Visual Scene Graphs for Audio Source Separation

Moitreya Chatterjee[1] Jonathan Le Roux[2] Narendra Ahuja[1] Anoop Cherian[2]
[1]University of Illinois at Urbana-Champaign, Champaign, IL 61820, USA
[2]Mitsubishi Electric Research Laboratories, Cambridge, MA 02139, USA
metro.smiles@gmail.com leroux@merl.com n-ahuja@illinois.edu cherian@merl.com

Abstract

State-of-the-art approaches for visually-guided audio source separation typically assume sources that have characteristic sounds, such as musical instruments. These approaches often ignore the visual context of these sound sources or avoid modeling object interactions that may be useful to better characterize the sources, especially when the same object class may produce varied sounds from distinct interactions. To address this challenging problem, we propose Audio Visual Scene Graph Segmenter *(AVSGS), a novel deep learning model that embeds the visual structure of the scene as a graph and segments this graph into subgraphs, each subgraph being associated with a unique sound obtained by co-segmenting the audio spectrogram. At its core, AVSGS uses a recursive neural network that emits mutually-orthogonal sub-graph embeddings of the visual graph using multi-head attention. These embeddings are used for conditioning an audio encoder-decoder towards source separation. Our pipeline is trained end-to-end via a self-supervised task consisting of separating audio sources using the visual graph from artificially mixed sounds.*

In this paper, we also introduce an "in the wild" video dataset for sound source separation that contains multiple non-musical sources, which we call Audio Separation in the Wild *(ASIW). This dataset is adapted from the AudioCaps dataset, and provides a challenging, natural, and daily-life setting for source separation. Thorough experiments on the proposed ASIW and the standard MUSIC datasets demonstrate state-of-the-art sound separation performance of our method against recent prior approaches.*

1. Introduction

Real-world events often encompass spatio-temporal interactions of objects, the signatures of which leave imprints both in the visual and auditory domains when captured as videos. Knowledge of these objects and the sounds that they produce in their natural contexts are essential when designing artificial intelligence systems to produce meaningful deductions. For example, the sound of a cell phone *ringing* is drastically different from that of one *dropping on the floor*; such distinct sounds of objects and their contextual interactions may be essential for an automated agent to assess the scene. The importance of having algorithms with such audio-visual capabilities is far reaching, with applications such as audio denoising, musical instrument equalization, audio-guided visual surveillance, or even in navigation planning for autonomous cars, for example by *visually localizing the sound of an ambulance*.

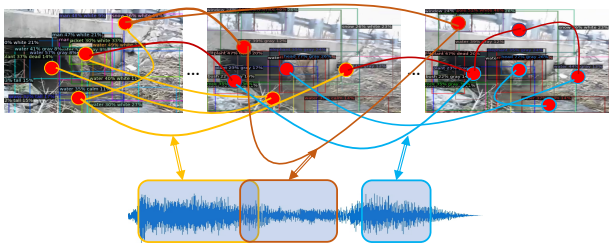

Figure 1. A schematic illustration of our Audio Visual Scene Graph Segmenter (AVSGS) framework on frames from our Audio Separation in the Wild (ASIW) dataset. Given an input video and the associated audio, our method builds a spatio-temporal (fully-connected) visual scene graph spanning across the video frames, and learns alignments between sub-graphs of this scene graph and the respective audio regions. The use of the scene graph allows rich characterization of the objects and their interactions, allowing effective identification of the sound sources for better separation.

Recent years have seen a surge in algorithms at the intersection of visual and auditory domains, among which visually-guided source separation – the problem of separating sounds from a mixture using visual cues – has made significant strides [5, 9, 58, 57]. State-of-the-art algorithms for this task [54, 58, 57] typically restrict the model design to only objects with unique sounds (such as musical instruments [5, 57]) or consider settings where there is only a single sound source, and the models typically lack the richness to capture spatio-temporal audio-visual context. For example, for a video with *"a guitar being played by a per-*

son" and one in which *"a guitar is kept against a wall"*, the context may help a sound separation algorithm to decide whether to look for the sound of the guitar in the audio spectrogram; however several of the prior works only consider visual patches of an instrument as the context to guide the separation algorithm [9], which is sub-optimal.

From a learning perspective, the problem of audio-visual sound source separation brings in several interesting challenges: (i) The association of a visual embedding of a sound source to its corresponding audio can be a one-to-many mapping and therefore ill-posed. For example, *a dog barking while splashing water in a puddle*. Thus, methods such as [5, 9] that assume a single visual source may be misled. (ii) It is desirable that algorithms for source separation are scalable to new sounds and their visual associations; i.e., the algorithm should be able to master the sounds of varied objects (unlike supervised approaches [48, 53]). (iii) Naturally occurring sounds can emanate out of a multitude of interactions – therefore, using *a priori* defined sources, as in [5, 9], can be limiting.

In this work, we rise up to the above challenges using our Audio Visual Scene Graph Segmenter (AVSGS) framework for the concrete task of sound source separation. Figure 1 presents the input-output setting for our task. Our setup represents the visual scene using spatio-temporal scene graphs [18] capturing visual associations between objects occurring in the video, towards the goal of training AVSGS to infer which of these visual associations lead to auditory grounding. To this end, we design a recursive source separation algorithm (implemented using a GRU) that, at each recurrence, produces an embedding of a sub-graph of the visual scene graph using graph multi-head attention. These embeddings are then used as conditioning information to an audio separation network, which adopts a U-Net style encoder-decoder architecture [39]. As these embeddings are expected to *uniquely* identify a *sounding interaction*, we enforce that they be mutually orthogonal. We train this system using a self-supervised approach similar to Gao *et al*. [9], wherein the model is encouraged to disentangle the audio corresponding to the conditioned visual embedding from a mixture of two or more different video sounds. Importantly, our model is trained to ensure consistency of each of the separated sounds by their type, across videos. Thus, two guitar sounds from two disparate videos should sound more similar than a guitar and a piano. Post separation, the separated audio may be associated with the visual sub-graph that induced its creation, making the sub-graph an *Audio-Visual Scene Graph* (AVSG), usable for other downstream tasks.

We empirically validate the efficacy of our method on the popular Multimodal Sources of Instrument Combinations (MUSIC) dataset [58] and a newly adapted version of the AudioCaps dataset [20], which we call Audio Separation in the Wild (ASIW). The former contains videos of performers playing musical instruments, while the latter features videos of naturally occurring sounds arising out of complex interactions *in the wild*, collected from YouTube. Our experiments demonstrate the importance of visual context in sound separation, and AVSGS outperforms prior state-of-the-art methods on both of these benchmarks.

