibiting impressive zero-shot performance: CLIP (Radford et al. 2021) is pre-trained on image-text data, contrastive language-audio pretraining (CLAP) (Elizalde et al. 2022) brings audio and text descriptions into a joint space.

We define the per-sample pair semantic contrast objective, where τ is a learnable temperature parameter for scaling the loss, and N is the number of data: $L = \frac{1}{2N}\sum_{i=1}^{N}\left(\log\frac{\exp(E_i^a\cdot E_i^v/\tau)}{\sum_{j=1}^{N}\exp(E_i^a\cdot E_j^v/\tau)} + \log\frac{\exp(E_i^v\cdot E_i^a/\tau)}{\sum_{j=1}^{N}\exp(E_i^v\cdot E_j^a/\tau)}\right)$

Following (Radford et al. 2021; Elizalde et al. 2023), two logarithmic terms consider either audio-to-video logits or video-to-audio logits. After training, we evaluate the model in downstream audio-video and video-audio retrieval tasks, where we compute the similarity between the audio and video embeddings. Take audio-video retrieval as an example, top-N descriptions are computed by picking the descriptions corresponding to the top N values in similarity.

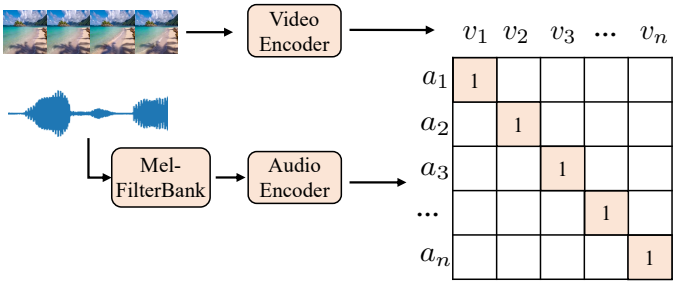


Figure 4: The contrastive audio-video pretraining process.

As can be seen in Table 6, the constrastive audio-video pretraining model (CAVP) achieves high retrieval accuracy. It indicates the outperformed capabilities in assessing the coherence of the audio about the video instruction. After training the CAVP model, we use it to evaluate video-guided audio generation models by calculating 1) CAVP video-audio similarity – measuring alignment between video and generated audio, and 2) CAAP audio-audio similarity - measuring the change between reference and generated audio.

Model Configurations

We list the model hyper-parameters of VisualAudio in Table 7.

Transformer condition mechanism

In Section , the multi-head cross-attention layers are attached to handle the challenges of video-synchronized audio generation, which is effective for learning various input modalities. We also explore two other methods (e.g., self-attention or concat) to integrate temporal local information for ablation and illustrate them in Figure 5. Ablation results in Table 4 demonstrate the effiectiveness of cross-attention mechanism to inject video condition.

VAE

The audio encoder E takes mel-spectrogram x_a as input and outputs compressed latent $z = E(x_a)$. The audio decoder D reconstructs the mel-spectrogram signals $\tilde{x_a} = D(z)$ from the compressed representation z. Different from other modalities, we use an audio VAE with 1D-convolution to improve the model's capacity for variable-length audio. VAE solves the problem of excessive smoothing in mel-spectrogram reconstruction through adversarial training with a discriminator.

Table 6: Retrieval accuracy using a contrastive audio-video pretraining model.

	Video → Audio				Audio → Video			
	R@1	R@5	R@10	mAP	R@1	R@5	R@10	mAP
AIST++	0.15	0.80	0.95	0.39	0.15	0.45	0.65	0.29
Landscape	0.08	0.25	0.36	0.14	0.08	0.22	0.30	0.14
MV100K	0.11	0.41	0.59	0.22	0.05	0.52	0.76	0.21

