Improving mix-and-separate training in audio-visual sound source separation with an object prior

Quan Nguyen, Julius Richter, Mikko Lauri, Timo Gerkmann, Simone Frintrop
Department of Informatics, Universität Hamburg
Email: {nguyen, jrichter, lauri, gerkmann, frintrop}@informatik.uni-hamburg.de

***Abstract*—The performance of an audio-visual sound source separation system is determined by its ability to separate audio sources given the images of the sources and the audio mixture. The goal of this study is to investigate the ability to learn the mapping between the sounds and the images of instruments in the self-supervisied mix-and-seperate training paradigm used by state-of-the-art audio-visual sound source separation methods. Theoretical and empirical analyses illustrate that the self-supervised mix-and-separate training does not automatically learn the 1-to-1 correspondence between visual and audio signals, leading to low audio-visual object classification accuracy. Based on this analysis, a weakly-supervised method called *Object-Prior* is proposed and evaluated on two audio-visual datasets. The experimental results show that the Object-Prior method outperforms state-of-the-art baselines in the audio-visual sound source separation task. It is also more robust against asynchronized data, where the frame and the audio do not come from the same video, and recognizes musical instruments based on their sound with higher accuracy. This indicates that learning the 1-to-1 correspondence between visual and audio features of an instrument improves the effectiveness of audio-visual sound source separation.**

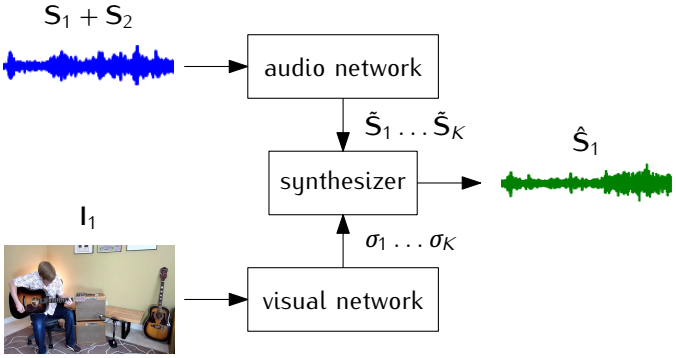

Fig. 1. Our audio-visual sound source separation framework at test time. The audio network takes an audio mixture, e.g., the sum of two spectrograms $\mathbf{S}_1$ and $\mathbf{S}_2$, and outputs a list of K spectrograms $\tilde{\mathbf{S}}_1 \ldots \tilde{\mathbf{S}}_K$. The visual network takes one video frame $\mathbf{I}_1$ as an object prior according to the sound source $\mathbf{S}_1$ and outputs a discrete probability distribution $p(\text{type} = i) = \sigma_i$. The synthesizer generates the separated sound source estimate $\hat{\mathbf{S}}_1$ according to its inputs.

I. Introduction

Humans are capable of recognizing an audio source from an audio mixture with high accuracy. For example, we can attend to the speech of a chosen speaker in a crowd or to the sound from a specific instrument in a duet. Sound source separation is essential for humans in everyday activities such as listening to a targeted speaker at a cocktail party, analyzing a natural scene comprised of many sounds, or in specialized activities such as transcribing individual instrumental tracks from a music piece.

Sound source separation remains challenging for machines. Current research attempts to overcome the difficulties by learning compact representations of audio and visual signals, facilitating the sound source separation and selection process. Because large datasets with object and audio labels are not available, state-of-the-art approaches rely on a combination of transfer learning with weakly-supervised learning [2], [3] or self-supervised learning [1], [4]–[6].

A typical architecture for audio-visual sound source separation consists of an audio network (ANet), a visual network (VNet) and a fusion module, as illustrated in Fig. 1. The ANet acts as a *separator* of the sound sources of an audio mixture. The VNet acts as a *selector* which selects a sound source based on its visual appearance. The fusion module acts as a *synthesizer* responsible for generating the final output audio signal. The synthesizer is either a linear combination module [1], a multi-instance multi-label learning module [2] or a deep neural network [4], [5].

The same architecture can be used for the *Audio-visual object classification* task [7], [8] in which the category of an object (e.g. saxophone or guitar) is recognized based on its sound and visual appearances. In Fig. 2, if each visual and audio channel in the softmax layer from the VNet and ANet consistently activate strongly to a particular type of instrument, then the system is capable of recognizing that instrument based on its visual and audio features. Given that the same network architecture can be used for both tasks, a natural question that arises is whether a late-fusion architecture can be trained to learn both sound source separation and audio-visual object classification simultaneously [1], especially when the latter requires human supervision on the object labels.

Recent works on sound source separation [1], [9] train deep learning models using the "mix-and-separate" paradigm. Two video clips of two different instruments are randomly selected, their audio tracks are mixed and the learning objective is to separate the sounds simultaneously based on the video frames. [1] suggested that given enough training data, audio-visual object classification will emerge during this self-supervised training procedure, even without any labels for the type of instruments (e.g. flute or trumpet). However, this conjecture is not supported by the experimental results of the PixelPlayer model [1] that achieves a low object classification accuracy

of 42%, indicating most instruments are misclassified by the VNet. It remains unclear whether the learning of audio-visual object classification can emerge from the self-supervised mix-and-separate training procedure for audio-visual sound source separation.

In this paper, we present a new model called *Object-Prior* for the audio-visual sound source separation. We use a simplified learning framework similar to the audio-visual sound separation task and use it to analyze how the activation channels in the VNet and ANet are trained in the mix-and-separate paradigm. We then provide theoretical analysis and empirical evidence that the learning objective in the audio-visual sound separation might inhibit the learning of audio-visual object classification. Our proposed Object-Prior method first trains the VNet for instrument classification based on visual images, and then uses this pre-trained VNet to learn the audio-visual sound source separation task. We find that the Object-Prior method achieves high performance on both audio-visual sound source separation and audio-visual object classification. The Object-Prior method is evaluated on the MUSIC [1] and AudioSet-SingleSource [2] datasets.

