The Sound of Motions

Hang Zhao[1], Chuang Gan[2], Wei-Chiu Ma[1], Antonio Torralba[1]
[1]MIT [2]MIT-IBM Watson AI Lab
{hangzhao,chuangg,weichium,torralba}@mit.edu

Abstract

Sounds originate from object motions and vibrations of surrounding air. Inspired by the fact that humans is capable of interpreting sound sources from how objects move visually, we propose a novel system that explicitly captures such motion cues for the task of sound localization and separation. Our system is composed of an end-to-end learnable model called Deep Dense Trajectory (DDT), and a curriculum learning scheme. It exploits the inherent coherence of audio-visual signals from a large quantities of unlabeled videos. Quantitative and qualitative evaluations show that comparing to previous models that rely on visual appearance cues, our motion based system improves performance in separating musical instrument sounds. Furthermore, it separates sound components from duets of the same category of instruments, a challenging problem that has not been addressed before.

1. Introduction

In a scorching afternoon, you relax under the shadow of a tree and enjoy the breeze. You notice that the tree branches are *vibrating* and you hear a *rustling sound*. Without a second thought, you realize that the sound is caused by the leaves *rubbing* one another. Despite a short notice, humans have the remarkable ability to connect and integrate signals from different modalities and perceptual inputs. In fact, the interplay among senses are one of the most ancient scheme of sensory organization in human brains [44] and is the key to understand the complex interaction of the physical world.

With such inspiration in mind, researchers have been painstakingly developing models that can effectively exploit signals from different modalities. Take audio-visual learning for example, various approaches have been proposed such as sound recognition [3, 1, 26], sound localization [22, 24, 33, 2, 12], *etc*. In this work, we are particularly interested in the task of sound source separation [12, 53, 17], where the goal is to distinguish the components of the sound and associate them with the corresponding objects. While current source separation methods achieve decent results on respective tasks, they often ignore the motion cues and simply rely on the static visual information. Motion signals, however, are of crucial importance for audio-visual learning, in particular when the objects making sounds are visu-

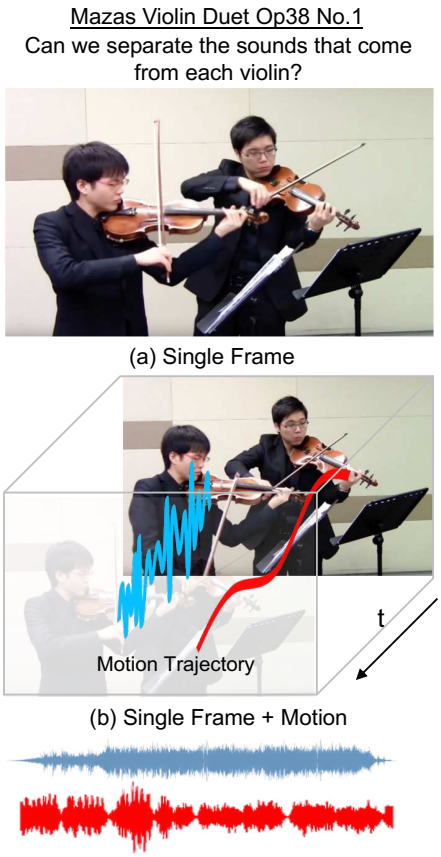

Figure 1. Motion matters: When watching a violin duet video, we can separate the melody from harmony. (a) Yet it is hard to tell the sources without looking or with only one glance. (b) By watching for a bit longer, we can differentiate who is playing the first violin and who is playing the second by associating their motions with the tempo of the music. In this work, we take inspirations from human to disambiguate and separate the sounds from multiple sources by exploring motion cues.

ally similar. Consider a case where two people are playing violin duets, as depicted in Figure 1. It is virtually impossible for humans to separate their melody from harmony by peaking at a single image. Yet if we see the movement of each person for a while, we can probably conjecture according to the temporal repetition of the motions and the beats of music. This illustration serves to highlight the importance of motion cues in the complex multi-modal reasoning. Our goal is to mimic, computationally, the ability to reason about the synergy between visual, audio, and motion signals[1].

We build our model upon previous success of Zhao *et al*. [53]. Instead of relying on image semantics, we explicitly consider the temporal motion information in the video. In particular, we propose an end-to-end learnable network architecture called Deep Dense Trajectory (DDT) to learn the motion cues necessary for the audio-visual sound separation. As the interplay among different modalities are very complex, we further develop a curriculum learning scheme. By starting from different instruments and then moving towards same types, we force the model to exploit motion cues for differentiation.

We demonstrate the effectiveness of our model on two recently proposed musical instrument datasets, MUSIC [53] and URMP [28]. Experiments show that by explicitly modeling the motion information, our approach improves prior art on the task of audio-visual sound source separation. More importantly, our model is able to handle extremely challenging scenarios, such as duets of the same instruments, where previous approaches failed significantly.

2. Related Work

Sound source separation. Sound source separation is a challenging classic problem, and is known as the "cocktail party problem" [30, 19] in the speech area. Algorithms based on Non-negative Matrix Factorization (NMF) [47, 10, 43] were the major solutions to this problem. More recently, several deep learning methods have been proposed, where Wang *et al*. gave an overview [48] on this series of approaches. Simpson *et al*. [42] and Chandna *et al*. [8] used CNNs to predict time-frequency masks for music source separation and enhancement. To solve the identity permutation problem in speech separation, Hershey *et al*. [21] proposed a deep learning-based clustering method, and Yu *et al*. [52] proposed a speaker-independent training scheme. While these solutions are inspiring, our setting is different from the previous ones in that we use additional visual signals to help with sound source separation.

[1]We encourage the readers to watch the video `https://www.youtube.com/watch?v=XDuKWUYfA_U` to get a better sense of the difficulty of this task.