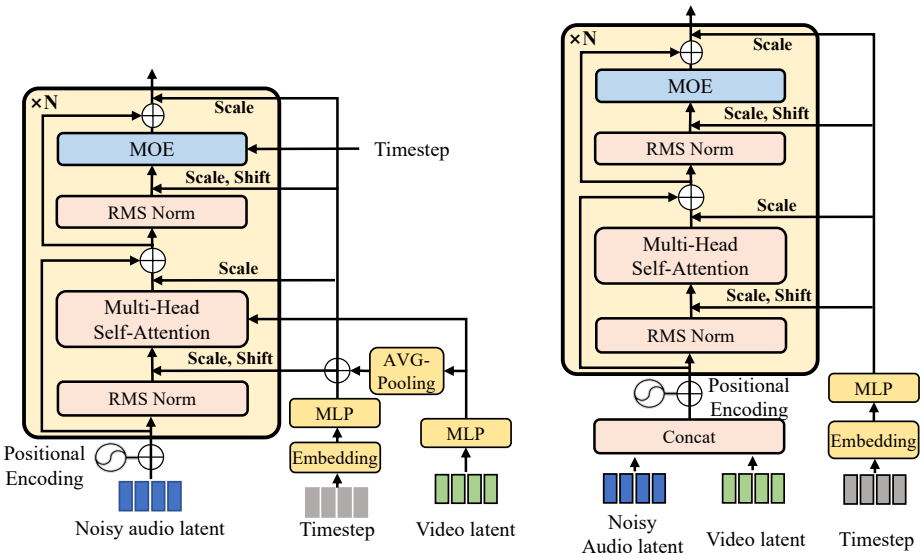


Figure 5: Condition mechanism. Left: with cross-attention module to inject visual information. Right: with the self-attention module, concatenating the video/audio latents along the channel/time dimension before feeding the input into the transformers.

The training objective is to minimize the weighted sum of reconstruction loss, GAN loss, and KL penalty loss. To this end, VisualAudio takes advantage of the VAE to predict self-supervised representations instead of waveforms. It largely alleviates the challenges of modeling long continuous data and guarantees high-level semantic understanding.

Vocoder

We train a BigVGAN (Lee et al. 2022a) vocoder from scratch for the spectrogram to waveform generation. The synthesizer includes the generator and multi-resolution discriminator (MRD). The generator is built from a set of look-up tables (LUT) that embed the discrete representation and a series of blocks composed of transposed convolution and a residual block with dilated layers. The transposed convolutions upsample the encoded representation to match the input sample rate.

Evaluation

To probe audio quality, we conduct the MOS-Q (mean opinion score) tests and explicitly instruct the raters to *"focus on examining the audio quality and naturalness."*. The testers present and rate the samples, and each tester is asked to evaluate the subjective naturalness on a 20-100 Likert scale.

To probe video-audio alignment, human raters are shown an audio and a video and asked *"Does the audio align with video faithfully?"*. They must respond with "completely", "mostly", or "somewhat" on a 20-100 Likert scale to score MOS-F.

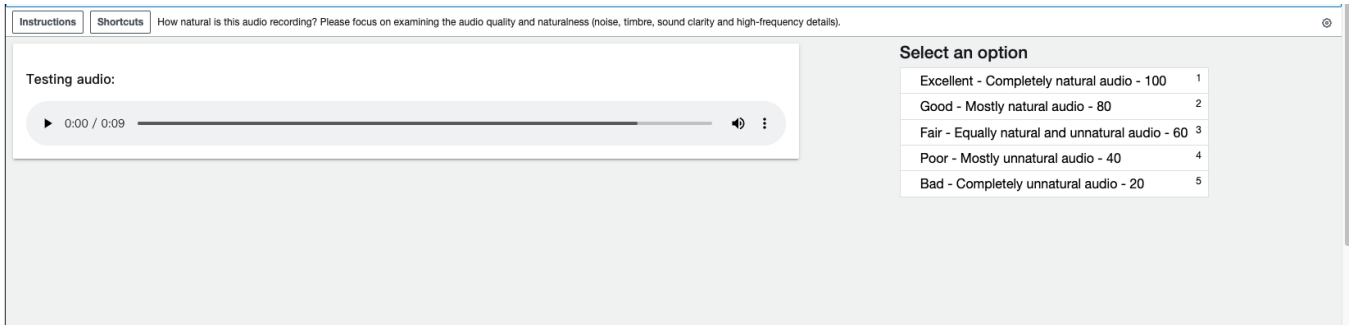
Our subjective evaluation tests are crowd-sourced and conducted via Amazon Mechanical Turk. These ratings are obtained independently for model samples and reference audio. The screenshots of instructions for testers have been shown in Figure 6. We paid $8 to participants hourly and totally spent about $600 on participant compensation. A small subset of audio samples used in the test is available at https://visual-audio-demo.github.io/.