We now summarize the key contributions of the paper:

- To the best of our knowledge, ours is the first work to employ the powerful scene graph representation [18] for the task of visually-guided audio source separation.
- We present AVSGS for this task, that is trained to produce mutually-orthogonal embeddings of the visual sub-graphs, allowing our model to infer representations of *sounding interactions* in a self-supervised way.
- We present ASIW, a large scale *in the wild* dataset adapted from *AudioCaps* for the source separation task. This dataset features sounds arising out of natural and complex interactions.
- Our AVSGS framework demonstrates state-of-the-art performance on both the datasets for our task.

2. Related Works

In this section, we review relevant prior works which we group into several categories for ease of readability.

Audio Source Separation has a very long history in the fields of signal processing and more recently machine learning [3, 6, 29, 46, 48, 47]. *Audio-only* methods have typically either relied on *a priori* assumptions on the statistics of the target sounds (such as independence, sparsity, etc.), or resorted to supervised training to learn these statistics [42] (and/or optimize the separation process from data [51]) via deep learning [55, 14, 50, 13, 56, 48]. Such supervised learning often involves the creation of synthetic training data by mixing known sounds together and training the model to recover target sounds from the mixture. Settings where isolated target sources are unavailable have recently been considered, either by relying on weak sound-event activity labels [36], or using unsupervised methods that learn to separate mixtures of mixtures [52].

Audio-Visual Source Separation considers the task of discovering the association between the acoustic signal and its corresponding signature in the visual domain. Such methods have been employed for tasks like speech separation [1, 4, 31], musical instrument sound separation [9, 5, 57, 58], and separation of on-screen sounds of generic objects [33, 43]. More recently, researchers have sought to integrate motion information into the visual representation of these methods, either in the form of pixel trajectories [57], or human pose [5]. However, these approaches adopt a video-level "mix-and-separate" training strategy which works best with clean, single-source videos. Differently, our approach is trained to disentangle sound sources within

a video. Gao *et al*. [9] proposed an approach in a similar regime, however they do not capture the visual context, which may be essential to separate sound that emanates as a result of potentially complex interactions between objects in the scene. Further, our proposed framework allows characterizing generic sounds that can arise from fairly unconstrained settings, unlike approaches that are tailored to tasks such as musical instrument sound separation.

Localizing Sound in Video Frames seeks to identify the pixels in a video frame that visually represent the sound source. Several approaches have been proposed for this task [2, 19, 40, 19, 12]. While such methods do visually ground the audio sources, they do not separate the audio, which is the task we consider.

Synthesizing Sound from Videos constitutes another class of techniques in the audio-visual paradigm [34, 59] that has become popular in recent years. For example, [8, 32] propose frameworks capable of generating both monaural and binaural audio starting from videos. However, we are interested in separating the audio from different sound sources, starting with a mixed audio.

Scene Graphs in Videos have proven to be an effective toolkit in representing the content of static images [18, 26] capable of capturing the relationship between different objects in the scene. These representations have only recently been deployed to videos for tasks such as action recognition [17] and visual dialog [11]. We employ these powerful representations to separate a mixed audio into its constituent sources, which can then be associated with their corresponding sub-graphs for other downstream tasks.

3. Proposed Method

We begin this section by first presenting a description of the problem setup along with an overview of our model. We then delve deeper into the details of the model and finish the section by providing the details of our training setup.

3.1. Problem Setup and Overview

Given an unlabeled video V and its associated discrete-time audio $x(t) = \sum_{i=1}^{N} \mathbf{s}_i(t)$ consisting of a linear mixture of N audio sources $\mathbf{s}_i(t)$, the objective in *visually-guided* source separation is to use V to disentangle $x(t)$ into its constituent sound sources $\mathbf{s}_i(t)$, for $i \in \{1, 2, \ldots, N\}$. In this work, we represent the video as a spatio-temporal visual scene graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, with nodes $\mathcal{V} = \{v_1, v_2, \ldots, v_K\}$ representing objects (including people) in the video V, and $\mathcal{E}$ denoting the set of edges e_{jk} capturing the pairwise interaction or spatial context between nodes v_j and v_k. Our main idea in AVSGS is to learn to associate each audio source $\mathbf{s}_i(t)$ with a visual sub-graph g_i of $\mathcal{G}$. We approach this problem from the perspective of graph attention pooling to produce mutually-orthogonal sub-graph embeddings auto-regressively; these embeddings are made to be aligned with the respective audio sources using an *Audio Separator* sub-network that is trained against a self-supervised *unmixing* task [9, 58, 57]. Figure 2 presents an overview of the algorithmic pipeline of our model.

3.2. Audio Visual Scene Graph Segmenter Model

Figure 2 presents an illustration of the algorithmic pipeline that we follow in order to obtain the separated sounds $\mathbf{s}_i(t)$ from their mixture $x(t)$. Below, we present the details of each step of this pipeline.

Object Detector: The process of representing a video V as a spatio-temporal scene graph starts with detecting a set of M objects and their spatial bounding boxes in each frame of the video. As is common practice, we use a Faster-RCNN (FRCNN) [38] model for this task, trained on the Visual Genome dataset [23]. As this dataset provides around 1600 object classes, denoted $\mathcal{C}$, it can detect a significant set of common place objects. Further, for detecting objects that are not in the Visual Genome classes (for example, the musical instruments in the MUSIC dataset which we consider later), we trained a separate FRCNN model with labeled images from the Open Images dataset [22], which contains those instrument annotations.

Given a video frame I, the object detector FRCNN produces a set of M quadruples $\left\{\left(C_I^k, B_I^k, F_I^k, S_I^k\right)\right\}_{k=1}^{M} = \text{FRCNN}(I)$, one for each detected object, consisting of the label $C \in \mathcal{C}$ of the detected object, its bounding box B in the frame, a feature vector F identifying the object, and a detection confidence score S.

Visual Scene Graph Construction: Once we have the object detections and their meta-data, our next sub-task is to use this information to construct our visual scene graph. While standard scene graph approaches [17] often directly use the object detections to build the graph (sometimes combined with a visual relationship detector [11]), our task of sound separation demands that the graph be constructed in adherence to the audio, so that the audio-visual correlations can be effectively learned. To this end, for every sound of interest, we associate a *principal object*, denoted p, among the classes in $\mathcal{C}$ (obtainable from the FRCNN) that could have produced the sound. For example, for the sound of a *piano* in an orchestra, the principal object can be the piano, while for the sound of *ringing*, the object could be a *telephone*. Let us denote the set of such principal object classes as $\mathcal{P} \subset \mathcal{C}$.

