Conditional Generation of Audio from Video via Foley Analogies

Yuexi Du[1,2] Ziyang Chen[1] Justin Salamon[3] Bryan Russell[3] Andrew Owens[1]

[1]University of Michigan [2]Yale University [3]Adobe Research

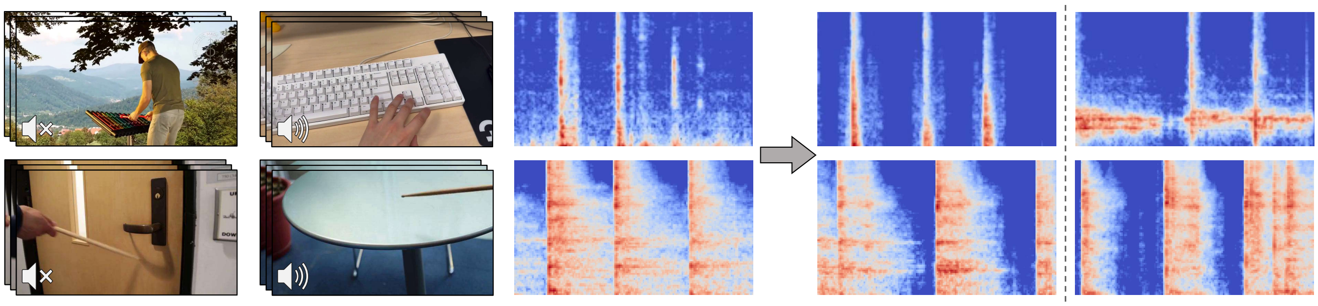


Figure 1. **Conditional Foley generation via analogy.** We generate a soundtrack for a silent input video, given a user-provided conditional example specifying what its audio should "sound like." In the first example, we make the xylophone strikes sound like the clicks of a mechanical keyboard. In the second, we generate a soundtrack for a video in which the drumstick striking a wooden door sounds as though it were made of metal. Notice that the shape of the sound events in the generated audio (*e.g.*, thin stripes in the top example) matches the conditional audio and the onsets match the input example's audio. For reference, we provide the input video's (held out) sound on the right. **We encourage the reader to watch and listen to the results on our project webpage**.

Abstract

The sound effects that designers add to videos are designed to convey a particular artistic effect and, thus, may be quite different from a scene's true sound. Inspired by the challenges of creating a soundtrack for a video that differs from its true sound, but that nonetheless matches the actions occurring on screen, we propose the problem of conditional Foley. *We present the following contributions to address this problem. First, we propose a pretext task for training our model to predict sound for an input video clip using a conditional audio-visual clip sampled from another time within the same source video. Second, we propose a model for generating a soundtrack for a silent input video, given a user-supplied example that specifies what the video should "sound like". We show through human studies and automated evaluation metrics that our model successfully generates sound from videos, while varying its output according to the content of a supplied example. Project site:* https://xypb.github.io/CondFoleyGen.

1. Introduction

When artists create sound effects for videos, they often "borrow" sounds from other sources, then manipulate them to match the on-screen actions. These artists' aim is not necessarily to convey the scene's true sound, but rather to achieve a desired artistic effect. Thus, the clunk of a coconut shell becomes a trotting horse, or the sizzle of cooking bacon becomes rain[1].

The problem of creating sound effects for video, known as Foley [1], has often been posed as predicting a video's co-occurring sound [29, 42, 68]. Yet the task that artists solve is subtly different. They create a soundtrack for a video that *differs* from its true sound, but that still plausibly matches the on-screen events. Also, these prior systems largely do not give artists control over the output sound.

To aid Foley artists while giving them artistic control, we propose a *conditional* Foley problem inspired by classic work on image analogies [27]. Our task is to generate a soundtrack for an input silent video from a user-provided conditional audio-visual example that specifies what the input video should "sound like." The generated soundtrack should relate to the input video in an analogous way as the provided example (Fig. 1). This formulation naturally separates the problem of selecting an exemplar sound, which arguably requires the artist's judgment, from the problem of manipulating that sound to match a video, such as by precisely adjusting its timing and timbre.

This proposed task is challenging, since a system must

[1]We encourage you to watch and listen to how sound artists work: https://www.youtube.com/watch?v=UO3N_PRIgX0

learn to adapt the exemplar (conditional) sound to match the timing of the visual content of a silent video while preserving the exemplar sound's timbre. While prior methods can predict a video's sound [29, 42, 68], they cannot incorporate an artist's exemplary conditional sound. Furthermore, while vision-to-sound methods can pose the problem as predicting a video's soundtrack from its images, it is less clear how supervision for conditional examples can be obtained.

To address these challenges, we contribute a self-supervised pretext task for learning conditional Foley, as well as a model for solving it. Our pretext task exploits the fact that natural videos tend to contain repeated events that produce closely related sounds. To train the model, we randomly sample two pairs of audio-visual clips from a video, and use one as the conditional example for the other. Our model learns to infer the types of actions within the scene from the conditional example, and to generate analogous sounds to match the input example. At test time, our model generalizes to conditional sounds obtained from other videos. To solve the task, we train a Transformer [58] to autoregressively predict a sequence of audio codes for a spectrogram VQGAN [13], while conditioning on the provided audio-visual example. We improve the model's performance at test time by generating a large number of soundtracks, then using an audio-visual synchronization model [8, 30, 41] to select the sound with the highest degree of temporal alignment with the video.

We evaluate our model on the *Greatest Hits* dataset [42], which contains videos that require an understanding of material properties and physical interactions, and via qualitative examples from the highly diverse *CountixAV* dataset [66]. Through perceptual studies and quantitative evaluations, we show that our model generates soundtracks that convey the physical properties of conditional examples while reflecting the timing and motions of the on-screen actions.

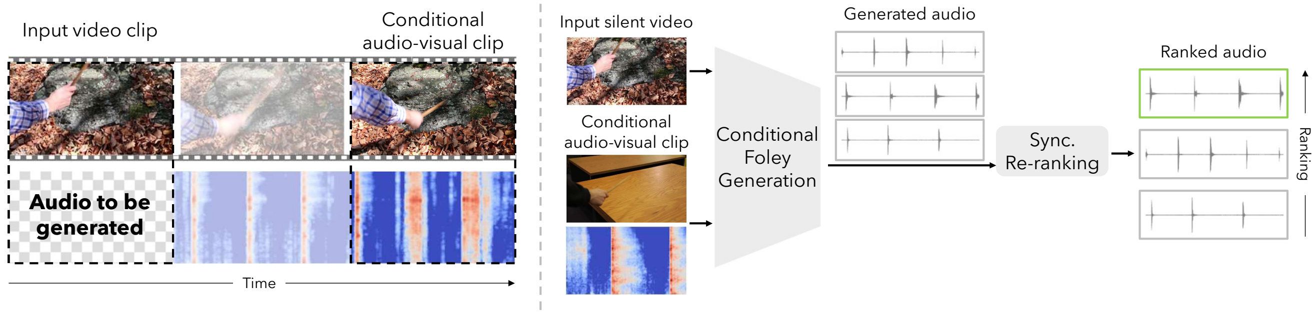
2. Related Work

Predicting sound from images and video. In early work, Van Den Doel *et al*. [56] generated sound for physical simulations. More recent examples include predicting soundtracks for videos in which someone strikes objects with a drumstick [42], generating music from piano [35], body motion [50], or dance videos [14, 51], and generating speech from lip motions [12, 44]. Other work predicted natural sounds (typically ambient sound) using an autoregressive vocoder [68], temporal relational networks [20], and visually guided generative adversarial network [21]. Iashin and Rahtu [29] recently used a VQGAN [13, 57] operating on mel spectrograms to generate sounds. We adopt this architecture to perform *conditioned* sound generation. In contrast to previous approaches, our goal is not simply to estimate the sound from a silent video, but to use a user-provided example to tailor the sound to the actions in a scene.