More Visualization

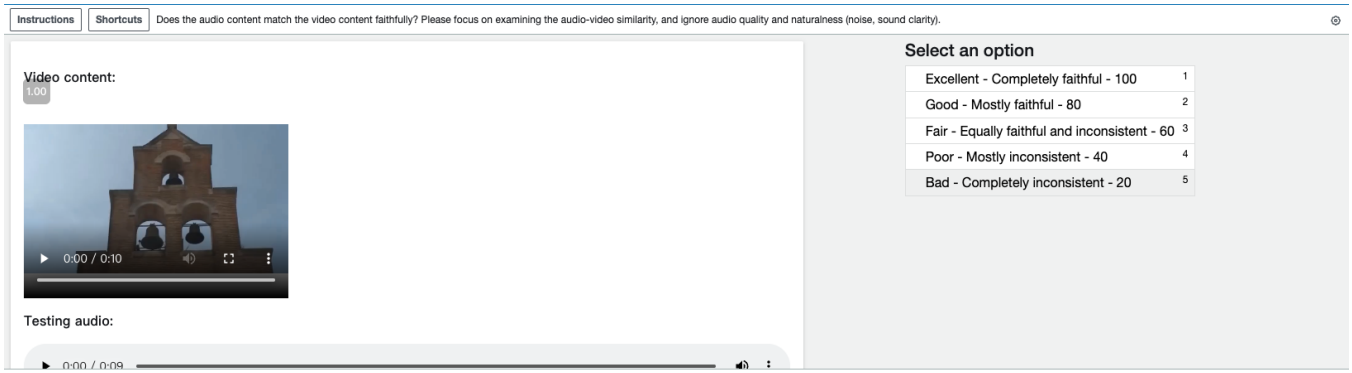
In this section, we put more visualizations of video-to-audio generation results.

Table 7: Hyperparameters of VisualAudio. We use T and F to denote the time and frequency moe layers respectively.

Hyperparameter		VisualAudio
M	Transformer Layer	8T+4F
	Transformer Embed Dim	768
	Transformer Attention Headers	12
	Number of Parameters	160 M
L	Transformer Layer	12T+12F
	Transformer Embed Dim	1024
	Transformer Attention Headers	16
	Number of Parameters	520 M
XL	Transformer Layer	16T+12F
	Transformer Embed Dim	1152
	Transformer Attention Headers	16
	Number of Parameters	750 M
BigVGAN Vocoder	Upsample Rates	[5, 4, 2, 2, 2, 2]
	Hop Size	320
	Upsample Kernel Sizes	[9, 8, 4, 4, 4, 4]
	Number of Parameters	121.6M



(a) Screenshot of MOS-Q evaluation.



(b) Screenshot of MOS-F evaluation.

Figure 6: Subjective evaluation.