The paper has three main contributions:

- Theoretical analysis and empirical evidence to support a hypothesis that the learning of audio-visual object classification does not necessarily emerge from the self-supervised mix-and-separate training procedure for audio-visual sound source separation.
- A weakly-supervised audio-visual sound source separation method called Object-Prior is proposed. Object-Prior learns audio-visual object classification and audio-visual sound source separation simultaneously and achieves state-of-the-art performance on the single sound source separation task.
- Experimental analyses on the effectiveness and robustness of the Object-Prior method on both sound source separation and audio-visual object recognition are presented.

The paper is structured as follows: Section II describes the related work on sound source separation and cross-modal learning. Section III develops a simplified mix-and-separate sound separation framework based on the PixelPlayer [1] and shows that the learning objective of this framework does not simultaneously train a model for audio-visual object classification. Section IV introduces the Object-Prior method and evaluates its performance on sound source separation and audio-visual object classification.

II. Related Work

Audio-only sound source separation using mix-and-separate training paradigm: The mix-and-separate training paradigm has long been used extensively in the sound source separation community. Early development of this technique uses the addition of two signals as the mixture and spectral clustering for the separation [10]. One of the recent formal discussions about mix-and-separate training paradigm for deep neural networks is the work on monaural source separation of [11], which argues that the mapping relationship between a mixture of two signals and the separated sources is nonlinear and should be modeled by nonlinear models such as deep neural networks. Several further developments such as [12]–[14] have been made in this direction. [13] combines deep neural networks and spectral clustering for the separation, while [14] proposes a new training objective that encourages the magnitude spectrograms of the original sources and the predicted sources to be similar. In general, our work differs from all these blind-source separation works because we focus on the setting in which the visual cues are essential in separating the sound sources.

Audio-visual sound source separation: Mixing two signals by summation works best when the two sources are independent, however this assumption is unrealistic for "true" mixed sounds [11]. To mitigate the effect of this artificial mixing process, [5] chooses component videos of more than one object (e.g. musical instruments) for the mixtures and defines a loss function on each object present in all frames of the component videos, thereby leverages more natural videos for training. Other works use only a subset of the frames of each component videos and aim at localizing the objects that produce the sounds [1], [2], [15], [16]. [6] attempts at separating sound sources recursively and avoids the constraint on the number of sources in the mixture. The visual cues in most of these works are static visual features of an object. Recently, an object's motion as a visual cue has also been investigated [4], [17]. We refer to [18] for an extensive review of recent deep audio-visual learning methods and applications. The sound source selection module in all of these approaches relies on the synchronization between the visual content and the audio of a video to select the correct sound source based on the visual cues. Our Object-Prior model extends these works to the asynchronized setting.

Robust and interpretable cross-modal representation learning: [9] regulates the training paradigm to make the activations of visual channels to be sparse, thereby forcing the VNet and ANet to have the 1-to-1 correspondence between the visual channels and audio channels. Their approach allowed the two networks to be trained simultaneously at the cost of additional complexity in a multi-stage training. However, the best accuracy of the VNet in [9] was still lower than 50%. In contrast, the training procedure in our work fine-tunes the VNet for object recognition and then learn the ANet in the original mix-and-separate manner without any regularization. [19] and [20] focused on the correspondence between a human speech and the fine-grained categorization of an object (e.g. yellow bird versus white bird), whereas our paper focuses on the characteristics of the sound of an object and the entry-level categorization of that object (e.g. flute versus trumpet).

III. Learning Dynamics of Mix-and-Separate Self-Supervised Training

In this section, we present a hypothesis that the self-supervised mix-and-separate training procedure does not automatically lead to the learning of 1-to-1 mapping between the visual and audio channels, thereby explaining the low