Audio-visual learning. Learning the correspondences between vision and sound has become a popular topic recently. One line of work has explored representation learning from audio-visual training. Owens *et al*. [35] used sound signals as supervision for vision model training; Aytar *et al*. [3] used vision as supervision for sound models; Arandjelovic *et al*. [1] and Korbar *et al*. [26] trained vision and sound models jointly and achieve superior results. Another line of work explored sound localization in the visual input [23, 22, 2, 40, 53]. More recently, researchers used voices and faces to do biometric matching [32], generated sounds for videos [56], generated talking faces [55], segmented images and audios jointly [39], and predicted stereo sounds [18] or 360 ambisonics [31] from videos.

Although a few recent papers have demonstrated how visual cues could help with music separation [53, 17], their visual cues mostly come from appearance, which can be obtained from a single video frame. Our work differentiate from those in that we explicitly model motion cues, to make good use of the video input.

Sounds and motions. Early works in vision and audition have explored the strong relations between sounds and motions. Fisher *et al*. [14] used a maximal mutual information approach and Kidron *et al*. [24, 23] proposed variations of canonical correlation methods to discover such relations.

Lip motion is a useful cue in the speech processing domain, Gabbay *et al*. [15] used it for speech denoising; Chung *et al*. [9] demonstrated lip reading from face videos. Ephrat *et al*. [12] and Owens *et al*. [34] demonstrated speech separation and enhancement from videos.

The most related work to ours is [4], which claimed the tight associations between audio and visual onset signals, and use the signals to perform audio-visual sound attribution. In this work, we generalize their idea by learning an aligned audio-visual representations for sound separation.

Motion representation for videos. Our work is in part related to motion representation learning for videos, as we are working on videos of actions. Traditional techniques mainly use handcrafted spatio-temporal features, like space-time interest points [27], HOG3D [25], dense trajectories [49], improved dense trajectories [50] as the motion representations of videos. Recently, works have shifted to learning representations using deep neural networks. There are three kinds of successful architectures to capture motion and temporal information in videos: (1) two-stream CNNs [41], where motion information is modeled by taking optical flow frames as network inputs; (2) 3D CNNs [46], which performs 3D convolutions over the spatio-temporal video volume; (3) 2D CNNs with temporal models on top such as LSTM [11], Attention [29, 5], Graph CNNs [51], *etc*. More recently, researchers proposed

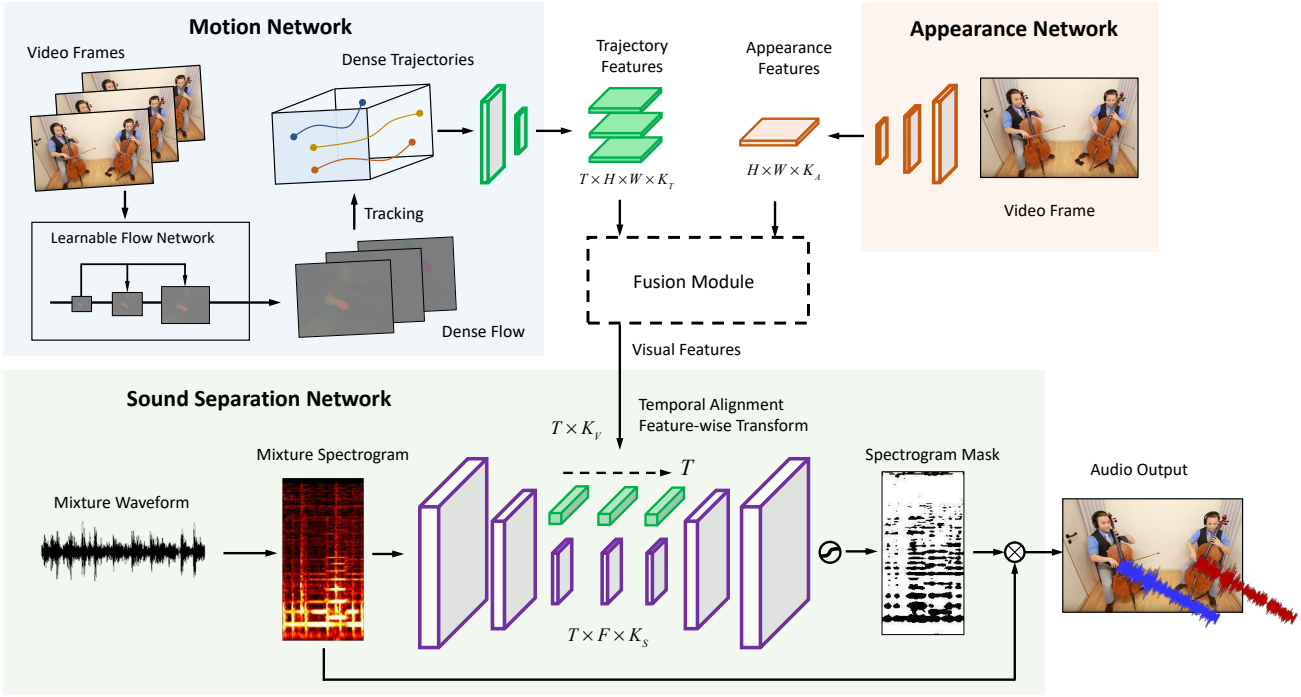

Figure 2. An overview of model architecture. Our framework is consist of four components: a motion network, an appearance network, a fusion module, and a sound separation network. The motion network takes a sequence of frames and outputs trajectory features; appearance network takes the first video frame and outputs appearance features; fusion module fuses appearance and trajectory features; sound separation network separates the input audio conditioned on the visual features.

to learn motion/trajectory representations for action recognition [13, 54, 16]. In contrast to action recognition or localization, our goal is to find correspondence between sound components and movements in videos.

