The Dissertation Committee for Ruohan Gao
certifies that this is the approved version of the following dissertation:

Look and Listen: From Semantic to Spatial Audio-Visual Perception

Committee:

Kristen Grauman, Supervisor

Andrew Zisserman

Raymond Mooney

Qixing Huang

Look and Listen:
From Semantic to Spatial Audio-Visual Perception

by

Ruohan Gao

DISSERTATION

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT AUSTIN

May 2021

Dedicated to my parents, Gao and Yang.

Acknowledgments

This thesis would not have been possible without the support and guidance from many incredible mentors, colleagues, friends, and family. I have a great many people to thank.

First and foremost, I would like to thank my advisor Kristen Grauman. I feel genuinely lucky to have her as my advisor. She has all the qualities that one could look for in a good advisor: passionate for research, knowledgeable in the field, dedicated to work, attentive to details, and always looking out for my best interests. Over the years, she guided me through all the hills and valleys in both research and life with her consistent and effective set of principles. Kristen has been the best advisor that I can ever dream of. She will continue to be the best role model for me in the rest of my life.

During my PhD study, I am grateful to have been guided by many great minds. Andrew Zisserman, Raymond Mooney, and Qixing Huang have provided very constructive feedback to strengthen this dissertation as my thesis committee members. Rogerio Feris was my first external collaborator, and working with him on my first audio-visual learning project is really fun and rewarding. In Summer 2019, I interned with Lorenzo Torresani at FAIR Boston. Many of his creative research ideas and insightful conversations made my internship a very enjoyable and fruitful experience. I also thank Tae-Hyun Oh for being such a wonderful collaborator in this internship.

I have also been very fortunate to be surrounded by many supportive labmates. They have made my graduate life much more enjoyable. I thank Chao-Yeh Chen, Suyog Dutt Jain, Dinesh Jayaraman, Aron Yu, Bo Xiong, Yu-Chuan Su, Wei-Lin Hsiao, Antonino Furnari, Ziad Al-Halah, Danna Gurari, Tushar Nagarajan, Santhosh Kumar Ramakrishnan, Changan Chen, Priyanka Mandikal, Rishabh Garg, and Sagnik Majumder. I will miss our game nights, discussions, and many deadlines we fight together.

I am also thankful to Wing Cheong Lau, Pili Hu, and Huanle Xu for their guidance in my undergraduate research. I am especially grateful for Pili, who taught me a lot in my first research project and encouraged me to pursue a PhD.

During my postdoc search, I was very fortunate to have received guidance from many extraordinary researchers. I want to thank Alexei Efros, Jim Glass, Leonidas Guibas, Abhinav Gupta, Fei-Fei Li, Jitendra Malik, Silvio Savarese, Jiajun Wu, and Dan Yamins for their encouragement. I also want to thank David Harwath, Andrew Owens, Shubham Tulsiani, and Yuke Zhu for their help and career advice in this process.

I have also been guided and helped by many other professors and staff members at UTCS, apart from the ones mentioned above. I want to thank Greg Durrett, Philipp Krähenbühl, Qiang Liu, Shyamal Mitra, Peter Stone, and William Young for their guidance at different stages of my PhD. I also want to thank Lydia Griffith and Katie Traughber for all the help they have provided as our graduate program coordinator.

In my last two years at UT Austin, I have been funded by the Google PhD Fellowship, Adobe Research Fellowship, and Facebook AI Research. I appreciate their support for my thesis research. I also thank the graduate school and the faculty review committees for recognizing this dissertation with the 2021 Michael H. Granof University's Top Dissertation Award.

Finally, I am deeply grateful to my parents. I couldn't have accomplished this without your unconditional love and support. I am also so blessed to have Xinying during this journey, filling love and inspirations into my life every day.

Look and Listen:
From Semantic to Spatial Audio-Visual Perception

Publication No. ____________________

Ruohan Gao, Ph.D.
The University of Texas at Austin, 2021

Supervisor: Kristen Grauman

Understanding scenes and events is inherently a multi-modal experience. We perceive the world by both looking and listening (and touching, smelling, and tasting). In particular, the sounds made by objects, whether actively generated or incidentally emitted, offer valuable signals about their physical properties and spatial locations—the cymbals crash on stage, the bird tweets up in the tree, the truck revs down the block, the silverware clinks in the drawer.

However, while recognition has made significant progress by "looking"—detecting objects, actions, or people based on their appearance—it often does not listen. In this thesis, I show that audio that accompanies visual scenes and events can be used as a rich source of training signal for learning (audio-)visual models. Particularly, I have developed computational models that leverage

both the *semantic* and *spatial* signals in audio to understand people, places, and things from continuous multi-modal observations. Below, I summarize my key contributions along these two themes:

Audio as a semantic signal: First, I develop methods that learn how different objects sound by both looking at and listening to unlabeled video containing multiple sounding objects. I propose an unsupervised approach to separate mixed audio into its component sound sources by disentangling the audio frequency bases for detected visual objects. Next, I further propose a new approach that trains audio-visual source separation models on pairs of training videos. This co-separation framework permits both end-to-end training and learning object-level sounds from unlabeled videos of multiple sound sources. As an extension of the co-separation approach, then I study the classic cocktail party problem to separate voices from the speech mixture by leveraging the consistency between the speaker's facial appearance and their voice. The two modalities, vision and audition, are mutually beneficial. While visual objects are indicative of the sounds they make to enhance audio source separation, audio can also be informative of the visual events in videos. Finally, I propose a framework that uses audio as a semantic signal to help visual events classification. I design a preview mechanism to eliminate both short-term and long-term visual redundancies using audio for efficient action recognition in untrimmed video.

Audio as a spatial signal: Both audio and visual data also convey signifi-

cant spatial information. The two senses naturally work in concert to interpret spatial signals. Particularly, the human auditory system uses two ears to extract individual sound sources from a complex mixture. Leveraging the spatial signal in videos, I devise an approach to lift a flat monaural audio signal to binaural audio by injecting the spatial cues embedded in the accompanying visual frames. When listening to the predicted binaural audio—the 2.5D visual sound—listeners can then feel the locations of the sound sources as they are displayed in the video. Beyond learning from passively captured video, I next explore the spatial signal in audio by deploying an agent to actively interact with the environment using audio. I propose a novel representation learning framework that learns useful visual features via echolocation by capturing echo responses in photo-realistic 3D indoor scene environments. Experimental results demonstrate that the image features learned from echoes are comparable or even outperform heavily supervised pre-training methods for multiple fundamental spatial tasks—monocular depth prediction, surface normal estimation, and visual navigation.

Our results serve as an exciting prompt for future work leveraging both the visual and audio modalities. Motivated by how we humans perceive and act in the world by making use of all our senses, the long-term goal of my research is to build systems that can perceive as well as we do by combining all the multisensory inputs. In the last chapter of my thesis, I outline the potential future research directions that I want to pursue beyond my Ph.D. dissertation.

Table of Contents

List of Tables

List of Figures

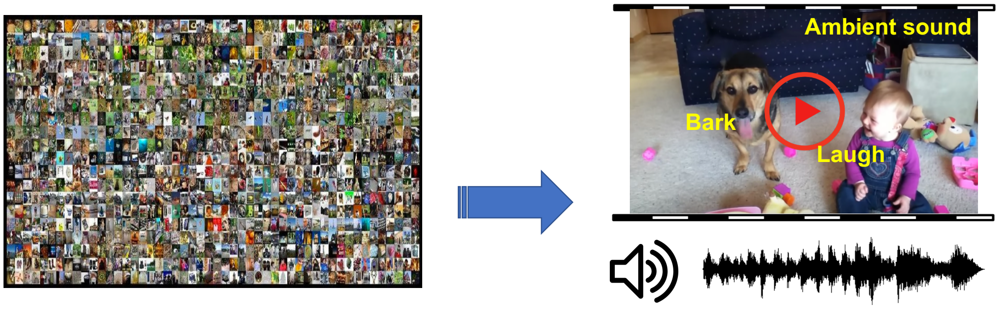
Chapter 1

Overview and Introduction

Multi-modal perception is essential to capture the richness of real-world sensory data for objects, scenes, and events. We perceive the world by making use of all our senses—especially looking and listening. Objects not only have their characteristic visual appearance, but also generate unique sounds due to their physical properties and interactions with other objects and the environment. For example, perception of a coffee shop scene may include seeing cups, saucers, people, and tables, but also hearing the dishes clatter, the espresso machine grind, and the barista shouting an order. Human developmental learning is also inherently multi-modal, with young children quickly amassing a repertoire of visual objects and their sounds: dogs bark, cats mew, phones ring.

However, the major successes in computer vision today mostly come from "looking" – detecting objects, actions, or people based on their visual appearance without paying attention to their accompanying sound. Particularly, objects in static snapshots are always analyzed as if they were silent entities in silent environments.

Cognitive science actually tells us that perception develops by making



Status quo: Learning from "silent" labeled snapshots

My research: Listening to learn about the visual world

Figure 1.1: While the status quo of most current computer vision systems learn from massive datasets of labeled images that are "silent", my thesis research aims to both listen and look to learn about the visual world.

use of all our senses without intensive supervision [199]. Towards this goal, my thesis research aims to break free from the status quo of learning from massive datasets of "silent" labeled snapshots, and make a system both look and listen to understand the visual world (Fig. 1.1). Audio that accompanies visual scenes and events can be used as a rich source of training signals for learning (audio-)visual models.

The importance of audio-visual learning is not only motivated by how we humans learn, but also because it can enable many useful applications in various fields: In the multimedia domain, audio-visual learning can be used to enhance audio events indexing, audio denoising for closed captioning, and instrument equalization. In healthcare, devices with both audio and visual signals can be used to assist visually or aurally impaired people. In AR/VR, avatars with synthesized visual and audio tracks can provide users with an immersive virtual experience.

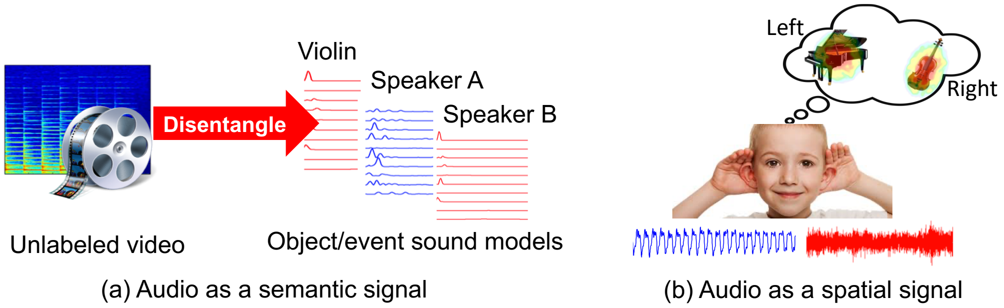

Violin
Speaker A
Speaker B
Disentangle
Unlabeled video
Object/event sound models
(a) Audio as a semantic signal
Left
Right
(b) Audio as a spatial signal

Figure 1.2: Audio itself is a supervision signal for *semantic* and *spatial* understanding of the world.

In spite of the great potential, joint learning with both audio and visual streams presents several challenges: 1) In a realistic video, object sounds are observed not as separate entities, but as a *single audio channel* that mixes all their frequencies together; 2) Not all objects can make sounds, and potential sound sources do not always make sounds; 3) Videos are often recorded in diverse scenes mixed with other background or ambient noises; 4) Objects are *spatially* located in the 3D world, and the sounds they make are also influenced by the geometry of the object/scene configurations.

The overarching goal of my thesis research is to recover audio-visual models from videos and embodied agents: *How can algorithms learn* what *and* where *the sound-making objects are when multiple sound sources are present? How can these audio-visual models benefit classic audio and vision tasks?* To address these questions, my research leverages both the *semantic* and *spatial* signals in audio to understand people, places, and things from continuous multimodal observations (Fig. 1.2).

As the first step, I propose to learn audio-visual object models from unlabeled video, then exploit the visual context to perform audio source separation in novel videos. Our approach relies on a deep multi-instance multi-label learning framework and non-negative matrix factorization (NMF) to disentangle the audio frequency bases that map to individual visual objects, even without observing (hearing) those objects in isolation. By leveraging both the visual and audio modalities in hundreds of thousands of unlabeled videos, we can learn object-level sound models that generalize to separate sounds for novel audio-visual instances. Chapter 3 discusses the approach for this work in more detail and presents our results. This work was first published at ECCV 2018 [75].

My original two-stage method heavily relies on NMF to perform separation, which limits its performance and practicability. Other prior or concurrent methods [55, 163, 258] for visually-guided audio source separation instead train with artificially mixed video clips, but this puts unwieldy restrictions on training data collection and may even prevent learning the properties of "true" mixed sounds. In Chapter 4, I introduce a new co-separation paradigm that permits both end-to-end training and learning object-level sounds from unlabeled multi-source videos.[1] Our novel training objective requires that the deep neural network's separated audio for similar-looking objects be consistently identifiable, while simultaneously reproducing accurate video-level audio

[1] Throughout, we use "multi-source video" as shorthand for video containing multiple sounds in its single-channel audio.

tracks for each source training pair. Our co-separation training paradigm allows training with “in the wild” sound mixes, and enhances the supervision beyond the commonly adopted “mix-and-separate” training strategy. Our method is able to learn well from multi-source videos, and it can successfully separate an object sound in a test video even if the object has never been observed individually during training. Chapter 4 discusses the approach for this work in more detail and presents experimental results. The work was first published at ICCV 2019 [77].

The co-separation approach introduced above makes use of the appearance cues of musical instruments to separate their respective sounds. However, in cases where the sound sources come from the same category as is the case with multiple human speakers, it can struggle to perform instance-level sound source separation. In Chapter 5, I extend our co-separation framework to the task of audio-visual speech separation. Given a video, the goal of audio-visual speech separation is to extract the speech associated with a face in spite of simultaneous background sounds and/or other human speakers. Whereas existing methods focus on learning the alignment between the speaker’s lip movements and the sounds they generate, we propose to leverage the speaker’s face appearance as an additional prior to isolate the corresponding vocal qualities they are likely to produce. Our approach jointly learns audio-visual speech separation and cross-modal speaker embeddings from unlabeled video. It yields state-of-the-art results on five benchmark datasets for audio-visual speech separation and enhancement, and generalizes well to challenging real-world videos of diverse scenarios. Chapter 5 discusses the approach for this work in more detail and presents experimental results. The work will be published at CVPR 2021 [78].

In Chapters 3, 4, and 5, I exploit the natural correspondence between visual objects and the sounds they make to learn audio-visual object sound models for the task of audio source separation. My three approaches obtain state-of-the-art performance on audio-visual source separation for human speakers, musical instruments, and other natural sounds. While visual objects are indicative of the sounds they make to enhance audio source separation, audio can also be informative to identify visual events in videos. Hence, next in Chapter 6, I explore how to leverage audio to help action recognition in untrimmed videos.

In the face of the video data deluge, today's expensive clip-level action classifiers are increasingly impractical. I propose a framework for efficient action recognition in untrimmed video that uses audio as a preview mechanism to eliminate both short-term and long-term visual redundancies. First, we devise an ImgAud2Vid framework that hallucinates clip-level features by distilling from lighter modalities—a single frame and its accompanying audio—reducing short-term temporal redundancy for efficient clip-level recognition. Second, building on ImgAud2Vid, we further propose ImgAud-Skimming, an attention-based long short-term memory network that iteratively selects useful moments in untrimmed videos, reducing long-term temporal redundancy for efficient video-level recognition. Extensive experiments

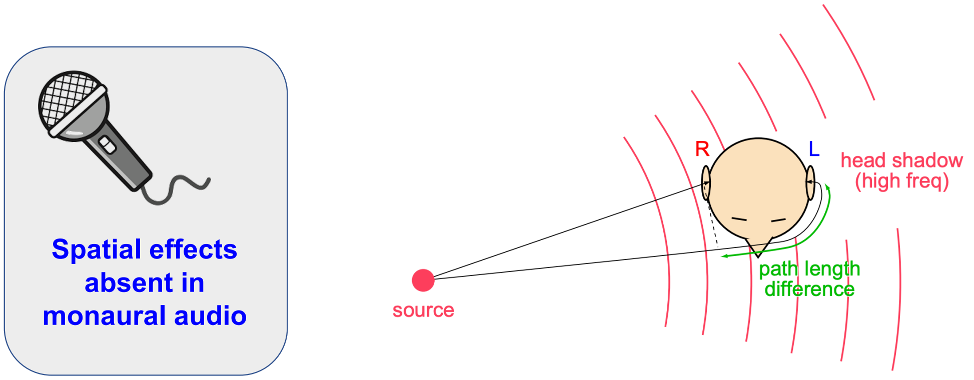


Figure 1.3: We use two ears to extract sound sources from a mixture. Two signals enter the left and right ears separately, causing an spatial effect [150].

on four action recognition datasets demonstrate that our method achieves the state-of-the-art in terms of both recognition accuracy and speed. Chapter 6 discusses the approach for this work in more detail and presents experimental results. This work was first published at CVPR 2020 [80].

So far, my methods learn from videos with *monaural audio* for audio-visual source separation and action recognition. They recognize objects based on their appearance without explicitly paying attention to their *spatial locations*. However, both audio and visual data also convey significant spatial information. We see where objects are and how the room is laid out, and we also hear them: sound-emitting objects indicate their location, and sound reverberations reveal the room's main surfaces, materials, and dimensions. Similarly, as in the famous cocktail party scenario, while having a conversation at a noisy party, one can hear another voice calling out and turn to face it. The two senses naturally work in concert to interpret spatial signals.

As shown in Fig. 1.3, the human auditory system uses *two* ears to extract individual sound sources from a complex mixture. The duplex theory proposed by Lord Rayleigh says that sound source locations are mainly determined by time differences between the sounds reaching each ear (Interaural Time Difference, ITD) and differences in sound level entering the ears (Interaural Level Difference, ILD) [179].

Starting from Chapter 7, I turn to use audio as a spatial signal for audio-visual learning. Firstly, I study the problem of visually-guided audio spatialization. I propose to convert common monaural audio into binaural audio by leveraging video. We devise a deep convolutional neural network that learns to decode the monaural soundtrack into its binaural counterpart by leveraging visual information presented in unlabeled videos. We call the resulting output—2.5D visual sound—the visual stream helps "lift" the flat single channel audio into spatialized sound. In addition to sound generation, we also demonstrate that our MONO2BINAURAL conversion process can benefit audio-visual source separation, a key challenge in audio-visual analysis. Chapter 7 discusses the approach for this work in more detail and presents results. This work was first published at CVPR 2019 [76].

My thesis research is motivated by how we humans perceive multisensory data. However, humans learn not only by watching passively captured videos of audio-visual streams, but also by actively interacting with the environment to learn about the world (Fig. 1.4). In the final piece of my thesis research presented in Chapter 8, I introduce our VISUALECHOES approach,

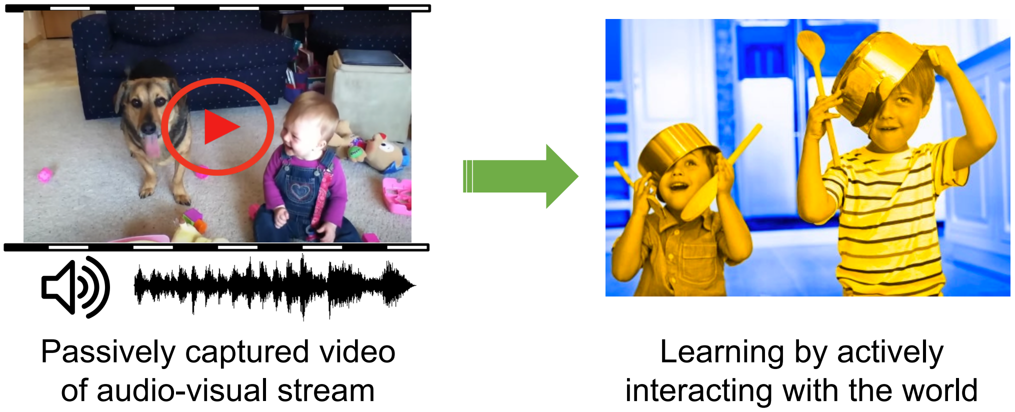


Figure 1.4: In addition to watching the passively captured videos of audio-visual streams, we humans learn by actively interacting with the environment to learn about the world.

which learns spatial image representations by using audio to actively interact with the physical world.

Several animal species (e.g., bats, dolphins, and whales) and even visually impaired humans have the remarkable ability to perform echolocation: a biological sonar used to perceive spatial layout and locate objects in the world. In Chapter 8, I explore the spatial cues contained in echoes and how they can benefit vision tasks that require spatial reasoning. First I capture echo responses in photo-realistic 3D indoor scene environments. Then I propose a novel interaction-based representation learning framework that learns useful *visual* features via echolocation. We show that the learned image features are useful for multiple downstream vision tasks requiring spatial reasoning—monocular depth estimation, surface normal estimation, and visual navigation—with results comparable or even better than heavily supervised pre-training. Our work opens a new path for representation learning for embodied agents, where supervision comes from interacting with the physical world. Chapter 8 discusses the approach for this work in more detail and presents experimental results. This work was first published at ECCV 2020 [74].

To summarize, leveraging audio as both a *semantic* signal and a *spatial* signal, I progressively approach my ultimate goal of comprehensive audio-visual scene understanding by studying the following four problems in this dissertation:

- Simultaneously looking at and listening to unlabeled video containing multiple sound sources to learn audio-visual source separation models [75, 77, 78] (Chapter 3, Chapter 4, and Chapter 5).

- Leveraging audio as a preview mechanism to enable efficient action recognition in untrimmed videos [80] (Chapter 6).

- Inferring binaural audio by exploiting the visual information in unlabeled video to "lift" the flat single-channel audio into spatialized sound [76] (Chapter 7).

- Learning spatial image representations via echolocation, where supervision comes from acoustically interacting with the physical world [74] (Chapter 8).

Next I review the important related work to the research presented in this dissertation. Then, in Chapters 3 through 8, I discuss the proposed methods outlined above. Finally, the last chapter summarizes my thesis research and highlights some potential future directions leading to my long-term research goal beyond this dissertation.

Chapter 2

Related Work

In this chapter, I review prior work relevant to the six components of my thesis research which will be discussed in Chapters 3 through 8. The material presented here serves both to set the stage to understand our proposed methods against their respective contexts and to overview the prior literature surrounding the topics of this dissertation.

2.1 Modes of Supervision

2.1.1 Self-Supervised Learning

Self-supervised learning leverages structured information within the data itself to generate "free" labels [40,91]. To this end, many "pretext" tasks have been explored—for example, predicting the rotation applied to an input image [6,85], discriminating image instances [60], colorizing images [136,255], solving a jigsaw puzzle from image patches [161], predicting unseen views of 3D objects [117], or multi-task learning using synthetic imagery [181]. Temporal information in videos also permits self-supervised tasks, for example, by predicting whether a frame sequence is in the correct order [61,153] or ensuring visual coherence of tracked objects [79,116,233]. Audio-visual data also offers a wealth of such tasks for self-supervised learning. Recent work explores self-supervision for visual [13,14,165] and audio [15] feature learning, cross-modal representations [16], and audio-visual alignment [98,130,163].

Whereas these methods aim to learn features generically useful for recognition, the objective of our VisualEchoes approach presented in Chapter 8 is to learn features generically useful for spatial estimation tasks. Accordingly, our echolocation objective is well-aligned with our target family of spatial tasks (depth, surfaces, navigation), consistent with findings that task similarity is important for positive transfer [254]. Furthermore, different from prior self-supervised learning methods, rather than learn from massive repositories of human-taken photos, our proposed approach learns from interactions with the scene via echolocation.

Our VisualVoice approach in Chapter 5 and the mono2binaural formulation discussed in Chapter 7 are also self-supervised, but unlike any of the above: VisualVoice learns cross-modal face-voice embeddings in a self-supervised way to enhance audio-visual speech separation; mono2binaural uses visual frames to supervise audio spatialization, while also learning better sound representations for audio-visual source separation.

2.1.2 Weakly-Supervised Visual Learning

Apart from self-supervised learning, my thesis research is also related to weakly-supervised learning. The audio-visual source separation problem described in Chapter 3 and 4 can be seen as a weakly-supervised visual learning problem, where the "supervision" in our case consists of automatically classified or detected visual objects. Given unlabeled video, our approach learns to disentangle which sounds within a mixed audio signal go with which recognizable objects. In Chapter 3, the weak supervision comes from a pre-trained image classifier; while in Chapter 4, our method uses localized object regions from a pre-trained object detector as weak supervision for audio-visual source separation.

Our approach of weakly-supervised audio-visual learning is entirely novel, but at a high level it follows the spirit of prior work leveraging weak annotations, including early "words and pictures" work [17, 49], internet vision methods [20, 218], training weakly-supervised object (activity) detectors [9,21,38,42,227], image captioning methods [45,120], or grounding acoustic units of spoken language to image regions [97, 98]. In contrast to any of these methods, our idea is to learn *sound* associations for objects from unlabeled video, and to exploit those associations for audio source separation on new videos.

2.1.3 Feature Learning by Interaction

Limited prior work explores feature learning through interaction. Unlike the self-supervised methods discussed in Sec. 2.1.1, this line of work fosters agents that learn from their own observations in the world, which can be critical for adapting to new environments and to realize truly "bottom-up" learning by experience. Existing methods explore touch and motion interactions.

In [164], objects are struck with a drumstick to facilitate learning material properties when they sound. In [172], the trajectory of a ball bouncing off surfaces facilitates learning physical scene properties. In [7, 169], a robot learns object properties by poking or grasping at objects. In [73], a drone learns not to crash after attempting many crashes. In [6,118], an agent tracks its egomotion in concert with its visual stream to facilitate learning visual categories. In contrast, our VisualEchoes approach presented in Chapter 8 is to learn visual features by *emitting audio* to acoustically interact with the scene. Our work offers a new perspective on interaction-based feature learning and has the advantages of not disrupting the scene physically and being ubiquitously available, i.e., reaching all surrounding surfaces.

2.2 Separation Methods

2.2.1 Audio Source Separation

In Chapter 3, 4, 5, and 7 of this dissertation, a recurring problem studied in my thesis research is the task called audio source separation, which has a rich history in the signal processing literature.

Some methods assume access to multiple microphones, which facilitates separation [48,159,251]. Others accept a single monoaural input [107,198,201, 223, 224] to perform "blind" separation. Popular approaches include Independent Component Analysis (ICA) [109], sparse decomposition [268], Computational Auditory Scene Analysis (CASA) [54], non-negative matrix factorization (NMF) [63,64,138,223], probabilistic latent variable models [103,197], and deep learning [101,107,195]. NMF is a traditional method that is still widely used for unsupervised source separation [94,111,115,201,222]. However, existing methods typically require supervision to get good results. Strong supervision in the form of isolated recordings of individual sound sources [198,224] is effective but difficult to secure for arbitrary sources in the wild. Alternatively, "informed" audio source separation uses special-purpose auxiliary cues to guide the process, such as a music score [100], text [137], or manual user guidance [23,47,224].

Compared to all the audio-only methods above, our approaches learn audio source separation models from unlabeled videos by leveraging the visual information. Unlike the audio-only methods, we use visual cues to guide the separation process (Chapters 3, 4, 5, and 7).

2.2.2 Audio-Visual Source Separation

There is series of work on audio-visual source separation. The idea of guiding audio source separation using *visual* information can be traced back to [39,65], where mutual information is used to learn the joint distribution of the visual and auditory signals, then applied to isolate human speakers. Subsequent work explores audio-visual subspace analysis [171,196], NMF informed by visual motion [167,190], statistical convolutive mixture models [182], and correlating temporal onset events [19,139]. The work of [171] attempts both localization and separation simultaneously; however, it assumes a moving object is present and only aims to decompose a video into background (assumed to be low-rank) and foreground sounds/pixels.

Whereas this prior work correlates low-level visual patterns—particularly motion and onset events—with the audio channel, we propose to learn from video how different *objects* look and sound, whether or not an object moves with obvious correlation to the sounds. Our methods assume access to an image classifier (Chapter 3), an object detector (Chapter 4), or a face detector (Chapter 5), but assume no side information about a novel test video. Furthermore, whereas existing methods analyze a single input video in isolation and are largely constrained to human speakers and instruments, our approach learns a valuable prior for audio separation from a large library of *unlabeled* videos.

Concurrently with our work, and independently of it, a growing body of work has recently begun to leverage deep learning for audio-visual source separation on speech [3, 4, 35, 55, 69, 163] and musical instruments [70, 77, 187, 246, 257, 258]. In contrast, in Chapter 3, we study a broader set of object-level sounds including instruments, animals, and vehicles. Moreover, our method's training data requirements are distinctly more flexible. We are the first to learn from uncurated "in the wild" videos that contain multiple objects and multiple audio sources. Furthermore, an important novelty of our co-separation approach in Chapter 4 is in an end-to-end system that allows more flexible training with multi-source data. Similar to audio-only methods, almost all audio-visual source separation methods use a "mix-and-separate" training paradigm to perform *video-level* separation by artificially mixing training videos. In contrast, our co-separation approach performs source separation at the *object level* to explicitly model sounds coming from different visual objects, and our model enforces separation *within* a video during training. Different from the prior work on audio-visual speech separation, our VisualVoice approach in Chapter 5 solves for speech separation by incorporating both lip motion and cross-modal face-voice attributes. In particular, we propose a multi-task learning framework to jointly learn audio-visual speech separation and cross-modal speaker embeddings. The latter helps learn separation from unlabeled video (i.e., no identity labels, no enrollment of users) by surfacing the sound properties consistent with different facial appearances, as we show in the results.

Finally, while all the methods above exploit mono audio cues to perform audio-visual source separation, our approach described in Chapter 7 proposes to predict binaural cues to enhance separation, which is entirely novel. By transforming mono to binaural through visual guidance, we can leverage the resulting representation to improve the separation quality.

2.3 Cross-Modal Learning with Faces and Voices

There are strong links between how a person's face looks and how their voice sounds. Leveraging this link, cross-modal learning methods explore a range of interesting tasks: face reconstruction from audio [162], talking face generation [263], emotion recognition [8], speaker diarization [34], speech recognition [36], and speaker identification [37,126,157,158,234]. Unlike any of the above, our VisualVoice approach in Chapter 5 tackles audio-visual speech separation.

Taking inspiration from prior work on cross-modal matching [36, 126, 157, 158, 234], we jointly learn cross-modal face-voice embeddings with audio-visual speech separation. However, our goal here is to enhance separation results instead of speaker identification, with the new insight that hearing voice elements consistent with a face's appearance can help disentangle speech from other overlapping sounds.

2.4 Action Recognition

Related to my work discussed in Chapter 6, action recognition in videos has been extensively studied in the past decades. Research has transitioned from initial methods using hand-crafted local spatiotemporal features [135, 226, 235] to mid-level descriptors [114, 178, 228], and more recently to deep video representations learned end-to-end [58, 121, 194]. Various deep networks have been proposed to model spatiotemporal information in videos [25, 57, 173, 209, 232]. Recent work includes capturing long-term temporal structure via recurrent networks [46, 253] or ranking functions [62], pooling across space and/or time [86, 229], modeling hierarchical or spatiotemporal information in videos [170, 214], building long-term temporal relations [237, 260], or boosting accuracy by treating audio as another (late-fused) input modality [124, 146, 231, 241].

The above work focuses on building powerful models to improve recognition without taking the computation cost into account, whereas our work aims to perform efficient action recognition in long untrimmed videos. Some work balances the accuracy-efficiency trade-off by using compressed video representations [192,238] or designing efficient network architectures [31,141,210,244,269]. In contrast, we propose to leverage audio to enable efficient clip-level and video-level action recognition in long untrimmed videos.

Our approach is most related to the limited prior work on selecting salient frames or clips for action recognition in untrimmed videos. Whereas we use only weakly labeled video to train, some methods assume strong human annotations, *i.e.*, ground truth temporal boundaries [250] or sequential annotation traces [10]. Several recent methods [56,204,240,242] propose reinforcement learning (RL) approaches for video frame selection. Without using guidance from strong human supervision, they ease the learning process by restricting the agent to a rigid action space [56], guiding the selection process of the agent with a global memory module [242], or using multiple agents to collaboratively perform frame selection [240].

Unlike any of the above, we introduce a video skimming mechanism to select the key moments in videos aided by audio. We use audio as an efficient way to preview dynamic events for fast video-level recognition. Furthermore, our approach requires neither strong supervision nor complex RL policy gradients, which are often unwieldy to train. SCSampler [131] also leverages audio to accelerate action recognition in untrimmed videos. However, they only consider video-level redundancy by sampling acoustically or visually salient clips.

In contrast, we address both clip-level and video-level redundancy, and we jointly learn the selection and recognition mechanisms. We include a comprehensive experimental comparison to methods in this genre.

2.5 Cross-Modal Distillation

Knowledge distillation [102] addresses the problem of training smaller models from larger ones. In Chapter 6, we propose to distill the knowledge from an expensive clip-based model to a lightweight image-audio based model. Other forms of cross-modal distillation consider transferring supervision from RGB to flow or depth [95] or from a visual network to an audio network, or vice versa [8, 15, 72, 165]. In the opposite direction of ours, DistInit [87] performs uni-modal distillation from a pre-trained image model to a video model for representation learning from unlabeled video. Instead, we perform multi-modal distillation from a video model to an image-audio model for efficient clip-based action recognition.

2.6 Auditory Scene Analysis using Echoes

Our VisualEchoes approach presented in Chapter 8 learns spatial image representations from echoes. Previous work also shows that using echo responses only, one can predict 2D [12] or 3D [44] room geometry and object shape [67]. Additionally, echoes can complement vision, especially when vision-based depth estimates are not reliable, e.g., on transparent windows or featureless walls [127, 249]. In dynamic environments, autonomous robots can leverage echoes for obstacle avoidance [213] or mapping and navigation [53] using a bat-like echolocation model. Concurrently with our work, a low-cost audio system called BatVision is used to predict depth maps purely from echo responses [32]. Our work explores a novel direction for auditory scene analysis by employing echoes for spatial visual feature learning, and unlike prior work, the resulting features are applicable in the absence of any audio.

2.7 Monocular Depth Estimation

Related to my VisualEchoes work discussed in Chapter 8 where we predict depth from echoes, recent methods on monocular depth estimation focus on improving neural network architectures [68] or graphical models [143, 230, 245], employing multi-scale feature fusion and multi-task learning [51, 106], leveraging motion cues from successive frames [212], or transfer learning [122]. However, these approaches rely on depth-labeled data that can be expensive to obtain. Hence, recent approaches leverage scenes' spatial and temporal structure to self-supervise depth estimation, by using the camera motion between pairs of images [82, 88] or frames [89, 119, 219, 265], or consistency cues between depth and features like surface normals [248] or optical flow [177]. Unlike any of these existing methods, we show that audio in the form of an echo response can be effectively used to recover depth, and we develop a novel feature learning method that benefits a purely visual representation (no audio) at test time.