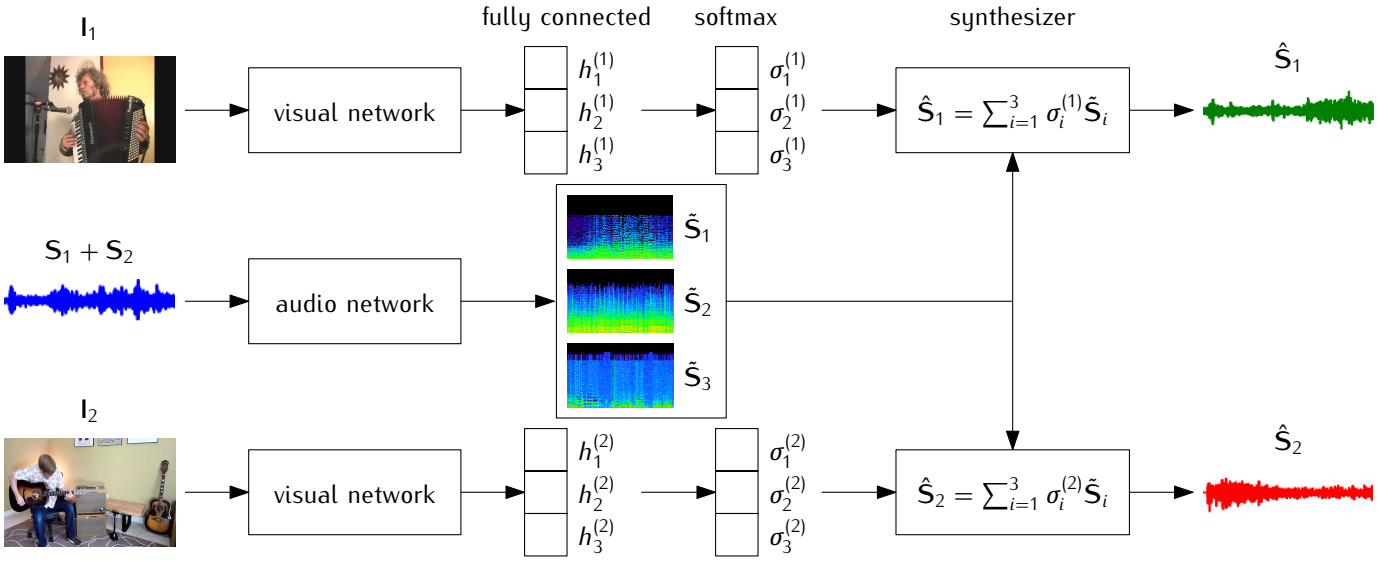

Fig. 2. Mix-and-separate training with an audio mixture $\mathbf{S}_1 + \mathbf{S}_2$ which is generated as the sum of the audio tracks of two different videos with their corresponding video frames $\mathbf{I}_1$ and $\mathbf{I}_2$. The audio network corresponds outputs a list of K spectrograms $\tilde{\mathbf{S}}_1 \dots \tilde{\mathbf{S}}_K$ resembling the spectrograms from different sources; for the purpose of illustration we set $K = 3$. The output of the visual network's softmax layer provides a discrete probability distribution over all K object types. The synthesizer assembles the separated audio source estimates $\hat{\mathbf{S}}_1$ and $\hat{\mathbf{S}}_2$ as a linear combination of the spectrograms $\tilde{\mathbf{S}}_1 \dots \tilde{\mathbf{S}}_K$ weighted by the output of the softmax layers $\sigma_i^{(1)}$ and $\sigma_i^{(2)}$.

object classification accuracy of PixelPlayer [1]. The late-fusion architecture is employed in which the ANet takes as input the magnitude spectrogram of an audio mixture and gives as output a list of K spectrograms corresponding to audio from K different sources; the VNet takes in one or more images indicating the targeted source and outputs a discrete probability distribution over all K possible types of audio source; the synthesizer is a linear combination of all the output spectrograms from the ANet weighted by the output probabilities from the VNet. Figure 2 illustrates this architecture with $K = 3$. In section III-A, a simplified learning framework based on PixelPlayer [1] is used to analyze how the activation channels of the ANet and VNet are updated after one forward pass and one backward pass of the mix-and-separate training procedure for the network architecture in Fig. 1 when the network is trained by gradient descent algorithms. Section III-B presents the analysis that once one of the two ANet or VNet confuses the two different audio or visual signals into one channel, that mistake is propagated to the other network. In section III-C, the PixelPlayer model is trained on a small dataset of three instruments and shown that it does not visually recognize one instrument, thereby empirically verifying this hypothesis.

A. A Simplified Learning Framework

Analyzing the learning dynamics of the deep neural networks in [1] is challenging because of the high dimensionality of the input signal, the model and the output signal. Instead, we employ a simplified framework where the input and output audio signals are represented by a scalar value. In the general framework, each scalar value represents the magnitude of each pixel on the spectrogram image of an audio signal.

We denote by $\mathbf{x} = (\mathbf{I}, \mathbf{S})$ an audio-visual sample with audio modality $\mathbf{S}$ and visual modality $\mathbf{I}$. For simplicity, each sample has exactly one audio source. Given a training set $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2 \dots, \mathbf{x}_N\}$ of N samples, the goal is to train a sound source separation network in Fig. 2 using gradient descent learning algorithm in the mix-and-separate procedure.

The network consists of a visual network VNet, an audio net ANet and a synthesizer. The VNet takes as input the visual signal $\mathbf{I}$ and outputs a discrete probability distribution over $K > 1$ possible types of instrument in $\mathbf{I}$. This probability distribution is computed at the final softmax layer of the VNet. We denote by $\mathbf{h} = \begin{bmatrix} h_1 & h_2 & \dots & h_K \end{bmatrix}^T$ the vector of activations of the layer before the softmax layer.

We denote $\sigma : \mathbb{R}^K \to \mathbb{R}^K$ the softmax function with ith component

$$\sigma_i(\mathbf{h}) = \frac{e^{h_i}}{\sum_{j=1}^{K} e^{h_j}}. \tag{1}$$

To simplify the notation, we omit the $\mathbf{h}$ vector and write $\sigma_i = \sigma_i(\mathbf{h})$ where it is clear on which activation vector $\mathbf{h}$ the softmax layer is being applied.

The ANet takes as input an audio mixture $\mathbf{S}_{mix}$ and outputs K audio spectrograms $\tilde{\mathbf{S}}_1, \tilde{\mathbf{S}}_2, \dots, \tilde{\mathbf{S}}_K$. Our analysis applies for networks whose activation function is the Rectified Linear Unit (ReLU). As a result, $\tilde{\mathbf{S}}_i$ is non-negative for every time-frequency bin. The synthesizer is set to be a dot product layer which computes the predicted audio signal $\hat{\mathbf{S}}$ as a linear combination of the K audio signals outputted from the ANet, weighted by the probability distribution from the VNet:

$$\hat{\mathbf{S}} = \sum_{i=1}^{K} \sigma_i \tilde{\mathbf{S}}_i. \tag{2}$$

The loss function per time-frequency bin $L : \mathbb{R}^2 \rightarrow \mathbb{R}^+$ is computed as the squared error between the true audio signal S and the predicted audio signal:

$$\ell(\mathbf{S}, \hat{\mathbf{S}}) = \frac{1}{2}(\mathbf{S} - \sum_{i=1}^{K} \sigma_i \tilde{\mathbf{S}}_i)^2. \tag{3}$$

The forward pass: In each learning iteration, a pair of audio-visual samples $\mathbf{x}_1 = (\mathbf{I}_1, \mathbf{S}_1)$ and $\mathbf{x}_2 = (\mathbf{I}_2, \mathbf{S}_2)$ is randomly selected from the training set. The two videos are chosen so that they contain two different types of musical instrument and each video contains one type of instrument. The forward pass is divided into two branches which output two predicted audio signals $\hat{\mathbf{S}}_1$ and $\hat{\mathbf{S}}_2$, respectively. Using Equation 2, the predicted audio signals is written as

$$\begin{aligned} \hat{\mathbf{S}}_1 &= \sum_{i=1}^{K} \sigma_i^{(1)} \tilde{\mathbf{S}}_i \\ \hat{\mathbf{S}}_2 &= \sum_{i=1}^{K} \sigma_i^{(2)} \tilde{\mathbf{S}}_i \end{aligned} \tag{4}$$

where $\sigma^{(1)}, \sigma^{(2)}$ denote the output of the softmax layer on the first and second branch, respectively.