3. Approach

In this section, we first introduce the mix-and-separate framework we used for the audio-visual sound separation. Then we present the model architecture we used for learning motion representations for audio-visual scene analysis. Finally, we introduce the curriculum training strategy for better sound separation results.

3.1. Mix-and-Separate for Self-supervised Learning

Our approach adopted the Mix-and-Separate framework [53] for vision guided sound separation. Mixture and separated audio ground truths are obtained by mixing the the audio signals from different video clips. And then the task of our model is to separate the audio tracks from mixture conditioned on their corresponding visual inputs. Critically, although the neural network is trained in a supervised fashion, it does not require labeled data. Thus the training pipeline can be considered as self-supervised learning.

During training, we randomly select N video clips with paired video frames and audios $\{V_n, S_n\}$, and then mix their audios to form a synthetic mixture $S_{mix} = \sum_{n=1}^{N} S_n$. Given one of the N video clips, our model f will extract visual features and audio features for source separation $\hat{S_n} = f(S_{mix}, V_n)$. The direct output of our model is a binary mask that will be applied on the input mixture spectrogram, where the ground truth mask of the n-th video is determined by whether the target sound is the dominant component in the mixture,

$$M_n(u,v) = [\![S_n(u,v) \geq S_m(u,v)]\!], \quad \forall m = (1, ..., N), \tag{1}$$

where (u, v) represents the time-frequency coordinates in the spectrogram S. The model is trained with per-pixel binary cross-entropy loss.

3.2. Learning Motions with Deep Dense Trajectories

We use pixel-wise trajectories as our motion features for its demonstrated superior performance in action recognition tasks [50].

Given a video, the dense optical flow for each frame of the video at time t is denoted as $\omega_t = (u_t, v_t)$, and we

represent the coordinate position of each tracked pixel as $P_t = (x_t, y_t)$. Then the pixels in adjacent frames can associated as $P_{t+1} = (x_{t+1}, y_{t+1}) = (x_t, y_t) + \omega|_{(x_t, y_t)}$, and the full trajectory of a pixel is the concatenation of its coordinates over time $(P_t, P_{t+1}, P_{t+2}, ...)$. We use position invariant displacement vectors as the trajectory representation $\mathcal{T} = (\Delta P_t, \Delta P_{t+1}, \Delta P_{t+2}, ...)$, where $\Delta P_t = (x_{t+1} - x_t, y_{t+1} - y_t)$.

We note that the aforementioned operators are all differentiable, so they can fit into a learnable neural network model. Given the recent advances on CNN-based optical flow estimation, we incorporate a state-of-the-art optical flow model PWC-Net [45] into our system. So our whole system is an end-to-end learnable pixel tracking model, we refer to it as Deep Dense Trajectory network (DDT).

In previous works on trajectories [50], people usually sub-sample, smooth and normalize pixel trajectories to get extra robustness. We do not perform these operations since we assume that the dense, noisy signals can be handled by the learning system. To avoid tracking drift, we first perform shot detection on the input untrimmed videos, and then track within each video shot.

3.3. Model Architectures

Our full model is shown in Figure 2. It is comprised of four parts: a motion network, an appearance network, a fusion module and a sound separation network. We detail them below.

Motion Network. The motion network is designed to capture the motion features in the input video, on which the sound separation outputs are conditioned. We introduce Deep Dense Trajectories (DDT) network here, which is an end-to-end trainable pixel tracking network. The DDT network is composed of three steps:

(i) Dense optical flow estimation. This step enables the followup trajectory estimation, and it can be achieved by an existing CNN-based optical flow network. We choose the state-of-the-art PWC-Net [45] for its lightweight design and fast speed. PWC-Net estimates optical flow at each level in the feature pyramid, then uses the estimated flow to warp the feature at the next level and constructs a cost volume.

(ii) Dense trajectory estimation. This step takes dense optical flows as input to form dense trajectories. As discussed in Section 3.2, the position of each pixel at the next time stamp is estimated as the current position added by the current optical flow field. So the whole trajectory is estimated by iteratively tracking the points according to optical flow fields. In our neural network model, this process is implemented as an iterative differentiable grid sampling process. Specifically, we start with a regular 2D grid G_0 for the first frame; then for each frame at time t, we sample its optical flow field ω_t according to current grid G_t to estimate the grid at next time stamp, $G_{t+1} = G_t + \text{grid\_sample}(\omega_t, G_t)$. After tracking, our dense trajectories are given by

$$\begin{aligned}\mathcal{T} &= (\Delta P_0, ..., \Delta P_t, ...) \\ &= (\text{grid\_sample}(\omega_0, G_0), ..., \text{grid\_sample}(\omega_t, G_t), ...),\end{aligned}$$

where $t = (1, ..., T)$. The dimension of trajectories $\mathcal{T}$ is $T \times H \times W \times 2$, where the last dimension represents the displacements in x and y direction.

(iii) Dense trajectory feature extraction. A CNN model is further applied to extract the deep features of these trajectories, the choice of architecture can be arbitrary. Here we use an I3D model, which demonstrated good capability in capturing spatiotemporal features [7]. It is a compact design which inflates 2D CNN into 3D so that 3D filters can be bootstrapped from pretrained 2D filters.