My dissertation focuses on audio-visual learning, and throughout I leverage both the semantic and spatial signals in audio for audio-visual source separation (Chapter 3, 4, 5), action recognition (Chapter 6), audio spatialization (Chapter 7), and spatial image representation learning (Chapter 8). The above sections have reviewed the important related work to these audio-visual learning tasks presented in this dissertation. Apart from these tasks, recent inspiring work also integrates both audio and visual cues on an array of other tasks including self-supervised representation learning [13, 15, 130, 163, 165], localizing sounds in video frames [14, 105, 191, 207], generating sounds from video [30, 164, 256, 264], and audio-visual navigation [28, 29, 71].

Starting from the next chapter, I will move on to present the technical details of the approaches together with the experimental results for each component of my dissertation.

Chapter 3

Disentangling Object Sounds from Unlabeled Video

[1] In this chapter, I introduce my first attempt to leverage the natural correspondence between visual objects and the sounds they make to learn audio-visual object sound models. I propose a method that learns how different objects sound by both looking at *and* listening to unlabeled video containing multiple sounding objects. This work was published in ECCV 2018 [75].

Perceiving a scene most fully requires all the senses. Yet modeling how objects look and sound is challenging: most natural scenes and events contain multiple objects, and the audio track mixes all the sound sources together. Audio source separation is the separation of a set of source signals from a set of mixed signals. Though studied extensively in the signal processing literature [63, 109, 223, 268], it remains a difficult problem with natural data outside of lab settings. Existing methods perform best by capturing the input with

[1] The work in this chapter was supervised by Prof. Kristen Grauman and was originally published in: "Learning to Separate Object Sounds by Watching Unlabeled Video". Ruohan Gao, Rogerio Feris, and Kristen Grauman. In Proceedings of the European Conference on Computer Vision, Munich, Germany, September 2018.

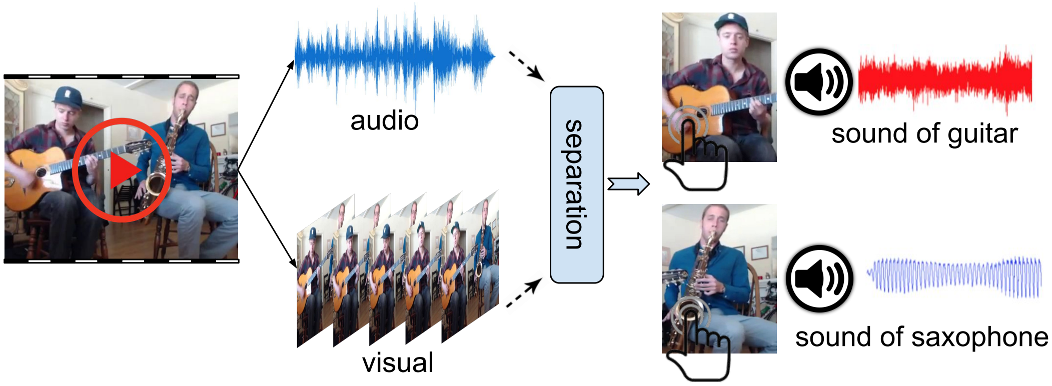


Figure 3.1: Learning to separate object sounds by watching unlabeled videos.

multiple microphones, or else assume a clean set of single source audio examples is available for supervision (e.g., a recording of only a violin, another recording containing only a drum), both of which are very limiting prerequisites. The blind audio separation task evokes challenges similar to image segmentation—and perhaps more, since all sounds overlap in the input signal.

Our goal is to learn how different objects sound by both looking at *and* listening to unlabeled video containing multiple sounding objects. In this dissertation, I propose two unsupervised approaches to disentangle mixed audio into its component sound sources. In this section, I introduce my first attempt— a two-stage approach based on multi-label multi-instance learning (MIML) and non-negative matrix factorization (NMF). Chapter 4 will introduce my end-to-end co-separation approach. The key insight of the two approaches is that observing sounds in a variety of visual contexts reveals the cues needed to isolate individual audio sources; the different visual contexts lend weak supervision for discovering the associations. For example, having experienced various instruments playing in various combinations before, then given a video with a guitar and a saxophone (Fig. 3.1), one can naturally anticipate what sounds could be present in the accompanying audio, and therefore better separate them. Indeed, neuroscientists report that the mismatch negativity of event-related brain potentials, which is generated bilaterally within auditory cortices, is elicited only when the visual pattern promotes the segregation of the sounds [176]. This suggests that synchronous presentation of visual stimuli should help to resolve sound ambiguity due to multiple sources, and promote either an integrated or segregated perception of the sounds.

Our two novel audio-visual source separation approaches realize this intuition. The first approach is a two-stage method that relies on non-negative matrix factorization (NMF) to perform separation. We first leverage a large collection of unannotated videos to discover a latent sound representation for each object. Specifically, we use state-of-the-art image recognition tools to infer the objects present in each video clip, and we perform NMF on each video's audio channel to recover its set of frequency basis vectors. At this point it is unknown which audio bases go with which visible object(s). To recover the association, we construct a neural network for multi-instance multi-label learning that maps audio bases to the distribution of detected visual objects. From this audio basis-object association network, we extract the audio bases linked to each visual object, yielding its prototypical spectral patterns. Finally, given a novel video, we use the learned per-object audio bases to steer audio source separation. To address the limitations of the two-stage approach described above, we propose a co-separation approach that permits end-to-end training from unlabeled multi-source videos, which will be introduced in the next chapter.

Prior attempts at visually-aided audio source separation tackle the problem by detecting low-level correlations between the two data streams for the input video [19,26,39,65,139,167,171,182], and they experiment with somewhat controlled domains of musical instruments in concert or human speakers facing the camera. In contrast, we propose to *learn object-level sound models* from hundreds of thousands of unlabeled videos, and generalize to separate new audio-visual instances. We demonstrate results for a broad set of "in the wild" videos. While a resurgence of research on cross-modal learning from images and audio also capitalizes on synchronized audio-visual data for various tasks [13–15, 125, 130, 164, 165], they treat the audio as a single monolithic input, and thus cannot associate different sounds to different objects in the same video. Concurrent with our work, other new methods for audio-visual source separation are being explored specifically for speech [3, 55, 69, 163] or musical instruments [258]. In contrast, we study a broader set of object-level sounds including instruments, animals, and vehicles. Moreover, our method's training data requirements are distinctly more flexible.

The main contributions in this component of my thesis research are as follows. Firstly, we propose to enhance audio source separation in videos by "supervising" it with visual information from image recognition results.[2] Sec-

[2] Our task can hence be seen as "weakly supervised", though the weak "labels" themselves

ondly, we propose a novel deep multi-instance multi-label learning framework to learn prototypical spectral patterns of different acoustic objects, and inject the learned prior into an NMF source separation framework. Thirdly, to our knowledge, we are the first to study audio source separation learned from large scale online videos. We demonstrate state-of-the-art results on visually-aided audio source separation and audio denoising.

I first describe our approach for learning object sound models from unlabeled video in Sec 3.1, before presenting the experimental results in Sec 3.2.

3.1 Approach

Our approach learns what objects sound like from a batch of unlabeled, multi-sound-source videos. Given a new video, our method returns the separated audio channels and the visual objects responsible for them.

We first formalize the audio separation task and overview audio basis extraction with NMF (Sec. 3.1.1). Then we introduce our framework for learning audio-visual objects from unlabeled video (Sec. 3.1.2) and our accompanying deep multi-instance multi-label network (Sec. 3.1.3). Next we present an approach to use that network to associate audio bases with visual objects (Sec. 3.1.4). Finally, we pose audio source separation for novel videos in terms of a semi-supervised NMF approach (Sec. 3.1.5).

are inferred from the video, not manually annotated.

3.1.1 Audio Basis Extraction

Single-channel audio source separation is the problem of obtaining an estimate for each of the J sources s_j from the observed linear mixture $x(t)$: $x(t) = \sum_{j=1}^{J} s_j(t)$, where $s_j(t)$ are time-discrete signals. The mixture signal can be transformed into a magnitude or power spectrogram $\mathbf{V} \in \mathbb{R}_+^{F \times N}$ consisting of F frequency bins and N short-time Fourier transform (STFT) [93] frames, which encode the change of a signal's frequency and phase content over time. We operate on the frequency domain, and use the inverse short-time Fourier transform (ISTFT) [93] to reconstruct the sources.

Non-negative matrix factorization (NMF) is often employed [63,64,138, 223] to approximate the (non-negative real-valued) spectrogram matrix $\mathbf{V}$ as a product of two matrices $\mathbf{W}$ and $\mathbf{H}$:

$$\mathbf{V} \approx \tilde{\mathbf{V}} = \mathbf{W}\mathbf{H}, \tag{3.1}$$

where $\mathbf{W} \in \mathbb{R}_+^{F \times M}$ and $\mathbf{H} \in \mathbb{R}_+^{M \times N}$. The number of bases M is a user-defined parameter. $\mathbf{W}$ can be interpreted as the non-negative audio spectral patterns, and $\mathbf{H}$ can be seen as the activation matrix. Specifically, each column of $\mathbf{W}$ is referred to as a *basis vector*, and each row in $\mathbf{H}$ represents the gain of the corresponding basis vector. The factorization is usually obtained by solving the following minimization problem:

$$\min_{\mathbf{W},\mathbf{H}} D(\mathbf{V}|\mathbf{W}\mathbf{H}) \text{ subject to } \mathbf{W} \geq 0, \mathbf{H} \geq 0, \tag{3.2}$$

where D is a measure of divergence, e.g., we employ the Kullback-Leibler (KL) divergence.

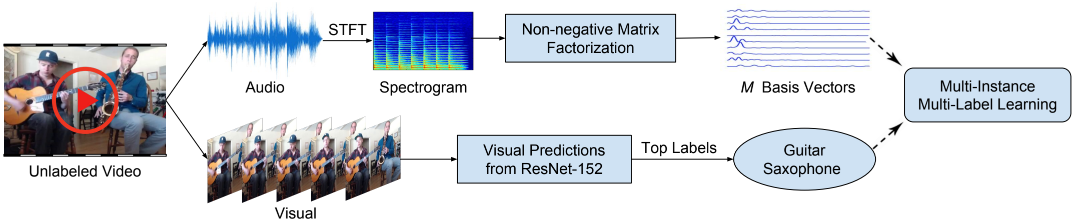

STFT
Non-negative Matrix Factorization
Audio
Spectrogram
M Basis Vectors
Multi-Instance Multi-Label Learning
Unlabeled Video
Visual Predictions from ResNet-152
Top Labels
Guitar
Saxophone
Visual

Figure 3.2: Unsupervised training pipeline. For each video, we perform NMF on its audio magnitude spectrogram to get M basis vectors. An ImageNet-trained ResNet-152 network is used to make visual predictions to find the potential objects present in the video. Finally, we perform multi-instance multi-label learning to disentangle which extracted audio basis vectors go with which detected visible object(s).

For each unlabeled training video, we perform NMF independently on its audio magnitude spectrogram to obtain its spectral patterns $\mathbf{W}$, and throw away the activation matrix $\mathbf{H}$. M audio basis vectors are therefore extracted from each video.

3.1.2 Weakly-Supervised Audio-Visual Object Learning Framework

Multiple objects can appear in an unlabeled video at the same time, and similarly in the associated audio track. At this point, it is unknown which of the audio bases extracted (columns of $\mathbf{W}$) go with which visible object(s) in the visual frames. To discover the association, we devise a multi-instance multi-label learning (MIML) framework that matches audio bases with the detected objects.

As shown in Fig. 3.2, given an unlabeled video, we extract its visual frames and the corresponding audio track. As defined above, we perform NMF

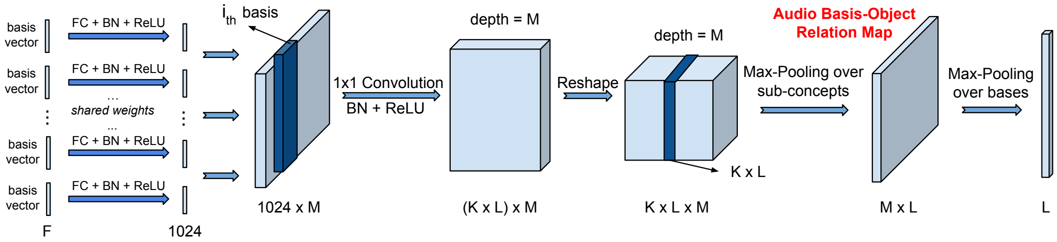


Figure 3.3: Our deep multi-instance multi-label network takes a bag of M audio basis vectors for each video as input, and gives a bag-level prediction of the objects present in the audio. The visual predictions from an ImageNet-trained CNN are used as weak "labels" to train the network with unlabeled video.

independently on the magnitude spetrogram of each audio track and obtain M basis vectors from each video. For the visual frames, we use an ImageNet pre-trained ResNet-152 network [99] to make object category predictions, and we max-pool over predictions of all frames to obtain a video-level prediction. The top labels (with class probability larger than a threshold) are used as weak "labels" for the unlabeled video. The extracted basis vectors and the visual predictions are then fed into our MIML learning framework to discover associations, as defined next.

3.1.3 Deep Multi-Instance Multi-Label Network

We cast the audio basis-object disentangling task as a multi-instance multi-label (MIML) learning problem. In single-label MIL [43], one has bags of instances, and a bag label indicates only that some number of the instances within it have that label. In MIML, the bag can have multiple labels, and there is ambiguity about which labels go with which instances in the bag.

We design a deep MIML network for our task. A bag of basis vectors $\{\mathbf{B}\}$ is the input to the network, and within each bag there are M basis vectors $\mathbf{B}_i$ with $i \in [1, M]$ extracted from one video. The "labels" are only available at the bag level, and come from noisy visual *predictions* of the ResNet-152 network trained for ImageNet recognition. The labels for each instance (basis vector) are unknown. We incorporate MIL into the deep network by modeling that there must be *at least one* audio basis vector from a certain object that constitutes a positive bag, so that the network can output a correct bag-level prediction that agrees with the visual prediction.

Fig. 3.3 shows the detailed network architecture. M basis vectors are fed through a Siamese Network of M branches with shared weights. The Siamese network is designed to reduce the dimension of the audio frequency bases and learns the audio spectral patterns through a fully-connected layer (FC) followed by batch norm (BN) [112] and a rectified linear unit (ReLU). The output of all branches are stacked to form a $1024 \times M$ dimension feature map. Each slice of the feature map represents a basis vector with reduced dimension. Inspired by [59], each label is decomposed to K sub-concepts to capture latent semantic meanings. For example, for drum, the latent sub-concepts could be different types of drums, such as bongo drum, tabla, and so on. The stacked output from the Siamese network is forwarded through a 1×1 Convolution-BN-ReLU module, and then reshaped into a feature cube of dimension $K \times L \times M$, where K is the number of sub-concepts, L is the number of object categories, and M is the number of audio basis vectors. The depth of the tensor equals the number of input basis vectors, with each $K \times L$ slice corresponding to one particular basis. The activation score of the $(k, l, m)_{\text{th}}$ node in the cube represents the matching score of the k_{th} sub-concept of the l_{th} label for the m_{th} basis vector.

To get a bag-level prediction, we conduct two max-pooling operations. Max pooling in deep MIL [59, 239, 247] is typically used to identify the positive instances within an aggregated bag. Our first pooling is over the sub-concept dimension (K) to generate an audio basis-object relation map. The second max-pooling operates over the basis dimension (M) to produce a video-level prediction. We use the following multi-label hinge loss to train the network:

$$\mathcal{L}(A, \mathcal{V}) = \frac{1}{L} \sum_{i=1, i \neq \mathcal{V}_j}^{L} \sum_{j=1}^{|\mathcal{V}|} \max[0, 1 - (A_{\mathcal{V}_j} - A_i)], \tag{3.3}$$

where $A \in \mathbb{R}^L$ is the output of the MIML network, and represents the object predictions based on audio bases; $\mathcal{V}$ is the set of visual objects, namely the indices of the $|\mathcal{V}|$ objects predicted by the ImageNet-trained model. The loss function encourages the prediction scores of the correct classes to be larger than incorrect ones by a margin of 1. We find these pooling steps in our MIML formulation are valuable to learn accurately from the ambiguously "labeled" bags (i.e., the videos and their object predictions).

3.1.4 Disentangling Per-Object Bases

The MIML network above learns from audio-visual associations, but does not itself disentangle them. The sounds in the audio track and objects present in the visual frames of unlabeled video are diverse and noisy (see Sec. 3.2.1 for details about the data we use). The audio basis vectors extracted from each video could be a component shared by multiple objects, a feature composed of them, or even completely unrelated to the predicted visual objects. The visual predictions from ResNet-152 network give approximate predictions about the objects that could be present, but are certainly not always reliable (see Fig. 3.6 for examples).

Therefore, to collect high quality representative bases for each object category, we use our trained deep MIML network as a tool. The audio basis-object relation map after the first pooling layer of the MIML network produces matching scores across all basis vectors for all object labels. We perform a dimension-wise softmax over the basis dimension (M) to normalize object matching scores to probabilities along each basis dimension. By examining the normalized map, we can discover links from bases to objects. We only collect the key bases that trigger the prediction of the correct objects (namely, the visually detected objects). Further, we only collect bases from an unlabeled video if multiple basis vectors strongly activate the correct object(s). See [75] for details, and see Fig. 3.6 for examples of typical basis-object relation maps. In short, at the end of this phase, we have a set of audio bases for each visual object, discovered purely from unlabeled video and mixed single-channel audio.

3.1.5 Object Sound Separation for a Novel Video

Finally, we present our procedure to separate audio sources in new

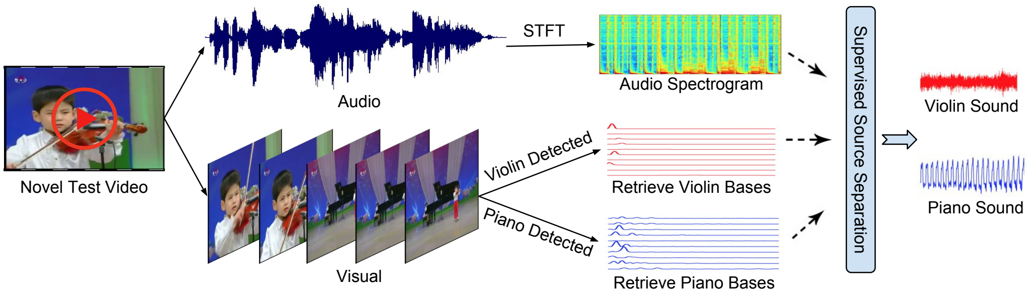


Figure 3.4: Testing pipeline. Given a novel test video, we detect the objects present in the visual frames, and retrieve their learnt audio bases. The bases are collected to form a fixed basis dictionary $\mathbf{W}$ with which to guide NMF factorization of the test video's audio channel. The basis vectors and the learned activation scores from NMF are finally used to separate the sound for each detected object, respectively.

videos. As shown in Fig. 3.4, given a novel test video q, we obtain its audio magnitude spectrogram $\mathbf{V}^{(q)}$ through STFT and detect objects using the same ImageNet-trained ResNet-152 network as before. Then, we retrieve the learnt audio basis vectors for each detected object, and use them to “guide” NMF-based audio source separation. Specifically,

$$\begin{aligned}\mathbf{V}^{(q)} \approx \tilde{\mathbf{V}}^{(q)} &= \mathbf{W}^{(q)}\mathbf{H}^{(q)} \\ &= \left[\mathbf{W}_1^{(q)} \quad \cdots \quad \mathbf{W}_j^{(q)} \quad \cdots \quad \mathbf{W}_J^{(q)}\right] \left[\mathbf{H}_1^{(q)} \cdots \mathbf{H}_j^{(q)} \cdots \mathbf{H}_J^{(q)}\right]^T,\end{aligned} \tag{3.4}$$

where J is the number of detected objects (J potential sound sources), and $\mathbf{W}_j^{(q)}$ contains the retrieved bases corresponding to object j in input video q. In other words, we concatenate the basis vectors learnt for each detected object to construct the basis dictionary $\mathbf{W}^{(q)}$. Next, in the NMF algorithm, we hold $\mathbf{W}^{(q)}$ fixed, and only estimate activations $\mathbf{H}^{(q)}$ with multiplicative update rules. Then we obtain the spectrogram corresponding to each detected object by $\mathbf{V}_j^{(q)} = \mathbf{W}_j^{(q)}\mathbf{H}_j^{(q)}$. We reconstruct the individual (compressed) audio source signals by soft masking the mixture spectrogram:

$$\mathbb{V}_j = \frac{\mathbf{V}_j^{(q)}}{\sum_{i=1}^{J} \mathbf{V}_i^{(q)}} \mathbb{V}, \tag{3.5}$$

where $\mathbb{V}$ contains both magnitude and phase. Finally, we perform ISTFT on $\mathbb{V}_j$ to reconstruct the audio signals for each detected object. If a detected object does not make sound, then its estimated activation scores will be low. This phase can be seen as a self-supervised form of NMF, where the detected visual objects reveal which bases (previously discovered from unlabeled videos) are relevant to guide audio separation.

3.2 Experiments

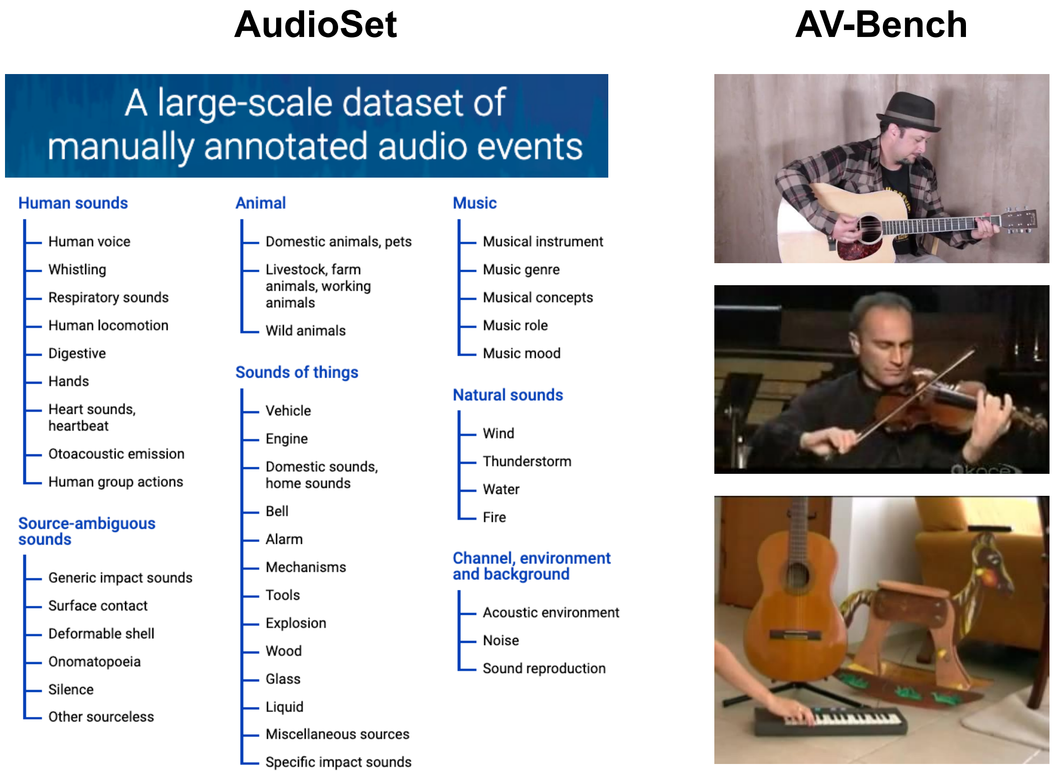
We now validate our approach and compare to existing methods.

3.2.1 Datasets

As shown in Fig. 3.5, we consider two public video datasets: AudioSet [84] and the benchmark videos from [113, 140, 171], which we refer to as AV-Bench.

- **AudioSet-Unlabeled:** We use AudioSet [84] as the source of unlabeled training videos.[3] The dataset consists of short 10 second video clips that often concentrate on one event. However, our method makes no

[3] AudioSet offers noisy video-level audio class annotations. However, we do not use any of its label information.



AudioSet

AV-Bench

A large-scale dataset of manually annotated audio events

Human sounds
- Human voice
- Whistling
- Respiratory sounds
- Human locomotion
- Digestive
- Hands
- Heart sounds, heartbeat
- Otoacoustic emission
- Human group actions

Source-ambiguous sounds
- Generic impact sounds
- Surface contact
- Deformable shell
- Onomatopoeia
- Silence
- Other sourceless

Animal
- Domestic animals, pets
- Livestock, farm animals, working animals
- Wild animals

Sounds of things
- Vehicle
- Engine
- Domestic sounds, home sounds
- Bell
- Alarm
- Mechanisms
- Tools
- Explosion
- Wood
- Glass
- Liquid
- Miscellaneous sources
- Specific impact sounds

Music
- Musical instrument
- Music genre
- Musical concepts
- Music role
- Music mood

Natural sounds
- Wind
- Thunderstorm
- Water
- Fire

Channel, environment and background
- Acoustic environment
- Noise
- Sound reproduction

Figure 3.5: We evaluate on AudioSet [84] and three benchmark videos from [113, 140, 171], which we refer to as AV-Bench.

particular assumptions about using short or trimmed videos, as it learns bases in the frequency domain and pools both visual predictions and audio bases from all frames. The videos are challenging: many are of poor quality and unrelated to object sounds, such as silence, sine wave, echo, and infrasound. As is typical for related experimentation in the literature [14, 266], we filter the dataset to those likely to display audio-visual events. In particular, we extract musical instruments, animals, and vehicles, which span a broad set of unique sound-making objects.

See [75] for a complete list of the object categories. Using the dataset's provided split, we randomly reserve some videos from the "unbalanced" split as validation data, and the rest as the training data. We use videos from the "balanced" split as test data. The final AudioSet-Unlabeled data contains 104k, 2.9k, 1k / 22k, 1.2k, 0.5k / 58k, 2.4k, 0.6k video clips in the train, val, test splits, for the instruments, animals, and vehicles, respectively.

- **AudioSet-SingleSource:** To facilitate quantitative evaluation (cf. Sec. 3.2.4), we construct a dataset of AudioSet videos containing only a single sounding object. We manually examine videos in the val/test set, and obtain 23 such videos. There are 15 musical instruments (accordion, acoustic guitar, banjo, cello, drum, electric guitar, flute, french horn, harmonica, harp, marimba, piano, saxophone, trombone, violin), 4 animals (cat, dog, chicken, frog), and 4 vehicles (car, train, plane, motorbike). Note that our method never uses these samples for training.
- **AV-Bench:** This dataset contains the benchmark videos (Violin Yanni, Wooden Horse, and Guitar Solo) used in previous studies [113, 140, 171].

3.2.2 Implementation Details

We extract a 10 second audio clip and 10 frames (every 1s) from each video. Following common settings [13], the audio clip is resampled at 48 kHz, and converted into a magnitude spectrogram of size 2401×202 through STFT of window length 0.1s and half window overlap. We use the NMF implementation of [64] with KL divergence and the multiplicative update solver. We extract $M = 25$ basis vectors from each audio. All video frames are resized to 256×256, and 224×224 center crops are used to make visual predictions. We use all relevant ImageNet categories and group them into 23 classes by merging the posteriors of similar categories to roughly align with the AudioSet categories; see [75]. A softmax is finally performed on the video-level object prediction scores, and classes with probability greater than 0.3 are kept as weak labels for MIML training. The deep MIML network is implemented in PyTorch with $F = 2,401$, $K = 4$, $L = 25$, and $M = 25$. We report all results with these settings and did not try other values. The network is trained using Adam [128] with weight decay 10^{-5} and batch size 256. The starting learning rate is set to 0.001, and decreased by 6% every 5 epochs and trained for 300 epochs.

3.2.3 Baselines

We compare to several existing methods [125,145,171,201] and multiple baselines:

- **MFCC Unsupervised Separation [201]:** This is an off-the-shelf unsupervised audio source separation method. The separated channels are first converted into Mel frequency cepstrum coefficients (MFCC), and then K-means clustering is used to group separated channels. This is an established pipeline in the literature [94, 111, 115, 222], making it a good representative for comparison. We use the publicly available code[4].

- **AV-Loc [171], JIVE [145], Sparse CCA [125]:** We refer to results reported in [171] for the AV-Bench dataset to compare to these methods.
- **AudioSet Supervised Upper-Bound:** This baseline uses AudioSet ground-truth labels to train our deep MIML network. AudioSet labels are organized in an ontology and each video is labeled by many categories. We use the 23 labels aligned with our subset (15 instruments, 4 animals, and 4 vehicles). This baseline serves as an upper-bound.
- **K-means Clustering Unsupervised Separation:** We use the same number of basis vectors as our method to initialize the **W** matrix, and perform unsupervised NMF. K-means clustering is then used to group separated channels, with K equal to the number of ground-truth sources. The sound sources are separated by aggregating the channel spectrograms belonging to each cluster.
- **Visual Exemplar for Supervised Separation:** We recognize objects in the frames, and retrieve bases from an exemplar video for each detected object class to supervise its NMF audio source separation. An exemplar video is the one that has the largest confidence score for a class among all unlabeled training videos.
- **Unmatched Bases for Supervised Separation:** This baseline is the same as our method except that it retrieves bases of the wrong class (at

[4] `https://github.com/interactiveaudiolab/nussl`

random from classes absent in the visual prediction) to guide NMF audio source separation.

- **Gaussian Bases for Supervised Separation:** We initialize the weight matrix $\mathbf{W}$ randomly using a Gaussian distribution, and then perform supervised audio source separation (with $\mathbf{W}$ fixed) as in Sec. 3.1.5.

3.2.4 Quantitative Results

Visually-Aided Audio Source Separation: For "in the wild" unlabeled videos, the ground-truth of separated audio sources never exists. Therefore, to allow quantitative evaluation, we create a test set consisting of combined single-source videos, following [19]. In particular, we take pairwise video combinations from AudioSet-SingleSource (cf. Sec. 3.2.1) and 1) compound their audio tracks by normalizing and mixing them and 2) compound their visual channels by max-pooling their respective object predictions. Each compound video is a test video; its reserved source audio tracks are the ground truth for evaluation of separation results.

To evaluate source separation quality, we use the widely used BSS-EVAL toolbox [220] and report the Signal to Distortion Ratio (SDR). We perform four sets of experiments: pairwise compound two videos of musical instruments (Instrument Pair), two of animals (Animal Pair), two of vehicles (Vehicle Pair), and two cross-domain videos (Cross-Domain Pair). For unsupervised clustering separation baselines, we evaluate both possible matchings

	Instrument Pair	Animal Pair	Vehicle Pair	Cross-Domain Pair
Upper-Bound	2.05	0.35	0.60	2.79
K-means Clustering	-2.85	-3.76	-2.71	-3.32
MFCC Unsupervised [201]	0.47	-0.21	-0.05	1.49
Visual Exemplar	-2.41	-4.75	-2.21	-2.28
Unmatched Bases	-2.12	-2.46	-1.99	-1.93
Gaussian Bases	-8.74	-9.12	-7.39	-8.21
Ours	**1.83**	**0.23**	**0.49**	**2.53**

Table 3.1: We pairwise mix the sounds of two single source AudioSet videos and perform audio source separation. Mean Signal to Distortion Ratio (SDR in dB, higher is better) is reported to represent the overall separation performance.

and take the best results (to the baselines' advantage).

Table 3.1 shows the results. Our method significantly outperforms the Visual Exemplar, Unmatched, and Gaussian baselines, demonstrating the power of our learned bases. Compared with the unsupervised clustering baselines, including [201], our method achieves large gains. It also has the capability to match the separated source to acoustic objects in the video, whereas the baselines can only return ungrounded audio signals. We stress that both our method as well as the baselines use no audio-based supervision. In contrast, other state-of-the-art audio source separation methods supervise the separation process with labeled training data containing clean ground-truth sources and/or tailor separation to music/speech (e.g., [101, 107, 144]). Such methods are not applicable here.

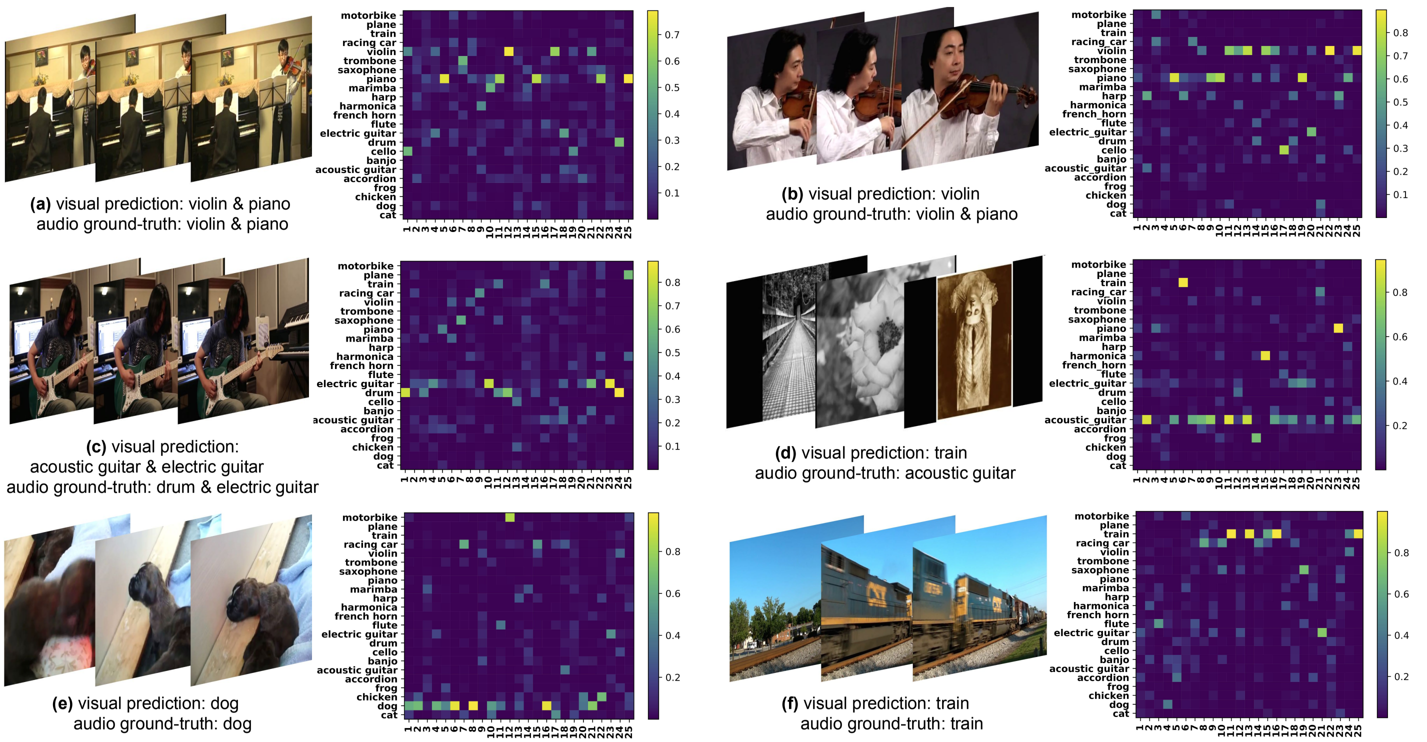
Visually-Aided Audio Denoising: To facilitate comparison to prior audio-visual methods (none of which report results on AudioSet), next we perform the same experiment as in [171] on visually-assisted audio denoising on AV-

	Wooden Horse	Violin Yanni	Guitar Solo	Average
Sparse CCA (Kidron et al. [125])	4.36	5.30	5.71	5.12
JIVE (Lock et al. [145])	4.54	4.43	2.64	3.87
Audio-Visual (Pu et al. [171])	8.82	5.90	**14.1**	9.61
Ours	**12.3**	**7.88**	11.4	**10.5**

Table 3.2: Visually-assisted audio denoising results on three benchmark videos, in terms of NSDR (in dB, higher is better).