To construct the visual scene graph for a given video V, we first identify the subset of principal objects $P = \{p_1, \ldots, p_N\} \subset \mathcal{P}$ that are associated with that video. This information is derived from the video metadata, such as for example the video captions or the class labels, if available. Next, we identify the video frames containing the most confident detections of each object $p_i \in P$. We refer to such frames as the *key frames* of the video – our scene graph is constructed using these key frames. For every principal ob-

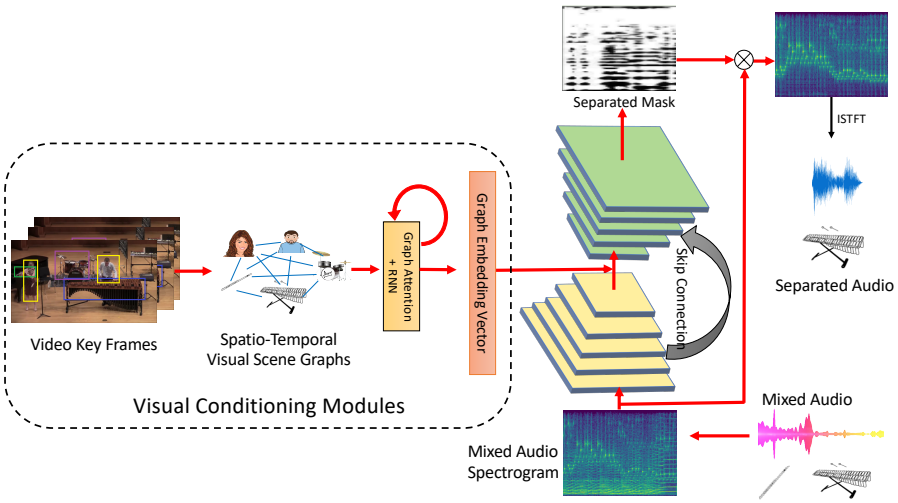

Figure 2. Detailed illustration of our proposed AVSGS model.

ject p_i, we then identify the subset of the M object bounding boxes (produced by FRCNN for that key frame), which have an Intersection Over Union (IoU) with the bounding box for p_i greater than a pre-defined threshold γ. We refer to this overlapping set of nodes as the *context nodes* of p_i, denoted as $\mathcal{V}_{p_i}$. The vertex set of the scene graph is then constructed as $\mathcal{V} = \bigcup_{i=1}^{N}(p_i \bigcup \mathcal{V}_{p_i})$. Note that each graph node v is associated with a feature vector F_v produced by FRCNN for the visual patch within the respective bounding box.

Our next sub-task for scene graph construction is to define the graph edges $\mathcal{E}$. Due to the absence of any supervision to select the edges (and rather than resorting to heuristics), we assume useful edges will emerge automatically from the audio-visual co-segmentation task, and thus, we decided to use a fully-connected graph between all the nodes in $\mathcal{V}$; i.e., our edges are given by: $\mathcal{E} = \{e_{jk}\}_{(j,k)\in\mathcal{V}\times\mathcal{V}}$. Since the scene graph is derived from multiple key frames in the video and its vertices span a multitude of objects in the key frames, our overall scene graph is thus spatio-temporal in nature.

Visual Embeddings of Sounding Interactions: The visual scene graph $\mathcal{G}$ obtained in the previous step is a holistic representation of the video, and thus characterizes the visual counterpart of the mixed audio $x(t)$. To separate the audio sources from the mixture, AVSGS must produce visual cues that can distinctly identify the sound sources. However, we neither know the sources nor do we know what part of the visual graph is producing the sounds. To resolve this dichotomy, we propose a joint learning framework in which the visual scene graph is segmented into sub-graphs, where each sub-graph is expected to be associated with a unique sound in the audio spectrogram, thus achieving source separation. To guide the model to learn to correctly achieve the audio-visual segmentation, we use a self-supervised task described in the next section. For now, let us focus on the modules needed to produce embeddings for the visual sub-graph.

For audio separation, there are two key aspects of the visual scene graph that we expect the ensuing embedding to encompass: (i) the nodes corresponding to sound sources and (ii) edges corresponding to sounding interactions. For the former, we use a multi-headed graph attention network [44], taking as input the features F_v associated with the scene graph nodes v and implement multi-head graph message passing, thereby parting attention weights to nodes that the framework (eventually) learns to be important in characterizing the sound. For the latter, i.e., capturing the interactions, we design an edge convolution network [49]. These networks are typically multi-layer perceptrons, $\boldsymbol{h}_\Lambda(\cdot,\cdot)$, which take as input the concatenated features corresponding to a pair of nodes v_j and v_k which are connected by an edge and produces an output vector $\boldsymbol{e}_{jk}$. Λ encapsulates the learnable parameters of this layer. The updated features of a node v_k are then obtained by averaging all the edge convolution embeddings incident on v_k. The two modules are implemented in a cascade with the node attention preceding the edge convolutions. Next, the attended scene graph is pooled using global max-pooling and global average pooling [25]; the pooled features from each operation are then concatenated, resulting in an embedding vector $\boldsymbol{\zeta}$ for the entire graph. As we need to produce N embedding vectors from $\boldsymbol{\zeta}$, one for each source and another additional one for background, we need to keep track of the embeddings generated thus far. To this end, we propose to use a recurrent neural network, implemented using a GRU. In more detail, our final set of visual sub-graph embeddings $Y = \{\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_N, , \boldsymbol{y}_{N+1}\}$, where each $\boldsymbol{y}_i \in \mathbb{R}^d$, is produced auto-regressively as:

$$\boldsymbol{y}_i = \text{GRU}(\boldsymbol{\zeta}; \Delta_{i-1}),\ i = 1, 2, \ldots, N, N+1 \qquad (1)$$

where Δ_{i-1} captures the bookkeeping that the GRU does to keep track of the embeddings generated thus far.