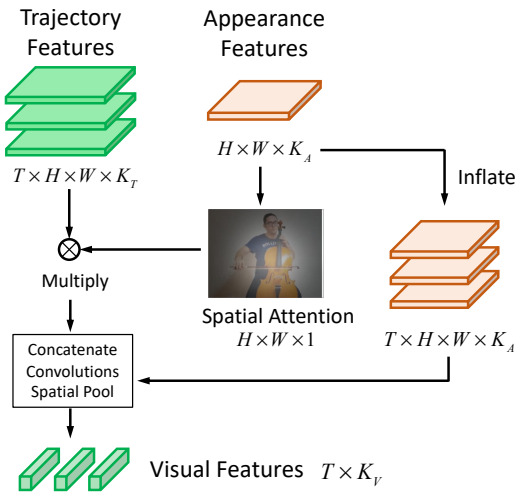

Figure 3. Fusion module of the model in Figure 2. A spatial attention map from appearance features is used to gate trajectory features.

Appearance Network The appearance network extracts semantic information from the input video. In terms of architecture, we use ResNet-18 [20] by removing the layers after spatial average pooling. We only take the first frame as input so that the trajectory feature maps are strictly registered with the appearance feature maps. The appearance and trajectory features are then fused to form the final visual features.

Attention based Fusion Module To fuse the appearance and trajectory features, we first predict a spatial attention map from the RGB features, and use it to modulate trajectory features. As shown in Figure 3, from the appearance feature we predict a single-channel map activated by `sigmoid`, with size $H \times W \times 1$. It is inflated in time and feature dimension, and multiplied with the trajectory feature from the Motion Network. Then appearance features are also inflated in time, and concatenated with the modulated trajectory features. After a couple of convolution layers, we perform max pooling to obtain the final visual feature. Such attention mechanism helps the model to focus on important trajectories.

Sound Separation Network The sound separation network takes in the spectrogram of sound, which is the 2D time-frequency representation; and predicts a spectrogram mask conditioned on the visual features. The architecture of sound separation network takes the form of a U-Net [38], so that the output mask size is the same as the input. In the middle part of the U-Net, where the feature maps are the smallest, condition signals from visual features are inserted. The way to incorporate visual features is by (1) aligning visual and sound features in time; (2) applying Feature-wise Linear Modulation (FiLM) [36] on sound features. FiLM refers to a feature-wise affine transformation, formally

$$\text{FiLM}(f_s) = \gamma(f_v) \cdot f_s + \beta(f_v), \tag{2}$$

where f_v and f_s are visual and sound features, $\gamma(\cdot)$ and $\beta(\cdot)$ are single linear layers which output scaling and bias on the sound features dependent on visual features.

The output spectrogram mask is obtained after a `sigmoid` activation on the network output. Then it is thresholded and multiplied with the input spectrogram to get a predicted spectrogram. Finally, an inverse Short Time Fourier Tranform (iSTFT) is applied to obtain the separated sound.

3.4. Curriculum Learning

Directly training sound separation on a single class of instruments suffers from overfitting due to the limited number training samples we have for each class. To remedy this drawback, we propose a 3-stage curriculum training by bootstrapping the model with easy tasks for good initializations, so that it converges better on the main tasks. The details are outlined as follows:

(i) Sound separation on mixture of different instruments. It shares similar settings as Section 4.2, where we randomly sample two video shots from the whole training set, mix their sounds as model input for separation;

(ii) Sound separation on mixture of the same kinds of instruments. Initializing from the model weights trained in Step 1, we then only train the model with mixtures from the same instruments, *e.g.* two videos of cellos;

(iii) Sound separation on mixture from the same video. To form the mixture, we sample two different video shots from the same long video. This is the hardest stage as semantic and context cues of those videos can be exactly the same, and the only useful cue is motions.

Note that we will only use this curriculum learning strategy in the same instrument sound separation task due to its challenging nature.

4. Experiments

4.1. Dataset

We perform vision guided sound separation tasks on the mixture of two video datasets: MUSIC [53] and URMP [28]. MUSIC is an unlabeled video dataset of instrument solos and duets by keyword query from Youtube; URMP is a small scale high quality multi-instrument video dataset recorded in studio.

To prevent the models from overfitting, we enlarge the MUSIC [53] dataset by collecting a larger number of musical instrument categories from web videos. Apart from the 11 instrument categories defined in MUSIC dataset: accordion, acoustic guitar, cello, clarinet, erhu, flute, saxophone, trumpet, tuba, violin and xylophone, we include another 10 common instrument categories: bagpipe, banjo, bassoon, congas, drum, electric bass, guzheng, piano, pipa and ukulele. We follow the procedure of [53] to collect the videos. Specifically, we construct a keyword with both instrument name with an additional "cover" and use it to retrieve videos from YouTube. We name the resulting dataset MUSIC-21, it contains 1365 untrimmed videos of musical solos and duets, where we split them into a training set of 1065 videos and a test set of 300 videos.

As our trajectory-based representation is sensitive to shot changes, we pre-process the raw videos into video shots, so that our training samples do not cross shot boundaries. Concretely, we densely sample the video frames and calculate the color histogram change of the adjacent frames over time, then we use a double thresholding approach [6] to find shot boundaries. After the processing, we obtain 5861 video shots in total.

4.2. Sound Separation for Different Instruments

To verify the effective of the learning motion representation for sound separation, we first evaluate the model performances in the task of separating sounds from different kinds of instruments, which has been explored in other works [47, 8, 53, 17].

Method	SDR	SIR	SAR
NMF [47]	2.78	6.70	9.21
Deep Separation [8]	4.75	7.00	10.82
MIML [17]	4.25	6.23	11.10
Sound of Pixels [53]	7.52	13.01	11.53
Ours, RGB single frame	7.04	12.10	11.05
Ours, RGB multi-frame	7.67	14.81	11.24
Ours, RGB+Flow	8.05	14.73	12.65
Ours, RGB+Trajectory	**8.31**	**14.82**	**13.11**

Table 1. Sound source separation performance ($N = 2$ mixture) of baselines and our model with different input modalities. Compared to Sound of Pixels, our models with temporal information perform better in sound separation.