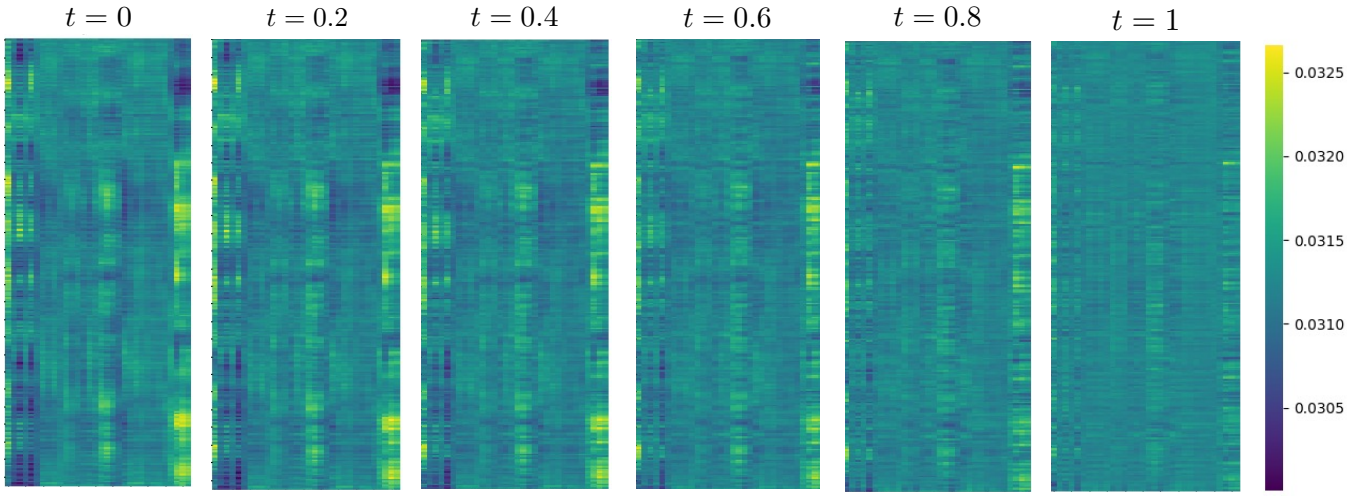
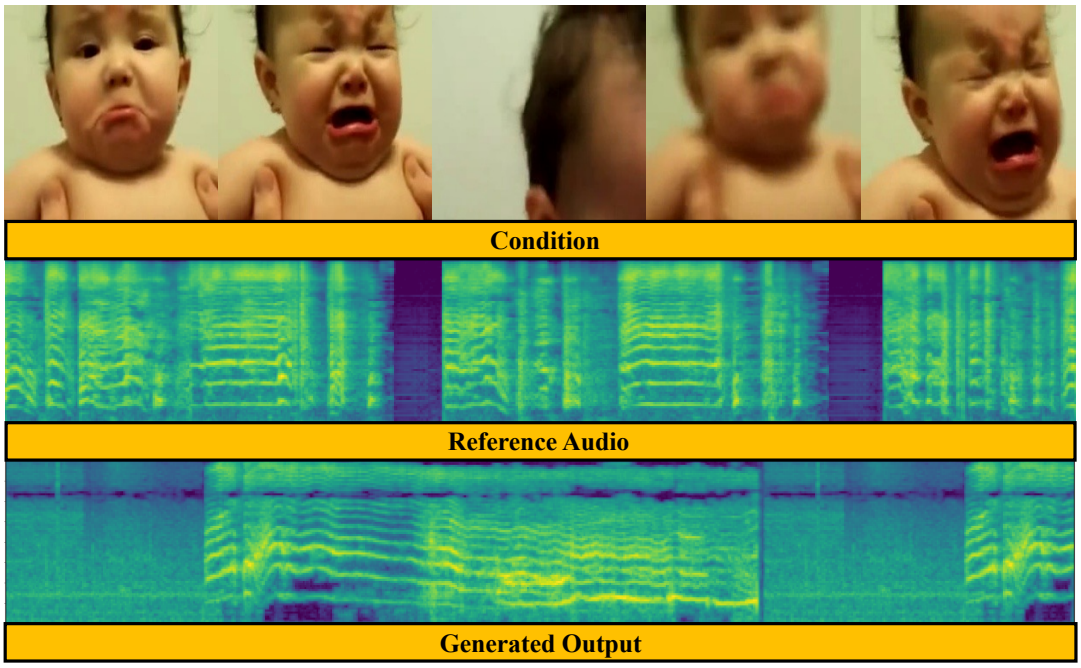
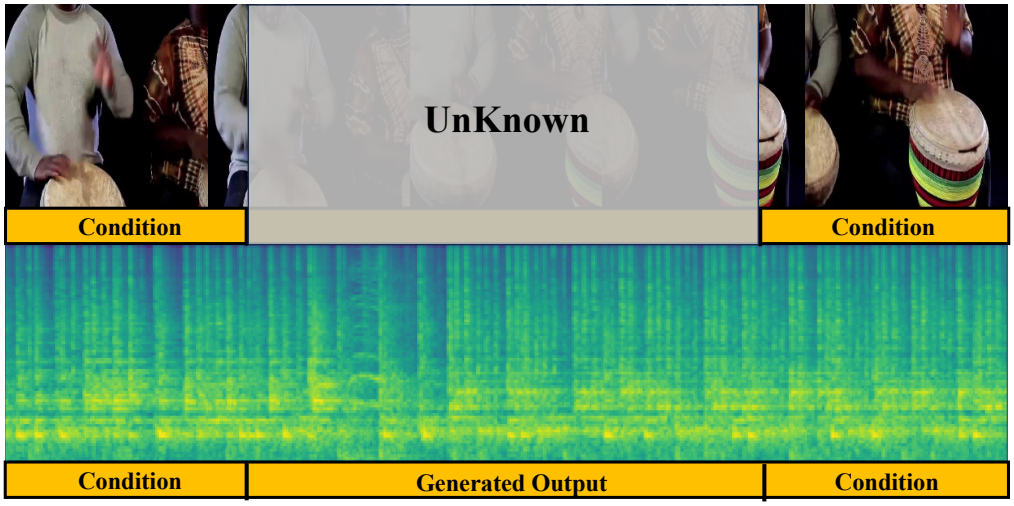