Mutual-Orthogonality of Visual Embeddings: A subtle but important technicality that needs to be addressed for the above framework to succeed is in allowing the GRU to know whether it has generated embeddings for all the audio sources in the mixture. This poses the question *how do we ensure the GRU does not repeat the embeddings?* Practically, we found that this is an important ingredient in our setup for audio source separation. To this end, we propose to enforce mutual orthogonality between the embeddings that the GRU produces. That is, for each recurrence of the GRU, it is expected to produce a unit-normalized embedding $\boldsymbol{y}_i$ that is orthogonal to each of the embeddings generated prior to it, i.e., $\{\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_{i-1}\}$. We include this constraint as a regularization in our training setup. Mathematically, we enforce a softer-version of this constraint given by:

$$\mathcal{L}_{\text{ortho}}(Y) = \sum_{i,j \in \{1,2,\ldots,N\}, i \neq j} (\boldsymbol{y}_i^\top \boldsymbol{y}_j)^2. \qquad (2)$$

One key attribute of this mechanism for deriving the feature representations $\boldsymbol{y}_i$ is that such embeddings could emerge from potentially complex interactions between the objects in the scene graph, unlike popular prior approaches, which resort to more simplistic visual embeddings, such as the whole frame [57] or a single object [9].

Audio Separator Network: The final component in our model is the *Audio Separator Network* (ASN). Given the success of U-Net [39] style encoder-decoder networks for separating audio mixtures into their component sound sources [16, 28], particularly in conditioned settings [9, 30, 58, 41], we adopt this architecture for inducing the source separation. Since we are interested in *visually guiding* the source separation, we condition the bottleneck layer of ASN with the subgraph embeddings $\boldsymbol{y}_i$ produced above. In detail, ASN takes as input the magnitude spectrogram $\mathbf{X} \in \mathbb{R}^{\Omega \times T}$ of a mixed audio $x(t)$, produced via the short-time Fourier transform (STFT), where Ω and T denote the number of frequency bins and the number of video frames, respectively. The spectrogram is passed through a series of 2D-convolution layers, each coupled with Batch Normalization and Leaky ReLU, until we reach the bottleneck layer. At this layer, we replicate each graph embedding $\boldsymbol{y}_i$ to match the spatial resolution of the U-Net bottleneck features, and concatenate along its channel dimension. This concatenated feature tensor is then fed to the U-Net decoder. The decoder consists of a series of up-convolution layers, followed by non-linear activations, each coupled with a skip connection from a corresponding layer in the U-Net encoder and matching in spatial resolution of its output. The final output of the U-Net decoder is a time-frequency mask, $\hat{\mathbf{M}}_i \in [0,1]^{\Omega \times T}$, which when multiplied with the magnitude spectrogram $\mathbf{X}$ of the mixture yields an estimate of the magnitude spectrogram of the separated source $\hat{\mathbf{S}}_i = \hat{\mathbf{M}}_i \odot \mathbf{X}$, where $\odot$ denotes element-wise product. An estimate $\hat{\mathbf{s}}_i(t)$ of the separated waveform signal for the i-th source can finally be obtained by applying an inverse short-time Fourier transform (iSTFT) to the complex spectrogram obtained by combining $\hat{\mathbf{S}}_i$ with the mixture phase. For architectural details, please refer to the supplementary.

3.3. Training Regime

Audio source separation networks are typically trained in a supervised setting in which a synthetic mixture is created by mixing multiple sound sources including one or more *known* target sounds, and training the network to estimate the target sounds when given the mixture as input [14, 13, 48, 50, 55, 56]. In the visually-guided source separation paradigm, building such synthetic data by considering multiple videos and mixing their sounds is referred to as "mix-and-separate" [9, 5, 57, 58]. We train our model in a similar fashion to Gao *et al*. [9], in which a co-separation loss is introduced to allow separation of multiple sources within a video without requiring ground-truth signals on the individual sources. In this training regime, we feed the ASN with a spectrogram representation $\mathbf{X}_{\text{m}}$ of the mixture $x_{\text{m}}(t) = x_1(t) + x_2(t)$ of the audio tracks from two videos, and build representative scene graphs, $\mathcal{G}_1$ and $\mathcal{G}_2$, for each of the two corresponding videos. We then extract unit-norm embeddings from each of these two scene graphs, $\boldsymbol{y}_i^1, i \in \{1, 2, \ldots, N_1\}$ and $\boldsymbol{y}_i^2, i \in \{1, 2, \ldots, N_2\}$. Next, each of these embeddings $\boldsymbol{y}_i^u$ are independently pushed into the bottleneck layer of ASN that takes as input $\mathbf{X}_{\text{m}}$. Once a separated spectrogram $\hat{\mathbf{S}}_i^u$ is obtained as output for the input pair $(\boldsymbol{y}_i^u, \mathbf{X}_{\text{m}})$, we feed this $\hat{\mathbf{S}}_i^u$ to a classifier which enforces the spectrogram signature to be classified as that belonging to one of the principal object classes in P_u. In contrast with [9], where there is a direct relationship between the conditioning by a visual object and the category of the sound to be separated, we here do not know in which order the GRU produced the conditioning embeddings, and thus which principal object class $l_c^u \in P_u$ should correspond to a given embedding $\boldsymbol{y}_i^u$. We therefore consider different permutations σ^u of the ground-truth class labels of video u, matching the ground-truth label of the c-th object to the $\sigma^u(c)$-th embedding, and use the one which yields the minimum cross-entropy loss, similarly to the permutation free (or invariant) training employed in speech separation [13, 15, 56]. Our loss is then:

$$\mathcal{L}_{\text{cons}} = -\sum_{u=1,2} \min_{\sigma^u \in \mathcal{S}_{N_u+1}} \sum_{c=1}^{N_u+1} \log(p^u_{\sigma^u(c)}(l^u_c)), \qquad (3)$$

where $\mathcal{S}_{N_u+1}$ indicates the set of all permutations on $\{1, \ldots, N_u + 1\}$, $p_i^u(l)$ denotes the predicted probability produced by the classifier for class l given $\hat{\mathbf{S}}_i^u$ as input, and l_c^u is the ground-truth class of the c-th object in video u.

Further, in order to restrict the space of plausible audio-visual alignments and to encourage the ASN to recover full

sound signals from the mixture (in contrast to merely what is required to minimize the consistency loss [36]), we also ensure that the sum of the predicted masks for separating the sound sources produce an estimated mask that is close to the ground truth ideal binary mask [27], using a co-separation loss similar to prior work [9, 36]:

$$\mathcal{L}_{\text{co-sep}} = \sum_{u=1,2} \Big\| \sum_{i=1}^{N_u+1} \hat{\mathbf{M}}_i^u - \mathbf{M}_{\text{ibm}}^{\mathbf{u}} \Big\|_{\mathbf{1}}, \qquad (4)$$

where $\mathbf{M}_{\text{ibm}}^{\mathbf{u}} = \mathbb{1}_{\mathbf{X}_{\text{m}}^{\mathbf{u}} > \mathbf{X}_{\text{m}}^{\neg \mathbf{u}}}$ denotes the ideal binary mask for the audio of video u within the mixture X_{m}.