Sound design. The sounds that occur in a film are often not recorded on-site, but instead are inserted by artists. Sound designers perform a number of steps, including "spotting" visual events that require sound, choosing or recording an appropriate sound, and manipulating the chosen sound with editing software [1]. Our work addresses this final manipulation step. Other work has sought computational approaches to re-target or match visual signals to audio in sound design problems. Davis and Agrawala [10] re-targeted video to match a soundtrack by aligning both signals at estimated beats. Langlois and James [37] addressed the task of synchronizing vision to match sounds. Other work learns to match relevant music with videos [52]. However, they do not address the proposed conditional Foley task.

Interactive stylization. We take inspiration from the classic work of Hertzmann *et al*. [27], which learned to restyle input images from a single user-provided example of an image and its stylization. Like this work, we seek to generalize from one piece of paired data (a video and its sound) to another. Zhang *et al*. [65] learned to colorize images with simulated user-provided hints, using a self-supervised training procedure. In contrast, our model is given user-supplied hints of audio-visual examples from *other* videos, rather than from annotations of the input. A variety of methods have been proposed to stylize images with user-provided conditions. Li *et al*. [39] stylized input images based on user-provided sound. Lee *et al*. [38] generated images from sound and pretrained language models. Chen *et al*. [3] learned to alter the acoustics of speech sound to match a visual scene. Many recent methods have applied style transfer to audio. [55] These include methods that separate style and content using feature statistics [55, 60], following Gatys *et al*. [19], and methods that transfer musical timbre [28] based on CycleGAN [69]. In contrast, Foley generation requires generating sound without a ground truth audio, since in general there is no existing recorded sound [1] in Foley artists' workflow. Nistal *et al*. [40] propose to synthesize drum sounds using a GAN [22] conditioned on audio perceptual features rather than a complete audio-visual conditional clip.

Self-supervised and few-shot audio-visual learning. Our work aims to learn from data without human annotations. There has been much recent progress in learning strong audio-visual representations from video with accompanying audio tracks for other downstream applications [2, 6, 9, 11, 15–17, 24, 41, 54, 63, 64, 67]. Example applications include using audio-visual data for source separation [11, 15, 24, 41, 54, 67] and for converting mono sound to stereo [16, 17, 63, 64]. Our work focuses on a different application, but is related to methods that adapt themselves using a small number of labeled examples, such as audio event detection [61, 62], talking head generation [5], and diarization [7].



(a) Pretext task for training (b) Inference

Figure 2. **Conditional video-to-audio synthesis via Foley analogy**. (a) For our pretext task, we extract two clips from a longer video and train our model to predict the soundtrack for one, given audio-visual information from the other. Through this process, our model learns to condition its soundtrack predictions on other videos. (b) At test time, we provide the model with a silent input video and an audio-visual clip (taken from another video). We can use an audio-visual synchronization model to re-rank the generated soundtracks and choose the one with the best temporal alignment.

3. Method

Our goal is to generate a soundtrack for a silent input video, given a user-provided *conditional* audio-visual example that specifies what the video should "sound like". We learn a function $\mathcal{F}_\theta(\mathbf{v}_q, \mathbf{v}_c, \mathbf{a}_c)$ parameterized by θ that generates a soundtrack from an input video $\mathbf{v}_q$, given a conditional video $\mathbf{v}_c$ and conditional audio $\mathbf{a}_c$. We now describe our pretext task for training $\mathcal{F}_\theta$ from unlabeled data, and our conditional vision-to-sound model.

3.1. Pretext task for conditional prediction

We desire a pretext task that results in the model obtaining the necessary information from each source. In particular, we would like the input video to specify the type of action (*e.g.*, hitting vs. scratching an object) and its timing, while the conditional audio-visual example should specify the timbre of the generated sound (*e.g.*, the type of the materials that are being interacted with).

We define our task as a video-to-audio prediction problem in which another clip from the same video is provided as the conditional example (Fig. 2a). During training, we sample two clips from a longer video, centered at times t and $t + \Delta t$ respectively, using one as the conditional example and the other as the input video. The model is tasked with predicting the sound from the silent input video, using conditional clips as an additional input.

According to this pretext task, we can define a loss $\mathcal{L}$ over an audio target $\mathbf{a}_g$ and a prediction $\mathcal{F}_\theta(\mathbf{v}_q, \mathbf{v}_c, \mathbf{a}_c)$ given corresponding input video $\mathbf{v}_q$ and conditional audio-visual clip $(\mathbf{v}_c, \mathbf{a}_c)$:

$$\mathcal{L}(\mathbf{a}_g, \mathcal{F}_\theta(\mathbf{v}_q, \mathbf{v}_c, \mathbf{a}_c)) \tag{1}$$

This formulation exploits the fact that the actions within a video tend to be closely related [66] (or "self-similar" [49]), such as when an action is performed repeatedly. Thus randomly sampled pairs of clips frequently contain related actions. When this occurs, the model can use conditional sound to improve its prediction. However, the model cannot solve the task by simply "copying and pasting" the conditional sound, since it must account for the content of the timing of actions (and the type of motion) in the input video. Since the model is trained to assume that the conditional example is informative about the input, we empirically find that it learns to base its prediction on the conditional sound. This finding allows for substituting in a conditional sound sampled from a completely *different* video at test time (Fig. 2b).

3.2. Conditional sound prediction architecture

We describe our architecture of $\mathcal{F}_\theta$ for, first, obtaining a code representation for a target sound via VQGAN [13, 29] and, second, predicting an output sound for a given input video and conditional audio-visual pair.

Vector-quantized audio representation. We follow Iashin and Rahtu [29] and represent the predicted sound as a sequence of discrete codes, using a VQGAN [13, 57] that operates on mel spectrograms[2]. We learn this code by training an autoencoder to reconstruct sounds in a dataset, using the codes as its latent vector. After training, a predicted code sequence can subsequently be converted to a waveform.