Figure 7: The visualization of cross-attention maps in different denoising timesteps $t \in [0, 1]$, where each value is the average of attention from this audio token to all video input.

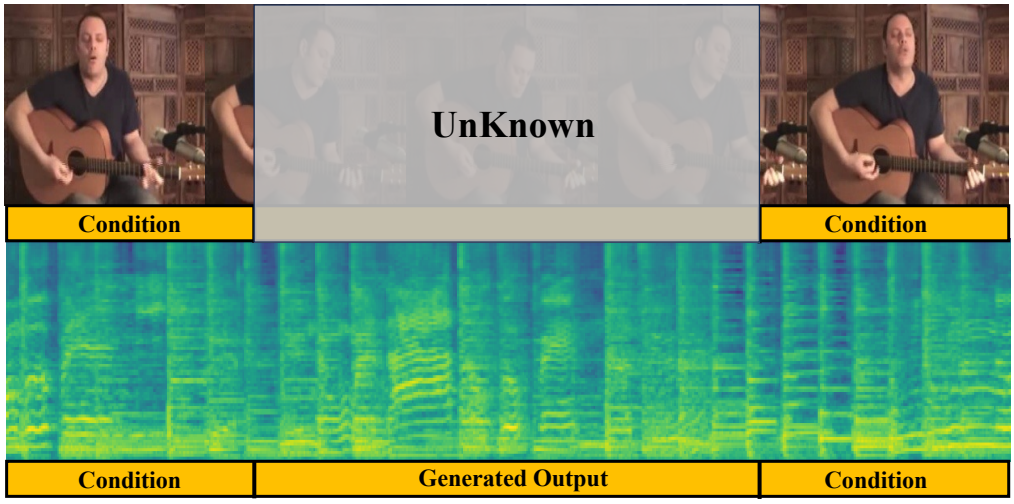


(a) Sample 1.

Figure 8: Visualizations of video-guided audio transfer.

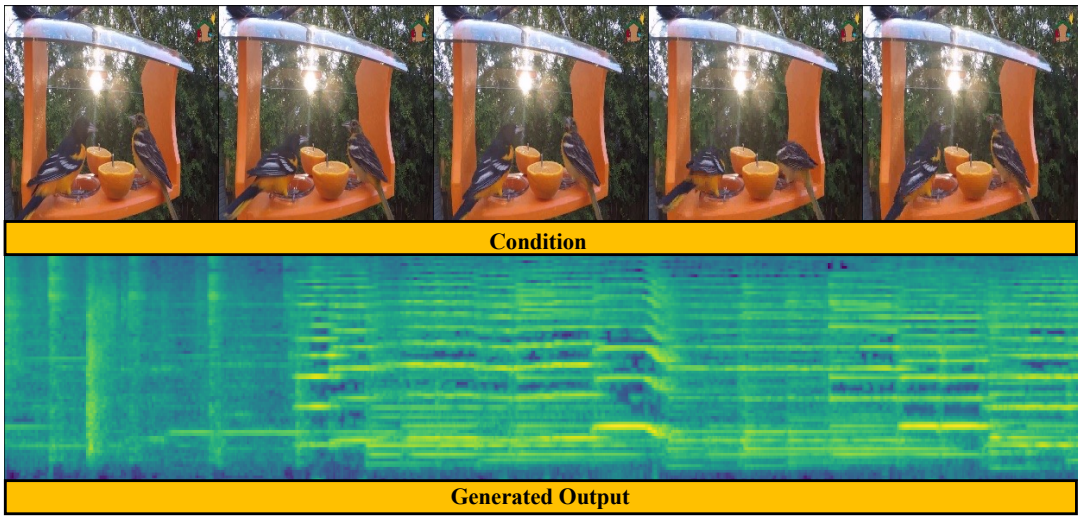


(a) Sample 1.