4.2.1 Experiment Configurations

During training, we randomly take 3-second video clips from the dataset, and then sample RGB frames at 8 FPS to get 24 frames, and sample audios at 11 kHz.

The motion network takes 24 RGB frames as input. The flow network (PWC-Net) in it estimates 23 dense optical flow fields; the trajectory estimator further extracts trajectories with length of 23; and then the trajectory features are extracted by I3D. The output feature maps are of size $T \times H \times W \times K_m$.

The appearance network takes the first frame of the clip, and outputs appearance feature of size $1 \times H \times W \times K_a$. This feature is fused with the trajectory features through the fusion module, and after spatial pooling, we obtain the appearance feature of size $T \times K_v$.

The sound separation network takes a 3-second mixed audio clip as input, and transforms it into spectrogram by Short Time Fourier Transform (STFT) with frame size of 1022 and hop size of 172. The spectrogram is then fed into a U-Net with 6 convolution and 6 deconvolution layers. In the middle of the sound separation network, visual features are aligned with the sound features, and the FiLM module modulates the sound features conditioned on visual features. The U-Net outputs a binary mask after `sigmoid` activation and thresholding. To obtain the final separated audio waveforms, iSTFT with the same parameters as the STFT is applied.

We use SGD optimizer with 0.9 momentum to train the our model. The Sound Separation Network and the fusion module use a learning rate of 1e-3; the Motion Network and Appearance Network use a learning rate of 1e-4, as they take pretrained ResNet and I3D on ImageNet and pretrained PWC-Net on MPI Sintel.

N	Method	SDR	SIR	SAR
3	NMF [47]	2.01	2.08	9.36
	Sound of Pixels [53]	3.65	8.77	8.48
	Ours, RGB+Trajectory	4.87	9.48	9.24
4	NMF [47]	0.93	-1.01	9.01
	Sound of Pixels [53]	1.21	6.58	4.19
	Ours, RGB+Trajectory	3.05	8.50	7.45

Table 2. Sound separation performances with $N = 3, 4$ mixtures. We compare our model against Sound of Pixels to show the advantage of motion features. Our model consistently improves separation metrics and outperforms in highly mixed cases.

4.2.2 Results

We evaluate the sound separation performance of our model with different variants. **RGB+Trajectory** is our full model as described in 3.3; **RGB+Flow** is the full model without the tracking module, so the motion feature is extracted from optical flow; **RGB multi-frame** further removes the flow network, so motion feature directly comes from RGB frame sequence; **RGB single frame** is a model without motion network, visual feature comes from appearance network only.

At the same time, we re-implement 4 models to compare against. **NMF** [47] is a classical approach based on matrix factorization, it uses ground truth labels for training; **Deep Separation** [8] is a CNN-variant supervised learning model, it also takes ground truth labels for training; **MIML** [17] is a model that combines NMF decomposition and multi-instance multi-label learning; **Sound of Pixels** [53] is a recently proposed self-supervised model which takes both sounds and video frames for source separation.

For fair comparisons, all the models are trained and tested with 3-second audios mixed from $N = 2$ input audios, and models dependent on vision take in 24 video frames. Model performances are evaluated on a validation set with 256 pairs of sound mixtures. We use the following metrics from the open-source `mir_eval` [37] library to quantify performance: Signal-to-Distortion Ratio (SDR), Signal-to-Interference Ratio (SIR), and Signal-to-Artifact Ratio (SAR). Their units are in dB.

Quantitative results are reported in Table 1. We observe that previous methods achieves reasonable performance in sound separation even though only appearance information is used [53]. It shows that appearance based models are already strong baselines for this task. In comparison, our RGB multi-frame, RGB+Flow and RGB+Trajectory models outperform all baseline methods, showing the effectiveness of encoding motion cues in the task of audio-visual source separation. And among them, RGB+Trajectory is best, and outperforms state-of-the-art Sound of Pixels model by ≈ 0.8dB. It demonstrates that among these mo-

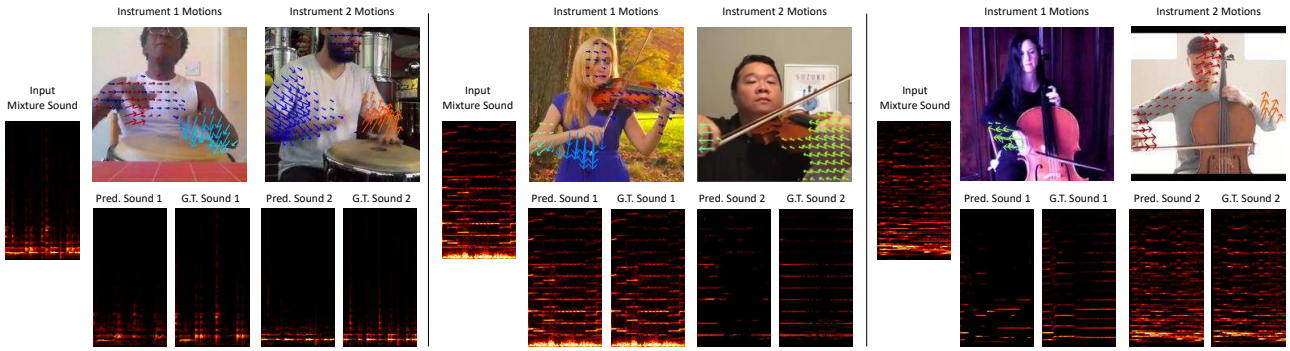

Figure 4. Results of sound separation on the same kinds of instruments. Our model can capture the motion information in videos to separate the sound. This visualization is only performed for quantitative model evaluation.

tion representations, trajectories has the strongest correlation with sound.