Given a waveform $\mathbf{a}$ and its mel spectrogram $\mathrm{MSTFT}(\mathbf{a}) \in \mathbb{R}^{T \times F}$, we compute embeddings $\hat{\mathbf{z}} = E(\mathrm{MSTFT}(\mathbf{a})) \in \mathbb{R}^{T' \times F' \times d}$, where T' and F' define a lower-resolution time-frequency grid, d is the dimensionality of the embedding at each patch, and E is a CNN. Each embedding vector is then replaced with the nearest entry in a codebook $\{\mathbf{c}_k\}_{k=1}^{K}$:

$$\mathbf{z}_{tf} = q(\hat{\mathbf{z}}_{tf}) = \underset{\mathbf{c}_k}{\mathrm{argmin}}\, ||\hat{\mathbf{z}}_{tf} - \mathbf{c}_k||, \tag{2}$$

where $\hat{\mathbf{z}}_{tf}$ is the embedding at time-frequency index (t, f). We train the model to reconstruct the input sound $\hat{\mathbf{S}} = D(q(E(\mathrm{MSTFT}(\mathbf{a}))))$, where D is a CNN-based decoder and q is applied to every embedding. We use the loss function from [29], which adapts the VQGAN loss [13] to spectrograms, jointly minimizing a mean-squared error reconstruction loss [57], a perceptual loss [32], and a patch-based discriminator loss [31]. We provide details in the supp.

[2] We use *log* mel spectrograms unless otherwise noted.

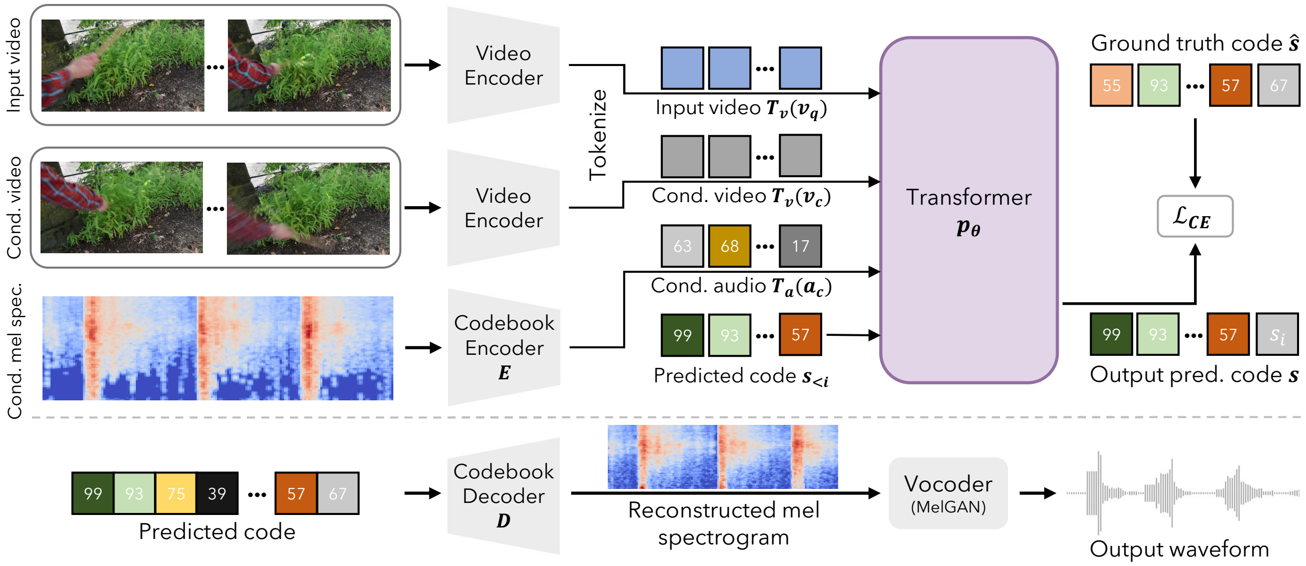


Figure 3. **Conditional Foley generation**. (Top) We predict the soundtrack for a silent video, conditioned on an audio-visual pair sampled from the same video. We encode and tokenize the video and audio signals, and feed them into a transformer. This transformer autoregressively predicts a code from a VQGAN [13, 29], representing the input example's sound. (Bottom) We generate a waveform by converting the code to a mel spectrogram, then using a MelGAN [36] vocoder to convert it to a waveform. Here, $\mathcal{L}_{CE}$ represents a cross-entropy loss.

Finally, we can obtain a *code* $s \in \{0, 1, \cdots, K-1\}^{T' \times F'}$ for a sound from the VQGAN by replacing each quantized vector in $\mathbf{z}$ with the index of its nearest codebook entry, *i.e.*, s_{tf} is the index of the selected codebook entry in Eq. (2).

Autoregressive sound prediction. With the predicted code s, we can now formulate the likelihood of generating code s from the silent input video and the conditional example. We order the indices of s in raster scan order [13] and predict them autoregressively:

$$p_\theta(s|\mathbf{v}_q, \mathbf{v}_c, \mathbf{a}_c) = \prod_i p_\theta(s_i|s_{<i}, \mathbf{v}_q, \mathbf{v}_c, \mathbf{a}_c), \qquad (3)$$

where $s_{<i}$ are the previous indices in the sequence. Given these probabilities, we formulate $\mathcal{L}$ (Eq. (1)) as the cross-entropy loss between the predicted token s_i and ground-truth token $\hat{s}_i$.

Having defined a code-based representation for sounds, we describe our architecture of $\mathcal{F}_\theta$ for conditional sound prediction (Fig. 3). Following [13, 29], we predict the code sequence (Eq. (3)) using a decoder-only transformer [59] based on GPT-2 [45]. The inputs to this transformer are tokenized versions of $\mathbf{v}_q$, $\mathbf{v}_c$, and $\mathbf{a}_c$. We now describe how these signals are converted into tokens.

Input representations. We represent each video signal using a ResNet (2+1)D-18 [53]. To preserve fine-grained temporal information, we remove all temporal striding, so that the final convolutional layer has the same temporal sampling rate as the input video. We perform average pooling over the spatial dimension, resulting in an embedding vector for each frame. Each such vector becomes a token. We denote this tokenization operation $T_v(\mathbf{v})$.

We represent the conditional audio signal using its vector-quantized embeddings. Specifically, we compute $\mathbf{z}^{(c)} = q(E(\mathrm{MSTFT}(\mathbf{a}_c))) \in \mathbb{R}^{T' \times F' \times d}$ (Eq. (2)) and extract its d-dimensional embedding vectors $\mathbf{z}_1^{(c)}, \mathbf{z}_2^{(c)}, \cdots, \mathbf{z}_N^{(c)}$ in raster-scan order. We denote this tokenization operation $T_a(\mathbf{a}_c)$.

We combine these tokens into a single sequence: $\mathcal{S} = \mathrm{Concat}(T_v(\mathbf{v}_c), T_v(\mathbf{v}_q), T_a(\mathbf{a}_c))$. Thus, we model $p_\theta(s|\mathbf{v}_q, \mathbf{v}_c, \mathbf{a}_c) = p_\theta(s|\mathcal{S})$. Following standard practice [13, 57], we generate the audio code autoregressively, feeding the previously generated codes back into the model using their vector-quantized representation.