The loss function on the first and second branches of the training process is

$$\begin{aligned} \ell_1 &= \frac{1}{2}(\mathbf{S}_1 - \sum_{i=1}^{K} \sigma_i^{(1)} \tilde{\mathbf{S}}_i)^2 \\ \ell_2 &= \frac{1}{2}(\mathbf{S}_2 - \sum_{i=1}^{K} \sigma_i^{(2)} \tilde{\mathbf{S}}_i)^2. \end{aligned} \tag{5}$$

The backward pass: The partial derivative of ℓ_1 and ℓ_2 with respect to the audio channel $\tilde{\mathbf{S}}_j$ are

$$\begin{aligned} \frac{\partial \ell_1}{\partial \tilde{\mathbf{S}}_j} &= -(\mathbf{S}_1 - \sum_{i=1}^{K} \sigma_i^{(1)} \tilde{\mathbf{S}}_i)\sigma_j^{(1)} \\ \frac{\partial \ell_2}{\partial \tilde{\mathbf{S}}_j} &= -(\mathbf{S}_2 - \sum_{i=1}^{K} \sigma_i^{(2)} \tilde{\mathbf{S}}_i)\sigma_j^{(2)}. \end{aligned} \tag{6}$$

The partial derivative of ℓ_1 and ℓ_2 with respect to the visual channel h_j are

$$\begin{aligned} \frac{\partial \ell_1}{\partial h_j} &= -(\mathbf{S}_1 - \sum_{i=1}^{K} \sigma_i^{(1)} \tilde{\mathbf{S}}_i)(\sum_{k=1}^{K} \tilde{\mathbf{S}}_k \frac{\partial \sigma_k^{(1)}}{\partial h_j}) \\ &= -(\mathbf{S}_1 - \sum_{i=1}^{K} \sigma_i^{(1)} \tilde{\mathbf{S}}_i)\sigma_j^{(1)}(\sum_{k=1}^{K} (\tilde{\mathbf{S}}_j - \tilde{\mathbf{S}}_k)\sigma_k^{(1)}) \\ \frac{\partial \ell_2}{\partial h_j} &= -(\mathbf{S}_2 - \sum_{i=1}^{K} \sigma_i^{(2)} \tilde{\mathbf{S}}_i)\sigma_j^{(2)}(\sum_{k=1}^{K} (\tilde{\mathbf{S}}_j - \tilde{\mathbf{S}}_k)\sigma_k^{(2)}). \end{aligned} \tag{7}$$

The derivation of Eq. (7) uses the fact that $\sum_{k=1}^{K} \sigma_k^{(1)} = \sum_{k=1}^{K} \sigma_k^{(2)} = 1$ and $\frac{\partial \sigma_j}{\partial h_i} = \sigma_j(\delta_{ji} - \sigma_i)$, where $\delta_{ji} = \mathbb{I}\{i = j\}$ is the Kronecker delta function.

Using gradient descent algorithm, the updated value for $\tilde{\mathbf{S}}_j$ and h_j are

$$\begin{aligned} \tilde{\mathbf{S}}'_j &= \tilde{\mathbf{S}}_j - \alpha(\frac{\partial \ell_1}{\partial \tilde{\mathbf{S}}_j} + \frac{\partial \ell_2}{\partial \tilde{\mathbf{S}}_j}) \\ h'_j &= h_j - \alpha(\frac{\partial \ell_1}{\partial h_j} + \frac{\partial \ell_2}{\partial h_j}), \end{aligned} \tag{8}$$

where $0 < \alpha < 1$ is the learning rate.

B. Mistakes in VNet and ANet

A desirable property of this model is to have discriminative activations in both VNet and ANet for different types of instruments. A mistake by the VNet and ANet during training is defined to be the case when at least one of them has the strongest activation on the same channel for both video samples x_1 and x_2. Concretely, the VNet makes a mistake if it has the same classification for the two input visual signals. Formally, there exists a channel j such that

$$j = \underset{t \in \{1,\ldots,K\}}{\arg\max}\ \sigma_t^{(1)} = \underset{t \in \{1,\ldots,K\}}{\arg\max}\ \sigma_t^{(2)}. \tag{9}$$

The ANet makes a mistake if one audio channel responds to the audio mixture while the rest are not activated. Formally, there exists a channel j such that

$$\forall t \in \{1, \ldots, K\} \text{ and } t \neq j : \tilde{\mathbf{S}}_t = 0 \text{ and } \tilde{\mathbf{S}}_j > 0. \tag{10}$$

Next, we show that in the self-supervised setting where both networks are trained simultaneously, when one of the two networks makes a mistake, that mistake is propagated in the other network.

When the VNet makes a mistake: We assume that the channel j of the VNet has strongest activations on both branches of the training procedure. Expanding Eq. (8) for $\tilde{\mathbf{S}}_j$ and using Eq. (6) we obtain:

$$\tilde{\mathbf{S}}'_j = \beta_j^{(1)}\mathbf{S}_1 + \beta_j^{(2)}\mathbf{S}_2 + (1 - \beta_j^{(1)}\sigma_j^{(1)} - \beta_j^{(2)}\sigma_j^{(2)})\tilde{\mathbf{S}}_j + C_j \tag{11}$$

where $\beta_j^{(i)} = \alpha\sigma_j^{(i)}$ and $C_j = \sum_{k=1, k\neq j}^{K} (\sigma_k^{(1)}\sigma_j^{(1)} + \sigma_k^{(2)}\sigma_j^{(2)})\tilde{\mathbf{S}}_k$ does not depend on $\tilde{\mathbf{S}}_j$. The weight $0 < \beta_j^{(i)} < 1$ indicates how much the audio channel $\tilde{\mathbf{S}}_j$ is pulled towards the audio signal $\mathbf{S}_i$. Since $\mathbf{S}_1$ and $\mathbf{S}_2$ are the audio signals of two different objects, the ANet is expected have one audio channel with the largest weight for $\mathbf{S}_1$ and another audio channel with the largest weight for $\mathbf{S}_2$. However, when the VNet makes a mistake, by the definition in Eq. (9) both $\beta_j^{(1)}$ and $\beta_j^{(2)}$ are the largest weights in both branches. Consequently, the audio channel $\tilde{\mathbf{S}}_j$ is pulled towards both $\mathbf{S}_1$ and $\mathbf{S}_2$ simultaneously. Hence, this channel responds to a combination of two audio sources $\mathbf{S}_1$ and $\mathbf{S}_2$ instead of one particular audio source.