Armed with the above three losses in Eq. 2, Eq. 3, and Eq. 4, the final training loss for our model is obtained as follows, with weights $\lambda_1, \lambda_2, \lambda_3 \geq 0$:

$$\mathcal{L} = \lambda_1 \mathcal{L}_{\text{cons}} + \lambda_2 \mathcal{L}_{\text{co-sep}} + \lambda_3 \mathcal{L}_{\text{ortho}}. \qquad (5)$$

4. Experiments

In order to validate the efficacy of our approach, we conduct experiments on two challenging datasets and compare its performance against competing and recent baselines.

4.1. Datasets

Audio Separation in the Wild (ASIW): Most prior approaches in visually-guided sound source separation report performances solely in the setting of separating the sounds of musical instruments [5, 58, 57]. Given musical instruments often have very characteristic sounds and most of the videos used for evaluating such algorithms often contain professional footages, they may not capture the generalizability of those methods to daily-life settings. While there have been recent efforts towards looking at more natural sounds [54], the categories of audio they consider are limited (∼10 classes). Moreover, most of the videos contain only a single sound source of interest, making the alignment straightforward. There are a few datasets that could be categorized as considering "in the wild" source separation, such as [7, 43], but they either only consider separating between on-screen and off-screen sounds [43], or provide only limited information about the nature of sounds featured [7], making the task of learning the audio-visual associations challenging.

To fill this gap in the evaluation benchmarks between "in the wild" settings and those with very limited annotations, we introduce a new dataset, called *Audio Separation in the Wild (ASIW)*. ASIW is adapted from the recently introduced large-scale AudioCaps dataset [20], which contains 49,838 training, 495 validation, and 975 test videos crawled from the AudioSet dataset [10], each of which is around 10 s long. In contrast to [7], these videos have been carefully annotated with human-written captions (English-speaking Amazon Mechanical Turkers – AMTs), emphasizing the auditory events in the video. We manually construct a dictionary of 306 frequently occurring *auditory words* from these captions. A few of our classes include: *splashing*, *flushing*, *eruptions*, or *giggling,* and these classes are almost always grounded to principal objects in the video generating the respective sound. The set of principal objects has 14 classes (baby, bell, birds, camera, clock, dogs, toilet, horse, man/woman, sheep/goat, telephone, trains, vehicle/car/truck, water) and an additional background class. The principal object list is drawn from the Visual Genome [23] classes. We retain only those videos which contain at least one of these 306 auditory words. Table 2 gives a distribution of the number of videos corresponding to each of these principal object categories. The resulting dataset features audio both arising out of standalone objects, such as *giggling of a baby*, as well as from inter-object interactions, such as *flushing of a toilet by a human*. The supplementary material lists all the 306 auditory words and the principal object associated to each word. After pre-processing this list, we use 147 validation and 322 test videos in our evaluation, while 10,540 videos are used for training.

MUSIC Dataset: Apart from our new ASIW dataset, we also report performance of our approach on the MUSIC dataset [58] which is often considered as the standard benchmark for visually-guided sound source seapartion. This dataset consists of 685 videos featuring humans performing musical solos and duets using 11 different instruments; 536 of these videos feature musical solos while the rest are duet videos. The instruments being played feature significant diversity in their type (for instance, guitar, erhu, violin are string instruments, flute, saxophone, trumpet are wind instruments, while xylophone is a percussion instrument). This makes the dataset a challenging one, despite its somewhat constrained nature. In order to conduct experiments, we split these videos into 10-second clips, following the standard protocol [9]. We ignore the first 10 seconds window of each of the untrimmed videos while constructing the dataset, since quite often the players do not really start playing their instruments right away. This results in 6,300/132/158 training, validation, and test videos respectively.

4.2. Baselines

We compare AVSGS against recently published approaches for visually-guided source separation, namely:

Sound of Pixel (SofP) [58]: one of the earliest deep learning based methods for this task.

Minus-Plus Net (MP Net) [54]: recursively removes the audio source that has the highest energy.

Co-Separation [9]: incorporates an object-level separation loss while training using the "mix-and-separate" framework. However, the visual conditioning is derived using only a single object in the scene.

Sound of Motion (SofM) [57]: integrates pixel-level motion trajectory and object/human appearances across

Table 1. SDR, SIR, and SAR [dB] results on the MUSIC and ASIW test sets. [Key: Best results in **bold** and second-best in blue.]

	MUSIC			ASIW		
	SDR↑	SIR↑	SAR↑	SDR↑	SIR↑	SAR↑
Sound of Pixel (SofP) [58]	6.1	10.9	10.6	6.2	8.1	10.6
Minus-Plus Net (MP Net) [54]	7.0	14.4	10.2	3.0	7.7	9.4
Sound of Motion (SofM) [57]	8.2	14.6	13.2	6.7	9.4	11.1
Co-Separation [9]	7.4	13.8	10.6	6.6	12.9	12.6
Music Gesture (MG) [5]	10.1	15.7	12.9	-	-	-
AVSGS (Ours)	**11.4**	**17.3**	**13.5**	**8.8**	**14.1**	**13.0**

Table 2. Number of videos for each of the principal object categories of ASIW dataset.

Baby	Bell	Birds	Camera	Clock	Dogs	Toilet	Horse	Man	Sheep	Telephone	Trains	Vehicle	Water
1616	151	2887	913	658	1407	838	385	6210	710	222	141	779	378

Table 3. SDR, SIR, and SAR [dB] results on the ASIW test set. [Key: Best results in **bold**.]

		ASIW		
Row		SDR↑	SIR↑	SAR↑
1	AVSGS (Full)	**8.8**	**14.1**	**13.0**
2	AVSGS - No orthogonality ($\lambda_3 = 0$)	7.4	13.3	11.6
3	AVSGS - No multi-lab. ($\lambda_1 = 0$)	6.4	11.2	11.7
4	AVSGS - No co-sep ($\lambda_2 = 0$)	1.1	1.3	**13.8**
5	AVSGS - N=3	8.4	13.5	12.2
6	AVSGS - No Skip Conn.	2.8	4.6	11.3
7	AVSGS - No GATConv	6.5	11.6	11.8
8	AVSGS - No EdgeConv	6.2	10.1	13.2
9	AVSGS - No GRU	6.5	12.3	10.6

video frames.