Bench. Following the same setup as [171], the audio signals in all videos are corrupted with white noise with the signal to noise ratio set to 0 dB. To perform audio denoising, our method retrieves bases of detected object(s) and appends the same number of randomly initialized bases as the weight matrix $\mathbf{W}$ to supervise NMF. The randomly initialized bases are intended to capture the noise signal. As in [171], we report Normalized SDR (NSDR), which measures the improvement of the SDR between the mixed noisy signal and the denoised sound.

Table 3.2 shows the results. Note that the method of [171] is tailored to separate noise from the foreground sound by exploiting the low-rank nature of background sounds. Still, our method outperforms [171] on 2 out of the 3 videos, and performs much better than the other two prior audio-visual methods [125, 145]. Pu *et al.* [171] also exploit motion in manually segmented regions. On Guitar Solo, the hand's motion may strongly correlate with the sound, leading to their better performance.



(a) visual prediction: violin & piano
audio ground-truth: violin & piano

(b) visual prediction: violin
audio ground-truth: violin & piano

(c) visual prediction:
acoustic guitar & electric guitar
audio ground-truth: drum & electric guitar

(d) visual prediction: train
audio ground-truth: acoustic guitar

(e) visual prediction: dog
audio ground-truth: dog

(f) visual prediction: train
audio ground-truth: train

Figure 3.6: In each example, we show the video frames, visual predictions, and the corresponding basis-label relation maps predicted by our MIML network. Please see our video[5] for more examples and the corresponding audio tracks.

3.2.5 Qualitative Results

Next we provide qualitative results to illustrate the effectiveness of MIML training and the success of audio source separation. Here we run our method on the real multi-source videos from AudioSet. They lack ground truth, but results can be manually inspected for quality (see our video[5]).

Fig. 3.6 shows example unlabeled videos and their discovered audio basis associations. For each example, we show sample video frames, ImageNet CNN visual object predictions, as well as the corresponding audio basis-object relation map predicted by our MIML network. We also report the AudioSet audio ground truth labels, but note that they are never seen by our method. The first example (Fig. 3.6-a) has both piano and violin in the visual frames, which are correctly detected by the CNN. The audio also contains the sounds of both instruments, and our method appropriately activates bases for both the violin and piano. Fig. 3.6-b shows a man playing the violin in the visual frames, but both piano and violin are strongly activated. Listening to the audio, we can hear that an out-of-view player is indeed playing the piano. This example accentuates the advantage of learning object sounds from thousands of unlabeled videos; our method has learned the correct audio bases for piano, and "hears" it even though it is off-camera in this test video. Fig. 3.6-c/d show two examples with inaccurate visual predictions, and our model correctly activates the label of the object in the audio. Fig. 3.6-e/f show two more examples of an animal and a vehicle, and the results are similar. These examples suggest that our MIML network has successfully learned the prototypical spectral patterns of different sounds, and is capable of associating audio bases with object categories.

Please see our video[5] for more results, where we use our system to detect and separate object sounds for novel "in the wild" videos.

3.3 Conclusions

In this chapter, I presented a framework to learn object sound models from thousands of unlabeled videos. Our deep multi-instance multi-label network automatically links audio bases to object categories. Using the disentangled bases to supervise non-negative matrix factorization, our approach successfully separates object-level sounds. We demonstrate its effectiveness on diverse data and object categories.

Overall, as my initial attempt to learn object sound models from unlabeled video, the results presented above are promising and constitute a noticeable step towards visually guided audio source separation for more realistic videos. However, the proposed approach is a two-stage method that heavily relies on non-negative matrix factorization (NMF) to perform separation, which limits its performance and practicability. NMF can be computationally expensive if a large quantity of audio bases are used to guide the separation process. Moreover, we assume that different classes do not share the same bases, and extract audio bases independently from each training video without enforcing

[5] http://vision.cs.utexas.edu/projects/separating_object_sounds/

them to be shared across videos. However, many audio frequency bases could be shared by several classes and audio bases extracted from different videos can be very different, both of which could be limiting factors that prevent clean separation. It would be desirable to have an end-to-end audio separation system, where not only the separation quality can be informed by visual information, but also it is not bounded by NMF. To address the above problem, in the next chapter, I will introduce my second attempt: an end-to-end co-separation approach for audio-visual source separation, and compare their results.

Chapter 4

Co-Separating Sounds of Visual Objects

[1]In the previous chapter, I introduced my initial approach to learn object sound models from unlabeled video. The results are encouraging, but the quality of audio separation of the proposed two-stage method is bounded by the performance of NMF, which limits its practicability. In this chapter, I propose a new end-to-end solution that permits both end-to-end training and learning object-level sounds from unlabeled multi-source videos. This was published in ICCV 2019 [77].

Recent methods tackle the audio(-visual) source separation problem using a "mix-and-separate" paradigm to train deep neural networks in a self-supervised manner [55, 163, 195, 252, 258]. Namely, different from traditional NMF-style methods that analyze structures in the mixture spectrogram to directly separate the component sounds, such methods randomly mix audio/video clips, and the learning objective is to recover the original unmixed

[1] The work in this chapter was supervised by Prof. Kristen Grauman and was originally published in: "Co-Separating Sounds of Visual Objects". Ruohan Gao and Kristen Grauman. In Proceedings of the IEEE International Conference on Computer Vision, Seoul, Korea, October 2019.

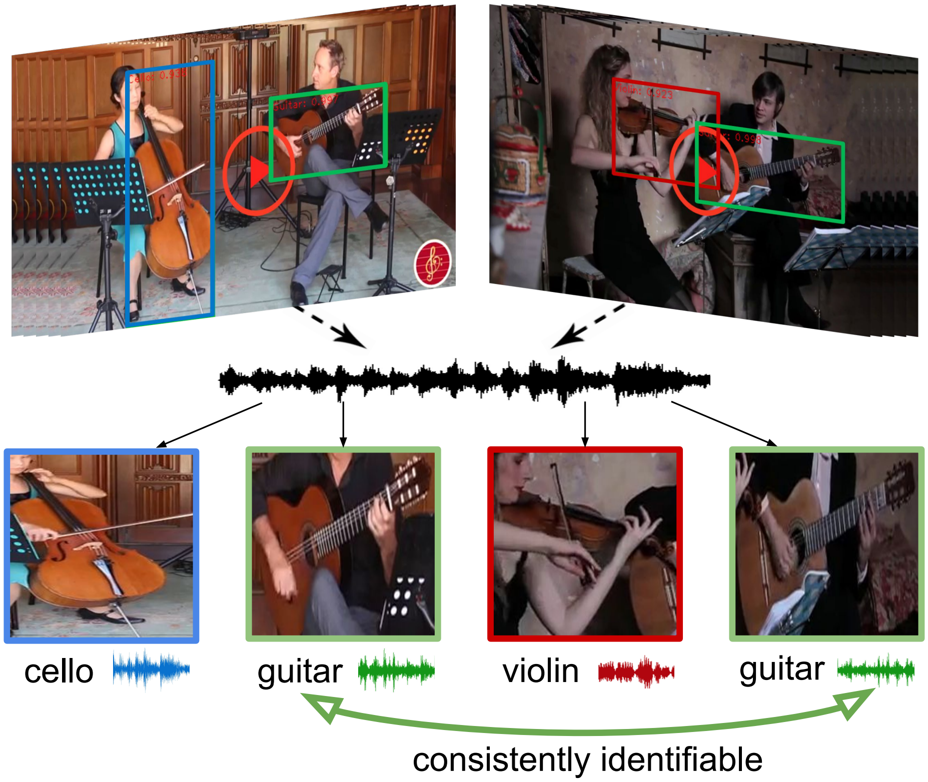


Figure 4.1: We propose a *co-separation* training objective to learn audio-source separation from unlabeled video containing multiple sound sources. Our approach learns to associate consistent sounds to similar-looking objects across pairs of training videos. Then, given a single novel video, it returns a separate sound track for each object.

signals. For example, one can create "synthetic cocktail parties" that mix clean speech with other sounds [55], add pseudo "off-screen" human speakers to other real videos [163], or superimpose audio from clips of musical instruments [258].

There are two key limitations with this current mix-and-separate training strategy. First, it implicitly assumes that the original real training videos are dominated by single-source clips containing one primary sound maker. However, gathering a large number of such clean "solo" recordings is expensive and will be difficult to scale beyond particular classes like human speakers and musical instruments. Second, it implicitly assumes that the sources in a recording are independent. However, it is precisely the correlations *between real sound sources* (objects) that make the source separation problem most challenging at test time. Such correlations can go uncaptured by the artificially mixed training clips.

Towards addressing these shortcomings, we introduce a new strategy for learning to separate audio sources. Our key insight is a novel *co-separation* training objective that learns from naturally occurring multi-source videos[2]. During training, our co-separation network considers pairs of training videos and, rather than simply separate their artificially mixed soundtracks, it must also generate audio tracks that are *consistently identifiable at the object level* across all training samples. In particular, using noisy object detections from the unlabeled training video, we devise a loss requiring that within an individual training video, each separated audio track should be distinguishable as its proper object. For example, when two training instances both contain a guitar plus other instruments, there is pressure to make the separated guitar tracks consistently identifiable by sound. See Fig. 4.1.

We draw a loose analogy to image co-segmentation [186] and call our

[2] Throughout, we use "multi-source video" as shorthand for video containing multiple sounds in its single-channel audio.

idea “co-separation”. For image co-segmentation, jointly segmenting two related images can be easier than segmenting them separately, since it allows disentangling a shared foreground object from differently cluttered backgrounds. Similarly, for our co-separation framework, we separate sounds for pairs of training videos and enforce the separated audio tracks to be consistently identifiable at the object level across all training samples. Note, however, that our co-separation operates during training only; unlike co-segmentation, at test time our method performs separation on an individual video input.

Compared to our previous NMF-based approach described in Chapter 3, our new co-separation method is end-to-end trainable and leverages localized object regions to guide the separation process. More generally, compared to other recent audio-visual source separation methods [3,163,257,258], our model design also offers the following advantages: First, co-separation allows training with “in the wild” sound mixes. It has the potential to benefit from the variability and richness of unlabeled multi-source video. Second, it enhances the supervision beyond “mix-and-separate”. By enforcing separation *within a single video* at the object-level, our approach exposes the learner to natural correlations between sound sources. Finally, objects with similar appearance from different videos can partner with each other to separate their sounds jointly, thereby regularizing the learning process. In this way, our method is able to learn well from multi-source videos, and successfully separate an object sound in a test video even if the object has never been observed individually during training.

We experiment on three benchmark datasets and demonstrate the advantages discussed above. Our approach yields state-of-the-art results on separation and denoising. Most notably, it outperforms the prior methods and baselines by a large margin when learning from noisy AudioSet [84] videos. Overall co-separation is a promising direction to learn audio-visual separation from multi-source videos.

In Sec 4.1, I describe our co-separation approach for learning audio-visual source separation. Then I present the experimental results in Sec 4.2.

4.1 Approach

Our approach learns to leverage localized object detection to visually guide audio source separation. In the following, we first formalize our object-level audio-visual source separation task (Sec. 4.1.1). Then we introduce our framework for learning object sound models from unlabeled video and our Co-Separation deep network architecture (Sec. 4.1.2). Finally, we present our training criteria and inference procedures (Sec. 4.1.3).

4.1.1 Problem Formulation

Given an unlabeled video clip V with accompanying audio $x(t)$, we denote $\mathcal{V} = \{O_1, \ldots, O_N\}$ the set of N objects detected in the video frames. Note that this is different in Chapter 3, where we use image-level labels predicted by a pre-trained image classifier. We treat each object as a potential sound source, and $x(t) = \sum_{n=1}^{N} s_n(t)$ is the observed single-channel linear mixture of these sources, where $s_n(t)$ are time-discrete signals responsible for each object. Our goal of object-level audio-visual source separation is to separate the sound $s_n(t)$ for each object O_n from $x(t)$.

Following [55, 76, 101, 108, 163, 252, 258], we start with the commonly adopted "mix-and-separate" idea to self-supervise source separation. Given two training videos V_1 and V_2 with corresponding audios $x_1(t)$ and $x_2(t)$, we use a pre-trained object detector to find objects in both videos. Then, we mix the audios of the two videos and obtain the mixed signal $x_m(t) = x_1(t)+x_2(t)$. The mixed audio $x_m(t)$ is transformed into a magnitude spectrogram $X^M \in \mathbb{R}_+^{F\times N}$ consisting of F frequency bins and N short-time Fourier transform (STFT) [93] frames, which encodes the change of a signal's frequency and phase content over time.

Our learning objective is to separate the sound each object makes from $x_m(t)$ conditioned on the localized object regions. For example, Fig. 4.3 illustrates a scenario of mixing two videos V_1 and V_2 with two objects O_1, O_2 detected in V_1 and one object O_3 detected in V_2. The goal is to separate $s_1(t)$, $s_2(t)$, and $s_3(t)$ for objects O_1, O_2, and O_3 from the mixture signal $x_m(t)$, respectively. To perform separation, we predict a spectrogram mask $\mathcal{M}_n$ for each object. We use real-valued ratio masks and obtain the predicted magnitude spectrogram by soft masking the mixture spectrogram: $X_n = X^M \times \mathcal{M}_n$. Finally, we use the inverse short-time Fourier transform (ISTFT) [93] to reconstruct the waveform sound for each object source.

Going beyond video-level mix-and-separation, the key insight of our approach is to simultaneously enforce separation *within a single video* at the object level. This enables our method to learn object sound models even from multi-source training videos. Our new co-separation framework can capture the correlations between sound sources and is able to learn from noisy Web videos, as detailed next.

4.1.2 Co-Separation Framework

Next we present our Co-Separation training framework and our network architecture to perform separation.

Object Detection: Firstly, we train an object detector for a vocabulary of C objects. In general, this detector should cover any potential sound-making object categories that may appear in training videos. Our implementation uses the Faster R-CNN [180] object detector with a ResNet-101 [99] backbone trained with Open Images [132]. For each unlabeled training video, we use the pre-trained object detector to find objects in all video frames. Then, we gather all object detections across frames to obtain a video-level pool of objects. See [77] for details.

Audio-Visual Separator: We use the localized object regions to guide the source separation process. Fig. 4.2 illustrates our audio-visual separator network that performs audio-visual feature aggregation and source separation. A related design for multi-modal feature fusion is also used in [76, 155, 163] for audio spatialization and separation. However, unlike those models, our

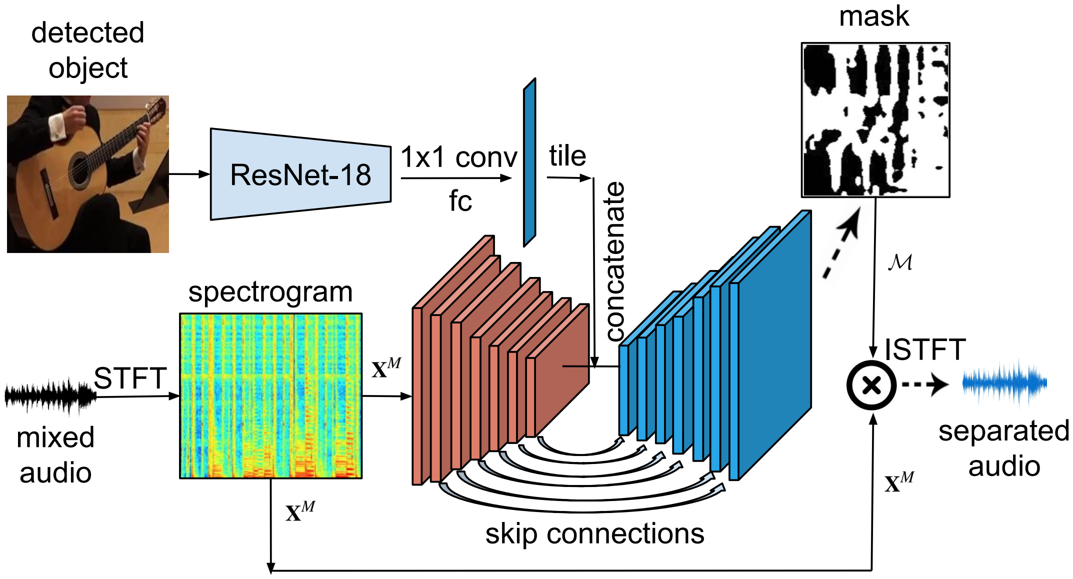


Figure 4.2: Our audio-visual separator network takes a mixed audio signal and a detected object from its accompanying video as input, and performs joint audio-visual analysis to separate the portion of sound responsible for the input object region.

separator network combines the visual features of a localized object region and the audio features of the mixed audio to predict a magnitude spectrogram mask for source separation.

The network takes a detected object region and the mixed audio signal as input, and separates the portion of the sound responsible for the object. We use a ResNet-18 network to extract visual features after the 4^{th} ResNet block with size $(H/32) \times (W/32) \times D$, where $H,\ W,\ D$ denote the frame and channel dimensions. We then pass the visual feature through a 1×1 convolution layer to reduce the channel dimension, and use a fully-connected layer to obtain an aggregated visual feature vector.

On the audio side, we adopt a U-Net [184] style network for its effectiveness in dense prediction tasks, similar to [76, 163, 258]. The network takes the magnitude spectrogram X^M as input and passes it through a series of convolution layers to extract an audio feature of dimension $(T/128) \times (F/128) \times D$. We replicate the visual feature vector $(T/128) \times (F/128)$ times, tile them to match the audio feature dimension, and then concatenate the audio and visual feature maps along the channel dimension. Then a series of up-convolutions are performed on the concatenated audio-visual feature map to generate a multiplicative spectrogram mask $\mathcal{M}$. We find spectrogram masks to work better than direct prediction of spectrograms or raw waveforms for source separation, confirming reports in [55, 76, 225]. The separated spectrogram for the input object is obtained by multiplying the mask and the spectrogram of the mixed audio:

$$X = X^M \times \mathcal{M}. \tag{4.1}$$

Finally, ISTFT is applied to the spectrogram to produce the separated real-time signal. See [77] for more details.

Co-Separation: Our proposed CO-SEPARATION framework first detects objects in a pair of videos, then mixes their audios at the video level, and finally separates the sounds for each detected object class. As shown in Fig. 4.3, for each video pair, we randomly sample a high confidence object window for each class detected in either video, and use the localized object region to guide audio source separation using the audio-visual separator network. For each object

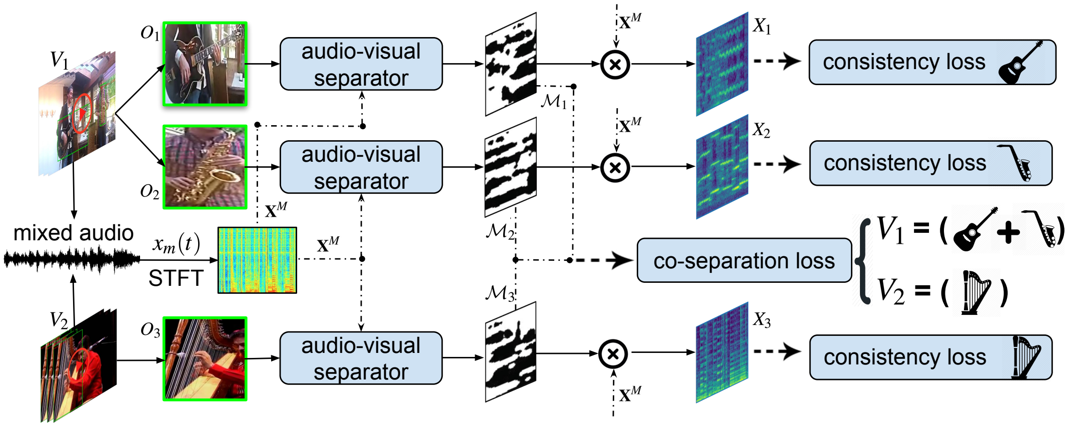


Figure 4.3: Co-separation training pipeline: our object-level co-separation framework first detects objects in a pair of videos, then mixes the audios at the video-level, and separates the sounds for each visual object. The network is trained by minimizing the combination of the co-separation and object-consistency losses defined in Sec. 4.1.2.

O_n, we predict a mask $\mathcal{M}_n$, and then generate the corresponding magnitude spectrogram.

Let $\mathcal{V}_1$ and $\mathcal{V}_2$ denote the set of objects for the two videos. We want to separate the sounds of their corresponding objects together from the audio mixture of V_1 and V_2. For each video, summing up the separated sounds of all objects should ideally reconstruct the audio signal for that video. Namely,

$$x_1(t) = \sum_{i}^{|\mathcal{V}_1|} s_i(t) \quad \text{and} \quad x_2(t) = \sum_{i}^{|\mathcal{V}_2|} s_i(t), \tag{4.2}$$

where $|\mathcal{V}_1|$ and $|\mathcal{V}_2|$ are the number of detected objects for V_1 and V_2. For simplicity of notation, we defer presenting how we handle background sounds (those unattributable to detected objects) until later in this section. Because we are operating in the frequency domain, the above relationship will only hold approximately due to phase interference. As an alternative, we approximate Eq. (4.2) by enforcing the following relationship on the predicted magnitude spectrograms:

$$X^{V_1} \approx \sum_{i}^{|\mathcal{V}_1|} X_i \quad \text{and} \quad X^{V_2} \approx \sum_{i}^{|\mathcal{V}_2|} X_i, \tag{4.3}$$

where X^{V_1} and X^{V_2} are the magnitude spectrograms for $x_1(t)$ and $x_2(t)$. Therefore, we minimize the following *co-separation loss* over the separated magnitude spectrograms:

$$L_{\textit{co-separation\_spectrogram}} = || \sum_{i=1}^{|\mathcal{V}_1|} X_i - X^{V_1} ||_1 + || \sum_{i=1}^{|\mathcal{V}_2|} X_i - X^{V_2} ||_1, \tag{4.4}$$

which approximates to minimizing the following loss function over their predicted ratio masks:

$$L_{\textit{co-separation\_mask}} = || \sum_{i=1}^{|\mathcal{V}_1|} \mathcal{M}_i - \mathcal{M}^{V_1} ||_1 + || \sum_{i=1}^{|\mathcal{V}_2|} \mathcal{M}_i - \mathcal{M}^{V_2} ||_1, \tag{4.5}$$

where $\mathcal{M}^{V_1}$ and $\mathcal{M}^{V_2}$ are the ground-truth spectrogram ratio masks for the two videos, respectively. Namely,

$$\mathcal{M}^{V_1} = \frac{X^{V_1}}{X^{V_1} + X^{V_2}} \quad \text{and} \quad \mathcal{M}^{V_2} = \frac{X^{V_2}}{X^{V_1} + X^{V_2}}. \tag{4.6}$$

In practice, we find that computing the loss over masks (vs. spectograms) makes the network easier to learn. We hypothesize that the sigmoid after the last layer of the audio-visual separator bounds the masks, making them more constrained and structured compared to spectrograms. In short, the proposed co-separation loss provides supervision to the network to only separate the audio portion responsible for the input visual object, so that the corresponding audios for each of the pair of input videos can be reconstructed.

In addition to the co-separation loss that enforces separation, we also introduce an *object-consistency loss* for each predicted audio spectrogram. The intuition is that if the sources are well-separated, the predicted "category" of the separated spectrogram should be consistent with the category of the visual object that initially guides its separation. Specifically, for the predicted spectrogram of each object, we introduce another ResNet-18 *audio* classifier that targets the weak labels of the input visual objects. We use the following cross-entropy loss:

$$L_{object\text{-}consistency} = \frac{1}{|\mathcal{V}_1| + |\mathcal{V}_2|} \sum_{i=1}^{|\mathcal{V}_1|+|\mathcal{V}_2|} \sum_{c=1}^{C} -y_{i,c} \log(p_{i,c}), \tag{4.7}$$

where C is the number of classes, $y_{i,c}$ is a binary indicator on whether class label c is the correct classification for predicted spectrogram X_i, and $p_{i,c}$ is the predicted probability for class c.

Not all sounds in a video will be attributable to a visually detected object. To account for ambient sounds, off-screen sounds, and noise, we incorporate a $C + 1^{st}$ "adaptable" audio class, as follows. During training, we pair each video with a visual scene feature in addition to the detected objects from the pre-trained object detector. Then an additional mask $\mathcal{M}_{\text{adapt}}$ responsible for the scene context is also predicted in Eq. (4.5) for both V_1 and V_2 to be optimized jointly. This step arms the network with the flexibility to assign noise or unrelated sounds to this "adaptable" class, leading to cleaner separation for sounds of the detected visual objects. These adaptable objects (ideally ambient sounds and noise) are collectively designated as having the "extra" $C + 1^{st}$ audio label. The separated spectrograms for these adaptable objects are also trained to match their category label by the object-consistency loss in Eq. (4.7).

Putting it all together, during training the network needs to discover separations for the multi-source videos that 1) minimize the co-separation loss, such that the two source videos' object sounds reassemble to produce their original video-level audio tracks, respectively, while also 2) minimizing the object consistency loss, such that separated sounds for *any* instances of the same visual object are reliably identifiable as that sound. We stress that our model achieves the latter *without* any pre-trained audio model and *without* any single-source audio examples for the object class. The object consistency loss only knows that same-object sounds should be similar after training the network—not what any given object is expected to sound like.

4.1.3 Training and Inference

We minimize the following combined loss function and train our network end to end:

$$L = L_{\textit{co-separation\_mask}} + \lambda L_{\textit{object-consistency}}, \tag{4.8}$$

where λ is the weight for the object-consistency loss.

We use per-pixel $L1$ loss for the co-separation loss, and weight the gradients by the magnitude of the spectrogram of the mixed audio. The network uses the weighted gradients to perform back-propagation, thereby emphasizing predictions on more informative parts of the spectrogram.

During testing, our model takes a *single* realistic multi-source video to perform source separation. Similarly, we first detect objects in the video frames by using the pre-trained object detector. For each detected object class, we use the most confident object region(s) as the visual input to separate the portion of the sound responsible for this object category from its accompanying audio. We use a sliding window approach to process videos segment by segment with a small hop size, and average the audio predictions on all overlapping parts.

4.2 Experiments

We now validate our approach for audio-visual source separation and compare to existing methods.

4.2.1 Datasets

We evaluate on four datasets: MUSIC, AudioSet-Unlabeled, AudioSet-SingleSource, and AV-Bench. The last three are the same datasets used in Chapter 3, and the details can be found in Sec. 3.2.1.

MUSIC: This MIT dataset contains YouTube videos crawled with keyword queries [258]. It contains 685 untrimmed videos of musical solos and duets, with 536 solo videos and 149 duet videos. The dataset is relatively clean and collected for the purpose of training audio-visual source separation models. It includes 11 instrument categories: accordion, acoustic guitar, cello, clarinet, erhu, flute, saxophone, trumpet, tuba, violin and xylophone. Following the authors' public dataset file of video IDs, we hold out the first/second video in each category as validation/test data, and the rest as training data. We split all videos into 10s clips during both training and testing, for a total of 8,928/259/269 train/val/test clips, respectively.

4.2.2 Implementation Details

Our Co-Separation deep network is implemented in PyTorch. For all experiments, we sub-sample the audio at 11kHz, and the input audio sample is approximately 6s long. STFT is computed using a Hann window size of 1022 and a hop length of 256, producing a 512×256 Time-Frequency audio representation. The spectrogram is then re-sampled on a log-frequency scale to obtain a $T \times F$ magnitude spectrogram of $T = 256, F = 256$. The settings are the same as [258] for fair comparison.

Our object detector is trained on images of $C = 15$ object categories from the Open Images dataset [132]. We filter out low confidence object detections for each video, and keep the top two[3] detected categories. See [77] for details. During co-separation training, we randomly sample 64 pairs of videos for each batch. We sample a confident object detection for each class

[3] This agrees with the number of objects detected by our pre-trained detector in most training video. We did not try any other values.

as its input visual object, paired with a random scene image sampled from the ADE dataset [262] as the adaptable object. The object window is resized to 256×256, and a randomly cropped 224×224 region is used as the input to the network. We use horizontal flipping, color and intensity jittering as data augmentation. λ is set to 0.05 in Eq. (4.8). The network is trained using an Adam optimizer with weight decay 1×10^{-4} with the starting learning rate set to 1×10^{-4}. We use a smaller starting learning rate of 1×10^{-5} for the ResNet-18 visual feature extractor because it is pre-trained on ImageNet.

4.2.3 Quantitative Results on Source Separation

We compare to the following baselines:

- **Sound-of-Pixels [258]:** We use the authors' publicly available code[4] to train 1-frame based models with ratio masks for fair comparison. Default settings are used for other hyperparameters.
- **AV-Mix-and-Separate:** A "mix-and-separate" baseline using the same audio-visual separation network as our model to do *video*-level separation. We use multi-label hinge loss to enforce video-level consistency, i.e., the class of each separated spectrogram should agree with the objects present in that training video.
- **AV-MIML [75]:** An existing audio-visual source separation method that uses audio bases learned from unlabeled videos to supervise an

[4] `https://github.com/hangzhaomit/Sound-of-Pixels`

NMF separation process. The audio bases are learned from a deep multi-instance multi-label (MIML) learning network. We use the results reported in [75] for AudioSet and AV-Bench; the authors do not report results in SDR and do not report results for MUSIC.

- **NMF-MFCC [201]:** An off-the-shelf audio-only method that performs NMF based source separation using Mel frequency cepstrum coefficients (MFCC). This non-learned baseline is a good representation of a well established pipeline for audio-only source separation [94, 111, 115, 222].
- **AV-Loc [171], JIVE [145], Sparse CCA [125]:** We use results reported in [75] to compare to these methods for the audio denoising benchmark AV-Bench.

We use the widely used mir eval library [175] to evaluate the source separation and report the standard metrics: Signal-to-Distortion Ration (SDR), Signal-to-Interference Ratio (SIR), and Signal-to-Artifact Ratio (SAR).

Separation Results: Tables 4.1 and 4.2 show the results for the MUSIC and AudioSet datasets, respectively.

Table 4.1 presents results on MUSIC as a function of the training source: single-source videos (solo) or multi-source videos (solo + duet). Our method consistently outperforms all baselines in separation accuracy, as captured by the SDR and SIR metrics.[5] While the SoP method [258] works well when

[5] Note that SAR measures the *artifacts* present in the separated signal, but not the

	Single-Source			Multi-Source		
	SDR	SIR	SAR	SDR	SIR	SAR
Sound-of-Pixels [258]	7.30	11.9	**11.9**	6.05	9.81	**12.4**
AV-Mix-and-Separate	3.16	6.74	8.89	3.23	7.01	9.14
NMF-MFCC [201]	0.92	5.68	6.84	0.92	5.68	6.84
Co-Separation (Ours)	**7.38**	**13.7**	10.8	**7.64**	**13.8**	11.3

Table 4.1: Average audio source separation results on a held out MUSIC test set. We show the performance of our method and the baselines when training on only single-source videos (solo) and multi-source videos (solo + duet). Note that NMF-MFCC is non-learned, so its results do not vary across training sets. Higher is better for all metrics. Note that SDR and SIR capture separation accuracy; SAR captures only the absence of artifacts (and hence can be high even if separation is poor). Standard error is approximately 0.2 for all metrics.

training only on solo videos, it fails to make use of the additional duets, and its performance degrades when training on the multi-source videos. In contrast, our method actually *improves* when trained on a combination of solos and multi-source duets, achieving its best performance. This experiment highlights precisely the limitation of the mix-and-separate training paradigm when presented with multi-source training videos, and it demonstrates that our co-separation idea can successfully overcome that limitation.

Our method also outperforms all baselines, including [258], when training on solos. Our better accuracy versus the AV-Mix-and-Separate baseline and [258] shows that our object-level co-separation idea is essential. The NMF-MFCC baseline can only return ungrounded separated signals. Therefore, we

separation accuracy. So, a less well-separated signal can achieve high(er) SAR values. In fact, naively copying the original input twice (i.e., doing no separation) results in SAR $\approx$ 80 in our setting.

	SDR	SIR	SAR
Sound-of-Pixels [258]	1.66	3.58	11.5
AV-MIML [75]	1.83	-	-
AV-Mix-and-Separate	1.68	3.30	12.2
NMF-MFCC [201]	0.25	4.19	5.78
Co-Separation (Ours)	**4.26**	**7.07**	**13.0**

Table 4.2: Average separation results on AudioSet test set. Standard error is approximately 0.3.

	Sound-of-Pixels [258]			Co-Separation (Ours)		
	SDR	SIR	SAR	SDR	SIR	SAR
Violin/Saxophone	1.52	1.48	12.9	8.10	11.7	11.2
Violin/Guitar	6.95	11.2	15.8	10.6	16.7	12.3
Saxophone/Guitar	0.57	0.90	16.5	5.08	7.90	9.34

Table 4.3: Toy experiment to demonstrate learning to separate sounds for objects never heard individually during training.

evaluate both possible matchings and take its best results (to the baseline's advantage). Our method still achieves large gains, and we also have the benefit of matching the separated sounds to semantically meaningful visual objects in the video.

Table 4.2 shows the results when training on AudioSet-Unlabeled and testing on mixes of AudioSet-SingleSource. Our method outperforms all prior methods and the baselines by a large margin on this challenging dataset. It demonstrates that our framework can better learn from the noisy and less curated "in the wild" videos of AudioSet, which contains many multi-source videos.