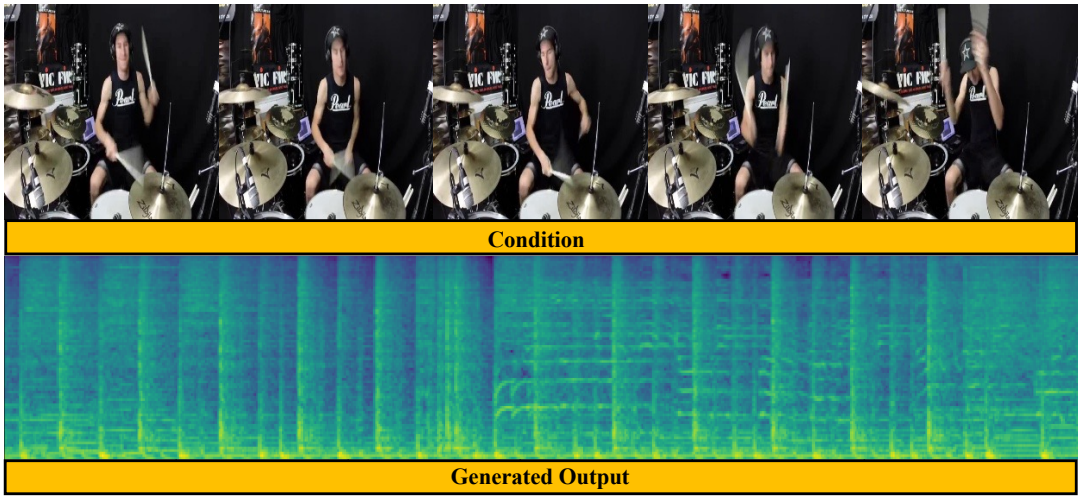


(b) Sample 2.

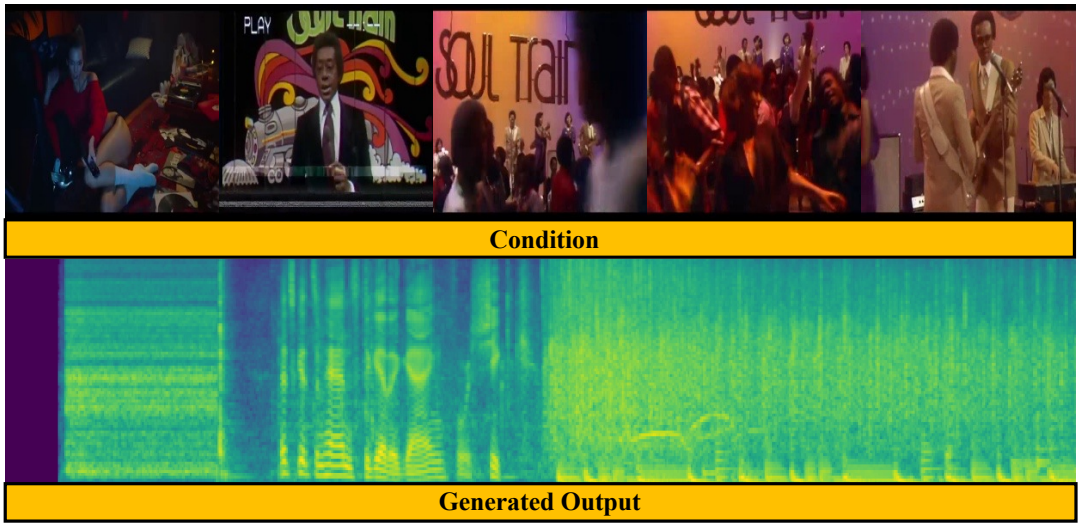
Figure 9: Visualizations of video-guided audio interpolation .



(a) Sample 1 (Landscape dataset).



(b) Sample 2 (AIST++ dataset).



(c) Sample 3 (MV100K dataset).

Figure 10: Visualizations of video-guided audio generation.