Music Gesture (MG) [5]: the most recent method on musical sound source separation, integrates appearance features from the scene along with human pose features. However this added requirement of human pose, limits its usability as a baseline to only the MUSIC dataset.

4.3. Evaluation Metrics

In order to quantify the performance of the different algorithms, we report the model performances in terms of the Signal-to-Distortion Ratio (SDR) [dB] [45, 37], where higher SDR indicates more faithful reproduction of the original signal. We also report two related measures, Signal-to-Interference Ratio (SIR) (which gives an indication of the amount of reduction of interference in the estimated signal) and Signal-to-Artifact Ratio (SAR) (which gives an indication of how much artifacts were introduced), as they were reported in prior audio-visual separation works [58, 9].

4.4. Implementation Details

We implement our model in PyTorch [35]. Following prior works [58, 9], we sub-sample the audio at 11 kHz, and compute the STFT of the audio using a Hann window of size 1022 and a hop length of 256. With input samples of length approximately 6 s, this yields a spectrogram of dimensions 512×256. The spectrogram is re-sampled according to a log-frequency scale to obtain a magnitude spectrogram of size $\Omega \times T$ with $\Omega = 256, T = 256$. The detector for the musical instruments was trained on the 15 musical object categories of the Open Images Dataset [24]. The FRCNN feature vectors F are 2048 dimensional. We detect up to two principal objects per video and use a set of up to 20 context nodes for a principal object. Additionally, a random crop from the image is considered as another principal object and is considered as belonging to the "background" class. The IoU threshold is set to $\delta = 0.1$, and 4 multi-head attention units are used in the graph attention network. The embedding dimension obtained from the graph pooling stage is set to 512. The GRU used is unidirectional with one hidden layer of 512 dimensions, and the visual representation vector thus has $d = 512$ dimensions. The weights on the loss terms are set to $\lambda_1 = 1, \lambda_2 = 0.05, \lambda_3 = 1$. The model is trained using the ADAM optimizer [21] with a weight decay of 1e-4, $\beta_1 = 0.9$, $\beta_2 = 0.999$. During training, the FRCNN model weights are frozen. An initial learning rate of 1e-4 is used and is decreased by a factor of 0.1 after every 15,000 iterations. These hyper-parameters and those of the baseline models are chosen based on the performances over the respective validation sets of the two datasets. At test time, the visual graph corresponding to a video is paired with a mixed audio (obtained from one or multiple videos) and fed as input to the network, which iteratively separates the audio sources from the input audio signal. We then apply an inverse STFT transform to map the separated spectrogram to the time domain, for evaluation.

4.5. Results

We present the model performances on the MUSIC and ASIW datasets in Table 1. From the results, we see that our proposed AVSGS model outperforms its closest competitor by large margins of around 1.3 dB on SDR and 1.6 dB on

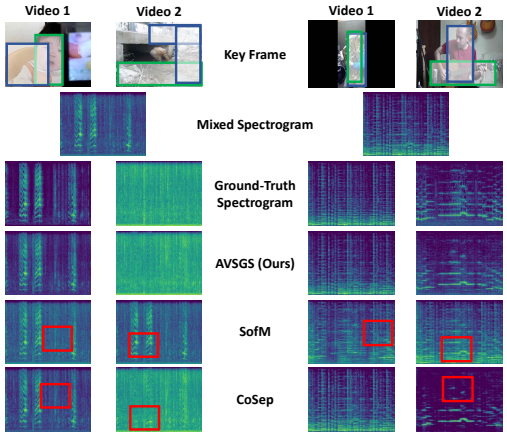

Figure 3. Qualitative separation results on ASIW (left) and MUSIC (right). Bounding boxes on frames show regions attended by AVSGS (green: principal object, blue: context nodes). Red boxes indicate regions of high differences between ground truth and predicted spectrograms.

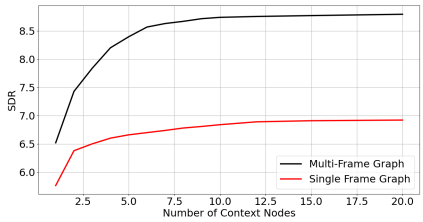

Figure 4. Performance plots of model variants with graphs constructed from multiple frames (black) and single frame (red).

SIR on the MUSIC dataset, and around 1.2 dB on SDR and 2 dB on SIR on the ASIW dataset, which reflects substantial gains, given that these metrics are in log scale. We note that our much higher SIR did not come at the expense of a lower SAR, as is often the case, with the SAR in fact surpassing MG's [5] by 0.6 dB. SofM is the non-MUSIC-specific baseline that comes closest to our model's performance, perhaps because it effectively combines motion and appearance, while the visual information of most other approaches is mainly appearance-based and holistic. In AVSGS, while motion is not explicitly encoded in the visual representation, the spatio-temporal nature of our graph G implicitly embeds this key element. MG's competitive performance on MUSIC gives credence to the hypothesis that for good audio separation, besides the embedding of the principal object, appropriate visual context is necessary. In their setting, this is limited to human pose. However, when the set of context nodes is expanded, richer interactions can be captured, which proves beneficial to the model performance, as is seen to be the case for our model. Importantly, our approach for incorporating this context information generalizes well across both datasets, unlike MG.

Ablations and Additional Results: In Table 3, we report performances of several ablated variants of our model on the ASIW dataset. The second, third, and fourth rows showcase the model performance obtained by turning off one loss term at a time. The results overwhelmingly point to the importance of the co-separation loss (row 3), without which the performance of the model drops significantly. We also tweaked the number of objects per video to 3 and observed very little change in model performance, as seen in row 5 of Table 3. Row 6 underscores the importance of having skip connection in the ASN network. In rows 7 and 8, we present results of ablating the different components of the scene graph. The results indicate that GATConv and EdgeConv are roughly equally salient. Finally, as seen in row 9, our model underperforms without the GRU.

Additionally, in Figure 4 we plot the performance of our AVSGS model with varying number of context nodes at test time, shown in black. This experiment is then repeated for a model where we build the graph from only a single frame. The performance plot of this variant is shown in red. The plots show a monotonically increasing trend underscoring the importance of constructing spatio-temporal graphs which capture the richness of the scene contexts.