When the ANet makes a mistake: The following proposition shows that under a specific condition, when ANet makes a mistake then the same mistake is propagated in the VNet.

Proposition 1: When $\mathbf{S}_1 \geq \sum_{i=1}^{K} \sigma_i^{(1)} \tilde{\mathbf{S}}_i$ and $\mathbf{S}_2 \geq \sum_{i=1}^{K} \sigma_i^{(2)} \tilde{\mathbf{S}}_i$, if the ANet makes a mistake on channel j then both $h_j^{(1)}$ and $h_j^{(2)}$ will be increased by the learning algorithm.

Proof: Since the ANet makes a mistake on channel j, using the definition in Eq. (10) we obtain $\forall k \neq j, \tilde{\mathbf{S}}_j - \tilde{\mathbf{S}}_k = \tilde{\mathbf{S}}_j > 0$. On the first branch, since $\mathbf{S}_1 \geq \sum_{i=1}^{K} \sigma_i^{(1)} \tilde{\mathbf{S}}_i$, it follows that in Eq. (7) the partial derivative $\frac{\partial \ell_1}{\partial h_j} < 0$. Similarly, $\frac{\partial \ell_2}{\partial h_j} < 0$ on the second branch. Consequently, by Eq. (8), $h'_j = h_j - \alpha(\frac{\partial \ell_1}{\partial h_j} + \frac{\partial \ell_2}{\partial h_j}) > h_j$. ■

The proposition states that when the predicted audio signals in both branches are smaller than the two input audio signals, if the ANet makes a mistake then the same visual channel in VNet in both branches is updated to respond more strongly toward two different objects. Due to the random initialization, both the ANet and VNet are expected to make mistakes at the beginning of the training phase. We hypothesize that these mistakes are amplified as the training progresses, and when the training finishes the ANet and VNet are unable to learn the 1-to-1 correspondence between audio and visual data.

C. Sound source separation experiment with three instruments

To provide empirical evidence for the aforementioned hypothesis, we replicate the sound source separation experiment for the PixelPlayer model in [1] on a smaller dataset of three instruments: accordion, trumpet and tuba. These three instrument types are chosen because their classification accuracy by the VNet and ANet of the PixelPlayer model reported in [1] are among the highest. We extract 143 solo videos from the MUSIC dataset [1] and divide them into a training set of 114 videos and a validation set of 29 videos. The number of videos of each instrument in the training set is 43 for accordion, 28 for trumpet and 43 for tuba. If the VNet in the mix-and-separate training procedure could learn object recognition, then on this simplified dataset, after training, it should be able to discriminate the three instrument classes by assigning one visual channel for each of the three instruments.

In contrast, after training we find that both the VNet and ANet are unable to differentiate between videos of trumpet and tuba. Specifically, for all samples in the validation set, the first channel (index 1) of the VNet is activated strongest towards images of both tuba and trumpet videos. The second channel (index 2) responds strongest to frames of accordion videos, whereas the third channel (index 3) does not respond to any instruments. The same behavior is observed on the audio channels of the ANet. Consequently, if the first visual channel is assigned to trumpet, then the classification accuracy of tuba is 0% and vice versa. Surprisingly, this behavior of the VNet and ANet does not prevent the whole model from achieving non-trivial performance on the sound source separation task. The sound source separation performance is reported in three metrics: Signal-to-Distortion Ratio (SDR), Signal-to-Inference (SIR) and Signal-to-Artifact Ratio (SAR). Methods with higher scores in these metrics are considered better[1]. The results are reported in Table I, showing that this newly trained model, denoted as 3-class PixelPlayer, is approximately 1 dB behind the performance of the pre-trained model provided by the PixelPlayer authors [1] in all three metrics. Note that this performance is higher to that of the Non-negative Matrix Factorization (NMF) baseline [22].

[1]A more detailed discussion of these metrics can be found in [21].

TABLE I
PERFORMANCE ON THE TEST SET OF 29 VIDEOS OF 3 INSTRUMENT CLASSES.

	SDR	SIR	SAR
3-class PixelPlayer	3.61	7.61	10.06
NMF [22]	3.14	6.70	10.10
Pre-trained PixelPlayer [1]	**4.85**	**8.81**	**11.16**

The discrepancy in the performance of the visual object recognition and audio source separation shows that the PixelPlayer model can obtain high performance in audio-visual sound source separation without learning the 1-to-1 correspondence between visual and audio channels. Consequently, the explainability of the model is limited.

IV. SOUND SOURCE SELECTION WITH OBJECT-PRIOR

In this section, we propose a method called Object-Prior for preventing the VNet and ANet from using the same channel for the audio signals of two different instruments. Section IV-A describes the motivation for this method based on the theoretical analysis in Section III. Section IV-B and IV-C provides the empirical performance of Object-Prior in the sound source separation task and the audio-based object recognition task.