Generating a waveform. Our complete model $\mathcal{F}_\theta$ works by first generating a code using a transformer, converting it to a mel spectrogram using the decoder D, then converting the mel spectrogram to a waveform. To perform this final step, we follow [29] and use a pretrained MelGAN vocoder [36]. We found that this produced significantly better results than standard Griffin-Lim [23].

Re-ranking based on audio-visual synchronization. Inspired by other work in cross-modal generation, we use *re-ranking* to improve our model's predictions [47]. We generate a large number of sounds, then select the best one, as judged by a separate classifier. Typically, these approaches use a model that judges the multimodal agreement between the input and output. In our case, however, such a classifier ought to consider conditionally generated sound to be a poor match for both the input and conditional videos. We instead propose to use an audio-visual synchronization model [8, 30, 41] to measure the temporal alignment between the predicted sound and the input video. These models predict a temporal offset that best aligns visual and audio data.

As shown in Fig. 2b, we use an off-the-shelf synchronization model [30] to estimate the offset t between the audio and video and the prediction's confidence. We find the min-

Model	Task							
	Material			Action			Onset	
	match Acc (%)	mismatch Acc (%)	overall Acc (%)	match Acc (%)	mismatch Acc (%)	overall Acc (%)	# onset Acc (%)	onset sync. AP (%)
Style transfer* [18, 55]	30.0	33.5	32.3	20.8	36.6	31.3	19.1	46.9
Onset transfer	**54.8**	**51.4**	**52.6**	**69.0**	**44.7**	**52.9**	**24.8**	**71.9**
Chance	5.9	5.9	5.9	50.0	50.0	50.0	–	–
SpecVQGAN [29]	25.4	26.8	26.1	52.3	43.1	46.2	11.3	51.0
SpecVQGAN - finetuned [29]	29.9	25.7	27.2	70.6	58.4	62.5	25.8	59.3
Ours - No cond.	21.3	24.9	23.7	61.4	55.1	57.2	24.6	59.3
Ours - Base	41.1	41.6	41.4	67.5	59.2	62.0	**26.5**	**60.0**
Ours - w/ re-rank	**43.4**	**45.2**	**44.0**	**78.2**	**61.3**	**66.7**	25.3	54.3

Table 1. **Automated evaluation metrics.** We measure the rate at that generated sounds have the material properties of the conditional examples, the actions of the input examples, and the number and the timing of the onsets in the generated sound with respect to the original sound. We further break down the automated metrics according to whether the conditional and input examples have the **matched** (or **mismatched**) actions and materials. The number of onsets is measured by whether the generated sound has the same number of onsets as the original sound. We measure the average precision of onset predictions that are within 0.1 seconds of the ground truth to evaluate the timing of the generated onsets. $*$ indicates that the model is an "oracle" and accesses the input example's sound.

imum absolute offset $\min|t|$ among all outputs. Then the outputs with an absolute offset greater than $\min|t| + \tau$ are removed, where τ is the offset tolerance. Finally, we select the sound with the highest confidence.

4. Experiments

To evaluate our method, we use a combination of automatic evaluation metrics and human perceptual studies.

4.1. Experiment Setup

Dataset. We train our conditional Foley generation model on datasets of video clips: *Greatest Hits* [42], which is composed of videos of a drumstick interacting with different objects in scenes, and *CountixAV* [66], which contains videos with as much as 23 different classes of repeated actions from in-the-wild YouTube video. These are challenging datasets for audio generation since they require precise timing, and varying sounds subtly based on fine-grained visual properties. In particular, the *Greatest Hits* task requires an understanding of the motion of the drumstick and the material properties of the objects. Since this dataset is straightforward to analyze in terms of actions and materials, we use it for our quantitative evaluation, while for *CountixAV* we provide qualitative results with permission to evaluate with similar videos in the wild. We provide more information about the dataset and the implementation details in the supplement.

Other models. We consider a variety of other models for comparison. First, we use the **SpecVQGAN** model of Iashin and Rahtu [29], a state-of-the-art vision-to-sound prediction method based on a two-stream visual network. We use the publicly released implementation and weight. We also finetuned the model on the *Greatest Hits* [42] dataset for the automated metrics as a fair comparison.

We also consider several ablations of our model: **No conditional example:** We remove all conditional information from the model. This model is a vision-to-sound prediction method that resembles SpecVQGAN [29] after controlling for architectural and data variations from our model. **No conditional video:** This model is provided with the conditional audio $\mathbf{a}_c$ but not the video $\mathbf{v}_c$, and hence cannot observe how the audio and visual events are connected in the conditional video. **No augmentation:** A model trained without audio augmentation. **Random conditional examples:** A model that is trained with conditional audio-visual clips that are unrelated to the input video, and hence uninformative. We select these clips randomly from other videos in the dataset. **Re-ranked examples:** We generate 100 outputs for each pair of input and condition, then re-rank them.

To better understand our model's behavior, we compare it against two "non-generative" methods. First, we propose a model called **Onset Transfer**. Instead of generating the sound, as our model does, this model uses a hand-crafted approach for transferring sounds from the conditional example. We train a ResNet (2+1)-D [53] model to detect audio onsets from video in both the conditional and input videos, then transfer sounds extracted from random onsets in the conditional example (see the supp. for more details). Second, we evaluated an audio **Style Transfer** method. We used the model of Ulyanov [55], which applies the stylization method of Gatys *et al.* [18] to spectrograms. We note that this method *requires audio* as input and is not designed for Foley generation. To address this, we provide the model with $\mathbf{a}_q$, the ground truth audio from the input video, thus giving it oracle information.

4.2. Automated Timbre Evaluation

A successful prediction method should accurately convey the actions in the input video but the material properties of the conditional example. To evaluate whether this is the case, trained classifiers to recognize the action (hit vs. scratch) and the material (a 17-way classification problem), using the labels in the *Greatest Hits* dataset [42]. We then used it to classify the predicted, conditional, and input sounds, and compared the estimated labels.