We further experiment on the task of separating larger number of sound mixtures, where $N = 3, 4$. Results are reported in Table 2. We observe that our best model outperforms Sound of Pixels by a larger margin in these highly mixed cases, $\approx$ 1.2dB at $N = 3$, and $\approx$ 1.8dB at $N = 4$.

4.3. Sound Separation for the Same Instruments

In this section, we evaluate the model performance in separating sounds from instruments of the same kind, which has rarely been explored before.

4.3.1 Experiment Configurations

To evaluate the performances of the our models, we select 5 kinds of musical instruments whose sounds are closely related to motions: violin, cello, congas, erhu and xylophone. All the training settings are similar to Section 4.2 except that we use curriculum learning strategy which is mentioned in Section 3.4.

4.3.2 Results

First we evaluate the effectiveness of our proposed curriculum learning strategy. With a fixed validation set, we compare **Single Stage** strategy, which is directly trained on mixtures of the same instruments, with our 3-stage training strategy. In **Curriculum Stage 1**, model is trained to separate sound mixtures of instruments of different categories; in **Curriculum Stage 2**, the task is to separate sound mixtures from the same kinds of instruments; in **Curriculum Stage 3**, the goal is to separate sound mixtures of different clips from the same long video. Results of our final model on the validation set are shown in Figure 4.

Results in Table 3 show that curriculum learning greatly improves the performance: it outperforms the **Single Stage** model in the **Curriculum Stage 1**, and further improves with the second and third stages. The total improvement in SDR is $\approx$ 4dB.

Schedule	SDR	SIR	SAR
Single Stage	1.91	5.73	8.83
Curriculum Stage 1	3.14	7.52	13.06
Curriculum Stage 2	5.72	13.89	11.92
Curriculum Stage 3	5.93	14.41	12.08

Table 3. Performance improvement with the proposed curriculum learning schedule.

Then we compare the performance of our model with Sound of Pixels model on the same instrument separation task. To make fair comparisons, Sound of Pixels model is trained with the same curriculum. Results on SDR metric are reported in Table 4. We can see that Sound of Pixels model gives much inferior results comparing to our model, the gap is $>$ 3dB.

Qualitative comparisons are presented in Figure 5, where we show pixel-level sound embeddings. To recover sounds spatially, we remove the spatial pooling operation in the fusion module in Figure 3 at test time, and then feed the visual feature at each spatial location to the Sound Separation Network. Therefore, we are able to get $H \times W$ number of separated sound components. We project those sound features (vectorized spectrogram values) into a 3 dimensional space using PCA, and visualize them in color. Different colors in the heatmaps refer to different sounds. We show that our model can tell the difference from duets of the same instruments, while Sound of Pixels model cannot.

4.3.3 Human Evaluation

Since the popular metrics (*e.g.* SDR, SIR and SAR) for sound separation might not reflect the actual perceptual quality of the sound separation results, we further compare the performances of these two methods on Amazon Me-

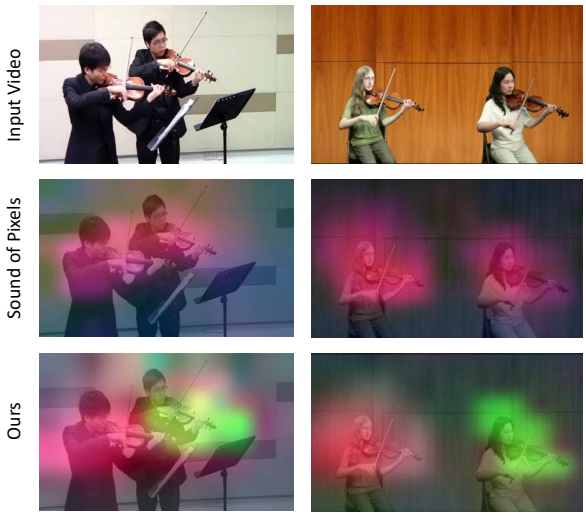

Figure 5. Pixel-level sound embedding results. To visualize the pixel-level sound separation results, we project sound features into a low dimensional space, and visualize them in RGB space. Different colors mean different sounds. Our model can tell the difference from duets of the same instruments, while Sound of Pixels model cannot.

Instrument	Sound of Pixels	Ours
violin	1.95	6.33
cello	2.62	5.48
congas	2.90	5.21
erhu	1.67	6.13
xylophone	3.56	6.50

Table 4. Sound source separation performance on duets of the same instruments. We show the SDR metric on each instrument. Our approach is consistently better than previous works.

chanical Turk (AMT) with subjective human evaluations.

Concretely, we collected 100 testing videos from each instrument, and got separation results of the Sound of Pixels baseline [53] and our best model. We also provide the ground truth results for references. To avoid shortcut, we randomly shuffle the orders of two models and ask the following question: `Which sound separation result is closer to the ground truth?` The workers are asked to choose one of the best sound separation results. We assign 3 independent AMT workers for each job.

Results are shown in Table 5, our proposed motion-based model consistently outperforms the Sound of Pixels systems for all the five instruments. We see the reasons lie in two folds: (1) motion information is crucial for the sound separation of the same instruments; (2) the Sound of Pixels model cannot capture motion cues effectively, while our model is better by design.

Instrument	Sound of Pixels	Ours
violin	38.75%	61.25%
cello	39.21%	60.79%
congas	35.42%	64.58%
erhu	44.59%	55.41%
xylophone	35.56%	64.44%

Table 5. Human evaluation result for the sound source separation on mixture of the same instruments.