A. Motivation for Object-Prior Method

Suppose that the VNet does not make any mistakes and without loss of generality, it assigns visual channel 1 to the object type in the first video $\mathbf{x}_1 = (\mathbf{I}_1, \mathbf{S}_1)$ and channel 2 to the object type in the second video $\mathbf{x}_2 = (\mathbf{I}_2, \mathbf{S}_2)$. In this case, the probability distribution given by the VNet on the first branch concentrates on index 1. Formally,

$$\sigma_j^{(1)} = \begin{cases} 1 & \text{if } j = 1 \\ 0 & \text{if } j \neq 1. \end{cases} \tag{12}$$

Similarly, on the second branch the probability distribution from VNet concentrates at index 2:

$$\sigma_j^{(2)} = \begin{cases} 1 & \text{if } j = 2 \\ 0 & \text{if } j \neq 2. \end{cases} \tag{13}$$

Plugging Eq. (12) and (13) into Eq. (11) and removing the terms equal to 0, the updated values for $\tilde{\mathbf{S}}_1$ and $\tilde{\mathbf{S}}_2$ are

$$\begin{aligned} \tilde{\mathbf{S}}'_1 &= \alpha \mathbf{S}_1 + (1-\alpha)\tilde{\mathbf{S}}_1 \\ \tilde{\mathbf{S}}'_2 &= \alpha \mathbf{S}_2 + (1-\alpha)\tilde{\mathbf{S}}_2. \end{aligned} \tag{14}$$

Equation 14 states that each audio channel is updated towards one distinct audio source. When the training converges, each audio channel gets assigned to the sound of one object type. The mistakes from ANet at the beginning of the training process are corrected in the end.

Motivated by this analysis, a two-step training procedure is developed. Initially, the VNet is trained to recognize the instrument type in a video frame with one-hot encoding labels.

Next, this pre-trained VNet is used in the mix-and-separate procedure to learn the whole sound source separation network. In the second step, the ANet and the synthesizer are trained while the VNet is frozen. Since this procedure uses an object recognition network trained *a priori* to perform sound source separation, we call this method *Object-Prior*. Because the first training step still requires human knowledge to label the types of instrument in a small number of videos, this approach is weakly-supervised rather than self-supervised.

B. Experiments

Experiments are carried out to measure the effectiveness of the Object-Prior method in three aspects:

- The performance on the audio-visual sound source separation task
- The robustness against asynchronized audio-visual data
- The ability to recognize an instrument based on its sound

Datasets: the sound separation performance is evaluated on the MUSIC [1] and AudioSet-SingleSource [2] datasets. The MUSIC dataset consists of 685 videos of 11 instruments, with 536 solo videos and 149 duet videos. In this study, we focus on solo videos. After eliminating videos that are no longer available for download, we obtain a set of 505 solo videos. All videos in MUSIC are guaranteed to contain the visual appearance of their instruments in their frames. The AudioSet-SingleSource contains 15 solo musical videos of over 100 instrument types. Each video has a 10-second length and contains one instrument. All the videos are selected so that each of them has at least one frame that contains the annotated instrument. The purpose of this selection step is to keep the comparison fair for PixelPlayer [1] and our Object-Prior methods which do not use all frames of a video.

Baseline: We compare the Object-Prior method to PixelPlayer [1], DeepConvSep [23], NMF-MFCC [24], Co-Separation [5] and AV-MIML [2]. DeepConvSep and NMF-MFCC are two audio-only based approaches and serve to illustrate the effectiveness of using visual cues for sound source separation. Co-Separation [5] is a mid-fusion method. AV-MIML [2] uses Non-negative Matrix Factorization technique instead of deep learning to process the audio signals.

Implementation Details: Similar to [1], we use the ResNet18 [25] architecture for VNet and the U-Net architecture with 7 down-convolutions and 7 up-convolutions for ANet. To ensure consistency, all videos and audios are pre-processed similarly to [1]. All videos are sub-sampled at the rate of 8 fps and all frames are cropped to the size 224×224 during training and testing. All audio signals are sub-sampled to 11kHz. The input audio mixture to the ANet is converted to a 256×256 Short-time Fourier Transform (STFT) representation. We use the log-frequency scale for the audio signal and binary masks for the ANet since they are reported to have the best SDR and SIR scores in [1]. The SDR, SIR and SAR metrics are computed using the `mir_eval` library. The implementation of the three networks is in PyTorch.

Training the instrument recognition network: The VNet is fine-tuned on the 11 instrument categories of MUSIC dataset. We use 90%, 5% and 5% of the solo videos in the MUSIC dataset for training, validation and testing, respectively. For each video in the training set, frames are extracted at the rate of 1 fps (i.e. 1 frame per second). For each video in the validation and test sets, frames are extracted at the rate of 0.5 fps (i.e. 1 frame per 2 seconds). We find empirically that such small sampling rates are effective in reducing the training time and preventing overfitting. In the end we have 55691 images for training, 1695 images for validation and 1903 images for testing. The VNet is fine-tuned for 20 epochs with a constant learning rate of 0.001. The VNet takes an image of size $224 \times 224 \times 3$ as input and outputs a feature vector of size of $K = 11$ which subsequently goes through a softmax layer. The ground truth index of each instrument is defined as its index in the sorted list of instrument names, i.e. the index of the accordion is 0 and the index of the xylophone is 11. The fine-tuning converges quickly after 5 epochs and obtains approximately 90% accuracy on validation and test set.

Training the sound source separation network: Having the VNet fine-tuned for instrument recognition, the ANet and synthesizer are trained with the prior information about object type from the VNet. From 505 usable solo videos in the MUSIC dataset, we split these videos into 300 videos for training, 130 videos for validation and 75 videos for testing. The performance of the PixelPlayer [1] was reported on a validation set, which can lead to an overly optimistic estimation of its performance [26]. We instead report the performance of the Object-Prior method and the PixelPlayer on the held-out test set of 75 videos.

We train the Object-Prior on 300 videos in the training set with the same hyper-parameters provided by the authors in [1]. The learning rate for VNet is set to 0. The number of training epochs is 100. The learning rate is set initially at 0.001 for both ANet and synthesizer and subsequently reduced by a factor of 10 at epochs 40 and 80.

C. Analysis

We first show that the Object-Prior method obtains higher sound separation performance than the baseline methods on the MUSIC and AudioSet-SingleSource datasets. Next, the robustness of the Object-Prior method on asynchronized data is illustrated. Finally, we compare the ability to classify an instrument from its sound of the Object-Prior method and the PixelPlayer [1] and show that the Object-Prior has a significantly higher classification accuracy.