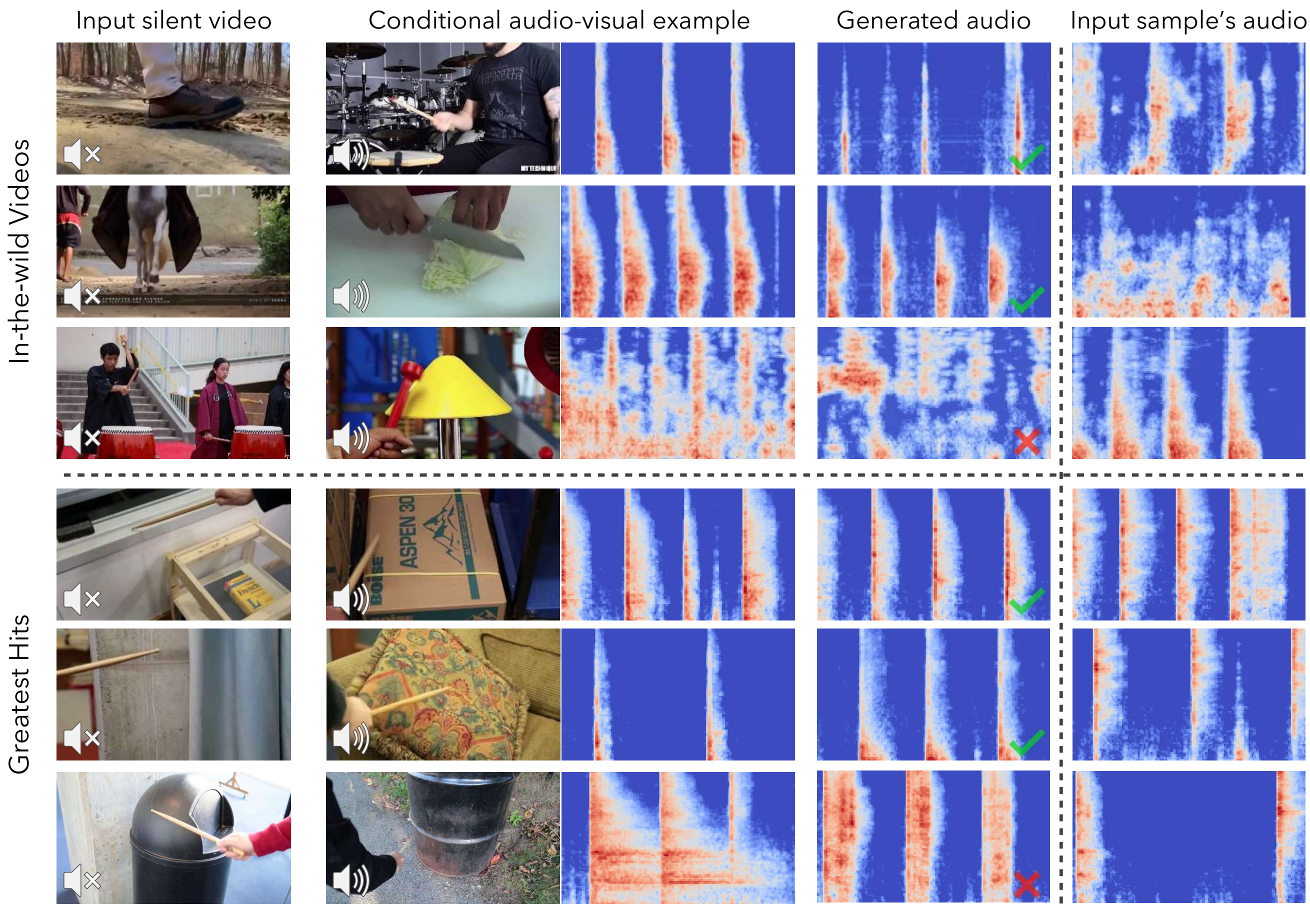


Figure 4. **Qualitative results**. We present results from our model. We show the result for the internet videos from the model trained on *CountixAV* dataset [66] (Row 1–3) and the *Greatest Hits* dataset [42] (Row 4–6). Rows 3 and 6 show failure cases with red crosses on generated audio. The timing and number of hits in the generated audio largely match that of the (held out) audio for the input video, suggesting the generated audio is matching the actions in the input video. The frequencies of the generated audio approximately match the conditional example, indicating a similar timbre. **To hear the sounds and see more examples, please refer to our project webpage**.

Sound classifier. We finetune a pretrained VGGish classifier [25,26] on the *Greatest Hits* [42] dataset to recognize the action or the material from a mel spectrogram. To avoid ambiguity, we only used clips that contained a single material or action type. We provide more details in the supplement.

Evaluation metrics. We used two evaluation metrics that capture our two criteria: *action accuracy*, the fraction of predicted sounds that have the same estimated action category as the (held out) input sound, and *material accuracy*, the fraction of predicted sounds that have the same estimated material category as the conditional example.

Results. We found (Tab. 1) that our model performed significantly better on the *material* metric than SpecVQGAN and than the variation without conditional examples, both of which are unconditional vision-to-sound prediction methods. As expected, the onset transfer model obtains near-optimal performance, since it simply transfers sounds from the conditional sound, which (trivially) are likely to have the same estimated category. On the other hand, this onset transfer baseline performs poorly on the *action* metric, since it has no mechanism for adapting the transferred sounds to the actions in the video (*e.g.*, converting hits to scratches). By contrast, our model obtains high performance on this metric.

To further understand the source of performance differences, we broke down the results according to whether the properties of the conditional example matched those of the input sound or not (Tab. 1). As expected, the onset transfer method performs strongly on material metrics, since "copy and pasting" conditioning sounds is a trivial solution. On the other hand, our generative approaches significantly outperform it when there is a mismatch between action types, since they can adapt the sound to match the action.

Additionally, we found that the synchronization reranking significantly boosted performance on both material and action tasks (Tab. 1). The re-ranked model outperforms the onset transfer method on all three action-related metrics. It also narrows the gap to the onset transfer method for material-related metrics. This demonstrates the effectiveness of the re-ranking method, as well as the advantage of posing our approach as a generative model.