Next we devise an experiment to test explicitly how well our method can learn to separate sound for objects it has not observed *individually* during training. We train our model and the best baseline [258] on the following four categories: violin solo, saxophone solo, violin+guitar duet, and violin+saxophone duet, and test by randomly mixing and separating violin, saxophone, and guitar test solo clips. Table 4.3 shows the results. We can see that although our system is not trained on any guitar solos, it can learn better from multi-source videos that contain guitar and other sounds. Our method consistently performs well on all three combinations, while [258] performs well only on the violin+guitar mixture. We hypothesize the reason is that it can learn by mixing the large quantity of violin solos and the guitar solo moments *within* the duets to perform separation, but it fails to disentangle other sound source correlations. Our method scores worse in terms of SAR, which again measures artifacts, but not separation quality.

Denoising Results: As a side product of our audio-visual source separation system, we can also use our model to perform visually-guided audio denoising. As mentioned in Sec. 4.1.3, we use an additional scene image to capture ambient/unseen sounds and noise. Therefore, given a test video with noise, we can use the top detected visual object in the video to guide our system to separate out the noise.

Table 4.4 shows the results on AV-Bench [75,171]. Though our method learns only from unlabeled video and does not explicitly model the low-rank nature of noise as in [171], we obtain state-of-the-art performance on 2 of the 3 videos. The method of [171] uses motion in manually segmented regions,

	Wooden Horse	Violin Yanni	Guitar Solo
Sparse CCA [125]	4.36	5.30	5.71
JIVE [145]	4.54	4.43	2.64
AV-Loc [171]	8.82	5.90	**14.1**
AV-MIML [75]	12.3	7.88	11.4
Ours	**14.5**	**8.53**	11.9

Table 4.4: Visually-assisted audio denoising on AV-Bench, in terms of NSDR (in dB, higher is better).

which may help on Guitar Solo, where the hand's motion strongly correlates with the sound.

4.2.4 Qualitative Results

Audio-Visual Separation Video Examples: Our video results[6] show qualitative separation results. We use our system to discover and separate object sounds for realistic multi-source videos. They lack ground truth, but the results can be manually inspected for quality.

Learned Audio Embedding: To *visualize* that our CO-SEPARATION network has indeed learned to separate sounds of visual objects, Fig. 4.4 displays a t-SNE [149] embedding of the discovered sounds for various input objects in 20K AudioSet clips. We use the features extracted at the last layer of the ResNet-18 audio classifier as the audio representation for the separated spectrograms. The sounds our method learned from multi-source videos tend to cluster by object category, demonstrating that the separator discovers sounds

[6] http://vision.cs.utexas.edu/projects/coseparation/

characteristic of the corresponding objects.

Using Discovered Sounds to Detect Objects: Finally, we use our trained audio-visual source separation network for *visual object discovery* using 912 noisy unseen videos from AudioSet. Given the pool of videos, we generate object region proposals using Selective Search [211]. Then we pass these region proposals to our network together with the audio of its accompanying video, and retrieve the *visual* proposals that achieve the highest *audio* classification scores according to our object consistency loss.

Fig. 4.5 shows the top retrieved proposals for several categories after removing duplicates from the same video. We can see that our method has learned a good mapping between the visual and audio modalities; the best visual object proposals usually best activate the audio classifier. The last column shows failure cases where the wrong object is detected with high confidence. They usually come from objects of similar texture or shape, like the stripes on the man's t-shirt and the shadow of the harp.

4.3 Conclusions

In this chapter, I presented an object-level audio-visual source separation framework that associates localized object regions in videos to their characteristic sounds. Similar to my first approach described in Chapter 3, we also use visual cues to guide the separation process. While the previous NMF-based approach only needs weak supervision from an image classifier to disen-

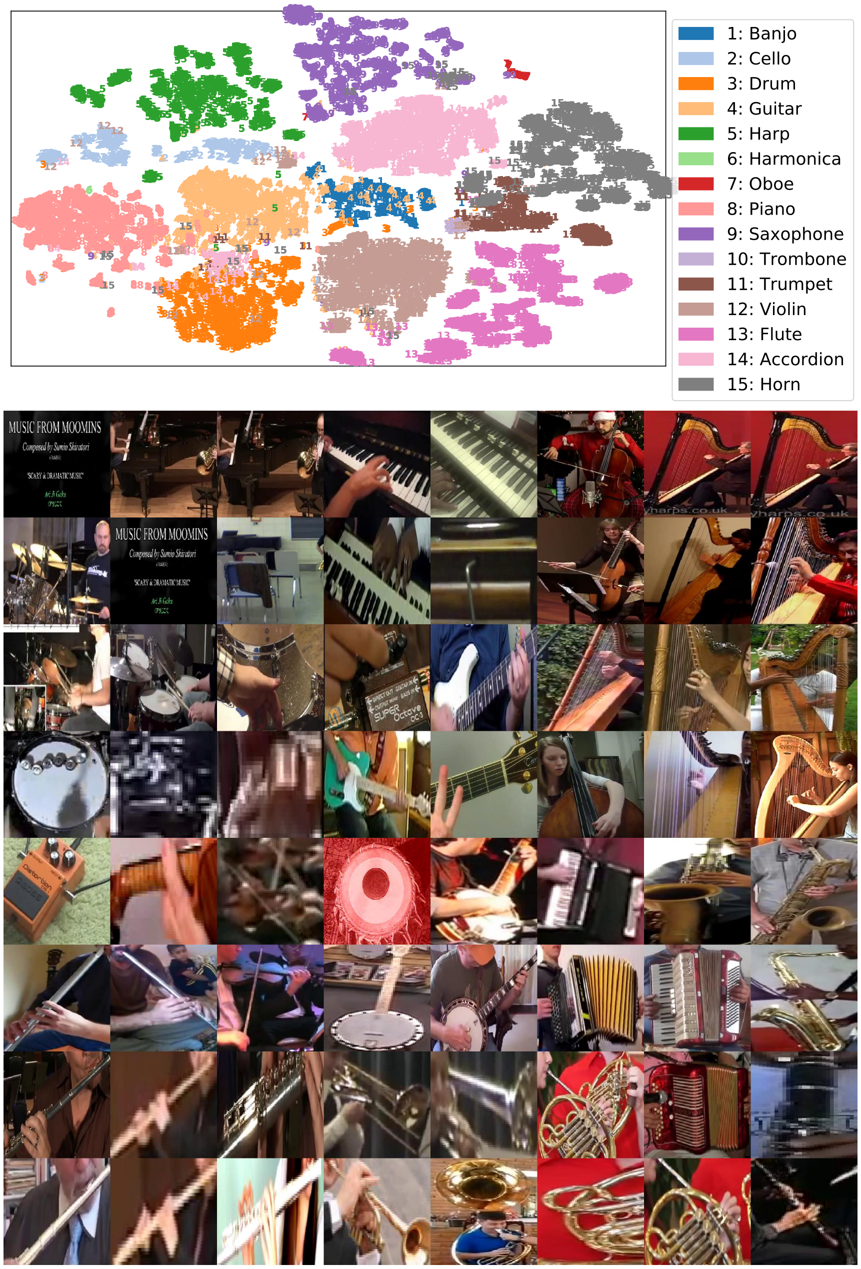


Figure 4.4: Embedding of separated sounds in AudioSet visualized with t-SNE in two ways: (top) categories are color-coded, and (bottom) visual objects are shown at their sound's embedding.

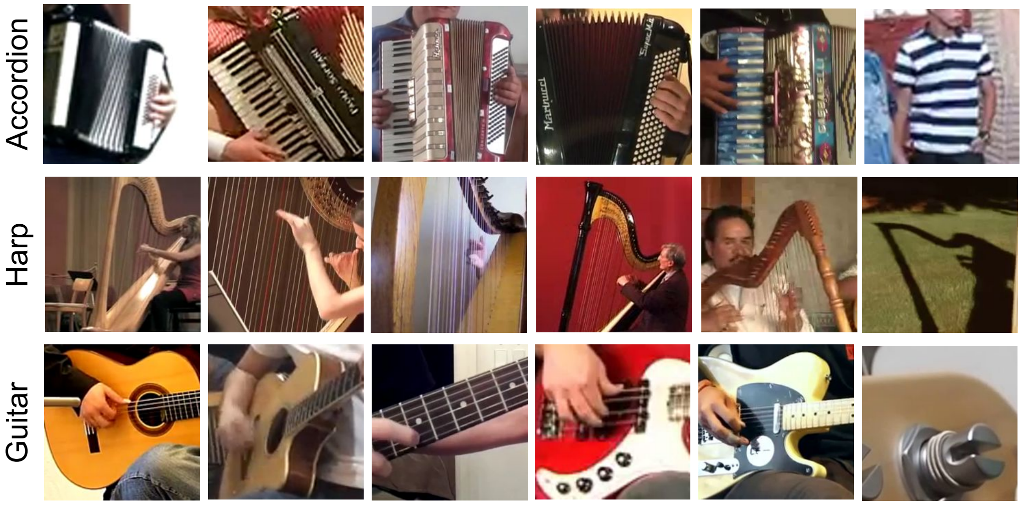


Figure 4.5: Top object proposals according to our discovered *audio* classifier. Last column shows typical failure cases.

tangle frequency bases at the categorty level, our end-to-end Co-Separation approach leverages noisy object detections as supervision to learn from large-scale unlabeled videos. We achieve state-of-the-art results on visually-guided audio source separation and audio denoising.

Although my new attempt addresses some of the limitations of our previous approach, it is of course not perfect. Our method can fail when the audio characteristics of detected objects are too similar or objects are incorrectly detected. Though the pre-trained object detector can recognize a wide array of objects, we are nonetheless constrained by its breadth. Furthermore, not all objects make sounds and not all sounds are within the camera's view. Motion analysis may also be valuable in order to perform instance-level source separation (*e.g.*, separate sounds for multiple human speakers). Our results above suggest that learning can be robust to such factors, yet it will be important to explicitly model them. In the next chapter, I take one further step to extend the co-separation framework to audio-visual speech separation, where we leverage the complementary cues between lip motion and facial appearance to guide the separation process.

Chapter 5

Audio-Visual Speech Separation with Cross-Modal Consistency

[1]In the previous chapter, I proposed a co-separation approach for audio-visual source separation that permits both end-to-end training and learning object-level sounds from unlabeled multi-source videos. In this chapter, I further extend the co-separation framework to incorporate both lip motion and facial appearance of speakers for the specific task of audio-visual speech separation. This work is going to be published at CVPR 2021 [78].

Human speech is rarely observed in a vacuum. Amidst the noisy din of a restaurant, we concentrate to parse the words of our dining partner; watching a heated presidential debate, we disentangle the words of the candidates as they talk over one another; on a Zoom call we listen to a colleague while our children chatter and play a few yards away. Presented with such corrupted and entangled sounds, the human perceptual system draws heavily on visual information to reduce ambiguities in the audio [176] and modulate attention on an active speaker in a busy environment [90]. Automating this process of *speech*

[1] The work in this chapter was supervised by Prof. Kristen Grauman. It will be published in: "VisualVoice: Audio-Visual Speech Separation with Cross-Modal Consistency". Ruohan Gao and Kristen Grauman. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Virtual, June 2021.

separation has many valuable applications, including assistive technology for the hearing impaired, superhuman hearing in a wearable augmented reality device, or better transcription of spoken content in noisy in-the-wild Internet videos.

While early work in automatic speech separation relied solely on the audio stream [48, 159, 251], recent work explores ways to leverage its close connections to the visual stream as well [3, 35, 55, 69, 163]. By analyzing the facial motion in concert with the emitted speech, these methods steer the audio separation module towards the relevant portions of the sound that ought to be separated out from the full audio track. For example, the mouth articulates in different shapes consistent with the phonemes produced in the audio, making it possible to mask a spectrogram for the target human speaker based on audio-visual (AV) consistency. However, solely relying on lip movements can fail when lip motion becomes unreliable, *e.g.*, the mouth region is occluded by the microphone or the speaker turns their head away.

While AV synchronization cues are powerful, we observe that the consistency between the speaker's facial appearance and their voice is also revealing for speech separation. Intuitively, attributes like gender, age, nationality, and body weight are often visible in the face *and* give a prior for what sound qualities (tone, pitch, timbre, basis of articulation) to listen for when trying to separate that person's speech from interfering sounds. For example, female speakers often register in higher frequencies, a heavier person may exhibit a wider range of sound intensities [18], and an American speaker may sound

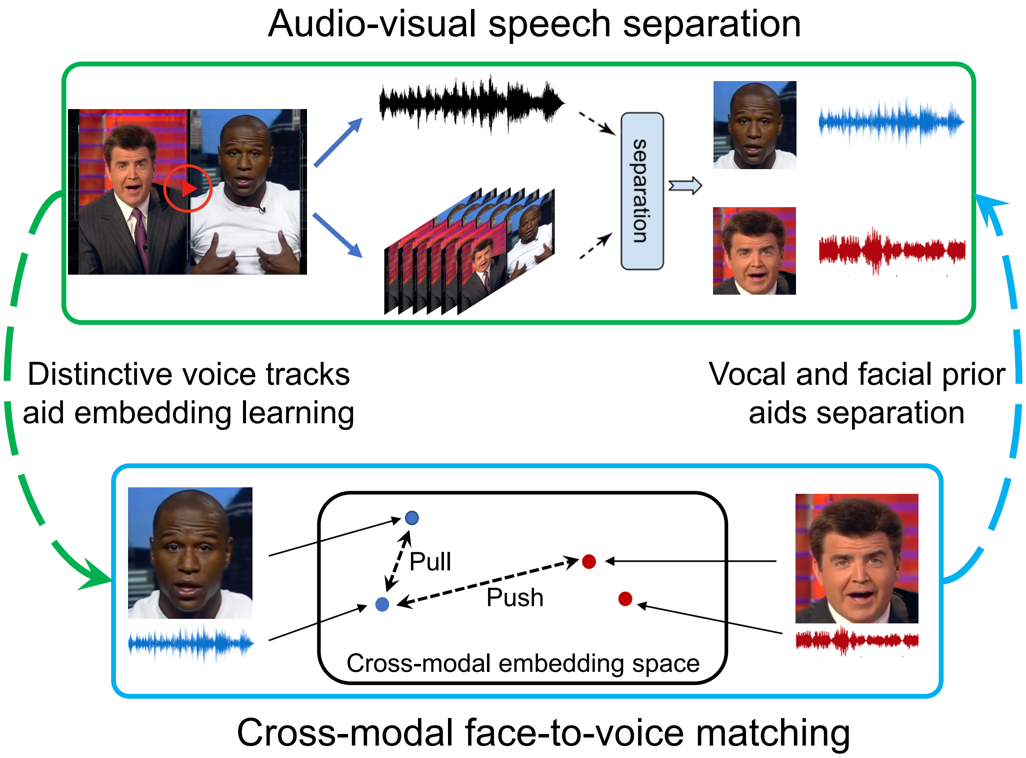

Audio-visual speech separation
separation
Distinctive voice tracks aid embedding learning
Vocal and facial prior aids separation
Pull
Push
Cross-modal embedding space
Cross-modal face-to-voice matching

Figure 5.1: The two tasks cross-modal face-to-voice matching and speech separation are mutually beneficial. The embeddings serve as a prior for the voice characteristics that enhances speech separation; the cleaner separated speech in turn produces more distinctive audio embeddings.

more nasal. The face-voice association, supported by cognitive science studies [22], is today often leveraged for speaker *identification* given the recording of a single speaker [126, 157, 158, 234]. In contrast, the speech *separation* problem demands discovering a cross-modal association in the presence of multiple overlapping sounds.

Our key insight is that these two tasks—cross-modal face-to-voice matching and speech separation—are mutually beneficial. The cleaner the sound separation, the more accurately an embedding can link the voice to a face; the better that embedding, the more distinctive is the prior for the voice characteristics which will in turn aid separation. We thus aim to “visualize” the voice of a person based on how they look to better separate that voice’s sound. See Figure 5.1.

To this end, we propose VisualVoice, a multi-task learning framework to jointly learn audio-visual speech separation together with cross-modal speaker embeddings. We introduce a speech separation network that takes video of a human speaker talking in the presence of other sounds (speech or otherwise) and returns the isolated sound track for just their speech. Our network relies on facial appearance, lip motion, and vocal audio to solve the separation task, augmenting the conventional “mix-and-separate” paradigm for audio-visual separation to account for a cross-modal contrastive loss requiring the separated voice to agree with the face. Notably, our approach requires no identity labels and no enrollment of speakers, meaning we can train and test with fully unlabeled video.

The main contributions for this part of my thesis are as follows. Firstly, we introduce an audio-visual speech separation framework that leverages complementary cues from facial motion and cross-modal face-voice attributes. Secondly, we devise a novel multi-task framework that successfully learns both separation and cross-modal embeddings in concert. Finally, through experiments on five benchmark datasets, we demonstrate state-of-the-art results for audio-visual speech separation and enhancement in challenging scenarios. The embedding learned by our model additionally improves the state of the art for unsupervised cross-modal speaker verification, emphasizing the yet-unexplored synergy of the two tasks.

In Sec 5.1, I describe our VisualVoice approach for learning audio-visual speech separation. Then I present experimental results in Sec 5.2.

5.1 Approach

Our goal is to perform audio-visual speech separation. We first formally define our problem (Sec. 5.1.1); then we present our audio-visual speech separation network that leverages both the lip motion and cross-modal facial attributes to guide the separation process (Sec. 5.1.2); next we introduce how we learn audio-visual speech separation and cross-modal face-voice embeddings in a multi-task learning framework (Sec. 5.1.3); finally we present our training criteria and inference procedures (Sec. 5.1.4).

5.1.1 Problem Formulation

Given a video V with multiple speakers, we denote $x(t) = \sum_{k=1}^{K} s_k(t)$ as the observed single-channel linear mixture of the voices for these K speakers, where $s_k(t)$ are time-discrete signals responsible for each speaker. Our goal in audio-visual speech separation is to separate the sound $s_k(t)$ for each speaker from $x(t)$ by leveraging the visual cues in the video. For simplicity we describe the sources as speakers throughout, but note that the mixed sound can be something other than speech, as we will demonstrate in results with speech enhancement evaluation.

To generate training examples, we follow the commonly adopted "mix-and-separate" paradigm [3, 55, 77, 163, 258] and generate synthetic audio mixtures by mixing human speech segments. These speech segments are accompanied by the face tracks[2] of the corresponding speakers, which are extracted automatically from "in the wild" videos with background chatter, laughter, pose variation, *etc.*.

Suppose we have two speech segments $s_{\mathcal{A}_1}(t)$, $s_{\mathcal{A}_2}(t)$ from video $V_{\mathcal{A}}$ for speaker $\mathcal{A}$, and $s_{\mathcal{B}}(t)$ from video $V_{\mathcal{B}}$ for speaker $\mathcal{B}$.[3] Let $F_{\mathcal{A}_1}, F_{\mathcal{A}_2}, F_{\mathcal{B}}$ denote the face tracks associated with the speech segments $s_{\mathcal{A}_1}(t)$, $s_{\mathcal{A}_2}(t)$, $s_{\mathcal{B}}(t)$, respectively. We create two mixture signals $x_1(t)$ and $x_2(t)$:

$$x_1(t) = s_{\mathcal{A}_1}(t) + s_{\mathcal{B}}(t), \quad x_2(t) = s_{\mathcal{A}_2}(t) + s_{\mathcal{B}}(t). \tag{5.1}$$

The mixture speech signals are then transformed into complex audio spectrograms X_1 and X_2.

Our training objective is to jointly separate $s_{\mathcal{A}_1}(t)$, $s_{\mathcal{A}_2}(t)$ and $s_{\mathcal{B}}(t)$ for face tracks $F_{\mathcal{A}_1}$, $F_{\mathcal{A}_2}$ and $F_{\mathcal{B}}$ from the two mixed signals $x_1(t)$ and $x_2(t)$. In Sec. 5.1.3 we present a speaker consistency loss that regularizes the separation process with the two mixtures. To perform separation, we predict complex ideal ratio masks (cIRM) [236] $M_{\mathcal{A}_1}$, $M_{\mathcal{A}_2}$, $M_{\mathcal{B}_1}$ and $M_{\mathcal{B}_2}$ to separate clean

[2] A face track is a tracklet of face detections, which is automatically obtained through an off-the-shelf face detector.

[3] No identity labels are used during training. $s_{\mathcal{A}_1}(t)$ and $s_{\mathcal{A}_2}(t)$ come from the same training video, so we assume they share the same identity.

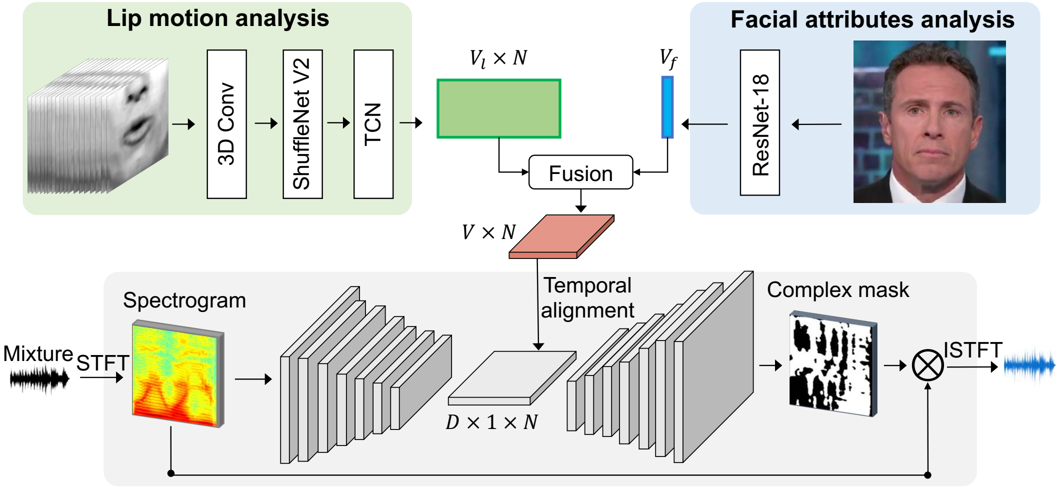


Figure 5.2: Our audio-visual speech separator network takes a mixed speech signal as input and analyses the lip motion and facial attributes in the face track to separate the portion of sound responsible for the corresponding speaker.

speech for the corresponding speakers from X_1 and X_2, respectively. Note that we separately predict a mask for speaker $\mathcal{B}$ from each mixture. The predicted spectrograms for the separated speech signals can be obtained by complex masking the mixture spectrograms:

$$S_{\mathcal{A}_i} = X_i * M_{\mathcal{A}_i}, \quad S_{\mathcal{B}_i} = X_i * M_{\mathcal{B}_i}, \quad i \in \{1, 2\}, \tag{5.2}$$

where $*$ indicates complex multiplication. Finally, using the inverse short-time Fourier transform (ISTFT) [93], we reconstruct the separated speech signals.

5.1.2 Audio-Visual Speech Separator Network

Next we present the architecture of our audio-visual speech separator network, which leverages the complementary visual cues of both lip motion and cross-modal facial attributes to guide the separation process. Later in Sec. 5.1.3 we will introduce our multi-task learning framework to learn both audio-visual speech separation and cross-modal face-voice embeddings, and describe how we jointly separate speech from $x_1(t)$ and $x_2(t)$.

We use the visual cues in the face track to guide the speech separation for each speaker. The visual stream of our network consists of two parts: a lip motion analysis network and a facial attributes analysis network (Figure 5.2).

Following the state-of-the-art in lip reading [148, 151], the lip motion analysis network takes N mouth regions of interest (ROIs)[4] as input and it consists of a 3D convolutional layer followed by a ShuffleNet v2 [147] network to extract a time-indexed sequence of feature vectors. They are then processed by a temporal convolutional network (TCN) to extract the final lip motion feature map of dimension $V_l \times N$.

For the facial attributes analysis network, we use a ResNet-18 [99] network that takes a single face image randomly sampled from the face track as input to extract a face embedding $\mathbf{i}$ of dimension V_f that encodes the facial attributes of the speaker. We replicate the facial attributes feature along the time dimension to concatenate with the lip motion feature map and obtain a final visual feature of dimension $V \times N$, where $V = V_l + V_f$.

The facial attributes feature represents an identity code whose role is to identify the space of expected frequencies or other audio properties for the

[4] The ROIs are derived from the face track through facial landmark detection and alignment to a mean reference face. See [78] for details.

speaker's voice, while the role of the lip motion is to isolate the articulated speech specific to that segment. Together they provide complementary visual cues to guide the speech separation process.

On the audio side, we use a U-Net [184] style network tailored to audio-visual speech separation. It consists of an encoder and a decoder network. The input to the encoder is the complex spectrogram of the mixture signal of dimension $2 \times F \times T$, where F, T are the frequency and time dimensions of the spectrogram. Each time-frequency bin contains the real and imaginary part of the corresponding complex spectrogram value. The input is passed through a series of convolutional layers with frequency pooling layers in between, which reduce the frequency dimension while preserving the time dimension. In the end we obtain an audio feature map of dimension $D \times 1 \times N$, where D is the channel dimension.

We then concatenate the visual and audio features along the channel dimension to generate an audio-visual feature map of dimension $(V + D) \times 1 \times N$. The decoder takes the concatenated audio-visual feature as input. It has symmetric structure with respect to the encoder, where the convolutional layer is replaced by an upconvolutional layer and the frequency pooling layer is replaced by a frequency upsampling layer. Finally, we use a `Tanh` layer followed by a `Scaling` operation on the output feature map to predict a bounded complex mask of the same dimension as the input spectrogram for the speaker.

We build an audio-visual feature map for each speaker in the mixture to separate their respective voices. Alternatively, to build a model tailored to two-speaker speech separation, we concatenate the visual features of both speakers in the mixture with the audio feature to generate an audio-visual feature map of dimension $(2V + D) \times 1 \times N$ and simultaneously separate their voices together. This leads to slightly better performance due to the additional context information of the other speaker provided (see [78] for a comparison), while a model trained with the visual feature of a single speaker can be used in the general case where the number of speakers is unknown at inference time. We use the applicable case in experiments. See [78] for the network details.

5.1.3 Cross-Modal Matching for Separation

Next we introduce our multi-task learning framework that simultaneously learns AV speech separation and cross-modal face-voice embeddings (see Fig. 5.3). The framework includes several novel loss functions to regularize learning.

Mask Prediction Loss: As shown in Fig. 5.4, we predict complex masks $M_{\mathcal{A}_1}$, $M_{\mathcal{A}_2}$, $M_{\mathcal{B}_1}$, $M_{\mathcal{B}_2}$ to separate speech for the corresponding speakers from X_1 and X_2, respectively. We compute the following loss on the predicted complex masks:

$$L_{\textit{mask-prediction}} = \sum_{i \in \{\mathcal{A}_1, \mathcal{A}_2, \mathcal{B}_1, \mathcal{B}_2\}} \|M_i - \mathcal{M}_i\|_2, \tag{5.3}$$

where $\mathcal{M}_i$ denotes the ground-truth complex masks, which are obtained by taking the complex ratio of the spectrogram of the clean speech to the corresponding mixture speech spectrogram. This loss provides the main supervision

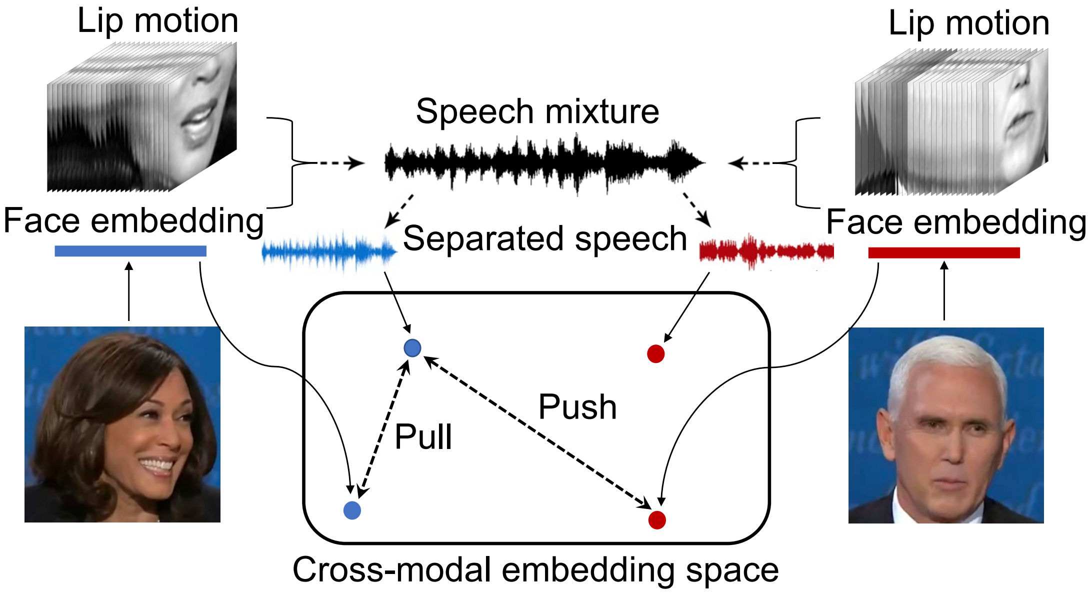


Figure 5.3: We propose a multi-task learning framework to jointly learn audio-visual speech separation and cross-modal face-voice embeddings. Our approach leverages the complementary cues between lip movements and cross-modal speaker embeddings for speech separation. The embeddings serve as a prior for the voice characteristics that enhances speech separation; the cleaner separated speech in turn produces more distinctive audio embeddings.

to enforce the separation of clean speech.

Cross-modal Matching Loss: To capture the desired cross-modal facial attributes to guide the separation process, we jointly learn cross-modal face-voice embeddings. The idea aligns with prior work on cross-modal matching [36, 37, 126, 157, 158, 234], but here our goal is audio separation—not person identification—and rather than a single-source input, in our case the audio explicitly contains *multiple* sound sources.

Similar to the facial attributes analysis network, we use a ResNet-18

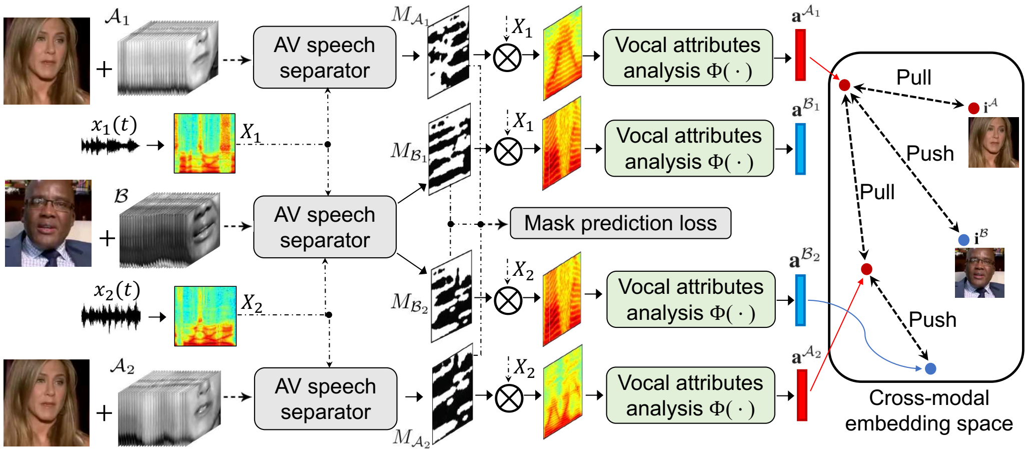

$\mathcal{A}_1$
$x_1(t)$
X_1
$\mathcal{B}$
$x_2(t)$
X_2
$\mathcal{A}_2$
AV speech separator
AV speech separator
AV speech separator
$M_{\mathcal{A}_1}$
$M_{\mathcal{B}_1}$
$M_{\mathcal{B}_2}$
$M_{\mathcal{A}_2}$
X_1
X_1
X_2
X_2
Mask prediction loss
Vocal attributes analysis $\Phi(\cdot)$
Vocal attributes analysis $\Phi(\cdot)$
Vocal attributes analysis $\Phi(\cdot)$
Vocal attributes analysis $\Phi(\cdot)$
$\mathbf{a}^{\mathcal{A}_1}$
$\mathbf{a}^{\mathcal{B}_1}$
$\mathbf{a}^{\mathcal{B}_2}$
$\mathbf{a}^{\mathcal{A}_2}$
Pull
$\mathbf{i}^{\mathcal{A}}$
Push
Pull
$\mathbf{i}^{\mathcal{B}}$
Push
Cross-modal embedding space

Figure 5.4: Our multi-task learning framework that jointly learns audio-visual speech separation and cross-modal face-voice embeddings. The network is trained by minimizing the combination of the mask prediction loss, the cross-modal matching loss, and the speaker consistency loss defined in Sec. 5.1.3.

network as the vocal attributes analysis network $\Phi(\cdot)$. We extract audio embeddings $\mathbf{a}^{\mathcal{A}_1}$, $\mathbf{a}^{\mathcal{A}_2}$, $\mathbf{a}^{\mathcal{B}_1}$, $\mathbf{a}^{\mathcal{B}_2}$ for each separated speech spectrogram:

$$\mathbf{a}^{\mathcal{A}_i} = \Phi(X_i * M_{\mathcal{A}_i}),\ \mathbf{a}^{\mathcal{B}_i} = \Phi(X_i * M_{\mathcal{B}_i}),\ i \in \{1, 2\}. \tag{5.4}$$

Let $\mathbf{i}^{\mathcal{A}}$ and $\mathbf{i}^{\mathcal{B}}$ denote the face image embeddings extracted from the facial attributes analysis network for speakers $\mathcal{A}$ and $\mathcal{B}$, respectively. We use the following triplet loss:

$$L_t(\mathbf{a}, \mathbf{i}^+, \mathbf{i}^-) = \max\{0, D(\mathbf{a}, \mathbf{i}^+) - D(\mathbf{a}, \mathbf{i}^-) + \mathtt{m}\}, \tag{5.5}$$

where $D(\mathbf{a}, \mathbf{i})$ is the cosine distance of the speech embedding and the face image embedding, and $\mathtt{m}$ represents the margin between the two distances.

The cross-modal matching loss is defined as follows:

$$\begin{aligned} L_{\textit{cross-modal}} = L_t(\mathbf{a}^{\mathcal{A}_1}, \mathbf{i}^{\mathcal{A}}, \mathbf{i}^{\mathcal{B}}) + L_t(\mathbf{a}^{\mathcal{A}_2}, \mathbf{i}^{\mathcal{A}}, \mathbf{i}^{\mathcal{B}}) \\ + L_t(\mathbf{a}^{\mathcal{B}_1}, \mathbf{i}^{\mathcal{B}}, \mathbf{i}^{\mathcal{A}}) + L_t(\mathbf{a}^{\mathcal{B}_2}, \mathbf{i}^{\mathcal{B}}, \mathbf{i}^{\mathcal{A}}). \end{aligned} \tag{5.6}$$

This loss forces the network to learn cross-modal face-voice embeddings such that the distance between the embedding of the separated speech and the face embedding for the corresponding speaker should be smaller than that between the separated speech embedding and the face embedding for the other speaker, by a margin `m`. It encourages the speech separation network to produce cleaner sounds so that a more accurate speech embedding can be obtained to link the voice to the face. Meanwhile, the better the face embedding, the more distinctive the facial attributes feature can be to guide the speech separation process.