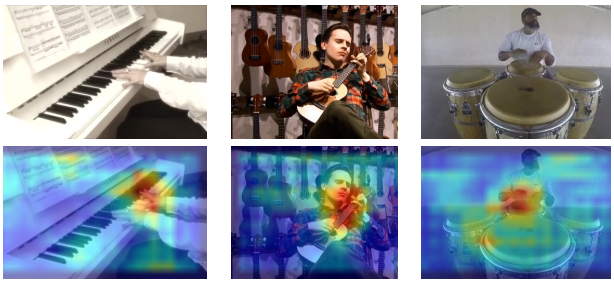

Figure 6. Sounding object localization. Overlaid heatmaps show the predicted sound volume at each pixel location. The model tends to predict the instrument parts where people are interacting with. Silent instruments such as the guitars on the wall are not detected as sounding objects.

4.4. Sounding object localization

As a further analysis, we explore the sounding object localization capability of our best model. We recover the sounds spatially similar to what we did in Section 4.3.2. And then we calculate the sound volume at each spatial location, and display them in heatmaps, as shown in Figure 6. We observe that (1) the model gives roughly correct predictions on the sounding object locations, but does not cover the whole instruments. Interestingly, it focuses on the parts where humans are interacting with; (2) Our model correctly predicts silent instruments, *e.g.* guitars on the wall, it demonstrates that sounding object localization is not only based on visual appearance, but also on audio input.

5. Conclusion

In this paper, we propose that motions are important cues in audio-visual tasks, and design a system that captures visual motions with deep dense trajectories (DDT) to separate sounds. Extensive evaluations show that, compared to previous appearance based models, we are able to perform audio-visual source separation of different instruments more robustly; we can also separate sounds of the same kind of instruments through curriculum learning, which seems impossible for the purely appearance based approaches.

References

[1] Relja Arandjelovic and Andrew Zisserman. Look, listen and learn. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 609–617. IEEE, 2017. 1, 2

[2] Relja Arandjelović and Andrew Zisserman. Objects that sound. *arXiv preprint arXiv:1712.06651*, 2017. 1, 2

[3] Yusuf Aytar, Carl Vondrick, and Antonio Torralba. Soundnet: Learning sound representations from unlabeled video. In *Advances in Neural Information Processing Systems*, pages 892–900, 2016. 1, 2

[4] Zohar Barzelay and Yoav Y Schechner. Harmony in motion. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8. IEEE, 2007. 2

[5] Yunlong Bian, Chuang Gan, Xiao Liu, Fu Li, Xiang Long, Yandong Li, Heng Qi, Jie Zhou, Shilei Wen, and Yuanqing Lin. Revisiting the effectiveness of off-the-shelf temporal modeling approaches for large-scale video classification. *arXiv preprint arXiv:1708.03805*, 2017. 2

[6] John Canny. A computational approach to edge detection. *IEEE Transactions on pattern analysis and machine intelligence*, (6):679–698, 1986. 5

[7] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*, pages 4724–4733. IEEE, 2017. 4

[8] Pritish Chandna, Marius Miron, Jordi Janer, and Emilia Gómez. Monoaural audio source separation using deep convolutional neural networks. In *ICLVASS*, pages 258–266, 2017. 2, 5, 6

[9] Joon Son Chung, Andrew W Senior, Oriol Vinyals, and Andrew Zisserman. Lip reading sentences in the wild. In *CVPR*, pages 3444–3453, 2017. 2

[10] Andrzej Cichocki, Rafal Zdunek, Anh Huy Phan, and Shunichi Amari. *Nonnegative matrix and tensor factorizations: applications to exploratory multi-way data analysis and blind source separation*. John Wiley & Sons, 2009. 2

[11] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *ICCV*, pages 2625–2634, 2015. 2

[12] 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. *arXiv preprint arXiv:1804.03619*, 2018. 1, 2

[13] Lijie Fan, Wenbing Huang, Stefano Ermon Chuang Gan, Boqing Gong, and Junzhou Huang. End-to-end learning of motion representation for video understanding. In *CVPR*, 2018. 3

[14] John W Fisher III, Trevor Darrell, William T Freeman, and Paul A Viola. Learning joint statistical models for audio-visual fusion and segregation. In *NIPS*, 2001. 2

[15] Aviv Gabbay, Ariel Ephrat, Tavi Halperin, and Shmuel Peleg. Seeing through noise: Speaker separation and enhancement using visually-derived speech. *arXiv preprint arXiv:1708.06767*, 2017. 2

[16] Chuang Gan, Boqing Gong, Kun Liu, Hao Su, and Leonidas J Guibas. Geometry-guided CNN for self-supervised video representation learning. 2018. 3

[17] Ruohan Gao, Rogerio Feris, and Kristen Grauman. Learning to separate object sounds by watching unlabeled video. In *ECCV*, 2018. 1, 2, 5, 6

[18] Ruohan Gao and Kristen Grauman. 2.5 d visual sound. *arXiv preprint arXiv:1812.04204*, 2018. 2

[19] Simon Haykin and Zhe Chen. The cocktail party problem. *Neural computation*, 17(9):1875–1902, 2005. 2

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 4

[21] John R Hershey, Zhuo Chen, Jonathan Le Roux, and Shinji Watanabe. Deep clustering: Discriminative embeddings for segmentation and separation. In *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, pages 31–35. IEEE, 2016. 2

[22] John R. Hershey and Javier R. Movellan. Audio vision: Using audio-visual synchrony to locate sounds. In S. A. Solla, T. K. Leen, and K. Müller, editors, *Advances in Neural Information Processing Systems 12*, pages 813–819. MIT Press, 2000. 1, 2

[23] Hamid Izadinia, Imran Saleemi, and Mubarak Shah. Multimodal analysis for identification and segmentation of moving-sounding objects. *IEEE Transactions on Multimedia*, 15(2):378–390, 2013. 2

[24] Einat Kidron, Yoav Y. Schechner, and Michael Elad. Pixels that sound. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1 - Volume 01*, CVPR '05, pages 88–95, Washington, DC, USA, 2005. IEEE Computer Society. 1, 2