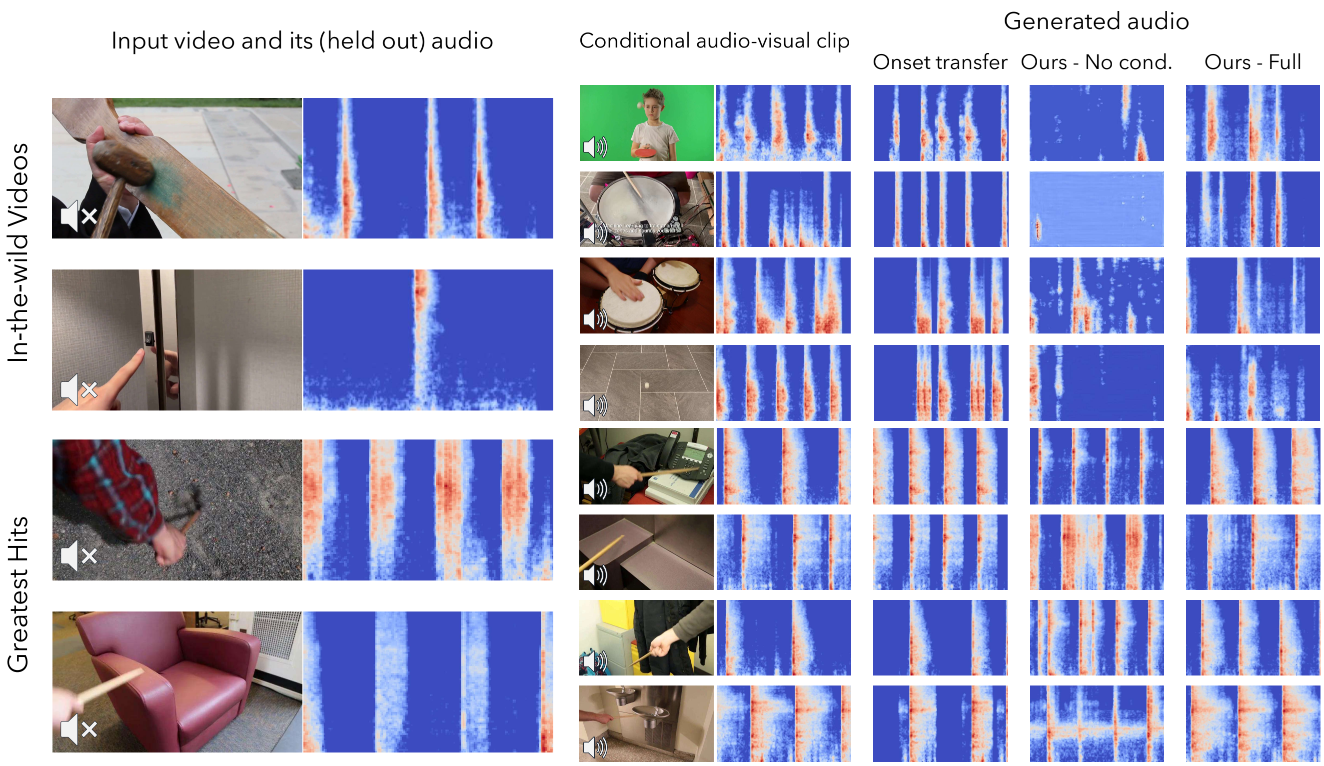


Figure 5. **Model comparison**. We show conditional Foley generation results for several models, using multiple conditional sounds. We show the result for publicly-sourced demonstration videos (Row 1–2) and the *Greatest Hits* dataset [42] (Row 3–4). Each of the input videos is paired with two different conditional videos. We provide 2 samples from the model variation with no conditional example.

4.3. Automated Onset Evaluation

To evaluate the quality of the generated timbre, we seek to evaluate whether the conditional Foley generation model generates sounds whose onsets match those of the (held-out) sound in the input video. We evaluate two criteria: whether the generated sound contains the correct number of onsets, and whether their timing matches those in the input video.

Evaluation metrics. We measure the fraction of video clips that contain the same number of onsets as the ground truth audio. Following Owens *et al*. [43], we report the average precision of detecting the correct onset where the relative wave amplitude provides the confidence of each onset. A detection is correct if it lies within a 0.1-second window of the ground truth.

Results. We notice (Tab. 1) that our model outperforms all the generative baselines and the Style transfer method in both metrics. It is not surprising that the onset transfer method obtains the best performance on the onset synchronization task, given that it is explicitly trained on a closely related task. The success of our model in generating the correct number of onset compared with the onset transfer model can be explained by the extra information about the audio-visual relation provided in the condition, which helps the model better understand the action in the video. As shown in Tab. 1, the accuracy in capturing the correct number of onsets drops to the same level as the onset transfer method if we remove conditional information. Interestingly, we find

Model	Variation	Task: Material Chosen(%) ↑	Task: Sync. Chosen(%) ↑
Style transfer* [18, 55]	–	9.9 (±2.3)	10.6 (±2.4)
Onset transfer	–	**64.7** (±3.8)	**57.3** (±3.9)
SpecVQGAN [29]	–	16.3 (±2.9)	18.0 (±3.0)
Ours	base	50.0 (±0.0)	50.0 (±0.0)
	- cond.	35.3 (±3.8)	40.1 (±4.0)
	- cond. video	46.1 (±4.0)	45.0 (±4.0)
	- augment	51.3 (±3.9)	49.7 (±4.0)
	w/ rand. cond.	45.5 (±4.0)	47.5 (±4.0)
	+ re-rank	**54.3** (±3.4)	**53.8** (±3.4)

Table 2. **Perceptual study results.** We report the rate at which participants chose a given method's results over our method (base) for the two questions in our study. For reference, we include the rate that our base method would obtain in the study (50%). We report results in terms of 95% confidence interval.

(Tab. 1) a drop in the performance for the re-ranked model. This may be due to the domain shift from the *VGG-Sound* dataset [4] that the synchronization network [30] was trained on, which may make it difficult to infer the precise timing of the sounds (rather than an overall assessment of whether the two streams are synchronized).

4.4. Perceptual Study

We also evaluated our model using a perceptual study, conducted using Amazon Mechanic Turk. We provide the participant with the conditional audio-visual clip and two input videos whose sound was generated by different models (our base model, plus a randomly chosen alternative).

We asked the participants to judge the generated sounds on two criteria that are similar to the automated metrics. Participants were asked to select: 1) in which result the audio is better synchronized with the actions in the video, 2) in which result the sound is most like that of the object or material in the conditional example. The 376 participants in our study were shown 21 sets of videos, randomly sampled from the evaluation set, the first 5 of which were used as practice and not counted.

Comparison to other variations of our model. We evaluate the influence of different variations of our model in terms of choosing the corresponding method in the perceptual study (Tab. 2). Our re-ranked model performs best overall and the base model without re-ranking also beats most of the baselines on both metrics. The model with no conditional example obtains poor performance on the material metric but obtains a relatively smaller decrease in synchronization performance. This is understandable, since the model did not use the conditional example but still was encouraged to be synchronized with the video. The model with no conditional video incurs a small drop in the material metrics and a relatively larger drop in synchronization, perhaps because it is unable to observe the relationship between images and sound in the conditional example.

Comparison to other approaches. We compare our model to other methods. Overall, the variation with no conditional example (which is a vision-to-sound method) outperforms SpecVQGAN [29]. The style transfer [18, 55] model performs poorly (and qualitatively often contains artifacts). Interestingly, the onset transfer model performs quite well on the perceptual study, outperforming our base model and the model with re-ranking, despite the fact that it does not tailor its output to the actions in the scene (Tab. 1). This is understandable, since it "copy and pastes" sounds from the conditional example at times that are chosen to be synchronized with impacts. Thus, the user is likely to observe that it exactly matches the conditional sounds and (trivially) conveys the same properties. However, this only occurs when the audio events are cleanly separated in time, and we expect the model to fail when sounds are not easily divided into discrete onsets, or when onsets are ambiguous.

4.5. Qualitative Results

We provide visualizations of predicted sounds from our test set in Fig. 4. Through this visualization, we can see that our model successfully generates sounds that resemble spectral properties of the conditional sound, while matching the timing and actions in the input video's (held out) sound. We also present the qualitative result (Fig. 4) of our method on the wild videos from the model trained on the *CountixAV* dataset [66]. We follow the same training and generation scheme as for the *Greatest Hits* [42]. The model again generates sounds with matching conditional spectral features and input action synchronization.