Speaker Consistency Loss: The audio segments $s_{\mathcal{A}_1}(t)$ and $s_{\mathcal{A}_2}(t)$ come from the same speaker from video $V_{\mathcal{A}}$, so the voice characteristics of $s_{\mathcal{A}_1}(t)$ and $s_{\mathcal{A}_2}(t)$ should be more similar compared to $s_{\mathcal{B}}(t)$. Therefore, the audio embeddings for the separated speech segments for speaker $\mathcal{A}$ should also be more similar compared to that of speaker $\mathcal{B}$. To capture this, we introduce a speaker consistency loss on the audio embeddings of the separated speech:

$$L_{\textit{consistency}} = L_t(\mathbf{a}^{\mathcal{A}_1}, \mathbf{a}^{\mathcal{A}_2}, \mathbf{a}^{\mathcal{B}_1}) + L_t(\mathbf{a}^{\mathcal{A}_1}, \mathbf{a}^{\mathcal{A}_2}, \mathbf{a}^{\mathcal{B}_2}). \tag{5.7}$$

This loss further regularizes the learning process by jointly separating sounds using the two mixtures.

5.1.4 Training and Inference

The overall objective function for training is as follows:

$$L = L_{\textit{mask-prediction}} + \lambda_1 L_{\textit{cross-modal}} + \lambda_2 L_{\textit{consistency}}, \tag{5.8}$$

where λ_1 and λ_2 are the weight for the cross-modal matching and speaker consistency losses, respectively. During testing, we first detect faces in the video frames and extract the mouth ROIs for each speaker. For each speaker, we use the mouth ROIs and one face image (a randomly selected frame) as the visual input and predict a complex mask to separate the speech from the mixture signal. We use a sliding window approach to perform separation segment by segment for videos of arbitrary length.

Our audio-visual speech separation network is trained from scratch without using any identity labels, whereas prior methods often assume access to a pre-trained lip reading model [3, 4] or a pre-trained face recognition model [55] that sees millions of labeled faces. Furthermore, we do not need to pre-enroll the voice of the speakers as in [4]. Our framework can train and test with fully unlabeled video.

5.2 Experiments

Using a total of six benchmark datasets, we validate our approach for 1) audio-visual speech separation, 2) speech enhancement (Sec. 5.2.3), and 3) cross-modal speaker verification (Sec. 5.2.4).

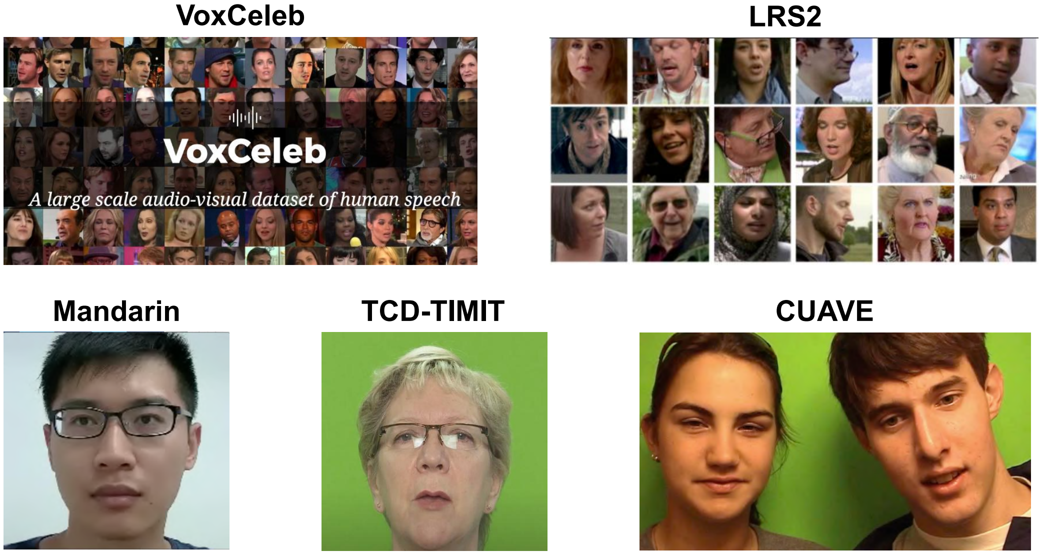


Figure 5.5: We evaluate on six challenging datasets: VoxCeleb1 [156], VoxCeleb2 [33], LRS2 [2], Mandarin [104], TCD-TIMIT [96] and CUAVE [168].

5.2.1 Datasets

We evaluate on six challenging datasets as shown in Fig 5.5, including VoxCeleb2 [33], Mandarin [104], TCD-TIMIT [96], CUAVE [168], LRS2 [2], and VoxCeleb1 [156].

VoxCeleb2 [33]: This dataset contains over 1 million utterances with the associated face tracks extracted from YouTube videos, with 5,994 identities in the training set and 118 identities in the test set. We hold out two videos for each identity in the training set as our seen-heard test set, and we use 59 identities in the original test set as our validation set and the other 59 identities as our unseen-unheard test set. Note that we make use of the identity labels only for the purpose of making these evaluation splits. During testing, we randomly mix two test clips from different speakers to create the synthetic mixture. This ensures the ground-truth of the separated speech is known for quantitative evaluation, following standard practice [3, 55]. We randomly sample 2,000 test parings each from the seen-heard and unseen-unheard test sets. For speech enhancement experiments, we additionally mix the speech mixture with non-speech audios from AudioSet [84] as background noise during both training and testing. The types of noise include music, laughter, crying, engine, wind, *etc.*. See [78] for details and video examples.

Mandarin [104], **TCD-TIMIT [96]**, **CUAVE [168], LRS2 [2]**: We evaluate on these four standard benchmark datasets to compare our model with a series of state-of-the-art audio-visual speech separation and enhancement methods in Sec. 5.2.3.2. See [78] for details.

VoxCeleb1 [156]: This dataset contains over 100,000 utterances for 1,251 celebrities extracted from YouTube videos. We evaluate on this dataset for cross-modal speaker verification in Sec. 5.2.4. We use the same train/val/test split as in [157] to compare with their reported results.

5.2.2 Implementation Details

Our AV speech separation framework is implemented in PyTorch. For all experiments, we sub-sample the audio at 16kHz, and the input speech segment is 2.55s long. STFT is computed using a Hann window length of 400 with a hop size of 160 and FFT window size of 512. The complex spectrogram X is of dimension $2 \times 257 \times 256$. The input to the lip motion analysis network is $N = 64$ mouth regions of interest (ROIs) of size of 88×88, and the input to the face attributes analysis network is a face image of size 224×224. The lip motion feature is of dimension $V_l \times N$ with $V_l = 512$, $N = 64$. The dimension for both the face and voice embeddings is 128. The entire network is trained using an Adam optimizer with weight decay of 0.0001, batchsize of 128 with the starting learning rate set to 1×10^{-4}. λ_1 and λ_2 are both set to 0.01 in Eq. 5.8. The margin $\mathtt{m}$ is set to 0.5 for the triplet loss. See [78] for details of the network architecture and other optimization hyperparameters.

5.2.3 Results on Audio-Visual Speech Separation

We first evaluate on audio-visual speech separation and compare to a series of state-of-the-art methods [3, 5, 26, 35, 55, 69, 104, 171] and multiple baselines:

- **Audio-Only**: This baseline uses the same architecture as our method except that no visual feature is used to guide the separation process. We use the permutation invariant loss (PIT) [252] to train the network.
- **Ours (lip motion)**: An ablation of our method where only the lip motion analysis network is used to guide the separation process.
- **Ours (static face)**: An ablation of our method where only the facial attributes analysis network is used to guide the separation process.
- **AV-Conv [3]**: A state-of-the-art audio-visual speech separation method that predicts the magnitude and phase of the spectrogram separately

through two subnetworks. Because the authors' code is available, we can use it for extensive experiments trained and evaluated on the same data as our method.

- **Ephrat *et al.* [55], Afouras *et al.* [5], Chung *et al.* [35], Gabbay *et al.* [69], Hou *et al.* [104], Casanovas *et al.* [26], Pu *et al.* [171]**: We directly quote results from [5,35,55] to compare to a series of prior state-of-the-art audio-visual speech separation and enhancement methods on standard benchmarks in Sec. 5.2.3.2.

We evaluate the speech separation results using a series of standard metrics including Signal-to-Distortion Ratio (SDR), Signal-to-Interference Ratio (SIR) and Signal-to-Artifacts Ratio (SAR) from the mir eval library [175]. We also evaluate using two speech-specific metrics: Perceptual Evaluation of Speech Quality (PESQ) [183], which measures the overall perceptual quality of the separated speech and Short-Time Objective Intelligibility (STOI) [206], which is correlated with the intelligibility of the signal.

5.2.3.1 Quantitative Results

Table 5.1 shows the speech separation results on the VoxCeleb2 dataset. We use the visual features of both speakers as input to guide the separation and simultaneously separate their voices. We present results separately for scenarios where the lip motion is reliable and unreliable. For the reliable case, we use the original mouth ROIs extracted automatically from the face tracks; for the unreliable case, we randomly shift the mouth ROI sequences in time

	Reliable lip motion					Unreliable lip motion				
	SDR	SIR	SAR	PESQ	STOI	SDR	SIR	SAR	PESQ	STOI
Audio-Only [252]	7.85	13.7	9.97	2.61	0.82	7.85	13.7	9.97	2.61	0.82
AV-Conv [3]	8.91	14.8	11.2	2.73	0.84	7.23	11.4	9.98	2.51	0.80
Ours (static face)	7.21	12.0	10.6	2.52	0.80	7.21	12.0	10.6	2.52	0.80
Ours (lip motion)	9.95	16.9	11.1	2.80	0.86	7.57	12.7	10.0	2.54	0.81
Ours	**10.2**	**17.2**	**11.3**	**2.83**	**0.87**	**8.53**	**14.3**	**10.4**	**2.64**	**0.84**

Table 5.1: Audio-visual speech separation results on the VoxCeleb2 dataset. We show the performance separately for testing examples where the lip motion is reliable (left) or unreliable (right). See text for details. Higher is better for all metrics.

by up to 1s and occlude the lip region for up to 1s per segment during both training and testing. These corruptions represent typical video artifacts (e.g., buffering lag) and mouth occlusions. Table 5.2 shows the speech enhancement results. The setting is the same as Table 5.1 except that the mixture contains additional background sounds (e.g., laughter, car engine, wind, *etc.*) sampled from AudioSet. The visual feature of only the target speaker is used to guide the separation for speech enhancement experiments.

Tables 5.1 and 5.2 show that in both scenarios, our method achieves the best separation results. It outperforms AV-Conv [3] by a good margin. The audio-only baseline benefits from our architecture design, and it has decent performance, though note that unlike AV methods, it cannot assign the separated speech to the corresponding speaker. We evaluate both possible matchings and report its best results (to the baseline's advantage). The ablations show that separation with our model is possible purely using one static face image, but it can be difficult especially when the facial attributes alone are not reliable or distinctive enough to guide separation (see Fig. 5.6). Lip

	Reliable lip motion					Unreliable lip motion				
	SDR	SIR	SAR	PESQ	STOI	SDR	SIR	SAR	PESQ	STOI
Audio-Only [252]	3.56	10.9	5.71	2.00	0.66	3.56	10.9	5.71	2.00	0.66
AV-Conv [3]	5.32	11.9	7.52	2.20	0.71	3.99	9.43	6.92	2.02	0.67
Ours (static face)	3.48	8.43	6.91	1.96	0.68	3.48	8.43	6.91	1.96	0.68
Ours (lip motion)	6.31	13.3	7.72	2.32	0.76	4.21	9.78	6.85	2.03	0.69
Ours	**6.55**	**13.7**	**7.84**	**2.34**	**0.77**	**4.95**	**11.0**	**7.02**	**2.12**	**0.72**

Table 5.2: Audio-visual speech enhancement results on the VoxCeleb2 dataset with audios from AudioSet used as non-speech background noise. Higher is better for all metrics.

motion is directly correlated with the speech content and is much more informative for speech separation when reliable. However, the performance of the lip motion-based model significantly drops when the lip motion is unreliable, as often the case in real-world videos. Our VisualVoice approach combines the complementary cues in both the lip motion and the face-voice embedding learned with cross-modal consistency, and thus is less vulnerable to unreliable lip motion.

5.2.3.2 Comparison to State-of-the-Art Methods

Table 5.3 compares our method to a series of state-of-the-art methods on AV speech separation and enhancement. We use the same evaluation protocols and the same metrics. Our approach improves the state-of-the-art on each of the five datasets.

Whereas Tables 5.1 and 5.2 use the exact same training sources for all methods, here we rely on the authors' reported results [5, 35, 55] in the literature to make comparisons, which draw on different sources. In Table 5.3a-5.3c,

	Gabbay *et al.* [69]	Hou *et al.* [104]	Ephrat *et al.* [55]	Ours
PESQ	2.25	2.42	2.50	**2.51**
STOI	–	0.66	0.71	**0.75**
SDR	–	2.80	6.10	**6.69**

(a) Results on Mandarin dataset.

	Gabbay *et al.* [69]	Ephrat *et al.* [55]	Ours
SDR	0.40	4.10	**10.9**
PESQ	2.03	2.42	**2.91**

(b) Results on TCD-TIMIT dataset.

	Casanovas *et al.* [26]	Pu *et al.* [171]	Ephrat *et al.* [55]	Ours
SDR	7.0	6.2	12.6	**13.3**

(c) Results on CUAVE dataset.

	Afouras *et al.* [3]	Afouras *et al.* [5]	Ours
SDR	11.3	10.8	**11.8**
PESQ	3.0	3.0	**3.0**

(d) Results on LRS2 dataset.

	Chung *et al.* [35]	Ours (static face)	Ours
SDR	2.53	7.21	**10.2**

(e) Results on VoxCeleb2 dataset.

Table 5.3: Comparing to prior state-of-the-art methods on audio-visual speech separation and enhancement. Baseline results are quoted from [5, 35, 55].

we evaluate on the Mandarin, TCD-TIMIT and CUAVE datasets using our speaker-independent model trained on VoxCeleb2 to test the cross-dataset generalization capability of our model. Note that this setting is similar to [55], where they also use a speaker-independent model trained on AV-Speech to test on these datasets. In comparison, the other prior methods require training a speaker-dependent model for each speaker in the test dataset. Our model significantly outperforms these methods, despite never seeing the speakers during training. In Table 5.3d, we train and test on the LRS2 dataset following [5]. Our method consistently outperforms all these prior methods. Notably, in Table 5.3e, our ablated static face model trained with cross-modal consistency significantly improves the prior static image-based model FaceFilter [35] by 4.68 in SDR. This shows that the cross-modal speaker embeddings learned through our VisualVoice framework can provide sufficient cues for separation, even without using any information on lip movements. This is important for a wide range of scenarios (*e.g.*, online social network platforms) where videos containing lip motion are absent, but a user's profile image is available to use for separation.

5.2.3.3 Qualitative Results

Real-World Speech Separation: To further test our method's success on real-world videos with mixed speech, we run our model on a variety of real-world test videos in various challenging scenarios including presidential debates, zoom calls, interviews, noisy restaurants, *etc.*. Note that these videos lack ground-truth, but can be manually checked for quality as shown in the supplementary video[5].

Best/Worst Performing Pairs: Fig. 5.6 illustrates the best and worst performing pairs for speech separation using synthetic pairs for our static face

[5] `http://vision.cs.utexas.edu/projects/VisualVoice/`

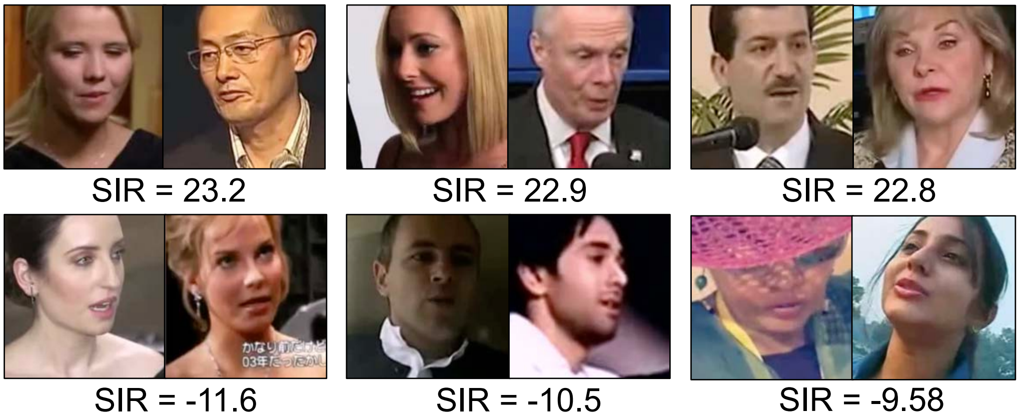
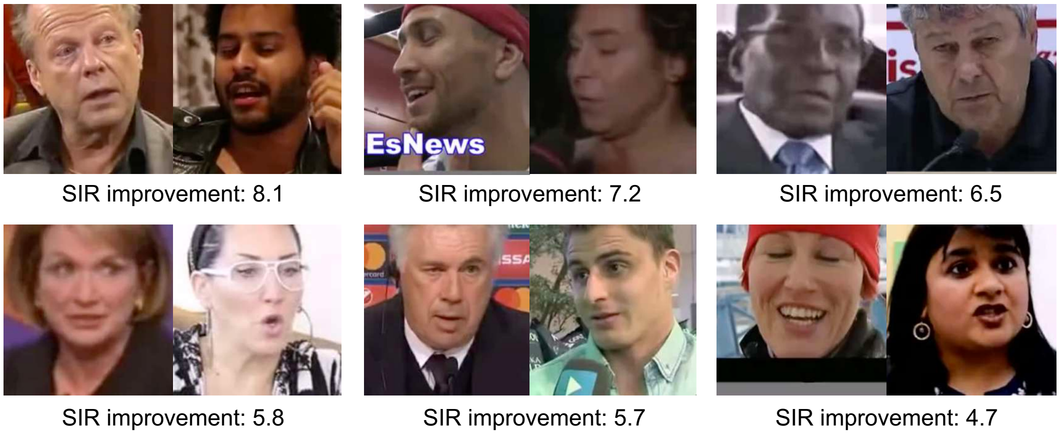


Figure 5.6: Qualitative examples of the best performing pairs (first row) and worst performing pairs (second row) for our static-face image based model.

model. Pairs that perform the best tend to be very different in terms of facial attributes like gender, age, and nationality (first row). Speech separation can be hard if the two mixed identities are visually similar or the facial attributes are hard to obtain from only a static face image due to occlusion or irregular pose (second row).

To further understand when the cross-modal face-voice embeddings help the most, we compare the per-pair performance of our model with only lip motion and our full model in Fig. 5.7. The pairs with the largest improvement from the cross-modal face-voice embeddings tend to be those that either have very different facial appearances or whose lip motion cues are difficult to extract (*e.g.*, non-frontal views).



SIR improvement: 8.1 SIR improvement: 7.2 SIR improvement: 6.5

SIR improvement: 5.8 SIR improvement: 5.7 SIR improvement: 4.7

Figure 5.7: Qualitative examples of the pairs with the largest improvement from cross-modal face-voice embeddings.

5.2.4 Learned Cross-Modal Embeddings

Our multi-task learning framework jointly learns both speech separation and cross-modal face-voice embeddings. Our results thus far show how the cross-modal embedding learning enhances speech separation, our primary goal. As a byproduct of our AV speech separation framework, cross-modal embedding learning may also benefit from the joint learning. Thus we next evaluate the cross-modal verification task, in which the system must decide if a given face and voice belong to the same person.

To compare with prior cross-modal learning work, we train and evaluate on the VoxCeleb1 dataset and compare with the following baselines: 1) **Learnable-Pins [33]**: A state-of-the-art cross-modal embedding learning method. We directly quote their reported results and follow the same evaluation protocols and data splits to compare with our method; 2) **Random**:

	Seen-Heard		Unseen-Unheard	
	AUC ↑	EER ↓	AUC ↑	EER ↓
Random	50.8	49.6	49.7	50.1
Learnable Pins [33]	73.8	34.1	63.5	39.2
Ours (single-task)	75.0	32.2	72.4	34.7
Ours	**84.9**	**23.6**	**74.2**	**32.3**

Table 5.4: Cross-modal verification results on the VoxCeleb1 dataset. ↓ lower better, ↑ higher better.

Embeddings extracted from a randomly initialized network of the same architecture as our method; 3) **Ours (single-task)**: Our cross-modal embedding network without jointly training for speech separation.

Table 5.4 shows the results. We use standard metrics for verification, *i.e.*, area under the ROC curve (AUC) and equal error rate (EER). Our cross-modal embedding network alone compares favorably with [33] on seen-heard speakers and generalizes much better to unseen-unheard speakers. When trained with speech separation in a multi-task setting, our method achieves large gains, demonstrating that our idea to jointly train for these two tasks is beneficial to learn more reliable cross-modal face-voice embeddings.

To *visualize* that our VisualVoice framework has indeed learned useful cross-modal face-voice embeddings, Figure 5.8 shows the t-SNE [149] embeddings of the voices for 15 random speakers from the VoxCeleb1 test set. The embeddings are extracted from our vocal attributes analysis network jointly trained with speech separation. The two sub-figures are color-coded with gen-

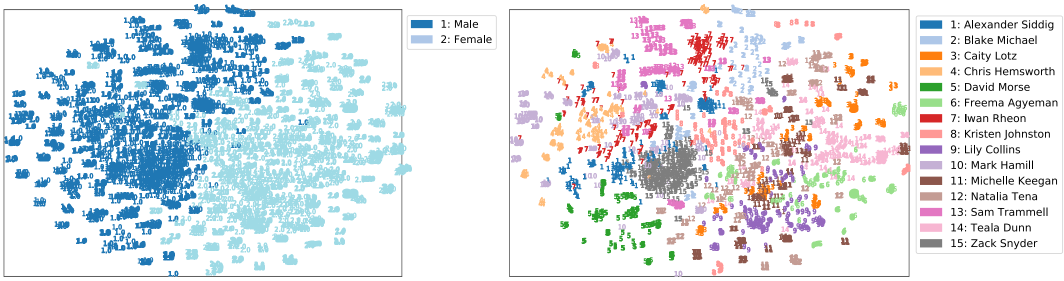


Figure 5.8: Our learned cross-modal embeddings of voices for 15 speakers from the VoxCeleb1 test set visualized with t-SNE. The two figures are color coded with gender and identity, respectively.

der and identity, respectively. Our method's learned voice embeddings tend to cluster speakers of the same cross-modal attributes together despite having access to no identity labels and no attribute labels during training.

5.3 Broader Impact

We are conscious of possible undesirable effects that can arise when working with data-driven approaches to human understanding in images and video. Specifically, a method's training data will guide the extent to which the model can generalize well and fairly to arbitrary inputs. To mitigate risks in this regard, we have taken several steps. First, we learn the cross-modal face-voice embeddings from the VoxCeleb2 dataset, which to our knowledge is the largest relevant available dataset with over 6,000 speakers spanning a range of different ethnicities, accents, professions, and ages. Second, we examine the speech separation results separately for a seen-heard test set and a unseen-unheard test set from VoxCeleb2 (see [78]) The results show our method achieves similar performance for seen-heard and unseen-unheard speakers. This shows that our model generalizes well to unseen-unheard speakers in VoxCeleb2 and is not limited to handling seen-heard speakers in the training data.

Finally, the output of our model consists of voices separated from the the original test video—in terms of masking the input spectrogram—as opposed to being generated or machine synthesized. This is important because it means our model is not free to hallucinate arbitrary voice sounds for the input speakers, e.g., the model cannot artificially conjure sounds or words often associated with training faces that happen to look like the input speaker unless they are consistent with the input sounds. Indeed, as shown in Sec. 5.2.3, lip motion continues to play a key role during speech separation, isolating words based on their visual agreement with what was physically spoken. The learned cross-modal face-voice embeddings complement lip motion cues to further enhance the separation results, particularly when lip motion is harder to read or the two input faces are very different in appearance.

To further explore the model's performance as a function of a person's race, gender, ethnicity, or other identity data, it would be interesting to sort results by the relative impact of our model along each dimension independently. However, existing meta-data does not permit this study (VoxCeleb2 only provides identity and gender labels). We hope to analyze the per-category performance of our models for these cross-modal speaker attributes when datasets as such meta-data and/or new dataset efforts become available.

5.4 Conclusions

In this chapter, I presented an audio-visual speech separation framework that simultaneously learns cross-modal speaker embeddings and speech separation in a multi-task setting. I extend the co-separation framework discussed in Chapter 4 to incorporate motion analysis for instance-level source separation of human speakers. Our VisualVoice approach exploits the complementary cues between the lip motion and cross-modal facial attributes. The cross-modal face-to-voice matching task and the speech separation task are mutually beneficial. We achieve state-of-the-art results on audio-visual speech separation and generalizes well to challenging real-world videos.

I have introduced several of my efforts in the previous three chapters to leverage the audio-visual correspondence of semantic objects for a classic audio task: audio source separation. In the next chapter, I also use audio as a semantic signal, but for a classic computer vision problem: action recognition.

Chapter 6

Action Recognition by Previewing Audio

[1]In the previous three chapters, I presented my approaches to audio-visual source separation, leveraging audio-visual correspondence of semantic objects in videos. While the visual appearance of objects can help to separate the sounds they make, audio and its associated semantic meaning in turn can also be informative to identify visual events in videos. In this chapter, I propose to use audio as a semantic signal for a classic vision problem: action recognition. This work was published in CVPR 2020 [80].

With the growing popularity of portable image recording devices as well as online social platforms, internet users are generating and sharing an ever-increasing number of videos every day. According to a recent study, it would take a person over 5 million years to watch the amount of video that will be crossing global networks each month in 2021 [1]. Therefore, it is imperative to devise systems that can recognize actions and events in these videos both ac-

[1] The work in this chapter was supervised by Prof. Kristen Grauman and Prof. Lorenzo Torresani. It was originally published in: "Listen to Look: Action Recognition by Previewing Audio". Ruohan Gao, Tae-Hyun Oh, Kristen Grauman, and Lorenzo Torresani. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Virtual, June 2020.

curately and efficiently. Potential benefits extend to many video applications, including video recommendation, summarization, editing, and browsing.

Recent advances in action recognition have mostly focused on building powerful clip-level models operating on short time windows of a few seconds [25, 57, 58, 194, 209, 232]. To recognize the action in a test video, most methods densely apply the clip classifier and aggregate the prediction scores of all the clips across the video. Despite encouraging progress, this approach becomes computationally impractical in real-world scenarios where the videos are untrimmed and span several minutes or even hours.

We contend that processing all frames or clips in a long untrimmed video may be unnecessary and even counter-productive. Our key insight is that there are two types of redundancy in video, manifested in both short-term clips as well as long-term periods. First, there is typically high temporal redundancy across the entire video (Fig. 6.1). Many clips capture the same event repetitively, suggesting it is unnecessary to process all the clips. Second, there is redundancy even within a clip: the visual composition within a short time span does not change abruptly; temporally adjacent frames are usually very similar, though there are temporal dynamics (motion) across frames. Therefore, it can be wasteful to process all clips and frames, especially when the video is very long. Moreover, for many activities, the actual actions taking place in the video can be very sparse. It is often a few important moments that are useful for recognition, while the rest actually distract the classifier. For example, in a typical video of surfing, a person might talk for a long time

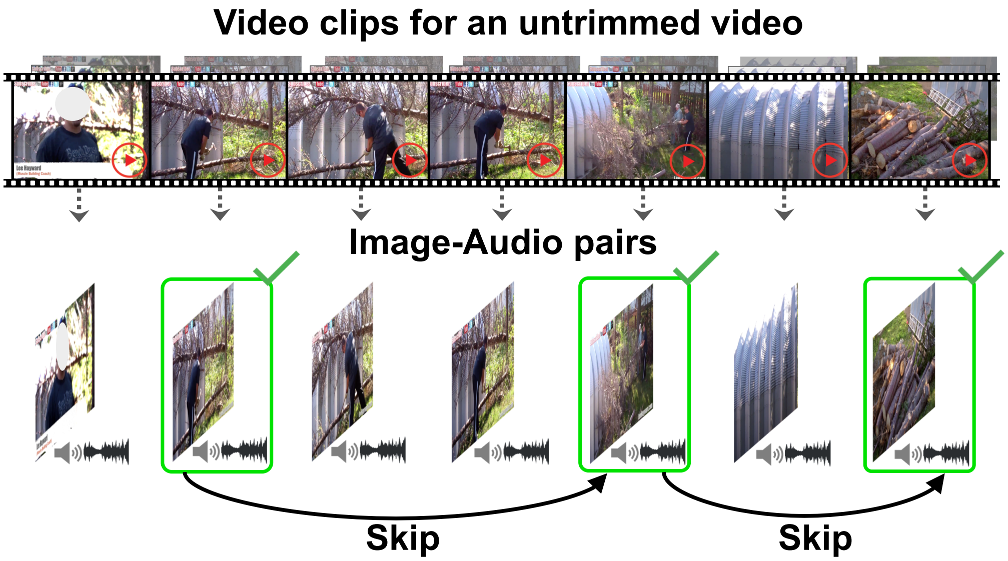


Figure 6.1: Our approach learns to use audio as an efficient preview of the accompanying visual content, at two levels. First we replace the costly analysis of video clips with a more efficient processing of image-audio pairs. A single image captures most of the appearance information within the clip, while the audio provides important dynamic information. Then our video skimming module selects the key moments (a subset of image-audio pairs) to perform efficient video-level action recognition.

and prepare the equipment before he/she begins to surf.

Our idea is to use *audio as an efficient video preview* to reduce both the clip-level and the video-level redundancy in long untrimmed videos. First, instead of processing a whole video clip, we propose an ImgAud2Vid teacher-student distillation framework to hallucinate a video descriptor (*e.g.*, an expensive 3D CNN feature vector) from a single video frame and its accompanying audio. Based on our lightweight image-audio network, we further propose a novel attention-based long short-term memory (LSTM) network, called ImgAud-Skimming, which scans through the entire video and selects the key moments for the final video-level recognition. Both ideas leverage audio as a fast preview of the full video content. Our distilled image-audio model efficiently captures information over short extents, while the skimming module performs fast long-term modeling by skipping over irrelevant and/or uninformative segments across the entire video.

Audio has ideal properties to aid efficient recognition in long untrimmed videos: audio contains dynamics and rich contextual temporal information [83] and, most importantly, it is much more computationally efficient to process compared to video frames. For example, as shown in Fig. 6.1, within a short clip of the action chopping wood, a single frame includes most of the appearance information contained in the clip, *i.e.*, {person, axe, tree}, while the accompanying audio (the sound of the axe hitting the tree in this case) contains useful cues of temporal dynamics. Across the entire video, audio can also be beneficial to select the key moments that are useful for recognition. For example, the sound of the person talking initially can suggest that the actual action has not started, while the sound of the electric saw may indicate that the action is taking place. Our approach automatically learns such audio signals.

We experiment on four datasets (Kinetics-Sounds, Mini-Sports1M, ActivityNet, UCF-101) and demonstrate the advantages of our framework. Our main contributions are threefold. Firstly, we are the first to propose to replace the expensive extraction of clip descriptors with an efficient proxy distilled from audio. Secondly, we propose a novel video-skimming mechanism that leverages image-audio indexing features for efficient long-term modeling in untrimmed videos. Thirdly, our approach pushes the envelope of the trade-off between accuracy and speed favorably; we achieve state-of-the-art results on action recognition in untrimmed videos with few selected frames or clips.

In Sec 6.1, I describe our co-separation approach for learning audio-visual source separation. Then I present key experiments and results in Sec 6.2.

6.1 Approach

Our goal is to perform accurate and efficient action recognition in long untrimmed videos. We first formally define our problem (Sec. 6.1.1); then we introduce how we use audio as a clip-level preview to hallucinate video descriptors based on only a single static frame and its accompanying audio segment (Sec. 6.1.2); finally we present how we leverage image-audio indexing features to obtain a video-level preview, and learn to skip over irrelevant or uninformative segments in untrimmed videos (Sec. 6.1.3).

6.1.1 Problem Formulation

Given a long untrimmed video $\mathcal{V}$, the goal of video classification is to classify $\mathcal{V}$ into a predefined set of C classes. Because $\mathcal{V}$ can be very long, it is often intractable to process all the video frames together through a single deep network due to memory constraints. Most current approaches [25, 57, 121, 173, 194, 209, 210, 232] first train a clip-classifier $\Omega(\cdot)$ to operate on a short fixed-length video clip $\mathbf{V} \in \mathbb{R}^{F \times 3 \times H \times W}$ of F frames with spatial resolution $H \times W$, typically spanning several seconds. Then, given a test video of arbitrary length, these methods densely apply the clip-classifier to N clips $\{\mathbf{V}_1, \mathbf{V}_2, \ldots, \mathbf{V}_N\}$ which are taken at a fixed hop size across the entire video. The final video-level prediction is obtained by aggregating the clip-level predictions of all N clips.

Such paradigms for video recognition are highly inefficient at two levels: (1) *clip-level*—within each short clip $\mathbf{V}$, temporally close frames are visually similar, and (2) *video-level*—across all the clips in $\mathcal{V}$, often only a few clips contain the key moments for recognizing the action. Our approach addresses both levels of redundancy via novel uses of audio.

Each video clip $\mathbf{V}$ is accompanied by an audio segment $\mathbf{A}$. The starting frame $\mathbf{I}$ among the F frames within the short clip $\mathbf{V}$ usually contains most of the appearance cues already, while the audio segment $\mathbf{A}$ contains rich contextual temporal information (recall the wood cutting example in Fig. 6.1). Our idea is to replace the powerful but expensive clip-level classifier $\Omega(\cdot)$ that takes F frames as input with an efficient image-audio classifier $\Phi(\cdot)$ that only takes the starting frame $\mathbf{I}$ and its accompanying audio segment $\mathbf{A}$ as input, while preserving the clip-level information as much as possible. Namely, we seek to learn $\Phi(\cdot)$ such that

$$\Omega(\mathbf{V}_j) \approx \Phi(\mathbf{I}_j, \mathbf{A}_j), \quad j \in \{1, 2, \ldots, N\}, \tag{6.1}$$

for a given pre-trained clip-classifier $\Omega(\cdot)$. In Sec. 6.1.2, we design an ImgAud2Vid distillation framework to achieve this goal. Through this step, we replace the processing of high-dimensional video clips $\{\mathbf{V}_1, \mathbf{V}_2, \ldots, \mathbf{V}_N\}$ with a lightweight model analyzing compact image-audio pairs $\{(\mathbf{I}_1, \mathbf{A}_1), (\mathbf{I}_2, \mathbf{A}_2), \ldots, (\mathbf{I}_N, \mathbf{A}_N)\}$.