1) Sound source separation performance: The results on the MUSIC dataset of the Object-Prior and the baselines are shown in Table II. All the baseline methods use one frame from each video as input to the VNet during training and testing, even though methods like Co-Separation [5] browse through all frames in a video to detect one suitable frame as input to the VNet. The performances of the PixelPlayer, NMF-MFCC and Co-Separation are taken from [5] and the performance of DeepConvSep is taken from [1]. The Object-Prior method outperforms the baselines by a large margin in both SDR and SIR on the MUSIC test set with over 1dB higher

TABLE II
SOUND SOURCE SEPARATION PERFORMANCE ON THE MUSIC DATASET

	SDR	SIR	SAR
PixelPlayer [1]	7.30	11.9	**11.9**
DeepConvSep [23]	6.12	8.38	11.02
NMF-MFCC [24]	0.92	5.68	6.84
Co-Separation [5]	7.38	13.7	10.8
Object-Prior (Ours)	**8.92**	**14.49**	11.44

TABLE III
PERFORMANCE ON THE FILTERED AUDIOSET TEST SET.

	SDR	SIR	SAR
PixelPlayer [1]	1.66	3.58	11.5
AV-MIML [2]	1.83	-	-
NMF-MFCC [24]	0.25	4.19	5.78
Co-Separation [5]	4.26	7.07	**13.0**
Object-Prior (Ours)	**6.58**	**12.33**	9.28

in both SDR and SIR compared to the second best method. Since the SDR and SIR measure sound separation quality, this result indicates that the prior information about object type improves the mix-and-separate training paradigm. The Object-Prior obtains a slightly smaller SAR than the PixelPlayer method, indicating that the signal-to-artifact ratio in the output of Object-Prior is higher but it does not significantly impact the sound separation quality.

The performance of Object-Prior and other methods on the AudioSet-SingleSource is reported in Table III. The reported results are the average value of five runs with five different random seeds from 0 to 4. Object-Prior outperforms the baselines significantly on SDR and SIR with approximately 2 dB higher in SDR and 5 dB higher in SIR. These results come from the same Object-Prior model in Table II, which is not trained on the AudioSet dataset, indicating that the VNet and ANet in Object-Prior generalize to the unseen videos in the AudioSet dataset.

2) Sound source separation performance on asynchronized data: We further evaluate the robustness of the Object-Prior method on asynchronized data where the frame and the audio do not come from the same video. We use the MUSIC dataset for this experiment. Before inputting into the VNet, each frame of the two videos in a mixture is replaced by another frame from a randomly chosen video in the test set of the same instrument. For example, if the frame in a video contains a flute then it is replaced by a frame from another video also containing the flute. The results of the PixelPlayer and Object-Prior methods are reported in Table IV. The Object-Prior outperforms the PixelPlayer by roughly 1 dB on SAR and 2 dB on SIR, indicating that the Object-Prior works much better on asynchronized data. This result shows that the Object-Prior model is applicable for asynchronized sound source separation on an audio recording whose visual stream is not available.

3) Audio-visual instrument classification: To evaluate the ability of the ANet to recognize an instrument based on its *sound*, for each instrument we compute the classification accuracy based on the strongest activated audio channel when presented with a video of that instrument. For each of the 130 videos in the validation set, we a perform mix-and-separate procedure with the input of both branches are the same video of interest. For each network, the index of the channel with the strongest activation is considered to be the classifying result. For VNet, the strongest activated channel is taken directly from the softmax layer. For ANet, since each audio channel is a 256×256 two-dimensional array, the activation of a channel is considered to be the average value of its elements. The ground truth index of an instrument is identical to its index in the fine-tuning process of the VNet. The confusion matrix is then computed from the classification results of all videos in the validation set. Fig. 3 illustrates the confusion matrix of the ANet in terms of the percentage of samples of each instrument. It can be observed that all of the 11 instruments have more than 50% classification accuracy based on the ANet, and 7 out of 11 instruments have more than 90% accuracy. Overall, the classification accuracy of the ANet is 92.31%, substantially higher than the reported 68.9% of the PixelPlayer [1]. An interesting observation from the confusion matrix in Fig. 3 is the discrepancy in the performance of the ANet on wind instruments: the three instruments with the lowest accuracy (saxophone, clarinet and flute) belong to the same category of woodwind instruments, whereas the two brasswind instruments (trumpet and tuba) are in the group of highest classification accuracy. Given that the numbers of woodwind and brasswind videos are similar [1], this result indicates that certain characteristics of woodwind instruments are more challenging to separate. Since the VNet of Object-Prior is explicitly trained for instrument recognition, it is expected to obtain a much higher classification accuracy than that of the PixelPlayer. Detailed results of both ANet and VNet are shown in Table V. We conclude that the prior information on instrument type indeed increases the discrimination in activations of the audio channel on different types of instruments.

TABLE IV
PERFORMANCE WITH ASYNCHRONIZED DATA ON A HELD-OUT TEST SET FROM THE MUSIC DATASET

	SDR	SIR	SAR
PixelPlayer [1]	6.43	11.18	**11.16**
Object-Prior (Ours)	**8.22**	**13.88**	11.03

TABLE V
CLASSIFICATION ACCURACY BASED ON VISUAL AND AUDIO CHANNELS OF THE PIXELPLAYER [1] AND OBJECT-PRIOR.