Qualitative Results: In Figure 3, we present example separation results on samples from the ASIW and MUSIC test sets, while contrasting the performance of our algorithm against two competitive baselines, Co-Separation and SofM. As is evident from the separated spectrograms, AVSGS is more effective in separating the sources than these baselines. Additionally, the figure also shows the regions attended to by AVSGS in order to induce the audio source separation. We find that AVSGS correctly chooses useful context regions/objects to attend to for both datasets. For more details, qualitative results, and user study, please see the supplementary materials.

5. Conclusions

We presented AVSGS, a novel algorithm that leverages the power of scene graphs to induce audio source separation. Our model leverages self-supervised techniques for training and does not require additional labelled training data. We show that the added context information that the scene graphs introduce allows us to obtain state-of-the-art results on the existing MUSIC dataset and a challenging new dataset of "in the wild" videos called ASIW. In future work, we intend to explicitly incorporate motion into the scene graph to further boost model performance.

Acknowledgements. The support of the Office of Naval Research under grant N00014- 20-1-2444, and USDA National Institute of Food and Agriculture under grant 2020-67021-32799/1024178 are gratefully acknowledged.

References

[1] Triantafyllos Afouras, Andrew Owens, Joon Son Chung, and Andrew Zisserman. Self-supervised learning of audio-visual objects from video. In *Proc. ECCV*, 2020. 2

[2] Relja Arandjelovic and Andrew Zisserman. Objects that sound. In *Proc. ECCV*, pages 435–451, 2018. 3

[3] Pierre Comon and Christian Jutten. *Handbook of Blind Source Separation: Independent component analysis and applications*. Academic press, 2010. 2

[4] 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. *ACM Trans. Graph. (TOG)*, 37(4):1–11, 2018. 2

[5] Chuang Gan, Deng Huang, Hang Zhao, Joshua B Tenenbaum, and Antonio Torralba. Music gesture for visual sound separation. In *Proc. CVPR*, pages 10478–10487, 2020. 1, 2, 5, 6, 7, 8

[6] Sharon Gannot, Emmanuel Vincent, Shmulik Markovich-Golan, and Alexey Ozerov. A consolidated perspective on multimicrophone speech enhancement and source separation. *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, 25(4):692–730, 2017. 2

[7] Ruohan Gao, Rogerio Feris, and Kristen Grauman. Learning to separate object sounds by watching unlabeled video. In *Proc. ECCV*, pages 35–53, Sept. 2018. 6

[8] Ruohan Gao and Kristen Grauman. 2.5 d visual sound. In *Proc. CVPR*, pages 324–333, 2019. 3

[9] Ruohan Gao and Kristen Grauman. Co-separating sounds of visual objects. In *Proc. ICCV*, pages 3879–3888, 2019. 1, 2, 3, 5, 6, 7

[10] Jort F Gemmeke, Daniel PW Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R Channing Moore, Manoj Plakal, and Marvin Ritter. Audio set: An ontology and human-labeled dataset for audio events. In *Proc. ICASSP*, pages 776–780, Mar. 2017. 6

[11] Shijie Geng, Peng Gao, Chiori Hori, Jonathan Le Roux, and Anoop Cherian. Spatio-temporal scene graphs for video dialog. In *Proc. AAAI*, 2021. 3

[12] John Hershey and Javier Movellan. Audio vision: Using audio-visual synchrony to locate sounds. In *Proc. NIPS*, pages 813–819, Dec. 1999. 3

[13] John R. Hershey, Zhuo Chen, Jonathan Le Roux, and Shinji Watanabe. Deep clustering: Discriminative embeddings for segmentation and separation. In *Proc. ICASSP*, Mar. 2016. 2, 5

[14] Po-Sen Huang, Minje Kim, Mark Hasegawa-Johnson, and Paris Smaragdis. Deep learning for monaural speech separation. In *Proc. ICASSP*, pages 1562–1566, May 2014. 2, 5

[15] Yusuf Isik, Jonathan Le Roux, Zhuo Chen, Shinji Watanabe, and John R. Hershey. Single-channel multi-speaker separation using deep clustering. In *Proc. Interspeech*, pages 545–549, Sept. 2016. 5

[16] Andreas Jansson, Eric Humphrey, Nicola Montecchio, Rachel Bittner, Aparna Kumar, and Tillman Weyde. Singing voice separation with deep U-net convolutional networks. In *Proc. ISMIR*, Oct. 2017. 5

[17] Jingwei Ji, Ranjay Krishna, Li Fei-Fei, and Juan Carlos Niebles. Action genome: Actions as compositions of spatio-temporal scene graphs. In *Proc. CVPR*, pages 10236–10247, 2020. 3

[18] Justin Johnson, Ranjay Krishna, Michael Stark, Li-Jia Li, David Shamma, Michael Bernstein, and Li Fei-Fei. Image retrieval using scene graphs. In *Proc. CVPR*, pages 3668–3678, 2015. 2, 3

[19] Einat Kidron, Yoav Y Schechner, and Michael Elad. Pixels that sound. In *Proc. CVPR*, volume 1, pages 88–95. IEEE, 2005. 3

[20] Chris Dongjoo Kim, Byeongchang Kim, Hyunmin Lee, and Gunhee Kim. Audiocaps: Generating captions for audios in the wild. In *Proc. NAACL HLT*, pages 119–132, 2019. 2, 6

[21] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proc. ICLR*, 2014. 7

[22] Ivan Krasin, Tom Duerig, Neil Alldrin, Vittorio Ferrari, Sami Abu-El-Haija, Alina Kuznetsova, Hassan Rom, Jasper Uijlings, Stefan Popov, Andreas Veit, et al. Openimages: A public dataset for large-scale multi-label and multi-class image classification. *Dataset available from https://github.com/openimages*, 2(3):18, 2017. 3

[23] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual Genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123(1):32–73, 2017. 3, 6

[24] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, et al. The Open Images dataset v4. *Int. J. Comput. Vis.*, pages 1–26, 2020. 7

[25] Junhyun Lee, Inyeop Lee, and Jaewoo Kang. Self-attention graph pooling. In *Proc. ICML*, pages 3734–3743, June 2019. 4

[26] Xiaowei Li, Changchang Wu, Christopher Zach, Svetlana Lazebnik, and Jan-Michael Frahm. Modeling and recognition of landmark image collections using iconic scene graphs. In *Proc. ECCV*, pages 427–440. Springer, 2008. 3