In Fig. 5, we visualize how our results vary as a function of the conditional sound and compare our model with the baselines on both datasets. For each input video, we show predictions from different models. We see that our model varies its output depending on the conditional sound (*e.g.*, varying its output based on whether the conditional example is plastic or metal). We see that the onset transfer method "copies and pastes" sounds from the conditional example at the correct times. We also observe the failure of the onset transfer baseline (see Fig. 5 Video in the wild part) in a more realistic scenario, where actions and sound in it are more abundant and complex. We show two random samples from the model with no conditioning. We see that the prediction generally matches the input sound, rather than the conditioning sound, and that there are large amounts of variation in the generated audio's timbre. Please refer to the supp. to listen to our outputs and for more qualitative results.

5. Discussion

In this paper, we proposed the *conditional Foley* task. We also proposed a method for solving this problem through self-supervised learning. We evaluated our method on the *Greatest Hits* dataset, finding through perceptual studies and automated metrics that our model successfully learns to transfer relevant information from a conditional sound, while matching the events within the silent input video. We also demonstrate the effectiveness of the model on more complex and realistic data from publicly-sourced videos with the model trained on the *CountixAV* dataset.

We see our work potentially opening several directions. Our work tackled one step of the sound design process—the process of manipulating sound to match a video. We see this as a step toward the broader goal of semi-automated "user in the loop" sound design. We also see our work as a step toward synthesis methods that can learn by analogy, in the tradition of classic work such as image analogies [27]. We will release code and models on our project site.

Limitations and Broader Impacts. While soundtrack generation is useful for creative applications, such as film making, it can also be used to create videos that can potentially be used to create disinformation, which can have negative outcomes. The field of image and audio forensics can help mitigate this outcome.

Acknowledgements. We thank Jon Gillick, Daniel Geng, and Chao Feng for the helpful discussions. This work was funded in part by DARPA Semafor and Cisco Systems, and by a gift from Adobe. The views, opinions and/or findings expressed are those of the authors and should not be interpreted as representing the official views or policies of the Department of Defense or the U.S. Government.

References

[1] Vanessa Theme Ament. *The Foley grail: The art of performing sound for film, games, and animation*. Routledge, 2014. 1, 2

[2] Relja Arandjelovic and Andrew Zisserman. Objects that sound. In *Proceedings of the European conference on computer vision (ECCV)*, pages 435–451, 2018. 2

[3] Changan Chen, Ruohan Gao, Paul Calamia, and Kristen Grauman. Visual acoustic matching. *arXiv preprint arXiv:2202.06875*, 2022. 2

[4] 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)*, pages 721–725. IEEE, 2020. 7

[5] Lele Chen, Guofeng Cui, Celong Liu, Zhong Li, Ziyi Kou, Yi Xu, and Chenliang Xu. Talking-head generation with rhythmic head motion. In *European Conference on Computer Vision*, pages 35–51. Springer, 2020. 2

[6] Ziyang Chen, Xixi Hu, and Andrew Owens. Structure from silence: Learning scene structure from ambient sound. In *5th Annual Conference on Robot Learning*, 2021. 2

[7] Joon Son Chung, Bong-Jin Lee, and Icksang Han. Who said that?: Audio-visual speaker diarisation of real-world meetings. *arXiv preprint arXiv:1906.10042*, 2019. 2

[8] Joon Son Chung and Andrew Zisserman. Out of time: automated lip sync in the wild. In *Asian conference on computer vision*, pages 251–263. Springer, 2016. 2, 4

[9] Chenye Cui, Yi Ren, Jinglin Liu, Rongjie Huang, and Zhou Zhao. Varietysound: Timbre-controllable video to sound generation via unsupervised information disentanglement. *arXiv preprint arXiv:2211.10666*, 2022. 2

[10] Abe Davis and Maneesh Agrawala. Visual rhythm and beat. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 2532–2535, 2018. 2

[11] Ariel Ephrat, Inbar Mosseri, Oran Lang, Tali Dekel, Kevin Wilson, Avinatan Hassidim, William T Freeman, and Michael Rubinstein. Looking to listen at the cocktail party: A speaker-independent audio-visual model for speech separation. *SIGGRAPH*, 2018. 2

[12] Ariel Ephrat and Shmuel Peleg. Vid2speech: speech reconstruction from silent video. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5095–5099. IEEE, 2017. 2

[13] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12873–12883, 2021. 2, 3, 4, 1

[14] Chuang Gan, Deng Huang, Peihao Chen, Joshua B Tenenbaum, and Antonio Torralba. Foley music: Learning to generate music from videos. In *European Conference on Computer Vision*, pages 758–775. Springer, 2020. 2

[15] Ruohan Gao, Rogerio Feris, and Kristen Grauman. Learning to separate object sounds by watching unlabeled video. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 35–53, 2018. 2

[16] Ruohan Gao and Kristen Grauman. 2.5 d visual sound. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 324–333, 2019. 2

[17] Rishabh Garg, Ruohan Gao, and Kristen Grauman. Geometry-aware multi-task learning for binaural audio generation from video. *arXiv preprint arXiv:2111.10882*, 2021. 2

[18] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. A neural algorithm of artistic style. *arXiv preprint arXiv:1508.06576*, 2015. 5, 7, 8, 2

[19] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2414–2423, 2016. 2

[20] Sanchita Ghose and John Jeffrey Prevost. Autofoley: Artificial synthesis of synchronized sound tracks for silent videos with deep learning. *IEEE Transactions on Multimedia*, 23:1895–1907, 2020. 2

[21] Sanchita Ghose and John J Prevost. Foleygan: Visually guided generative adversarial network-based synchronous sound generation in silent videos. *arXiv preprint arXiv:2107.09262*, 2021. 2

[22] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, pages 2672–2680, 2014. 2

[23] 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. 4

[24] John Hershey and Michael Casey. Audio-visual sound separation via hidden markov models. *Advances in Neural Information Processing Systems*, 14, 2001. 2

[25] Shawn Hershey, Sourish Chaudhuri, Daniel PW Ellis, Jort F Gemmeke, Aren Jansen, R Channing Moore, Manoj Plakal, Devin Platt, Rif A Saurous, Bryan Seybold, et al. Cnn architectures for large-scale audio classification. *arXiv preprint arXiv:1609.09430*, 2016. 6, 2

[26] Shawn Hershey, Sourish Chaudhuri, Daniel P. W. Ellis, Jort F. Gemmeke, Aren Jansen, Channing Moore, Manoj Plakal, Devin Platt, Rif A. Saurous, Bryan Seybold, Malcolm Slaney, Ron Weiss, and Kevin Wilson. Cnn architectures for large-scale audio classification. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2017. 6, 2

[27] Aaron Hertzmann, Charles E Jacobs, Nuria Oliver, Brian Curless, and David H Salesin. Image analogies. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 327–340, 2001. 1, 2, 8

[28] Sicong Huang, Qiyang Li, Cem Anil, Xuchan Bao, Sageev Oore, and Roger B Grosse. Timbretron: A wavenet (cyclegan (cqt (audio))) pipeline for musical timbre transfer. *arXiv preprint arXiv:1811.09620*, 2018. 2