[25] Alexander Klaser, Marcin Marszałek, and Cordelia Schmid. A spatio-temporal descriptor based on 3d-gradients. In *BMVC*, pages 275–1, 2008. 2

[26] Bruno Korbar, Du Tran, and Lorenzo Torresani. Co-training of audio and video representations from self-supervised temporal synchronization. *arXiv preprint arXiv:1807.00230*, 2018. 1, 2

[27] Ivan Laptev. On space-time interest points. *International journal of computer vision*, 64(2-3):107–123, 2005. 2

[28] Bochen Li, Xinzhao Liu, Karthik Dinesh, Zhiyao Duan, and Gaurav Sharma. Creating a multitrack classical music performance dataset for multimodal music analysis: Challenges, insights, and applications. *IEEE Transactions on Multimedia*, 21(2):522–535, 2019. 2, 5

[29] Xiang Long, Chuang Gan, Gerard de Melo, Jiajun Wu, Xiao Liu, and Shilei Wen. Attention clusters: Purely attention based local feature integration for video classification. In *CVPR*, 2018. 2

[30] Josh H McDermott. The cocktail party problem. *Current Biology*, 19(22):R1024–R1027, 2009. 2

[31] Pedro Morgado, Nuno Nvasconcelos, Timothy Langlois, and Oliver Wang. Self-supervised generation of spatial audio for 360 video. In *NIPS*, 2018. 2

[32] Arsha Nagrani, Samuel Albanie, and Andrew Zisserman. Seeing voices and hearing faces: Cross-modal biometric matching. *arXiv preprint arXiv:1804.00326*, 2018. 2

[33] Jiquan Ngiam, Aditya Khosla, Mingyu Kim, Juhan Nam, Honglak Lee, and Andrew Y. Ng. Multimodal deep learning. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, ICML'11, pages 689–696, 2011. 1

[34] Andrew Owens and Alexei A Efros. Audio-visual scene analysis with self-supervised multisensory features. *arXiv preprint arXiv:1804.03641*, 2018. 2

[35] Andrew Owens, Jiajun Wu, Josh H McDermott, William T Freeman, and Antonio Torralba. Ambient sound provides supervision for visual learning. In *European Conference on Computer Vision*, pages 801–816. Springer, 2016. 2

[36] Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. *arXiv preprint arXiv:1709.07871*, 2017. 5

[37] 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 *In Proceedings of the 15th International Society for Music Information Retrieval Conference, ISMIR*. Citeseer, 2014. 6

[38] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 5

[39] Andrew Rouditchenko, Hang Zhao, Chuang Gan, Josh McDermott, and Antonio Torralba. Self-supervised audio-visual co-segmentation. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2357–2361. IEEE, 2019. 2

[40] Arda Senocak, Tae-Hyun Oh, Junsik Kim, Ming-Hsuan Yang, and In So Kweon. Learning to localize sound source in visual scenes. *arXiv preprint arXiv:1803.03849*, 2018. 2

[41] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *NIPS*, pages 568–576, 2014. 2

[42] Andrew JR Simpson, Gerard Roma, and Mark D Plumbley. Deep karaoke: Extracting vocals from musical mixtures using a convolutional deep neural network. In *International Conference on Latent Variable Analysis and Signal Separation*, pages 429–436. Springer, 2015. 2

[43] Paris Smaragdis and Judith C Brown. Non-negative matrix factorization for polyphonic music transcription. In *Applications of Signal Processing to Audio and Acoustics, 2003 IEEE Workshop on.*, pages 177–180. IEEE, 2003. 2

[44] Barry E Stein and M Alex Meredith. *The merging of the senses*. The MIT Press, 1993. 1

[45] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8934–8943, 2018. 4

[46] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*, pages 4489–4497, 2015. 2

[47] Tuomas Virtanen. Monaural sound source separation by nonnegative matrix factorization with temporal continuity and sparseness criteria. *IEEE transactions on audio, speech, and language processing*, 15(3):1066–1074, 2007. 2, 5, 6

[48] DeLiang Wang and Jitong Chen. Supervised speech separation based on deep learning: an overview. *arXiv preprint arXiv:1708.07524*, 2017. 2

[49] Heng Wang, Alexander Kläser, Cordelia Schmid, and Cheng-Lin Liu. Action recognition by dense trajectories. In *CVPR*, pages 3169–3176, 2011. 2

[50] Heng Wang and Cordelia Schmid. Action recognition with improved trajectories. In *Proceedings of the IEEE international conference on computer vision*, pages 3551–3558, 2013. 2, 3, 4

[51] Xiaolong Wang and Abhinav Gupta. Videos as space-time region graphs. *ECCV*, 2018. 2

[52] Dong Yu, Morten Kolbæk, Zheng-Hua Tan, and Jesper Jensen. Permutation invariant training of deep models for speaker-independent multi-talker speech separation. In *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*, pages 241–245. IEEE, 2017. 2

[53] Hang Zhao, Chuang Gan, Andrew Rouditchenko, Carl Vondrick, Josh McDermott, and Antonio Torralba. The sound of pixels. In *The European Conference on Computer Vision (ECCV)*, September 2018. 1, 2, 3, 5, 6, 8

[54] Yue Zhao, Yuanjun Xiong, and Dahua Lin. Trajectory convolution for action recognition. In *NIPS*, 2018. 3

[55] Hang Zhou, Yu Liu, Ziwei Liu, Ping Luo, and Xiaogang Wang. Talking face generation by adversarially disentangled audio-visual representation. In *AAAI*, 2019. 2

[56] Yipin Zhou, Zhaowen Wang, Chen Fang, Trung Bui, and Tamara L Berg. Visual to sound: Generating natural sound for videos in the wild. *arXiv preprint arXiv:1712.01393*, 2017. 2