[27] Yipeng Li and DeLiang Wang. On the optimality of ideal binary time-frequency masks. *Speech Communication*, 51(3):230–239, 2009. 6

[28] Yuzhou Liu and DeLiang Wang. Divide and conquer: A deep CASA approach to talker-independent monaural speaker separation. *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, 27(12):2092–2102, 2019. 5

[29] Philipos C Loizou. *Speech enhancement: theory and practice*. CRC press, 2013. 2

[30] Gabriel Meseguer-Brocal and Geoffroy Peeters. Conditioned-U-Net: Introducing a control mechanism in the U-Net for multiple source separations. *arXiv preprint arXiv:1907.01277*, 2019. 5

[31] Daniel Michelsanti, Zheng-Hua Tan, Shi-Xiong Zhang, Yong Xu, Meng Yu, Dong Yu, and Jesper Jensen. An overview of deep-learning-based audio-visual speech enhancement and separation. *arXiv preprint arXiv:2008.09586*, 2020. 2

[32] Pedro Morgado, Nuno Vasconcelos, Timothy Langlois, and Oliver Wang. Self-supervised generation of spatial audio for 360° video. In *Proc. NeurIPS*, pages 360–370, 2018. 3

[33] Andrew Owens and Alexei A Efros. Audio-visual scene analysis with self-supervised multisensory features. In *Proc. ECCV*, pages 631–648, 2018. 2

[34] Andrew Owens, Phillip Isola, Josh McDermott, Antonio Torralba, Edward H Adelson, and William T Freeman. Visually indicated sounds. In *Proc. CVPR*, pages 2405–2413, 2016. 3

[35] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An imperative style, high-performance deep learning library. In *Proc. NeurIPS*, pages 8024–8035, Dec. 2019. 7

[36] Fatemeh Pishdadian, Gordon Wichern, and Jonathan Le Roux. Finding strength in weakness: Learning to separate sounds with weak supervision. *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, 28:2386–2399, 2020. 2, 6

[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 *Proc. ISMIR*, 2014. 7

[38] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(6):1137–1149, 2016. 3

[39] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Proc. MICCAI*, pages 234–241. Springer, 2015. 2, 5

[40] Arda Senocak, Tae-Hyun Oh, Junsik Kim, Ming-Hsuan Yang, and In So Kweon. Learning to localize sound source in visual scenes. In *Proc. CVPR*, pages 4358–4366, 2018. 3

[41] Olga Slizovskaia, Leo Kim, Gloria Haro, and Emilia Gomez. End-to-end sound source separation conditioned on instrument labels. In *Proc. ICASSP*, pages 306–310, May 2019. 5

[42] Paris Smaragdis, Cedric Fevotte, Gautham J Mysore, Nasser Mohammadiha, and Matthew Hoffman. Static and dynamic source separation using nonnegative factorizations: A unified view. *IEEE Signal Process. Mag.*, 31(3):66–75, 2014. 2

[43] Efthymios Tzinis, Scott Wisdom, Aren Jansen, Shawn Hershey, Tal Remez, Daniel PW Ellis, and John R Hershey. Into the wild with AudioScope: Unsupervised audio-visual separation of on-screen sounds. In *Proc. ICLR*, 2021. 2, 6

[44] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. In *Proc. ICLR*, Apr. 2018. 4

[45] Emmanuel Vincent, Rémi Gribonval, and Cédric Févotte. Performance measurement in blind audio source separation. *IEEE Trans. Audio, Speech, Lang. Process.*, 14(4):1462–1469, July 2006. 7

[46] Emmanuel Vincent, Tuomas Virtanen, and Sharon Gannot. *Audio source separation and speech enhancement*. John Wiley & Sons, 2018. 2

[47] DeLiang Wang and Guy J Brown. *Computational auditory scene analysis: Principles, algorithms, and applications*. Wiley-IEEE press, 2006. 2

[48] DeLiang Wang and Jitong Chen. Supervised speech separation based on deep learning: An overview. *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, 26(10):1702–1726, 2018. 2, 5

[49] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E Sarma, Michael M Bronstein, and Justin M Solomon. Dynamic graph CNN for learning on point clouds. *ACM Trans. Graph. (TOG)*, 38(5):1–12, 2019. 4

[50] Felix Weninger, Jonathan Le Roux, John R. Hershey, and Björn Schuller. Discriminatively trained recurrent neural networks for single-channel speech separation. In *Proc. GlobalSIP*, Dec. 2014. 2, 5

[51] Felix Weninger, Jonathan Le Roux, John R Hershey, and Shinji Watanabe. Discriminative NMF and its application to single-channel source separation. In *Proc. Interspeech*, Sept. 2014. 2

[52] Scott Wisdom, Efthymios Tzinis, Hakan Erdogan, Ron J Weiss, Kevin Wilson, and John R Hershey. Unsupervised sound separation using mixtures of mixtures. In *Proc. NeurIPS*, Dec. 2020. 2

[53] Shasha Xia, Hao Li, and Xueliang Zhang. Using optimal ratio mask as training target for supervised speech separation. In *Proc. APSIPA ASC*, pages 163–166, 2017. 2

[54] Xudong Xu, Bo Dai, and Dahua Lin. Recursive visual sound separation using minus-plus net. In *Proc. ICCV*, pages 882–891, Oct. 2019. 1, 6, 7

[55] Yong Xu, Jun Du, Li-Rong Dai, and Chin-Hui Lee. An experimental study on speech enhancement based on deep neural networks. *IEEE Signal Process. Lett.*, 21(1):65–68, 2013. 2, 5

[56] Dong Yu, Morten Kolbæk, Zheng-Hua Tan, and Jesper Jensen. Permutation invariant training of deep models for speaker-independent multi-talker speech separation. In *Proc. ICASSP*, pages 241–245, Mar. 2017. 2, 5

[57] Hang Zhao, Chuang Gan, Wei-Chiu Ma, and Antonio Torralba. The sound of motions. In *Proc. ICCV*, pages 1735–1744, 2019. 1, 2, 3, 5, 6, 7

[58] Hang Zhao, Chuang Gan, Andrew Rouditchenko, Carl Vondrick, Josh McDermott, and Antonio Torralba. The sound of pixels. In *Proc. ECCV*, pages 570–586, 2018. 1, 2, 3, 5, 6, 7

[59] Yipin Zhou, Zhaowen Wang, Chen Fang, Trung Bui, and Tamara L Berg. Visual to sound: Generating natural sound for videos in the wild. In *Proc. CVPR*, pages 3550–3558, 2018. 3