Next, building on our efficient image-audio classifier $\Phi(\cdot)$, to address video-level redundancy we design an attention-based LSTM network called ImgAud-Skimming. Instead of classifying every image-audio pair using $\Phi(\cdot)$ and aggregating all their prediction results, our ImgAud-Skimming framework iteratively selects the most useful image-audio pairs. Namely, our method efficiently selects a small subset of T image-audio pairs from the entire set of N pairs in the video (with $T \ll N$) and only aggregates the predictions from these selected pairs. We present our video skimming mechanism in Sec. 6.1.3.

6.1.2 Clip-Level Preview

We present our approach to perform efficient clip-level recognition and our ImgAud2Vid distillation network architecture. As shown in Fig. 6.2, the clip-based model takes a video clip $\mathbf{V}$ of F frames as input and based on that video volume generates a clip descriptor $\mathbf{z}^{\mathbf{V}}$ of dimensionality D. A fully-connected layer is used to make predictions among the C classes in Kinetics. For the student model, we use a two-stream network: the image stream takes the first frame $\mathbf{I}$ of the clip as input and extracts an image descriptor $\mathbf{z}^{\mathbf{I}}$; the audio stream takes the audio spectrogram $\mathbf{A}$ as input and extracts an audio feature vector $\mathbf{z}^{\mathbf{A}}$. We concatenate $\mathbf{z}^{\mathbf{I}}$ and $\mathbf{z}^{\mathbf{A}}$ to generate an image-audio feature vector of dimensionality D using a fusion network $\Psi(\cdot)$ that consists of two fully-connected layers. A final fully-connected layer is used to produce

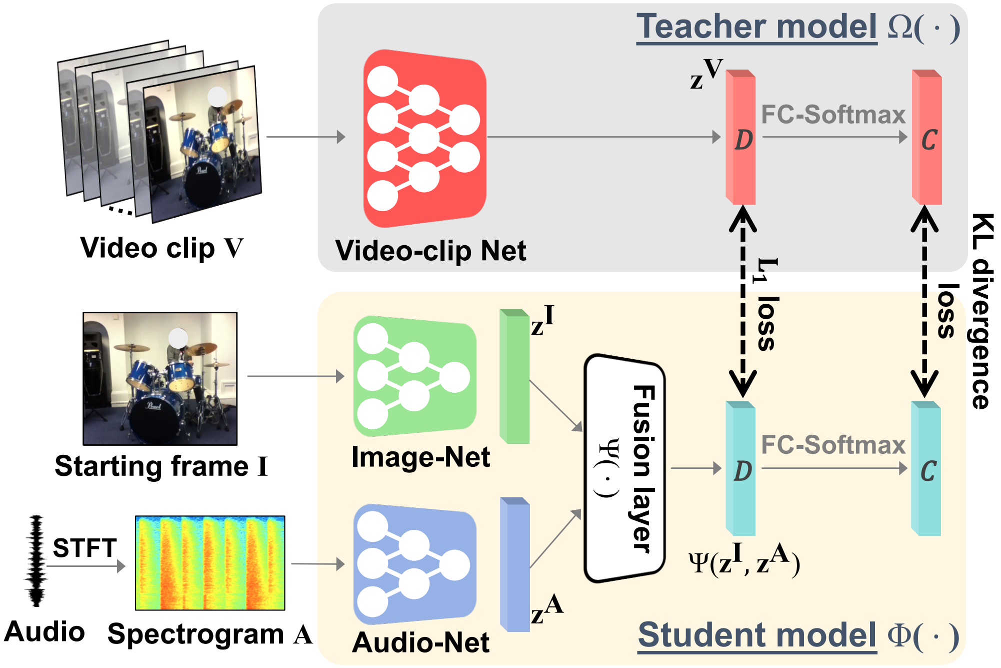

Teacher model $\Omega(\cdot)$
Video clip $\mathbf{V}$
Video-clip Net
$\mathbf{z}^{\mathbf{V}}$
D
FC-Softmax
C
L_1 loss
KL divergence loss
Starting frame $\mathbf{I}$
Image-Net
$\mathbf{z}^{\mathbf{I}}$
Fusion layer $\Psi(\cdot)$
D
FC-Softmax
C
$\Psi(\mathbf{z}^{\mathbf{I}}, \mathbf{z}^{\mathbf{A}})$
STFT
Audio
Spectrogram $\mathbf{A}$
Audio-Net
$\mathbf{z}^{\mathbf{A}}$
Student model $\Phi(\cdot)$

Figure 6.2: ImgAud2Vid distillation framework: The teacher model is a video-clip classifier, and the student model consists of a visual stream that takes the starting frame of the clip as input and an audio stream that takes the audio spectrogram as input. By processing only a single frame and the clip's audio, we get an estimate of what the expensive video descriptor would be for the full clip.

a C-class prediction like the teacher model.

The teacher model $\Omega(\cdot)$ returns a softmax distribution over C classification labels. These predictions are used as soft targets for training weights associated with the student network $\Phi(\cdot)$ using the following objective:

$$\mathcal{L}_{\text{KL}} = -\sum\nolimits_{\{(\mathbf{V},\mathbf{I},\mathbf{A})\}} \sum\nolimits_{c} \Omega_c(\mathbf{V}) \log \Phi_c(\mathbf{I}, \mathbf{A}), \tag{6.2}$$

where $\Omega_c(\mathbf{V})$ and $\Phi_c(\mathbf{I}, \mathbf{A})$ are the softmax scores of class c for the teacher model and the student model, respectively. We further impose an $\mathcal{L}_1$ loss on the clip descriptor $\mathbf{z^V}$ and the image-audio feature to regularize the learning process:

$$\mathcal{L}_1 = \sum\nolimits_{\{(\mathbf{z^V}, \mathbf{z^I}, \mathbf{z^A})\}} \|\mathbf{z^V} - \Psi(\mathbf{z^I}, \mathbf{z^A})\|_1. \tag{6.3}$$

The final learning objective for ImgAud2Vid distillation is a combination of these two losses:

$$\mathcal{L}_{\text{Dist.}} = \mathcal{L}_1 + \lambda L_{\text{KL}}, \tag{6.4}$$

where λ is the weight for the KL divergence loss. The training is done over the image and audio student networks (producing representations $\mathbf{z^I}$ and $\mathbf{z^A}$, respectively) and the fusion model $\Psi(\cdot)$ with respect to a fixed teacher video-clip model. The teacher model we use is a R(2+1)D-18 [210] video-clip classifier, which is pre-trained on Kinetics [123]. Critically, processing the audio for a clip is substantially faster than processing all its frames, making audio an efficient preview. See Sec. 6.2.1 for cost comparisons. After distillation, we fine-tune the student model on the target dataset to perform efficient clip-level action recognition.

6.1.3 Video-Level Preview

ImgAud2Vid distills knowledge from a powerful clip-based model to an efficient image-audio based model. Next, we introduce how we leverage the distilled image-audio network to perform efficient video-level recognition. Recall that for long untrimmed video, processing only a subset of clips is desirable both for speed and accuracy, *i.e.*, to ignore irrelevant content.

We design ImgAud-Skimming, an attention-based LSTM network

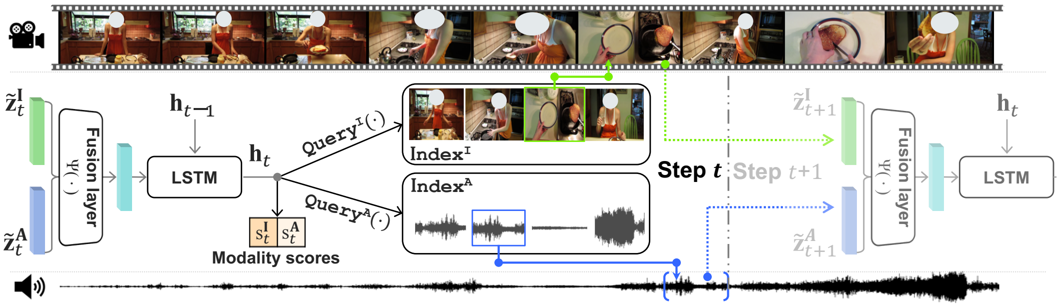

$\tilde{\mathbf{z}}_t^{\mathbf{I}}$
$\tilde{\mathbf{z}}_t^{\mathbf{A}}$
Fusion layer
$\Psi(\cdot)$
$\mathbf{h}_{t-1}$
LSTM
$\mathbf{h}_t$
Query$^{\mathrm{I}}(\cdot)$
Query$^{\mathrm{A}}(\cdot)$
Index$^{\mathrm{I}}$
Index$^{\mathrm{A}}$
$s_t^{\mathbf{I}}$ $s_t^{\mathbf{A}}$
Modality scores
Step t
Step $t+1$
$\tilde{\mathbf{z}}_{t+1}^{\mathbf{I}}$
$\tilde{\mathbf{z}}_{t+1}^{A}$
Fusion layer
$\Psi(\cdot)$
$\mathbf{h}_t$
LSTM

Figure 6.3: Our ImgAud-Skimming network is an LSTM network that interacts with the sequences of image and audio indexing features to select where to "look at" and "listen to" next. At each time step, it takes the image feature and audio feature for the current time step as well as the previous hidden state and cell output as input, and produces the current hidden state and cell output. The hidden state for the current time step is used to make predictions about the next moment to focus on in the untrimmed video through the querying operation illustrated in Fig. 6.4. The average-pooled ImgAud2Vid features of all selected time steps is used to make the final prediction of the action in the video.

(Fig. 6.3), which interacts with the sequence of image-audio pairs $\{(\mathbf{I}_1, \mathbf{A}_1), (\mathbf{I}_2, \mathbf{A}_2), \ldots, (\mathbf{I}_N, \mathbf{A}_N)\}$, whose features are denoted as $\{\mathbf{z}_1^{\mathbf{I}}, \mathbf{z}_2^{\mathbf{I}}, \ldots, \mathbf{z}_N^{\mathbf{I}}\}$ and $\{\mathbf{z}_1^{\mathbf{A}}, \mathbf{z}_2^{\mathbf{A}}, \ldots, \mathbf{z}_N^{\mathbf{A}}\}$, respectively. At the t-th time step, the LSTM cell takes the *indexed* image feature $\tilde{\mathbf{z}}_t^{\mathbf{I}}$ and the *indexed* audio feature $\tilde{\mathbf{z}}_t^{\mathbf{A}}$, as well as the previous hidden state $\mathbf{h}_{t-1}$ and the previous cell output $\mathbf{c}_{t-1}$ as input, and produces the current hidden state $\mathbf{h}_t$ and the cell output $\mathbf{c}_t$:

$$\mathbf{h}_t, \mathbf{c}_t = \texttt{LSTM}\big(\Psi(\tilde{\mathbf{z}}_t^{\mathbf{I}}, \tilde{\mathbf{z}}_t^{\mathbf{A}}), \mathbf{h}_{t-1}, \mathbf{c}_{t-1}\big), \tag{6.5}$$

where $\Psi(\cdot)$ is the same fusion network used in ImgAud2Vid with the same parameters. To fetch the indexed features $\tilde{\mathbf{z}}_t^{\mathbf{I}}$ and $\tilde{\mathbf{z}}_t^{\mathbf{A}}$ from the feature sequences, an indexing operation is required. This operation is typically nondifferentiable. Instead of relying on approximating policy gradients as in prior work [56, 240, 242], we propose to deploy a differentiable soft indexing mechanism, detailed below.

We predict an image query vector $\mathbf{q}_t^{\mathbf{I}}$ and an audio query vector $\mathbf{q}_t^{\mathbf{A}}$ from the hidden state $\mathbf{h}_t$ at each time step through two prediction networks $\mathtt{Query}^{\mathbf{I}}(\cdot)$ and $\mathtt{Query}^{\mathbf{A}}(\cdot)$. The query vectors, $\mathbf{q}_t^{\mathbf{I}}$ and $\mathbf{q}_t^{\mathbf{A}}$, are used to query the respective sequences of image indexing features $\{\mathbf{z}_1^{\mathbf{I}}, \mathbf{z}_2^{\mathbf{I}}, \ldots, \mathbf{z}_N^{\mathbf{I}}\}$ and audio indexing features $\{\mathbf{z}_1^{\mathbf{A}}, \mathbf{z}_2^{\mathbf{A}}, \ldots, \mathbf{z}_N^{\mathbf{A}}\}$. The querying operation is intended to predict which part of the untrimmed video is more useful for recognition of the action in place and decide where to "look at" and "listen to" next. It is motivated by attention mechanisms [92, 205, 216, 221], but we adapt this scheme to the problem of selecting useful moments for action recognition in untrimmed video.

Figure 6.4 illustrates our querying mechanism. First, we use one fully-connected layer $\mathtt{Key}(\cdot)$ to transform indexing features $\mathbf{z}$ to indexing keys $\mathbf{k}$. Then, we get an attention score $\frac{\mathbf{k}^\top \mathbf{q}}{\sqrt{d}}$ for each indexing key in the sequence, where d is the dimensionality of the key vector. A `Softmax` layer normalizes the attention scores and generates an attention weight vector $\mathbf{w}$ by:

$$\mathbf{w} = \mathtt{Softmax}\Big(\frac{[\mathbf{k}_1 \mathbf{k}_2 \ldots \mathbf{k}_N]^\top \cdot \mathbf{q}}{\sqrt{d}}\Big), \tag{6.6}$$

where $\mathbf{k}_j = \mathtt{Key}(\mathbf{z}_j)$, $j \in \{1, 2, \ldots, N\}$.

At each time step t (we omit t for simplicity if deducible), one could obtain the frame index for the next time step by $\arg\max(\mathbf{w})$. However, this op-

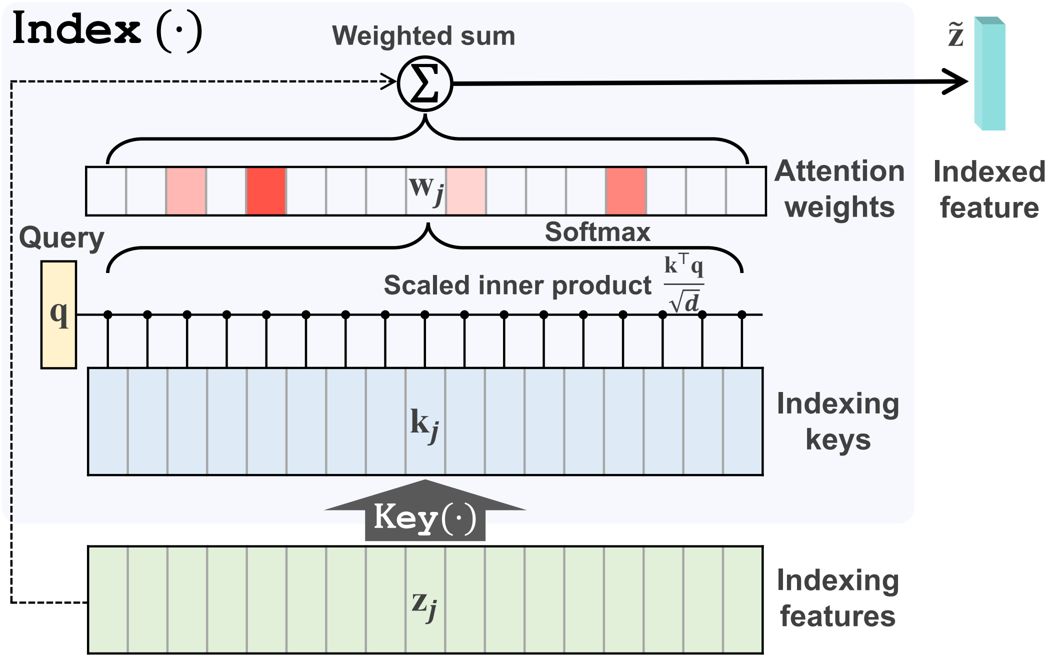


Figure 6.4: Attention-based frame selection mechanism.

eration is not differentiable. Instead of directly using the image and audio features of the selected frame index, we use the weighted average of the sequence of indexing features to generate an aggregated feature vector $\tilde{\mathbf{z}}^{\mathbf{I}}_{t+1} = \mathtt{Index}^{\mathbf{I}}(\mathbf{w}_t)$ and $\tilde{\mathbf{z}}^{\mathbf{A}}_{t+1} = \mathtt{Index}^{\mathbf{A}}(\mathbf{w}_t)$ as input to the fusion network $\Psi(\cdot)$, where

$$\begin{aligned} \mathtt{Index}^{\mathbf{I}}(\mathbf{w}) &:= \textstyle\sum_{j=1}^{N} w_j \mathbf{z}^{\mathbf{I}}_j, \\ \mathtt{Index}^{\mathbf{A}}(\mathbf{w}) &:= \textstyle\sum_{j=1}^{N} w_j \mathbf{z}^{\mathbf{A}}_j, \quad w_{j\in\{1,\cdots,N\}} \in \mathbb{R}_+. \end{aligned} \tag{6.7}$$

The querying operations are performed independently on the visual and audio modalities, and produce distinct weight vectors $\mathbf{w}^{\mathbf{I}}_t$ and $\mathbf{w}^{\mathbf{A}}_t$ to find the visually-useful and acoustically-useful moments, respectively. These two weight vectors may give importance to different moments in the sequence. We fuse this information by dynamically adjusting how much to rely on each modality at each step. To this end, we predict two modality scores $s_t^{\mathbf{I}}$ and $s_t^{\mathbf{A}}$, from the hidden state $\mathbf{h}_t$ through a two-way classification layer. $s_t^{\mathbf{I}}$ and $s_t^{\mathbf{A}}$ ($s_t^{\mathbf{I}}, s_t^{\mathbf{A}} \in [0,1], s_t^{\mathbf{I}} + s_t^{\mathbf{A}} = 1$) indicate how much the system decides to rely on the visual modality versus the audio modality, respectively, at time step t. Then, the image and audio feature vectors for the next time step are finally obtained by aggregating the feature vectors predicted both visually and acoustically, as follows:

$$\begin{aligned} \tilde{\mathbf{z}}_{t+1}^{\mathbf{I}} &= s_t^{\mathbf{I}} \cdot \mathtt{Index}^{\mathbf{I}}(\mathbf{w}_t^{\mathbf{I}}) + s_t^{\mathbf{A}} \cdot \mathtt{Index}^{\mathbf{I}}(\mathbf{w}_t^{\mathbf{A}}), \\ \tilde{\mathbf{z}}_{t+1}^{\mathbf{A}} &= s_t^{\mathbf{I}} \cdot \mathtt{Index}^{\mathbf{A}}(\mathbf{w}_t^{\mathbf{I}}) + s_t^{\mathbf{A}} \cdot \mathtt{Index}^{\mathbf{A}}(\mathbf{w}_t^{\mathbf{A}}). \end{aligned} \tag{6.8}$$

Motivated by iterative attention [154], we repeat the above procedure for T steps, and average the image-audio features obtained. Namely,

$$\mathbf{m} = \tfrac{1}{T}\textstyle\sum_{j=1}^{T} \Psi(\tilde{\mathbf{z}}_j^{\mathbf{I}}, \tilde{\mathbf{z}}_j^{\mathbf{A}}). \tag{6.9}$$

$\mathbf{m}$ is a feature summary of the useful moments selected by ImgAud-Skimming. A final fully-connected layer followed by $\mathtt{Softmax}(\cdot)$ takes $\mathbf{m}$ as input and makes predictions of action categories. The network is then trained with cross-entropy loss and video-level action label annotations.

While we optimize the ImgAud-Skimming network for a fixed number of T steps during training, at inference time we can stop early at any step depending on the computation budget. Moreover, instead of using all indexing features, we can also use a subset of indexing features to accelerate inference with the help of feature interpolation. See Sec. 6.2.2 for details about the efficiency and accuracy trade-off when using sparse indexing features and early stopping.

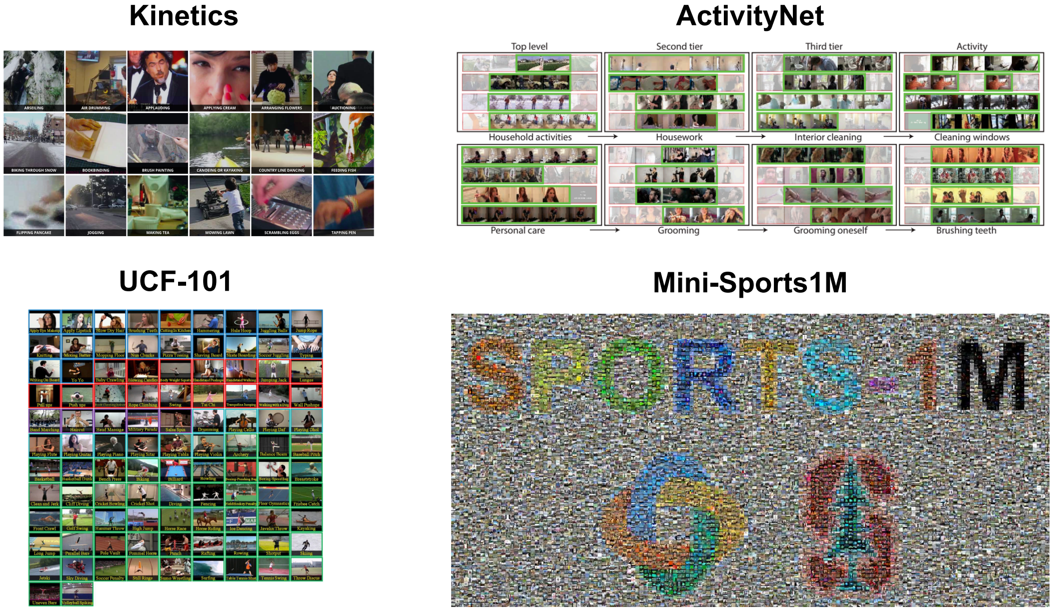


Figure 6.5: We train on Kinetics [123] and evaluate on four other datasets: Kinetics-Sounds [13] (a subset of Kinetics), UCF-101 [200], ActivityNet [24], and Mini-Sports1M [121].

6.2 Experiments

Using a total of four datasets, we evaluate our approach for accurate and efficient clip-level action recognition (Sec. 6.2.1) and video-level action recognition (Sec. 6.2.2).

Datasets: Our distillation network is trained on Kinetics [123], and we evaluate on four other datasets: Kinetics-Sounds [13], UCF-101 [200], ActivityNet [24], and Mini-Sports1M [121]. See Fig. 6.5.

- **Kinetics-Sounds** is a subset of Kinetics and consists of only action classes that are potentially recognizable both visually and aurally. It is assembled by [13] and consists of 34 classes. However, 3 classes were removed from the original Kinetics dataset. Therefore, we use the remaining 31 classes in our experiments. The 31 action classes are: blowing nose, blowing out candles, bowling, chopping wood, dribbling basketball, laughing, mowing lawn, playing accordion, playing bagpipes, playing bass guitar, playing clarinet, playing drums, playing guitar, playing harmonica, playing keyboard, playing organ, playing piano, playing saxophone, playing trombone, playing trumpet, playing violin, playing xylophone, ripping paper, shoveling snow, shuffling cards, singing, stomping grapes, tap dancing, tapping guitar, tapping pen, and tickling.

- **UCF-101** is a dataset of about 13K short trimmed clips of 101 human actions. We use the official training/validation/testing splits (split1) in our experiments.

- **ActivityNet** contains videos of various lengths with average duration of 117 seconds. We use the latest release (version 1.3), which consists of around 20K videos of 200 classes. We use the official training/validation/testing splits in our experiments.

- **Mini-Sports1M** is a subset of Sports1M dataset containing an equal number of videos for each class. It is assembled by us to facilitate comparisons of video-level action recognition following [131]. We only take videos of length 2-5 mins, and randomly sample 30 videos for each class for training, and 10 videos for each class for testing.

Kinetics-Sounds and UCF-101 contain only short trimmed videos, so we only test on them for clip-level recognition; ActivityNet contains videos of various lengths, so it is used as our main testbed for both clip-level and video-level recognition; Mini-Sports1M contains only long untrimmed videos, and we use it for evaluation of video-level recognition.

Implementation Details: We implement in PyTorch. For ImgAud2Vid, the R(2+1)D-18 [210] teacher model takes 16 frames of size 112 × 112 as input. The student model uses a ResNet-18 network for both the visual and audio streams, which take the starting RGB frame of size 112 × 112 and a 1-channel audio-spectrogram of size 101 × 40 (1 sec. audio segment) as input, respectively. We use $\lambda = 100$ for the distillation loss in Equation 6.4. For ImgAud-Skimming, we use a one-layer LSTM with 1,024 hidden units and a dimension of 512 for the indexing key vector. We use $T = 10$ time steps during training. See [80] for details.

6.2.1 Clip-Level Action Recognition

First, we directly evaluate the performance of the image-audio network distilled from the video model. We fine-tune on each of the three datasets for clip-level recognition and compare against the following baselines:

- **Clip-based Model:** The R(2+1)D-18 [210] teacher model.
- **Image-based Model (distilled/undistilled):** A ResNet-18 frame-based model. The undistilled model is pre-trained on ImageNet, and the distilled

model is similar to our method except that the distillation is performed using only the visual stream.

- **Audio-based Model (distilled/undistilled):** The same as the image-based model except here we only use the audio stream for recognition and distillation. The model is pre-trained on ImageNet to accelerate convergence.
- **Image-Audio Model (undistilled):** The same image-audio network as our method but without distillation.

For each baseline, we use the corresponding model as initialization and fine-tune on the same target dataset for clip-based action recognition. Note that our purpose here is not to compete on recognition accuracy using R(2+1)D-18 (or any other expensive video features), but rather to demonstrate our distilled image-audio features can approximate its recognition accuracy much more efficiently.

Figure 6.6 compares the accuracy vs. efficiency for our approach and the baselines. Our distilled image-audio network achieves accuracy comparable to that of the clip-based teacher model, but at a much reduced computational cost. Moreover, the models based on image-only or audio-only distillation produce lower accuracy. This shows that the image or audio alone is not sufficient to hallucinate the video descriptor, but when combined they provide sufficiently complementary information to reduce the accuracy gap with the true (expensive) video-clip descriptor.

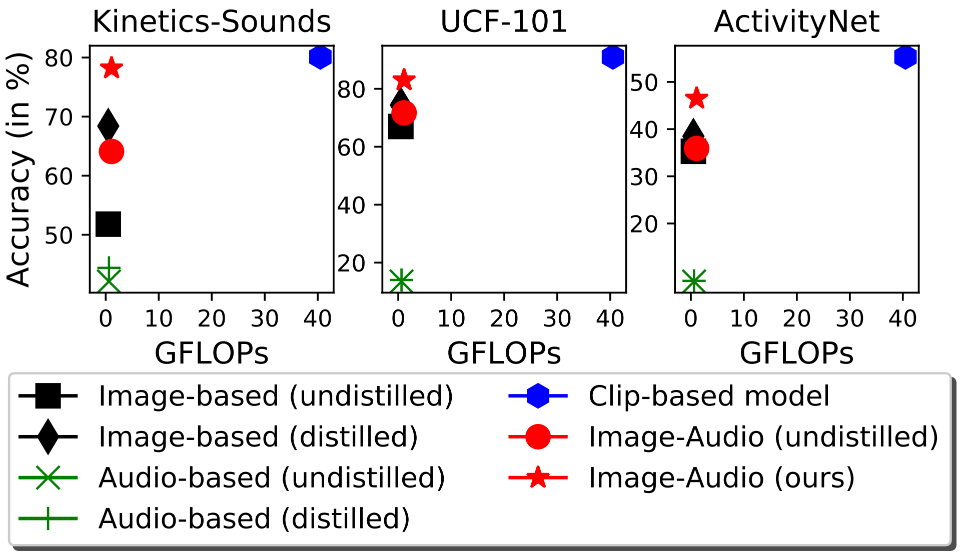


Figure 6.6: Clip-level action recognition on Kinetics-Sounds, UCF-101, and ActivityNet. We compare the recognition accuracy and the computational cost of our model against a series of baselines. Our ImgAud2Vid approach strikes a favorable balance between accuracy and efficiency.

To understand when audio helps the most, we compute the $\mathcal{L}_1$ distance of the hallucinated video descriptor to the ground-truth video descriptor by our ImgAud2Vid distillation and the image-based distillation. As shown in Fig. 6.7, the top-ranked clips (first row) for which we best match the ground-truth tend to be dynamic scenes that have informative audio information, *e.g.*, grinding meat, jumpstyle dancing, playing cymbals, playing bagpipes, wrestling and welding. The bottom-ranked clips (second row) tend to be clips where the audio either contains just silence, narration, and background music, or are too difficult to perceive, *e.g.*, answering questions, bee keeping, clay pottery making, getting a haircut, tossing coin and extinguishing fire.

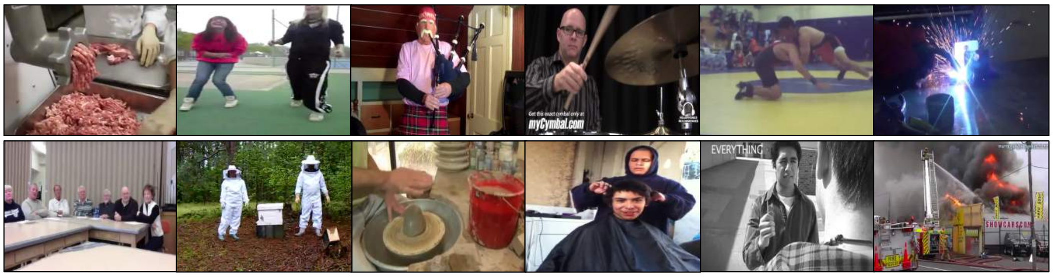
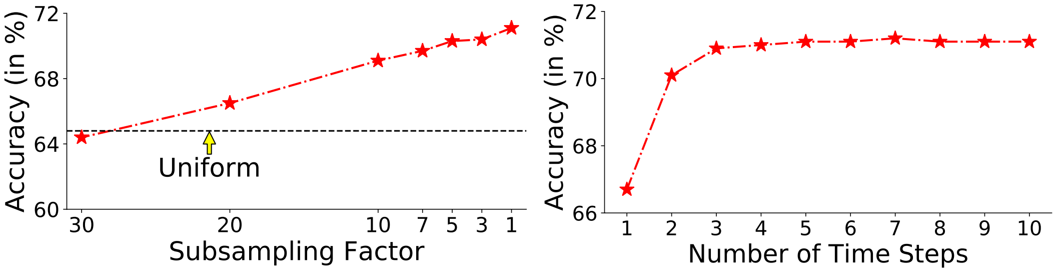


Figure 6.7: Top-ranked/bottom-ranked clips where audio helps the most/least for our ImgAud2Vid distillation. The top-ranked clips (first row) belong to classes: grinding meat, jumpstyle dancing, playing cymbals, playing bagpipes, wrestling and welding; The bottom-ranked clips (second row) belong to classes: answering questions, bee keeping, clay pottery making, getting a haircut, tossing coin and extinguishing fire.

6.2.2 Untrimmed Video Action Recognition

Having demonstrated the clip-level performance of our distilled image-audio network, we now examine the impact of the ImgAud-Skimming module on video-level recognition. We evaluate on ActivityNet [24] and Mini-Sports1M [121], which contain long untrimmed videos.

Efficiency & Accuracy Trade-off: Before showing the results, we introduce how we use feature interpolation to further enhance the efficiency of our system. Apart from using features from all N time stamps as described in Sec. 6.1.3, we experiment with using *sparse* indexing features extracted from a subset of image-audio pairs, *i.e.*, subsampling along the time axis. Motivated by the locally-smooth action feature space [50] and based on our empirical observation that neighboring video features can be linearly approximated well, we synthesize the missing image and audio features by computationally cheap



(a) Feature interpolation

(b) Early stopping

Figure 6.8: Trade-off between efficiency and accuracy when using sparse indexing features or early stopping on ActivityNet. Uniform denotes the UNIFORM baseline in Table 6.1.

linear interpolation to generate the full feature sequences of length N. Figure 6.8a shows the recognition results when using different subsampling factors. We can see that recognition remains robust to even aggressive subsampling of the indexing features.

Next we investigate early stopping as an additional means to reduce the computational cost. Instead of repeating the skimming procedure for 10 times as in the training stage, we can choose to stop early after a few recurrent steps. Figure 6.8b shows the results when stopping at different time steps. We can see that the first three steps yield sufficient cues for recognition. This suggests that we can stop around the third step with negligible accuracy loss. See [80] for a similar observation on Mini-Sports1M.

Results: We compare our approach to the following baselines and several existing methods [56, 131, 240, 242, 250]:

- RANDOM: We randomly sample 10 out of the N time stamps, and average the predictions of the image-audio pairs from these selected time stamps using the distilled image-audio network.

- Uniform: The same as the previous baseline except that we perform uniform sampling.
- Front / Center / End: The same as before except that the first / center / last 10 time stamps are used.
- Dense: We average the prediction scores from all N image-audio pairs as the video-level prediction.
- SCSampler [131]: We use the idea of [131] and select the 10 image-audio pairs that yield the largest confidence scores from the image-audio classifier. We average their predictions to produce the video-level prediction.
- LSTM: This is a one-layer LSTM as in our model but it is trained and tested using all N image-audio features as input sequentially to predict the action label from the hidden state of the last time step.
- Non-Recurrent: The same as our method except that we only use a single query operation without the recurrent iterations. We directly obtain the 10 time stamps from the indexes of the 10 largest attention weights.

Table 6.1 shows the results. Our method outperforms all the baselines. The low accuracy of Random / Uniform / Front / Center / End indicates the importance of the context-aware selection of useful moments for action recognition. Using sparse indexing features (with a subsampling factor of 5), our method outperforms Dense (the status quo of how most current methods obtain video-level predictions) by a large margin using only about 1/5

	Random	Uniform	Front	Center	End	SCSampler [131]	Dense	LSTM	Non-Recurrent	Ours (sparse / dense)
ActivityNet	63.7	64.8	39.0	59.0	38.1	69.1	66.3	63.5	67.5	**70.3** / **71.1**
Mini-Sports1M	35.4	35.6	17.1	29.7	17.4	38.4	37.3	34.1	38.0	**39.2** / **39.9**

Table 6.1: Video-level action recognition accuracy (in %) on ActivityNet (# classes: 200) and Mini-Sports1M (# classes: 487). Kinetics-Sounds and UCF-101 consist of only short trimmed videos, so they are not applicable here. Our method consistently outperforms all baseline methods. Ours (sparse) uses only about 1/5 the computation cost of the last four baselines, while achieving large accuracy gains. See Table 6.2 for more computation cost comparisons.

of its computation cost. Our method is also better and faster than SCSampler [131], despite their advantage of densely evaluating prediction results on all clips. LSTM performs comparably to Random. We suspect that it fails to aggregate the information of all time stamps when the video gets very long. Non-Recurrent is an ablated version of our method, and it shows that the design of recursive prediction of the "next" interesting moment in our method is essential. Both LSTM and Non-Recurrent support our contribution as a whole framework, *i.e.*, iterative attention based selection.