	By Visual Channel	By Audio Channel
PixelPlayer [1]	46.2%	68.9%
Object-Prior (Ours)	**96.15**%	**92.31**%

V. CONCLUSION

In this paper, we have proposed the *Object-Prior* model for audio-visual sound source separation. This model emerges as

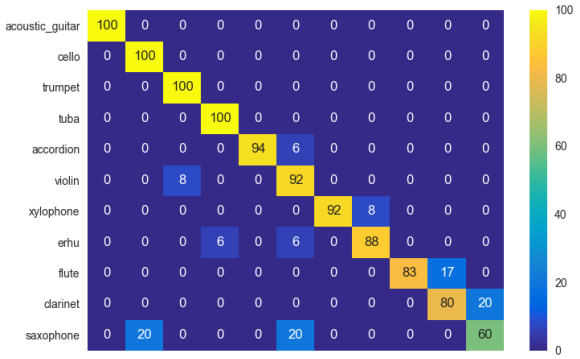

Fig. 3. Confusion matrix of object classifier based on audio channels. All eleven instrument types have more than 50% accuracy.

a solution to an apparent weakness of the mix-and-separate training procedure for audio-visual sound source separation. We provided both theoretical and empirical analyses on the learning process for a late-fusion mix-and-separate training procedure, showing that the VNet and ANet are unable to learn the proper 1-to-1 correspondence between the visual and audio channels when they are trained simultaneously. The effectiveness and robustness of the Object-Prior model are evaluated empirically on two challenging datasets. The results indicate that the Object-Prior model achieves state-of-the-art performance on the sound source separation task while being robust to asynchronized audio-visual data. However, the Object-Prior model still relies on human supervision to train the VNet and therefore is not a self-supervised method. Future work includes regularization methods which can enforce the 1-to-1 correspondence between visual and audio channels without relying on human supervision.

ACKNOWLEDGMENT

This work has been funded by ahoi.digital, the alliance of the Hamburg universities for computer science, and by the German Research Foundation (DFG) in project Transregio Crossmodal Learning (TRR 169).

REFERENCES

[1] H. Zhao, C. Gan, A. Rouditchenko, C. Vondrick, J. McDermott, and A. Torralba, "The sound of pixels," in *The European Conference on Computer Vision (ECCV)*, September 2018.

[2] R. Gao, R. Feris, and K. Grauman, "Learning to separate object sounds by watching unlabeled video," in *ECCV*, 2018.

[3] S. Parekh, A. Ozerov, S. Essid, N. Q. K. Duong, P. Prez, and G. Richard, "Identify, locate and separate: Audio-visual object extraction in large video collections using weak supervision," in *2019 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, Oct 2019, pp. 268–272.

[4] A. Owens and A. A. Efros, "Audio-visual scene analysis with self-supervised multisensory features," in *Computer Vision – ECCV 2018*, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds. Cham: Springer International Publishing, 2018, pp. 639–658.

[5] R. Gao and K. Grauman, "Co-separating sounds of visual objects," in *The IEEE International Conference on Computer Vision (ICCV)*, October 2019.

[6] X. Xu, B. Dai, and D. Lin, "Recursive visual sound separation using minus-plus net," in *The IEEE International Conference on Computer Vision (ICCV)*, October 2019.

[7] A. Pieropan, G. Salvi, K. Pauwels, and H. Kjellstrm, "Audio-visual classification and detection of human manipulation actions," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2014, pp. 3045–3052.

[8] A. Sterling, J. Wilson, S. Lowe, and M. C. Lin, "Isnn: Impact sound neural network for audio-visual object classification," in *Computer Vision – ECCV 2018*, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds. Cham: Springer International Publishing, 2018, pp. 578–595.

[9] A. Rouditchenko, H. Zhao, C. Gan, J. H. McDermott, and A. Torralba, "Self-supervised audio-visual co-segmentation," *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2357–2361, 2019.

[10] F. R. Bach and M. I. Jordan, "Learning spectral clustering, with application to speech separation," *J. Mach. Learn. Res.*, vol. 7, p. 19632001, Dec. 2006.

[11] P.-S. Huang, M. Kim, M. Hasegawa-Johnson, and P. Smaragdis, "Joint optimization of masks and deep recurrent neural networks for monaural source separation," *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, vol. 23, no. 12, p. 21362147, Dec. 2015.

[12] A. J. R. Simpson, G. Roma, and M. 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*, 2015.

[13] J. R. Hershey, Z. Chen, J. Le Roux, and S. Watanabe, "Deep clustering: Discriminative embeddings for segmentation and separation," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 31–35.

[14] D. Yu, M. Kolbk, Z. Tan, and J. Jensen, "Permutation invariant training of deep models for speaker-independent multi-talker speech separation," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 241–245.

[15] J. Ramaswamy and S. Das, "See the sound, hear the pixels," in *The IEEE Winter Conference on Applications of Computer Vision (WACV)*, March 2020.

[16] R. Arandjelovic and A. Zisserman, "Objects that sound," in *The European Conference on Computer Vision (ECCV)*, September 2018.

[17] H. Zhao, C. Gan, W.-C. Ma, and A. Torralba, "The sound of motions," in *The IEEE International Conference on Computer Vision (ICCV)*, October 2019.

[18] H. Zhu, M. Luo, R. Wang, A. Zheng, and R. He, "Deep audio-visual learning: A survey," *ArXiv*, vol. abs/2001.04758, 2020.

[19] G. Chrupala, L. Gelderloos, and A. Alishahi, "Representations of language in a model of visually grounded speech signal," in *Proceedings of the 55th of the Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2017, p. 613622.

[20] A. Ephrat, I. Mosseri, O. Lang, T. Dekel, K. Wilson, A. Hassidim, W. T. Freeman, and M. Rubinstein, "Looking to listen at the cocktail party: A speaker-independent audio-visual model for speech separation," *ACM Trans. Graph.*, vol. 37, no. 4, Jul. 2018.

[21] J. L. Roux, S. Wisdom, H. Erdogan, and J. R. Hershey, "Sdr half-baked or well done?" *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 626–630, 2019.

[22] T. Virtanen, "Monaural sound source separation by nonnegative matrix factorization with temporal continuity and sparseness criteria," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 3, pp. 1066–1074, 2007.

[23] P. Chandna, M. Miron, J. Janer, and E. Gómez, "Monoaural audio source separation using deep convolutional neural networks," in *LVA/ICA*, 2017.

[24] M. Spiertz and V. Gnann, "Source-filter based clustering for monaural blind source separation," in *in Proceedings of International Conference on Digital Audio Effects DAFx09*, 2009.

[25] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[26] Y. Xu and R. Goodacre, "On splitting training and validation set: A comparative study of cross-validation, bootstrap and systematic sampling for estimating the generalization performance of supervised learning," *Journal of analysis and testing*, vol. 2, no. 3, pp. 249–262, 2018.