[29] Vladimir Iashin and Esa Rahtu. Taming visually guided sound generation. *arXiv preprint arXiv:2110.08791*, 2021. 1, 2, 3, 4, 5, 7, 8

[30] Vladimir Iashin, Weidi Xie, Esa Rahtu, and Andrew Zisserman. Sparse in space and time: Audio-visual synchronisation

with trainable selectors. *arXiv preprint arXiv:2210.07055*, 2022. 2, 4, 7

[31] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017. 3, 1

[32] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution, 2016. 3, 1

[33] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. International Conference on Learning Representation, 2015. 1, 2

[34] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 1

[35] A Sophia Koepke, Olivia Wiles, Yael Moses, and Andrew Zisserman. Sight to sound: An end-to-end approach for visual piano transcription. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1838–1842. IEEE, 2020. 2

[36] Kundan Kumar, Rithesh Kumar, Thibault de Boissiere, Lucas Gestin, Wei Zhen Teoh, Jose Sotelo, Alexandre de Brébisson, Yoshua Bengio, and Aaron C Courville. Melgan: Generative adversarial networks for conditional waveform synthesis. *Advances in neural information processing systems*, 32, 2019. 4

[37] Timothy R Langlois and Doug L James. Inverse-foley animation: synchronizing rigid-body motions to sound. *ACM Transactions on Graphics (TOG)*, 33(4):41, 2014. 2

[38] Seung Hyun Lee, Wonseok Roh, Wonmin Byeon, Sang Ho Yoon, Chan Young Kim, Jinkyu Kim, and Sangpil Kim. Sound-guided semantic image manipulation. *arXiv preprint arXiv:2112.00007*, 2021. 2

[39] Tingle Li, Yichen Liu, Andrew Owens, and Hang Zhao. Learning visual styles from audio-visual associations. *arXiv*, 2022. 2

[40] Javier Nistal, Stefan Lattner, and Gael Richard. Drumgan: Synthesis of drum sounds with timbral feature conditioning using generative adversarial networks. *arXiv preprint arXiv:2008.12073*, 2020. 2

[41] Andrew Owens and Alexei A Efros. Audio-visual scene analysis with self-supervised multisensory features. *European Conference on Computer Vision (ECCV)*, 2018. 2, 4

[42] Andrew Owens, Phillip Isola, Josh McDermott, Antonio Torralba, Edward H Adelson, and William T Freeman. Visually indicated sounds. *CVPR*, 2016. 1, 2, 5, 6, 7, 8

[43] Andrew Owens, Phillip Isola, Josh McDermott, Antonio Torralba, Edward H Adelson, and William T Freeman. Visually indicated sounds. In *Computer Vision and Pattern Recognition (CVPR)*, 2016. 7

[44] KR Prajwal, Rudrabha Mukhopadhyay, Vinay P Namboodiri, and CV Jawahar. Learning individual speaking styles for accurate lip to speech synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13796–13805, 2020. 2

[45] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. 2019. 4

[46] Colin Raffel, Brian McFee, Eric J Humphrey, Justin Salamon, Oriol Nieto, Dawen Liang, Daniel PW Ellis, and C Colin Raffel. Mir_eval: A transparent implementation of common mir metrics. In *ISMIR*, pages 367–372, 2014. 2

[47] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pages 8821–8831. PMLR, 2021. 4

[48] Tim Sainburg, Marvin Thielk, and Timothy Q Gentner. Finding, visualizing, and quantifying latent structure across diverse animal vocal repertoires. *PLoS computational biology*, 16(10):e1008228, 2020. 1

[49] Eli Shechtman and Michal Irani. Matching local self-similarities across images and videos. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2007. 3

[50] Kun Su, Xiulong Liu, and Eli Shlizerman. Multi-instrumentalist net: Unsupervised generation of music from body movements. *arXiv preprint arXiv:2012.03478*, 2020. 2

[51] Kun Su, Xiulong Liu, and Eli Shlizerman. How does it sound? *Advances in Neural Information Processing Systems*, 34, 2021. 2

[52] Dídac Surís, Carl Vondrick, Bryan Russell, and Justin Salamon. It's time for artistic correspondence in music and video. *CVPR*, 2022. 2

[53] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6450–6459, 2018. 4, 5, 2

[54] Efthymios Tzinis, Scott Wisdom, Aren Jansen, Shawn Hershey, Tal Remez, Daniel PW Ellis, and John R Hershey. Into the wild with audioscope: Unsupervised audio-visual separation of on-screen sounds. *arXiv preprint arXiv:2011.01143*, 2020. 2

[55] Dmitry Ulyanov. Audio texture synthesis and style transfer. https://dmitryulyanov.github.io/audio-texture-synthesis-and-style-transfer, 2016. 2, 5, 7, 8

[56] Kees Van Den Doel, Paul G Kry, and Dinesh K Pai. Foleyautomatic: physically-based sound effects for interactive simulation and animation. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 537–544. ACM, 2001. 2

[57] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017. 2, 3, 4, 1

[58] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017. 2

[59] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Neural Information Processing Systems (NIPS)*, 2017. 4

[60] Prateek Verma and Julius O Smith. Neural style transfer for audio spectograms. *arXiv preprint arXiv:1801.01589*, 2018. 2

[61] Yu Wang, Nicholas J Bryan, Justin Salamon, Mark Cartwright, and Juan Pablo Bello. Who calls the shots? rethinking few-shot learning for audio. In *2021 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pages 36–40. IEEE, 2021. 2

[62] Yu Wang, Justin Salamon, Nicholas J Bryan, and Juan Pablo Bello. Few-shot sound event detection. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 81–85. IEEE, 2020. 2

[63] Xudong Xu, Hang Zhou, Ziwei Liu, Bo Dai, Xiaogang Wang, and Dahua Lin. Visually informed binaural audio generation without binaural audios. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15485–15494, 2021. 2

[64] Karren Yang, Bryan Russell, and Justin Salamon. Telling left from right: Learning spatial correspondence of sight and sound. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9932–9941, 2020. 2

[65] Richard Zhang, Jun-Yan Zhu, Phillip Isola, Xinyang Geng, Angela S Lin, Tianhe Yu, and Alexei A Efros. Real-time user-guided image colorization with learned deep priors. *arXiv preprint arXiv:1705.02999*, 2017. 2

[66] Yunhua Zhang, Ling Shao, and Cees GM Snoek. Repetitive activity counting by sight and sound. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14070–14079, 2021. 2, 3, 5, 6, 8, 1

[67] Hang Zhao, Chuang Gan, Andrew Rouditchenko, Carl Vondrick, Josh McDermott, and Antonio Torralba. The sound of pixels. In *Proceedings of the European conference on computer vision (ECCV)*, pages 570–586, 2018. 2

[68] Yipin Zhou, Zhaowen Wang, Chen Fang, Trung Bui, and Tamara L Berg. Visual to sound: Generating natural sound for videos in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3550–3558, 2018. 1, 2

[69] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. 2017. 2