Comparison to State-of-the-Art Frame Selection Methods: Fig. 6.9 compares our approach to state-of-the-art frame selection methods given the same computational budget. The results of the baselines are quoted from AdaFrame [242] and MultiAgent [240], where they both evaluate on ActivityNet. For fair comparison, we test a variant of our method with only the visual modality, and we use the same ResNet-101 features for recognition. Our framework also has the flexibility of using cheaper features for indexing (frame selection). See [80] for details about the single-modality architecture of our ImgAud-Skimming network and how we use different features for in-

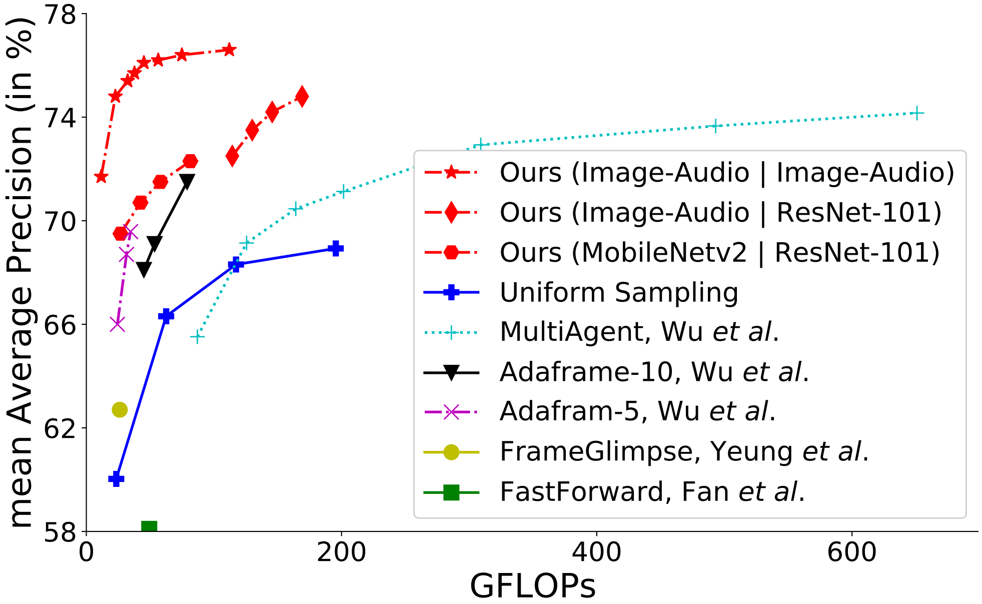


Figure 6.9: Comparisons with other frame selection methods on ActivityNet. We directly quote the numbers reported in AdaFrame [242] and MultiAgent [240] for all the baseline methods and compare the mAP against the average GFLOPs per test video. See text for details.

dexing and recognition. We use three different combinations denoted as *Ours ("indexing features" | "recognition features")* in Fig. 6.9, including using MobileNetv2 [188] features for efficient indexing similar to [242]. Moreover, to gauge the impact of our ImgAud2Vid step, we also report the results obtained by using image-audio features for recognition.

Our method consistently outperforms all existing methods and achieves the best balance between speed and accuracy when using the same recognition features, suggesting the accuracy boost can be attributed to our novel differentiable indexing mechanism. Furthermore, with the aid of ImgAud2Vid distillation, we achieve much higher accuracy with much less computation cost;

	Backbone	Pre-trained	Accuracy	mAP
IDT [226]	–	ImageNet	64.7	68.7
C3D [209]	–	Sports1M	65.8	67.7
P3D [173]	ResNet-152	ImageNet	75.1	78.9
RRA [267]	ResNet-152	ImageNet	78.8	83.4
MARL [240]	ResNet-152	ImageNet	79.8	83.8
Ours	ResNet-152	ImageNet	**80.3**	**84.2**

(a) Comparison to prior work with ResNet-152 features.

	Indexing	Recognition	mAP	TFLOPs
Dense	–	R(2+1)D-152	88.9	25.9
Uniform	–	R(2+1)D-152	87.2	1.26
Ours	Image-Audio	R(2+1)D-152	88.5	1.31
Ours	R(2+1)D-152	R(2+1)D-152	**89.9**	2.64

(b) Accuracy vs. Efficiency with R(2+1)D-152 features.

Table 6.2: ActivityNet comparison to SOTA methods.

this scheme combines the efficiency of our image-audio clip-level recognition with the speedup and accuracy enabled by our ImgAud-Skimming network for video-level recognition.

Comparison to the State-of-the-Art on ActivityNet: Having compared our skimming approach to existing methods for frame selection, now we compare to state-of-the-art activity recognition models that forgo frame selection. For fair comparison, we use the ResNet-152 model provided by [240]. This model is pre-trained on ImageNet and fine-tuned on ActivityNet with TSN-style [229] training. As shown in Table 6.2a, our method consistently outperforms all the previous state-of-the-art methods. To show that the benefits of our method extend even to more powerful but expensive features, we use R(2+1)D-152 features for recognition in Table 6.2b. When using R(2+1)D-152 features for both indexing and recognition, we outperform the dense approach while being 10× *faster*. We can still achieve comparable performance to the dense approach if using our image-audio features for indexing, while being *20× faster*.

6.2.3 Qualitative Analysis

Figure 6.10 shows frames selected by our method using the visual modality versus those obtained by uniform sampling. The frames chosen by our method are much more informative of the action in the video compared to those uniformly sampled. See Supp. video[2] for examples of acoustically useful moments selected by our method.

We can inspect per-class performance to understand what are the classes that benefit the most from our skimming mechanism compared to uniform sampling. The top classes in descending order of accuracy gain are: cleaning sink, beer pong, gargling mouthwash, painting furniture, archery, laying tile, and triple jump—classes where the action is sporadic and is often exhibited over a short segment of the video. See [80] for more analysis.

[2] `http://vision.cs.utexas.edu/projects/listen_to_look/`

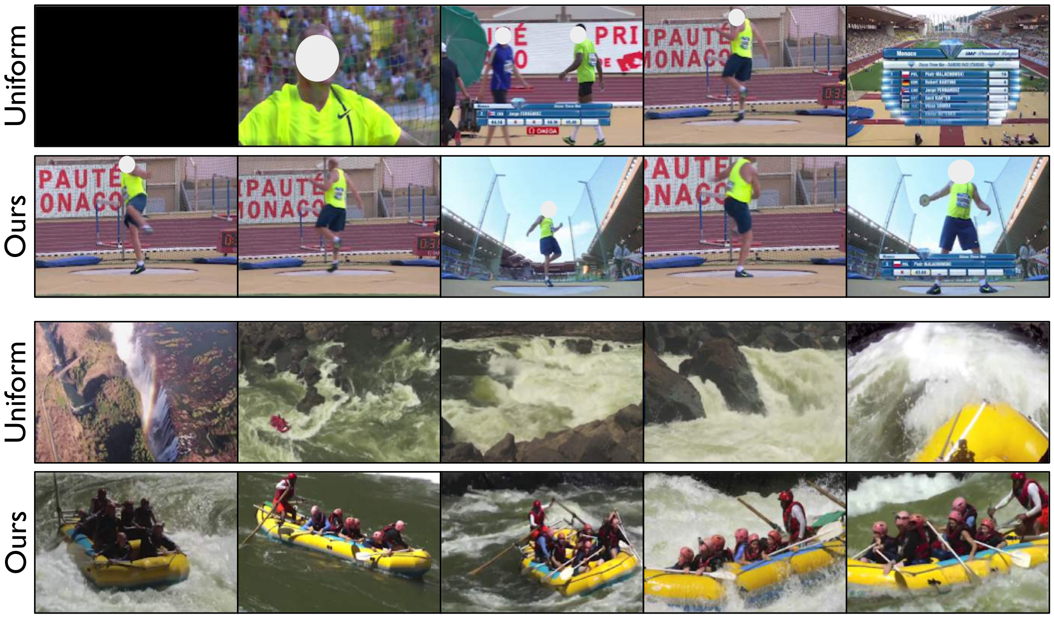


Figure 6.10: Qualitative examples of 5 uniformly selected moments and the first 5 visually useful moments selected by our method for two videos of the actions *throwing discus* and *rafting* in ActivityNet. The frames selected by our method are more indicative of the corresponding action. See the supplementary video[2] for examples of acoustically useful moments selected by our method.

6.3 Conclusions

In this chapter, I presented an approach to achieve both accurate and efficient action recognition in long untrimmed videos by leveraging audio as a previewing tool. Our IMGAUD2VID distillation framework replaces the expensive clip-based model by a lightweight image-audio based model, enabling efficient clip-level action recognition. Moreover, we propose an IMGAUD-SKIMMING network that iteratively selects useful image-audio pairs, enabling efficient video-level action recognition. Our work strikes a favorable balance between speed and accuracy, and we achieve state-of-the-art results for video action recognition using few selected frames or clips.

However, the current framework cannot handle cases where multiple actions are happening in an untrimmed video. It would be important future work to consider multi-label video action classification by adapting the video-preview module to select useful moments for each individual action category. Moreover, we separately train the clip-based model and the video-level model in two stages. It would be interesting to train an end-to-end model that simultaneously performs cross-modal distillation and video-level action recognition.

So far, my work presented in this dissertation use audio as a semantic signal. My approaches recognize objects based on their appearance without explicitly paying attention to their *spatial* locations. However, both audio and visual data also convey significant spatial information. Starting from the next chapter, I will present my work on leveraging audio as a spatial signal for audio-visual learning.

Chapter 7

Visually-Guided Audio Spatialization

[1]The audio-visual source separation and action recognition tasks presented so far center around learning sounds associated with objects or events and their semantics. In this chapter, I study the problem of audio spatialization, which further requires reasoning about objects' locations. I devise an approach that learns to decode the monaural (single-channel) soundtrack into its binaural counterpart by injecting visual information about object and scene configurations. This work was published in CVPR 2019 [76].

The human auditory system uses *two* ears to extract individual sound sources from a complex mixture. Accordingly, to mimic human hearing, *binaural audio* is usually recorded using two microphones attached to the two ears of a dummy head (see Fig. 7.2). The rig's two microphones, their spacing, and the physical shape of the ears are all significant for approximating how humans receive sound signals. As a result, when playing binaural au-

[1] The work in this chapter was supervised by Prof. Kristen Grauman and was originally published in: "2.5D Visual Sound". Ruohan Gao and Kristen Grauman. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, June 2019.

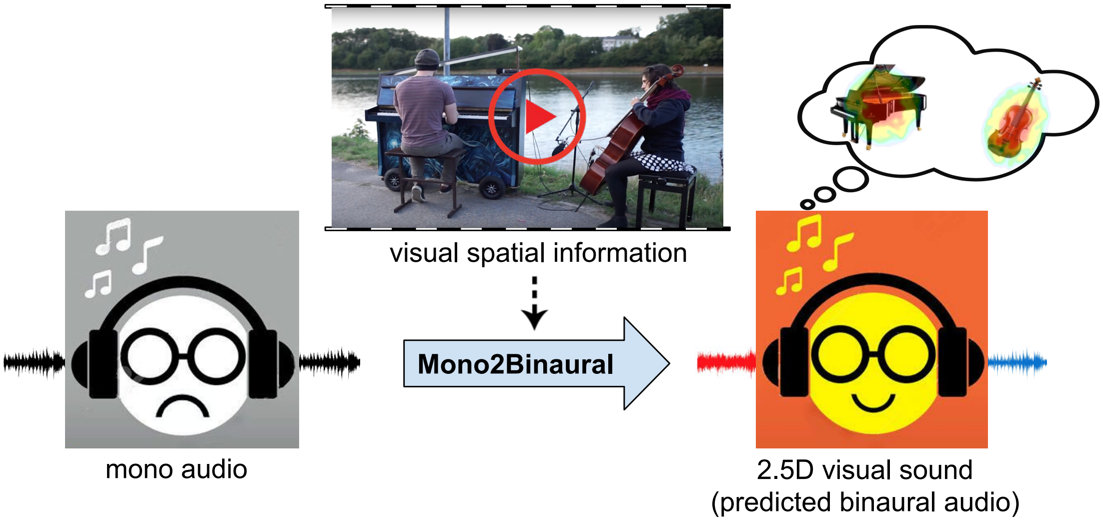


Figure 7.1: Binaural audio creates a 3D soundscape for listeners, but such recordings remain rare. The proposed approach infers *2.5D visual sound* by injecting the spatial information contained in the video frames accompanying a typical monaural audio stream.

dio through headphones, listeners feel the *3D sound sensation* of being in the place where the recording was made and can easily localize the sounds. The immersive spatial sound is valuable for audiophiles, AR/VR applications, and social video sharers alike.

However, binaural recordings are difficult to obtain in daily life due to the high price of the recording device and the required expertise. Consumer-level cameras typically only record monaural audio with a single microphone, or stereo audio recorded using two microphones with arbitrary arrangement and without physical representation of the pinna (outer ear). We contend that for both machines and people, monaural or even stereo auditory input has very limited dimension. Monaural audio collapses all independent audio streams to the same spatial point, and the listener cannot sense the spatial locations of the sound sources.

Our key insight is that video accompanying monaural audio has the potential to unlock spatial sound, lifting a flat audio signal into what we call "2.5D visual sound". Although a single channel audio track alone does not encode any spatial information, its accompanying visual frames do contain object and scene configurations. For example, as shown in Fig. 7.1, we observe from the video frame that a man is playing the piano on the left and a man is playing the cello on the right. Although we cannot sense the locations of the sound sources by listening to the mono recording, we can nonetheless anticipate what we *would* hear if we were personally in the scene by inference from the visual frames.

We introduce an approach to realize this intuition. Given unlabeled training video, we devise a MONO2BINAURAL deep convolutional neural network to convert monaural audio to binaural audio by injecting the spatial cues embedded in the visual frames. Our encoder-decoder style network takes a mixed single-channel audio and its accompanying visual frames as input to perform joint audio-visual analysis, and attempts to predict a two-channel binaural audio that agrees with the spatial configurations in the video. When listening to the predicted binaural audio—the 2.5D visual sound—listeners can then feel the locations of the sound sources as they are displayed in the video.

Moreover, we show that apart from binaural audio generation, the MONO2BINAURAL conversion process can also benefit audio-visual source separation, a key challenge in audio-visual analysis, as introduced in Chapter 3, 4 and 5. State-of-the-art systems [3, 55, 75, 163, 258] aim to separate a mixed monaural audio recording into its component sound sources, and thus far they rely solely on the spatial cues evident in the visual stream. For example, our two approaches for audio-visual source separation both operate on monaural audio, where spatial information is missing. We show that the proposed audio-visual binauralization can *self-supervise* representation learning to elicit spatial signals relevant to separation from the audio stream as well. Critically, gaining this new learning signal requires neither semantic annotations nor single-source data preparation, only the same unlabeled binaural training video.

The main contributions of this final component of my thesis proposal are threefold: Firstly, we propose to convert monaural audio to binaural audio by leveraging video frames, and we design a MONO2BINAURAL deep network to achieve that goal; Secondly, we collect FAIR-Play, a 5.2 hour video dataset with binaural audio—the first dataset of its kind to facilitate research in both the audio and vision communities; Thirdly, we propose to perform audio-visual source separation on predicted binaural audio, and show that it provides a useful self-supervised representation for the separation task. We validate our approach on four challenging datasets spanning a variety of sound sources (e.g., instruments, street scenes, travel, sports).

In Sec 7.1, I describe our MONO2BINAURAL approach for audio spatialization. Then I present experimental results in Sec 7.2.

7.1 Approach

Our approach learns to map monaural audio to binaural audio via video. In the following, we first describe our binaural audio video dataset (Sec. 7.1.1). Then we present our MONO2BINAURAL formulation (Sec. 7.1.2), and our network and training procedure to solve it (Sec. 7.1.3). Finally we introduce our approach to leverage inferred binaural sound to perform audio-visual source separation (Sec. 7.1.4).

7.1.1 FAIR-Play Data Collection

Training our method requires binaural audio and accompanying video. Since no large public video datasets contain binaural audio, we collect a new dataset we call FAIR-Play with a custom rig. As shown in Fig. 7.2, we assembled a rig consisting of a 3Dio Free Space XLR binaural microphone, a GoPro HERO6 Black camera, and a Tascam DR-60D recorder as the audio pre-amplifier. We mounted the GoPro camera on top of the 3Dio binaural microphone to mimic a person's embodiment for seeing and hearing, respectively. The 3Dio binaural microphone records binaural audio, and the GoPro camera records videos at 30fps with stereo audio. We simultaneously record from both devices so the streams are roughly aligned.

Note that both the ear shaped housing (pinnae) for the microphones and their spatial separation are significant; professional binaural mics like 3Dio simulate the physical manner in which humans receive sound. In contrast, stereo sound is captured by two mics with an arbitrary separation that

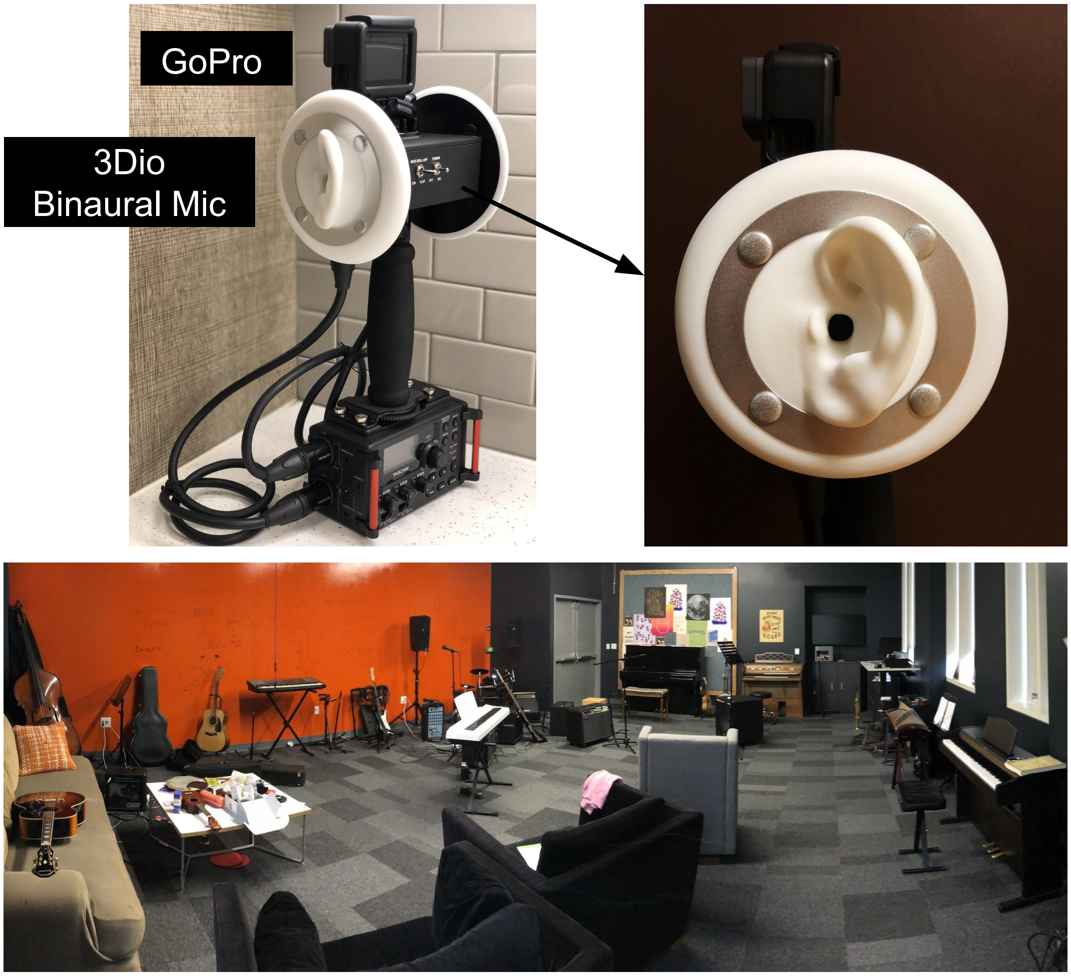


Figure 7.2: Binaural rig and data collection in a music room.

varies across capture devices (phones, cameras), and so lacks the spatial nuances of binaural. The limit of binaural capture, however, is that a single rig inherently assumes a single *head-related transfer function*, whereas individuals have slight variations due to inter-person anatomical differences. Personalizing head-related transfer functions is an area of active research [110, 208].

We captured videos with our custom rig in a large music room (about 1,000 square feet). Our intent was to capture a variety of sound making objects in a variety of spatial contexts, by assembling different combinations of instruments and people in the room. The room contains various instruments including cello, guitar, drum, ukelele, harp, piano, trumpet, upright bass, and banjo. We recruited 20 volunteers to play and recorded them in solo, duet, and multi-player performances. We post-process the raw data into 10s clips. In the end, our FAIR-Play[2] dataset consists of 1,871 short clips of musical performances, totaling 5.2 hours. In experiments we use both the music data as well as ambisonics datasets [155] for street scenes and YouTube videos of sports, and travel.

7.1.2 Mono2Binaural Formulation

Binaural cues let us infer the location of sound sources. The interaural time difference (ITD) and the interaural level difference (ILD) play an essential role. ITD is caused by the difference in travel distances between the two ears. When a sound source is closer to one ear than the other, there is a time delay between the signals' arrival at the two ears. ILD is caused by a "shadowing" effect—a listener's head is large relative to certain wavelengths of sound, so it serves as a barrier, creating a shadow. The particular shape of the head, pinnae, and torso also act as a filter depending on the locations of the sound sources (distance, azimuth, and elevation). All these cues are missing

[2] `https://github.com/facebookresearch/FAIR-Play`

in monaural audio, thus we cannot sense any spatial effect by listening to single-channel audio.

We denote the signal received at the left and right ears by $x^L(t)$ and $x^R(t)$, respectively. If we mix the two channels into a single channel $x^M(t) = x^L(t) + x^R(t)$, then all spatial information collapses. We can formulate a self-supervised task to take the mixed monaural signal $x^M(t)$ as input and split it into two separate channels $\tilde{x}^L(t)$ and $\tilde{x}^R(t)$, using the original $x^L(t)$, $x^R(t)$ as ground-truth during training. However, this is a highly under-constrained problem, as $x^M(t)$ lacks the necessary information to recover both channels. Our key idea is to guide the MONO2BINAURAL process with the accompanying video frames, from which *visual* spatial information can serve as supervision.

Instead of directly predicting the two channels, we predict the difference of the two channels:

$$x^D(t) = x^L(t) - x^R(t). \tag{7.1}$$

More specifically, we operate on the frequency domain and perform short-time Fourier transform (STFT) [93] on $x^M(t)$ to obtain the complex-valued spectrogram $\mathbf{X}^M$, and the objective is to predict the complex-valued spectrogram $\mathbf{X}^D$ for $x^D(t)$:

$$\mathbf{X}^M = \{\mathbf{X}^M_{t,f}\}^{T,F}_{t=1,f=1}, \qquad \mathbf{X}^D = \{\mathbf{X}^D_{t,f}\}^{T,F}_{t=1,f=1}, \tag{7.2}$$

where t and f are the time frame and frequency bin indices, respectively, and T and F are the numbers of bins. Then we obtain the predicted difference

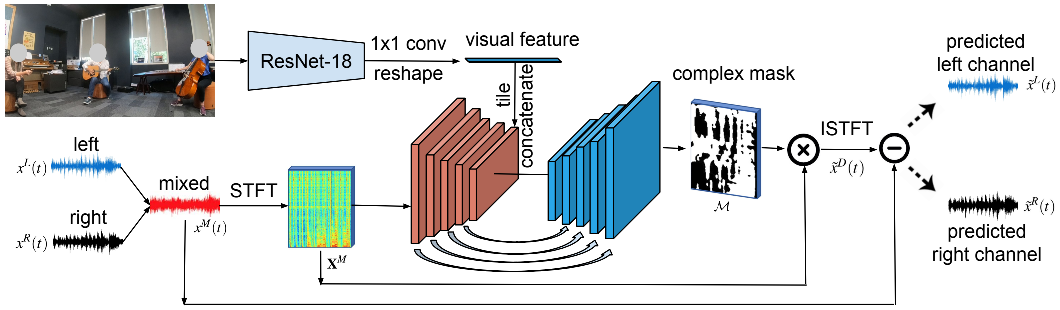


Figure 7.3: Our MONO2BINAURAL deep network takes a mixed monaural audio and its accompanying visual frame as input, and predicts a two-channel binaural audio output that satisfies the visual spatial configurations. An ImageNet pre-trained ResNet-18 network is used to extract visual features, and a U-Net is used to extract audio features and perform joint audio-visual analysis. We predict a complex mask for the audio difference signal, then combine it with the input mono audio to restore the left and right channels, respectively. At test time, the input is single-channel monaural audio.

signal $\tilde{x}^D(t)$ by the inverse short-time Fourier transform (ISTFT) [93] of $\mathbf{X}^D$. Finally, we recover both channels—the binaural audio output:

$$\tilde{x}^L(t) = \frac{x^M(t) + \tilde{x}^D(t)}{2}, \qquad \tilde{x}^R(t) = \frac{x^M(t) - \tilde{x}^D(t)}{2}. \tag{7.3}$$

7.1.3 Mono2Binaural Network

Next we present our MONO2BINAURAL deep network to perform audio spatialization. The network takes the mono audio $x^M(t)$ and visual frames as input and predicts $x^D(t)$.

As shown in Fig. 7.3, we extract visual features from the center frame of the audio segment using ResNet-18 [99], which is pre-trained on ImageNet. The ResNet-18 network extracts per-frame features after the 4^{th} ResNet block with size $(H/32) \times (W/32) \times C$, where H, W, C denote the frame and channel dimensions. We then pass the visual feature through a 1×1 convolution layer to reduce the channel dimension, and flatten it into a single visual feature vector.

On the audio side, we adopt a U-Net [184] style architecture. The U-Net encoder-decoder network adopted here is ideal for our dense prediction task where the input and output have the same dimension, as used in Chapter 4 and 5 to predict spectrogram masks for audio separation. We mix the left and right channels of the binaural audio, and extract a sequence of STFT frames to generate an audio spectrogram $\mathbf{X}^M$. We use the complex spectrogram: each time-frequency bin contains the real and imaginary part of the corresponding complex spectrogram value. Then it is passed through a series of convolution layers to extract an audio feature of dimension $(T/32) \times (F/32) \times C$. We replicate the visual feature vector $(T/32) \times (F/32)$ times, tile them to match the audio feature dimension, and then concatenate the audio and visual feature maps along the channel dimension. Through the series of operations, each audio feature dimension is injected with the visual feature to perform joint audio-visual analysis.

Finally, we perform up-convolutions on the concatenated audio-visual feature map to generate a complex multiplicative spectrogram mask $\mathcal{M}$. In source separation tasks, spectrogram masks have proven better than alternatives such as direct prediction of spectrograms or raw waveforms [225]. Similarly, here we also adopt the idea of masking, but our goal is to mask the spectrogram of the mixed mono audio and predict the spectrogram of the difference signal, rather than perform separation. The real and imaginary components of the complex mask are separately estimated in the real domain. We add a sigmoid layer after the up-convolution layers to bound the complex mask values to [-1, 1], similar to [55]. The series of convolutions and up-convolutions maps the input mono spectrogram to a complex mask that encodes the predicted binaural audio.

Initially, we attempted to directly predict the left and right channels. However, we found that direct prediction makes the network fall back on a "safe" but useless solution of copying and pasting the input audio, without reasoning with the visual features. Instead, predicting the difference signal forces the deep network to analyze the visual information and learn the subtle difference between the two channels, as required by the binaural audio target.

The spectrogram of the difference signal is then obtained by complex multiplying the input spectrogram with the predicted complex mask:

$$\tilde{\mathbf{X}}^D = \mathcal{M} \cdot \mathbf{X}^M. \tag{7.4}$$

We train our MONO2BINAURAL network using L2 loss to minimize the distance between the ground-truth complex spectrogram and the predicted one. Finally, using ISTFT, we obtain the predicted difference signal $\tilde{x}^D(t)$, through which we recover the two channels $\tilde{x}^L(t)$ and $\tilde{x}^R(t)$ as defined in Eq. 7.3. See [76] for network details.

At test time, the network is presented with monaural audio and a video frame and infers the binaural output, i.e., the 2.5D visual sound. To process a full video stream, each video is decomposed into many short audio segments. Video frames usually do not change much within such a short segment. We use a sliding window to perform spatialization segment by segment with a small hop size, and average predictions on overlapping parts. Thus, our method is able to handle moving sound sources and cameras.

Our approach expects a similar field of view (FoV) between training and testing, and assumes the microphone is near the camera. Our experiments demonstrate we can learn MONO2BINAURAL for both normal FoV and 360° video, and furthermore the same system can cope with mono inputs from variable hardware (e.g., YouTube videos).

7.1.4 Audio-Visual Source Separation

So far we have defined our MONO2BINAURAL approach to convert monaural audio to binaural audio by introducing visual spatial cues from video. Recall that we have two goals: to predict binaural audio for sound generation itself, and to explore its utility for audio-visual source separation.

Audio source separation is the problem of obtaining an estimate for each of the J sources s_j from the observed linear mixture $x(t) = \sum_{j=1}^{J} s_j(t)$. For binaural audio source separation, the problem is to obtain an estimate for each of the J sources s_j from the observed binaural mixture $x^L(t)$ and $x^R(t)$:

$$x^L(t) = \sum_{j=1}^{J} s_j^L(t), \qquad x^R(t) = \sum_{j=1}^{J} s_j^R(t), \tag{7.5}$$

where $s_j^L(t)$ and $s_j^R(t)$ are time-discrete signals received at the left ear and the right ear for each source, respectively.

Interfering sound sources are often located at different spatial positions in the physical space. Human listeners exploit the spatial information from the coordination of both ears to resolve sound ambiguity caused by multiple sources. This ability is greatly diminished when listening with only one ear, especially in reverberant environments [129]. Audio source separation by machine listeners is similarly handicapped, typically lacking access to binaural audio [55, 75, 163, 258]. However, we hypothesize that our MONO2BINAURAL *predicted* binaural audio can aid separation. Intuitively, by forcing the network to learn how to lift mono audio to binaural, its representation is encouraged to expose the very spatial cues that are valuable for source separation. Thus, even though the MONO2BINAURAL features see the same video as any other audio-visual separation method, they may better decode the latent spatial cues because of their binauralization "pre-training" task.

In particular, we expect two main effects. First, binaural audio embeds information about the spatial distribution of sound sources, which can act as a regularizer for separation. Second, binaural cues may be especially helpful in cases where sound sources have similar acoustic characteristics, since the spatial organization can reduce source ambiguities. Related regularization effects are observed in other vision tasks. For example, hallucinating motion enhances static-image action recognition [81], or predicting semantic segmentation informs depth estimation [142].

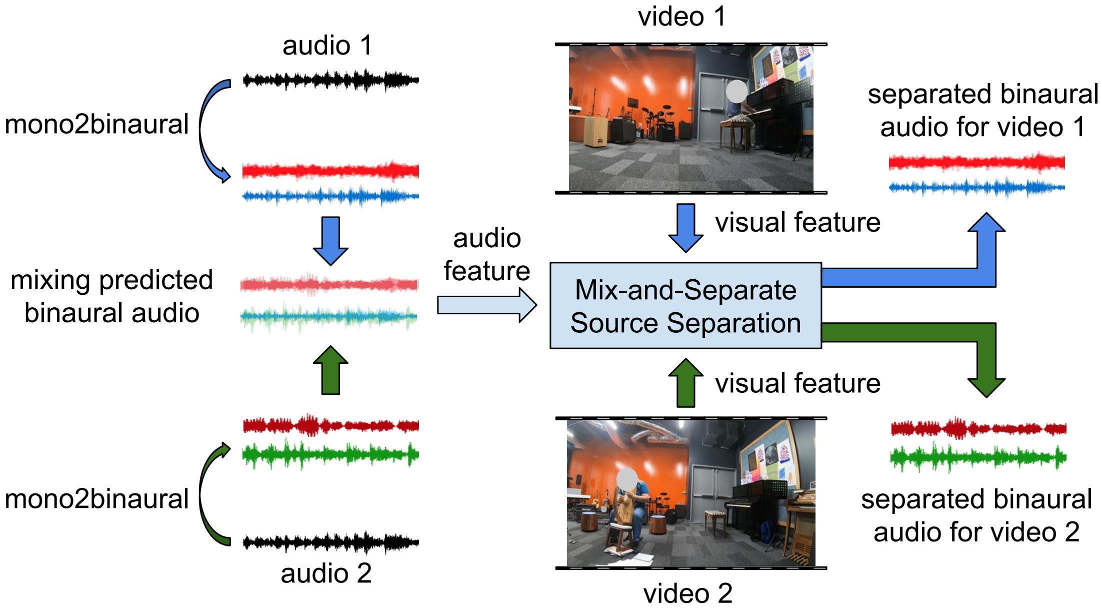

audio 1
mono2binaural
video 1
separated binaural audio for video 1
visual feature
audio feature
mixing predicted binaural audio
Mix-and-Separate Source Separation
visual feature
mono2binaural
audio 2
video 2
separated binaural audio for video 2

Figure 7.4: Mix-and-Separate [55,163,258]-inspired framework for audio-visual source separation. During training, we mix the binaural audio tracks for a pair of videos to generate a mixed audio input. The network learns to separate the sound for each video conditioned on their visual frames.

To implement a testbed for audio-visual source separation, we adopt the Mix-and-Separate idea [55,163,258]. We use the same base architecture as our MONO2BINAURAL network except that now the input to the network is a pair of training video clips. Fig. 7.4 illustrates the separation framework. We mix the sounds of the predicted binaural audio for the two videos to generate a complex audio input signal, and the learning objective is to separate the binaural audio for each video conditioned on their corresponding visual frames. Following [258], we only use spectrogram magnitude and predict a ratio mask for separation. Per-pixel L1 loss is used for training. See [76] for details.

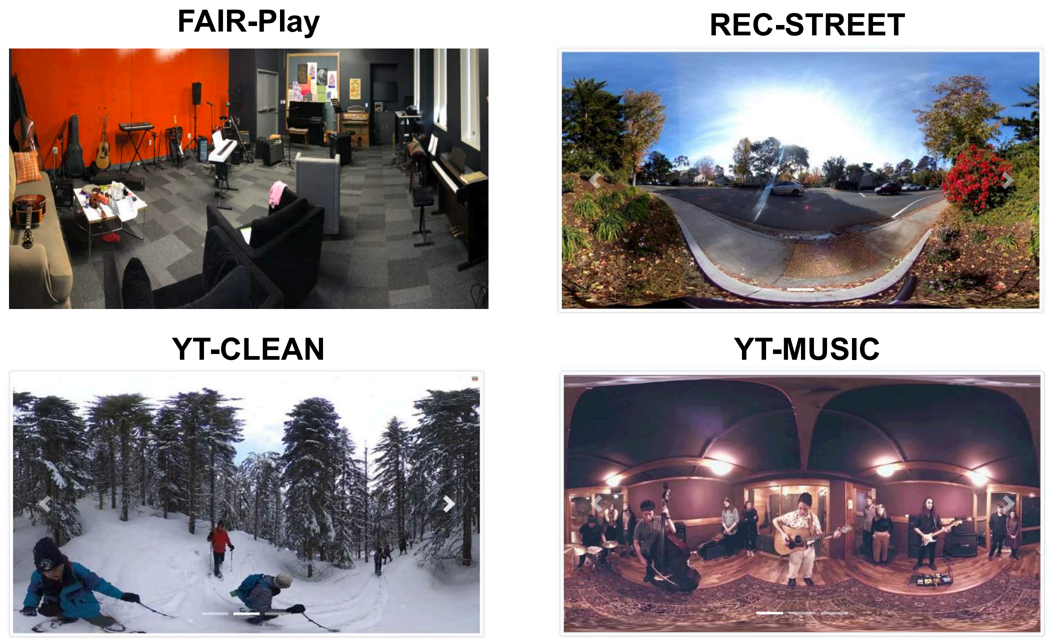


Figure 7.5: Four challenging datasets: FAIR-Play, REC-STREET [155], YT-CLEAN [155], and YT-MUSIC [155].

7.2 Experiments

We validate our approach for generation and separation.

7.2.1 Datasets

We use four challenging datasets (see Fig. 7.5) spanning a wide variety of sound sources, including musical instruments, street scenes, travel, and sports.

- **FAIR-Play:** Our new dataset consists of 1,871 10s clips of videos recorded in a music room (Fig. 7.2). The videos are paired with binaural audios of high quality recorded by a professional binaural microphone.

We create 10 random splits by splitting the data into train/val/test splits of 1,497/187/187 clips, respectively.

- **REC-STREET:** A dataset collected by [155] using a Theta V 360° camera with TA-1 spatial audio microphone. It consists of 43 videos (3.5 hours) of outdoor street scenes.
- **YT-CLEAN:** This dataset contains in-the-wild 360° videos from YouTube crawled by [155] using queries related to spatial audio. It consists of 496 videos of a small number of super-imposed sources, such as people talking in a meeting room, and outdoor sports.
- **YT-MUSIC:** A dataset that consists of 397 YouTube videos of music performances collected by [155]. It is their most challenging dataset due to the large number of mixed sources (voices and instruments).

To our knowledge, FAIR-Play is the first dataset of its kind that contains videos of professional recorded binaural audio. For REC-STREET, YT-CLEAN and YT-MUSIC, we split the videos into 10s clips and divide them into train/val/test splits based on the provided split1. These datasets only contain ambisonics, so we use a binaural decoder to convert them to binaural audio. Specifically, we use the head related transfer function (HRTF) from NH2 subject in the ARI HRTF Dataset[3] to perform decoding. For our FAIR-Play dataset, half of the training data is used to train the MONO2BINAURAL

[3] `http://www.kfs.oeaw.ac.at/hrtf`

network, and the other half is reserved for audio-visual source separation experiments.

7.2.2 Mono2Binaural Generation Accuracy

We evaluate the quality of our predicted binaural audio by using common metrics as well as two user studies. We compare to the following baselines:

- **Ambisonics [155]**: We use the pre-trained models provided by [155] to predict ambisonics. The models are trained on the same data as our method. Then we use the binaural decoder to convert the predicted ambisonics to binaural audio. This baseline is not available for the BINAURAL-MUSIC-ROOM dataset.
- **Audio-Only:** To determine if visual information is essential to perform MONO2BINAURAL conversion, we remove the visual stream and implement a baseline using only audio as input. All other settings are the same except that only audio features are passed to the up-convolution layers for binaural audio prediction.
- **Flipped-Visual:** During testing, we flip the accompanying visual frames of the mono audios to perform prediction using the wrong visual information.
- **Mono-Mono:** A straightforward baseline that copies the mixed monaural audio onto both channels to create a fake binaural audio.

We report two metrics: 1) **STFT Distance:** The euclidean distance

	FAIR-Play		REC-STREET		YT-CLEAN		YT-MUSIC	
	STFT	ENV	STFT	ENV	STFT	ENV	STFT	ENV
Ambisonics [155]	-	-	0.744	0.126	1.435	0.155	1.885	0.183
Audio-Only	0.966	0.141	0.590	0.114	1.065	0.131	1.553	0.167
Flipped-Visual	1.145	0.149	0.658	0.123	1.095	0.132	1.590	0.165
Mono-Mono	1.155	0.153	0.774	0.136	1.369	0.153	1.853	0.184
Mono2Binaural (Ours)	**0.836**	**0.132**	**0.565**	**0.109**	**1.027**	**0.130**	**1.451**	**0.156**

Table 7.1: Quantitative results of binaural audio prediction on four diverse datasets. We report the STFT distance and the envelope distance; lower is better. For FAIR-Play, we report the average results across 10 random splits. The results have a standard error of approximately 5×10^{-2} for STFT distance and 3×10^{-3} for ENV distance on average.

between the ground-truth and predicted complex spectrograms of the left and right channels:

$$\mathcal{D}_{\{\text{STFT}\}} = ||\mathbf{X}^L - \tilde{\mathbf{X}}^L||_2 + ||\mathbf{X}^R - \tilde{\mathbf{X}}^R||_2.$$

2) **Envelope (ENV) Distance:** Direct comparison of raw waveforms may not capture perceptual similarity well. Following [155], we take the envelope of the signals, and measure the euclidean distance between the envelopes of the ground-truth left and right channels and the predicted signals. Let $E[x(t)]$ denote the envelope of signal $x(t)$. The envelope distance is defined as:

$$\mathcal{D}_{\{\text{ENV}\}} = ||E[x^L(t)] - E[\tilde{x}^L(t)||_2 + ||E[x^R(t)] - E[\tilde{x}^R(t)||_2.$$

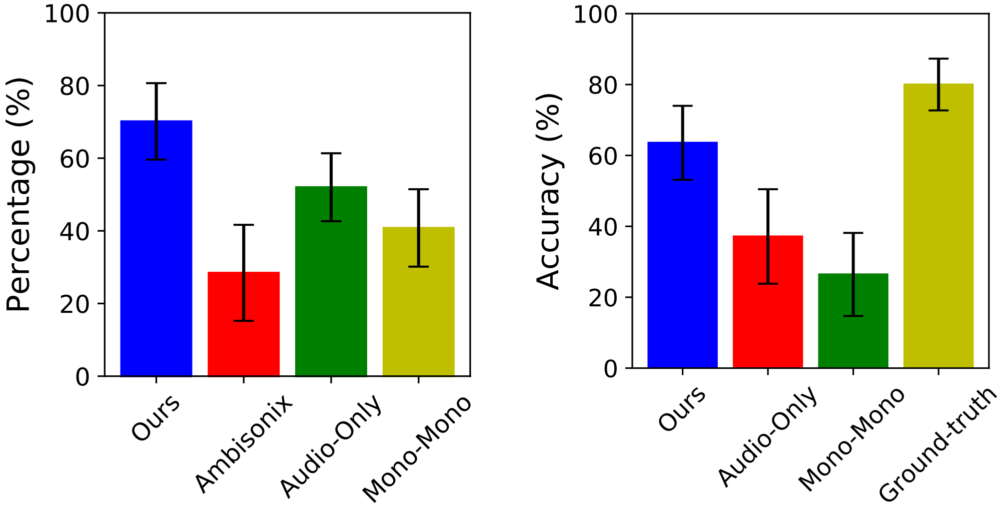
Results: Table 7.1 shows the binaural generation results. Our method outperforms all baselines consistently on all four datasets. Our mono2binaural approach performs better than the Audio-Only baseline, indicating the visual stream is essential to guide conversion. Note that the Audio-Only baseline uses the same network design as our method, so it has reasonably good performance. Still, we find our method outperforms it most when object(s) are not simply located in the center. Flipped-Visual performs much worse, demonstrating that our network properly learns to localize sound sources to predict binaural audio correctly.

The Ambisonics [155] approach does not do as well. We hypothesize several reasons. The method predicts four channel ambisonics directly, which must be converted to binaural audio. While ambisonics have the advantage of being a more general audio representation that is ideal for 360° video, predicting ambisonics first and then decoding to binaural audio for deployment can introduce artifacts that make the binaural audio less realistic. Better head-related transfer functions could help to render more realistic binaural audio from ambisonics, but this remains active research [133, 160].[4] Furthermore, manually inspecting the results, we find that the decoded binaural audio by [155] conveys spatial sensation, but it is less accurate and stable than our method. Our approach directly formulates the audio spatialization problem in terms of the two-channel binaural audio that listeners ultimately hear, which yields better accuracy.

Our video results[5] show qualitative results including failure cases. Our system can fail when there are multiple objects of similar appearance, e.g. multiple human speakers. Our model incorrectly spatializes the audio, because the people are too visually similar. However, when there is only one human

[4] We experimented with multiple ambisonics-binaural decoding solutions and report the best results for [155] in Table 7.1.

[5] `http://vision.cs.utexas.edu/projects/2.5D_visual_sound/`



(a) User study 1

(b) User study 2

Figure 7.6: User studies to test how listeners perceive the predicted binaural audio. For the first study, the participants are asked to compare two predicted binaural audios generated by our method and a baseline; For the second study, the participants are asked to name the direction they hear a particular sound coming from. Both studies suggest that the predicted binaural audio form our method presents listeners a much more accurate spatial audio experience.

speaker amidst other sounds, it can successfully perform audio spatialization. Future work incorporating motion may benefit instance-level spatialization.

User Studies: Having quantified the advantage of our method in Table 7.1, we now report real user studies. To test how well the predicted binaural audio makes a listener feel the 3D sensation, we conduct two user studies.

For the first study, the participants listen to a 10s ground-truth binaural audio and see the visual frame. Then they listen to two predicted binaural audios generated by our method and a baseline (Ambisonics, Audio-Only, or Mono-Mono). After listening to each pair, participants are asked which of the two creates a better 3D sensation that matches the ground-truth binaural audio. We recruited 18 participants with normal hearing. Each listened to 45 pairs spanning all the datasets. Fig. 7.6a shows the results. We report the percentage of times each method is chosen as the preferred one. We can see that the binaural audio generated by our method creates a more realistic 3D sensation.

For the second user study, we ask participants to name the direction they hear a particular sound coming from. Using the FAIR-Play data, we randomly select 10 instrument video clips where some player is located in the left/center/right of the visual frames. We ask every participant to *only listen* to the ground-truth or predicted binaural audio from our method or a baseline, and then choose the direction the sound of a specified instrument is coming from. Note that for this study, we input real mono audio recorded by the GoPro mic for binaural audio prediction. Fig. 7.6b shows the results from the 18 participants. The true recorded binaural audio is of high quality, and the listeners can often easily perceive the correct direction. However, our predicted binaural audio also clearly conveys directionality. Compared to the baselines, ours presents listeners a much more accurate spatial audio experience.

7.2.3 Localizing the Sound Sources

Does the network attend to the locations of the sound sources when performing binauralization? As a byproduct of our MONO2BINAURAL training, we can use the network to perform sound source localization. We use a

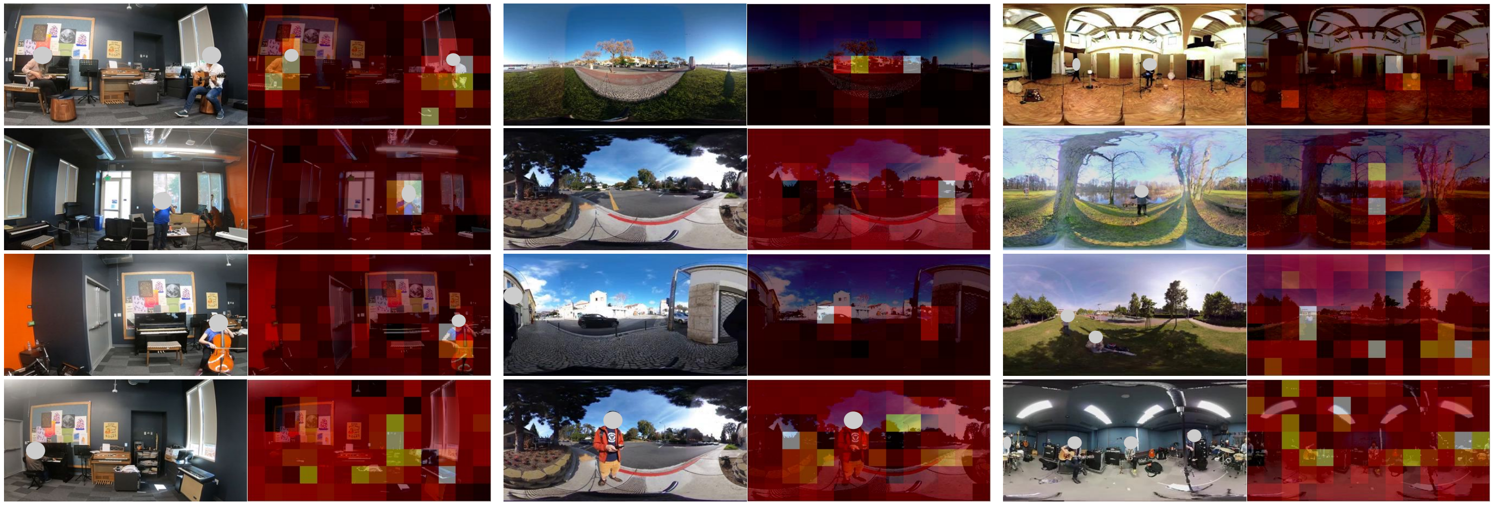


Figure 7.7: Visualizing the key regions the visual network focuses on when performing MONO2BINAURAL conversion. Each pair of images shows the frame accompanying the monaural audio (left) and the heatmap of the key regions overlaid (right).

mask of size 32 × 32 to replace image regions with image mean values, and forward the masked frame through the network to predict binaural audio. Then we compute the loss, and repeat by placing the mask at different locations of the frame. Finally, we highlight the regions which, when replaced, lead to the largest losses. They are considered the most important regions for MONO2BINAURAL conversion, and are expected to align with sound sources.

Fig. 7.7 shows examples. The highlighted key regions correlate quite well with sound sources. They are usually the instruments playing in the music room, the moving cars in street scenes, the place where an activity is going on, *etc.* The final row shows some failure cases. The model can be confused when there are multiple similar instruments in view, or silent or noisy scenes. Sound sources in YT-Clean and YT-Music are especially difficult to spatialize and localize due to diverse and/or large number of sound sources.

7.2.4 Audio-Visual Source Separation

In Chapters 3, 4, and 5, we introduced our three approaches for audio-visual source separation, a key problem in audio-visual analysis. In this chapter, having demonstrated our predicted binaural audio creates a better 3D sensation, we now examine its impact on audio-visual source separation using the FAIR-Play dataset. The dataset contains object-level sounds of diverse sound making objects (instruments), which is well-suited for the Mix-and-Separate audio-visual source separation approach we adopt. We train on the held-out data of FAIR-Play, and test on 10 typical single-instrument video

	SDR	SIR	SAR
Mono	2.57	4.25	10.12
Mono-Mono	2.43	4.01	10.15
Predicted Binaural (Ours)	**3.01**	**5.03**	**10.24**
GT Binaural (upper bound)	3.25	5.32	10.60

Table 7.2: Audio-visual source separation results. SDR, SIR, SAR are reported in dB; higher is better.

clips from the val/test set, with each representing one unique instrument in our dataset. We pairwise mix each video clip and perform separation, for a total of 45 test videos.

In addition to the ground truth binaural (upper bound) and the Mono-Mono baseline defined above, we compare to a **Mono** baseline that takes monaural audio as input and separates monaural audios for each source. Mono represents the current norm of performing audio-visual source separation using only single-channel audio [75, 163, 258]. We stress that all other aspects of the networks are the same, so that any differences in performance can be attributed to our binauralization self-supervision. To evaluate source separation quality, we use the widely used mir eval library [175], and the standard metrics: Signal-to-Distortion Ratio (SDR), Signal-to-Interference Ratio (SIR), and Signal-to-Artifact Ratio (SAR). Table 7.2 shows the results. We obtain large gains by inferring binaural audio. The inferred binaural audio offers a more informative audio representation compared to the original monaural audio, leading to cleaner separation. See our qualitative video[5] for examples.

7.3 Conclusions

In this chapter, I presented an approach to convert single channel audio into binaural audio by leveraging object/scene configurations in the visual frames. The predicted 2.5D visual sound offers a more immersive audio experience. Our MONO2BINAURAL framework achieves state-of-the-art performance on audio spatialization. Moreover, using the predicted binaural audio as a better audio representation, we boost a modern model for audio-visual source separation.

However, our frame-based model can get confused when there are multiple instruments of similar appearance in view. Motion analysis may be needed in order to perform instance-level spatialization. Scene sounds are not explicitly modeled for our current model, so it can find it difficult to predict spatial audio in complex environments of diverse sounds. Our model performs the best in the music room setting, but does not generalize as well to other novel domains. It would be important and interesting to train a more generalizable model that can infer realistic spatial audio for "in the wild" videos of normal field of view. Nevertheless, as the first approach to perform visually-guided MONO2BINAURAL audio spatialization, the obtained results constitute a noticeable step towards offering listeners more immersive 3D sound sensation. Generating binaural audio for off-the-shelf video can potentially close the gap between transporting audio and visual experiences, enabling new applications in VR/AR.

My work presented in the previous several chapters of this dissertation all leverage videos for audio-visual learning. However, humans learn not only by watching these passively captured videos of audio-visual streams, but also by actively interacting with the environment to learn about the world. In the final component of my dissertation, I will present VisualEchoes, an approach to learn by using audio to actively interact with the physical world through echolocation.

Chapter 8

Spatial Image Representation Learning through Echolocation

[1]In the previous chapters, I introduced several approaches for audio-visual learning from videos, which are passively captured. In this chapter, I present VisualEchoes, an approach to actively interact with the physical world using audio for spatial image representation learning. This work was published in ECCV 2020 [74].

The perceptual and cognitive abilities of embodied agents are inextricably tied to their physical being. We perceive and act in the world by making use of all our senses—especially looking and listening. We see our surroundings to avoid obstacles, listen to the running water tap to navigate to the kitchen, and infer how far away the bus is once we hear it approaching.

As discussed in the last chapter, by using *two* ears, we perceive spatial sound. Not only can we identify the sound-emitting object (e.g., the revving

[1] The work in this chapter was supervised by Prof. Kristen Grauman and was originally published in: "VisualEchoes: Spatial Image Representation Learning through Echolocation". Ruohan Gao, Changan Chen, Ziad Al-Halah, Carl Schissler, and Kristen Grauman. In Proceedings of the European Conference on Computer Vision, Virtual, August 2020.

engine corresponds to a bus), but also we can determine that object's location. Critically, even beyond objects, audio is also rich with information about the *environment* itself. The sounds we receive are a function of the geometric structure of the space around us and the materials of its major surfaces [12]. In fact, some animals capitalize on these cues by using *echolocation*—actively emitting sounds to perceive the 3D spatial layout of their surroundings [185].

We propose to learn image representations from echoes. Motivated by how animals and blind people obtain spatial information from echo responses, first we explore to what extent the echoes of chirps generated in a scanned 3D environment are predictive of the depth in the scene. Then, we introduce VisualEchoes, a novel image representation learning method based on echolocation. Given a first-person RGB view and an echo audio waveform, our model is trained to predict the correct camera orientation at which the agent would receive those echoes. In this way, the representation is forced to capture the alignment between the sound reflections and the (visually observed) surfaces in the environment. At test time, we observe only pixels—no audio. Our learned VisualEchoes encoder better reveals the 3D spatial cues embedded in the pixels, as we demonstrate in three downstream tasks.

Our approach offers a new way to learn image representations without manual supervision by *interacting* with the environment. In pursuit of this high-level goal there is exciting—though limited—prior work that learns visual features by touching objects [7, 164, 169, 172] or moving in a space [6, 73, 118]. Unlike mainstream "self-supervised" feature learning work that crafts pretext tasks for large static repositories of human-taken images or video (e.g., colorization [255], jigsaw puzzles [161], audio-visual correspondence [13, 130]), in *interaction-based feature learning* an embodied agent[2] performs physical actions in the world that dynamically influence its own first-person observations and possibly the environment itself. Both paths have certain advantages: while conventional self-supervised learning can capitalize on massive static datasets of human-taken photos, interaction-based learning allows an agent to "learn by acting" with rich multi-modal sensing. This has the advantage of learning features adaptable to new environments. Unlike any prior work, we explore feature learning from echoes.

Our contributions are threefold: 1) We explore the spatial cues contained in echoes, analyzing how they inform depth prediction; 2) We propose VisualEchoes, a novel interaction-based feature learning framework that uses echoes to learn an image representation and does not require audio at test time; 3) We successfully validate the learned spatial representation for the fundamental downstream vision tasks of monocular depth prediction, surface normal estimation, and visual navigation, with results comparable to or even outperforming heavily supervised pre-training baselines.

In Sec 8.1, I describe our co-separation approach for learning audio-visual source separation. Then I present key experiments and results in Sec 8.2.

[2] person, robot, or simulated robot

8.1 Approach

Our goals are to show that echoes convey spatial information, to learn visual representations by echolocation, and to leverage the learned representations for downstream tasks. In the following, we first describe how we simulate echoes in 3D environments (Sec. 8.1.1). Then we perform a case study to demonstrate how echoes can benefit monocular depth prediction (Sec. 8.1.2). Next, we present VisualEchoes, our interaction-based feature learning formulation to learn image representations (Sec. 8.1.3). Finally, we exploit the learned visual representation for monocular depth, surface normal prediction, and visual navigation (Sec. 8.1.4).

8.1.1 Echolocation Simulation

Our echolocation simulation is based on recent work on audio-visual navigation that builds a realistic acoustic simulation called SoundSpaces [28] on top of the Habitat [189] platform and Replica environments [202]. Habitat [189] is an open-source 3D simulator that supports efficient RGB, depth, and semantic rendering for multiple datasets [27,202,243]. Replica is a dataset of 18 apartment, hotel, office, and room scenes with 3D meshes and high definition range (HDR) textures and renderable reflector information. SoundSpaces [28] simulates acoustics by pre-computing room impulse responses (RIR) between all pairs of possible source and receiver locations, using a form of audio ray-tracing [217]. An RIR is a transfer function between the sound source and the sound microphone, and it is influenced by the room geometry, materials, and the sound source location [134]. The sound received at the listener location is computed by convolving the appropriate RIR with the waveform of the source sound.

We use the binaural RIRs for all Replica environments to generate echoes for our approach. As the source audio “chirp” we use a sweep signal from 20Hz-20kHz (the human-audible range) within a duration of 3ms. While technically any emitted sound could provide some echo signal from which to learn, our design (1) intentionally provides the response for a wide range of frequencies and (2) does so in a short period of time to avoid overlap between echoes and direct sounds. We place the source at the *same* location as the receiver and convolve the RIR for this source-receiver pair with the sweep signal. In this way, we compute the echo responses that would be received at the agent’s microphone locations. We place the agents at all navigable points on the grid (every 0.5m [28]) and orient the agent in four cardinal directions (0°, 90°, 180°, 270°) so that the rendered egocentric views (RGB and depth) and echoes capture room geometry from different locations and orientations.

Fig. 8.1 illustrates how we perform echolocation for one scene environment. The agent goes to the densely sampled navigable locations marked with yellow dots and faces four orientations at each location. It actively emits omnidirectional chirp signals and records the echo responses received when facing each direction. Note that the spectrograms of the sounds received at the left (L) and right (R) ears reveal that the agent first receives the direct sound (strong bright curves), and then receives different echoes for the left and right

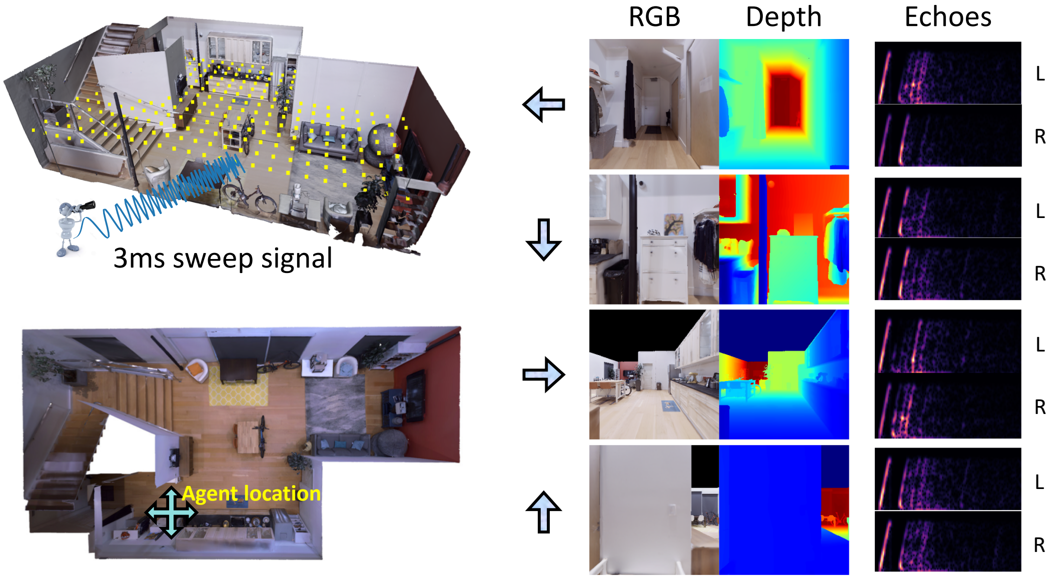


Figure 8.1: Echolocation simulation in real-world scanned environments. During training, the agent goes to the densely sampled locations marked with yellow dots. The left bottom figure illustrates the top-down view of one Replica scene where the agent's location is marked. The agent actively emits 3 ms omnidirectional sweep signals to get echo responses from the room. The right column shows the corresponding RGB and depth of the agent's view as well as the echoes received in the left and right ears when the agent faces each of the four directions.

microphones due to ITD, ILD, and pinnae reflections. The subtle difference in the two spectrograms conveys cues about the spatial configuration of the environment, as can be observed in the last column of Fig. 8.1.

8.1.2 Case Study: Spatial Cues in Echoes

With the synchronized egocentric views and echo responses in hand, we now conduct a case study to investigate the spatial cues contained in echo responses in these realistic indoor 3D environments. We have two questions:

(1) can we directly predict depth maps purely from echoes? and (2) can we use echoes to augment monocular depth estimation from RGB? Answering these questions will inform our ultimate goal of devising a interaction-supervised visual feature learning approach leveraging echoes only at training time (Sec. 8.1.3). Furthermore, it can shed light on the extent to which low-cost audio sensors can replace depth sensors, which would be especially useful for navigation robots under severe bandwidth or sensing constraints, e.g., nano drones [152, 166].

Note that these two goals are orthogonal to that of prior work performing depth prediction from a single view [52, 68, 106, 143, 245]. Whereas they focus on developing sophisticated loss functions and architectures, here we explore how an agent *actively interacting with the scene acoustically* may improve its depth predictions. Our findings can thus complement existing monocular depth models.

We devise an RGB+Echo2Depth network (and its simplified variants using only RGB or echo) to test the settings of interest. The RGB+Echo2Depth network predicts a depth map based on the agent's egocentric RGB input and the echo response it receives when it emits a chirp standing at that position and orientation in the 3D environment. The core model is a multi-modal U-Net [184]; see Fig. 8.2. To directly measure the spatial cues contained in echoes alone, we also test a variant called Echo2Depth. Instead of performing up-sampling based on the audio-visual representation, this model drops the RGB input, reshapes the audio feature, and directly upsamples from the audio rep-

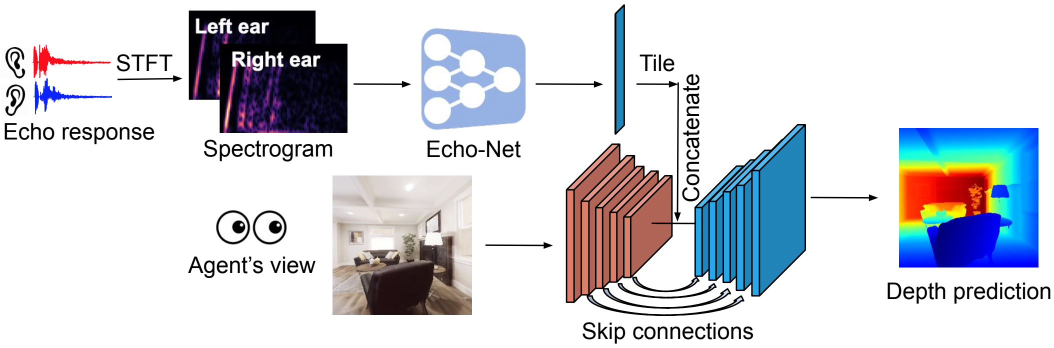


Figure 8.2: Our RGB+ECHO2DEPTH network takes the echo responses and the corresponding egocentric RGB view as input, and performs joint audio-visual analysis to predict the depth map for the input image. The injected echo response provides additional cues of the spatial layout of the scene. Note: in later sections we define networks that do not have access to the audio stream at test time.

	RMS ↓	REL ↓	log 10 ↓	$\delta < 1.25$ ↑	$\delta < 1.25^2$ ↑	$\delta < 1.25^3$ ↑
AVERAGE	1.070	0.791	0.230	0.235	0.509	0.750
ECHO2DEPTH	0.713	0.347	0.134	0.580	0.772	0.868
RGB2DEPTH	0.374	0.202	0.076	0.749	0.883	0.945
RGB+ECHO2DEPTH	**0.346**	**0.172**	**0.068**	**0.798**	**0.905**	**0.950**

Table 8.1: Case study depth prediction results. ↓ lower better, ↑ higher better.

resentation. Similarly, to measure the cues contained in the RGB alone, a variant called RGB2DEPTH drops the echoes and predicts the depth map purely based on the visual features. The RGB2DEPTH model represents existing monocular depth prediction approaches that predict depth from a single RGB image, in the context of the same architecture design as RGB+ECHO2DEPTH to allow apples-to-apples calibration of our findings. We use RGB images of spatial dimension 128 × 128. See [78] for network details and loss functions used to train the three models.

Table 8.1 shows the quantitative results of predicting depth from only

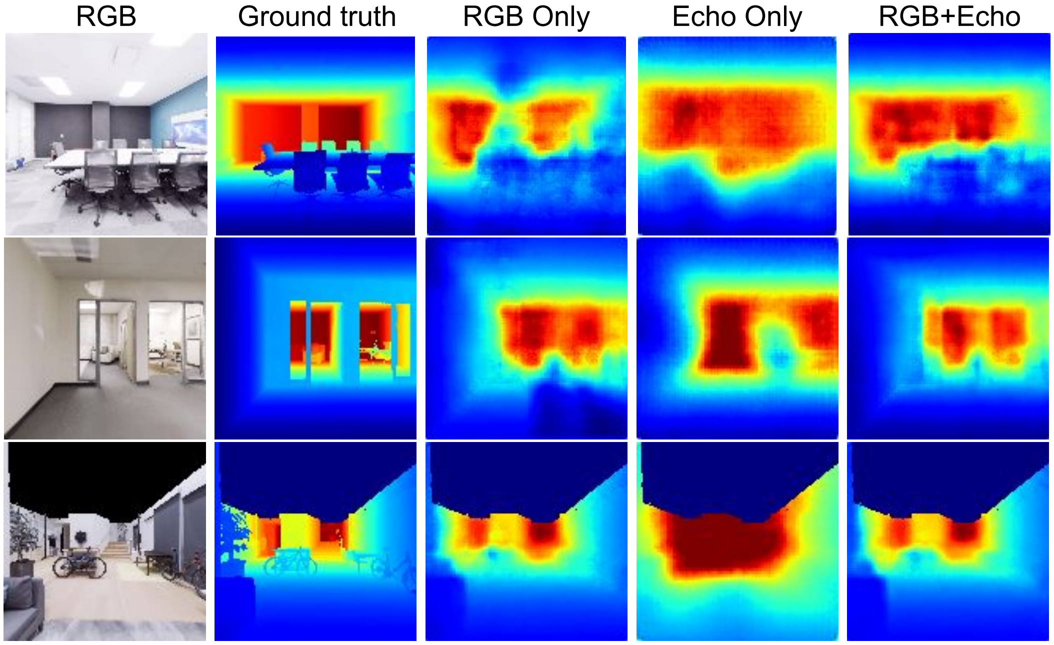


Figure 8.3: Qualitative results of our case study on monocular depth estimation in unseen environments using echoes. Together with the quantitative results (Tab. 8.1), these examples show that echoes contain useful spatial cues that inform a visual spatial task. For example, in row 1, the RGB+Echo model better infers the depth of the column on the back wall, whereas the RGB-Only model mistakenly infers the strong contours to indicate a much closer surface. The last row shows a typical failure case (see text). See [74] for more examples.

echoes, only RGB, or their combination. We evaluate on a heldout set of three Replica environments (comprising 1,464 total views) with standard metrics: root mean squared error (RMS), mean relative error (REL), mean log 10 error (log 10), and thresholded accuracy [52, 106]. We can see that depth prediction is possible purely from echoes. Augmenting traditional single-view depth estimation with echoes (bottom row) achieves the best performance by leveraging the additional acoustic spatial cues. Echoes alone are naturally weaker than

RGB alone, yet still better than the simple AVERAGE baseline that predicts the average depth values in all training data.

Fig. 8.3 shows qualitative examples. It is clear that echo responses indeed contain cues of the spatial layout; the depth map captures the rough room layout, especially its large surfaces. When combined with RGB, the predictions are more accurate. The last row shows a typical failure case, where the echoes alone cannot capture the depth as well due to far away surfaces with weaker echo signals.

8.1.3 VisualEchoes Spatial Representation Learning Framework

Having established the scope for inferring depth from echoes, we now present our VISUALECHOES model to leverage echoes for visual representation learning. We stress that our approach assumes audio/echoes are available only during training; at test time, an RGB image alone is the input.

The key insight of our approach is that the echoes and visual input should be consistent. This is because both are functions of the same latent variable—the 3D shape of the environment surrounding the agent's co-located camera and microphones. We implement this idea by training a network to predict their correct association.

In particular, as described in Sec. 8.1.1, at any position in the scene, we suppose the agent can face four orientations, i.e., at an azimuth angle of 0°, 90°, 180°, and 270°. When the agent emits the sweep signal (chirp) at a certain position, it will hear different echo responses when it faces each

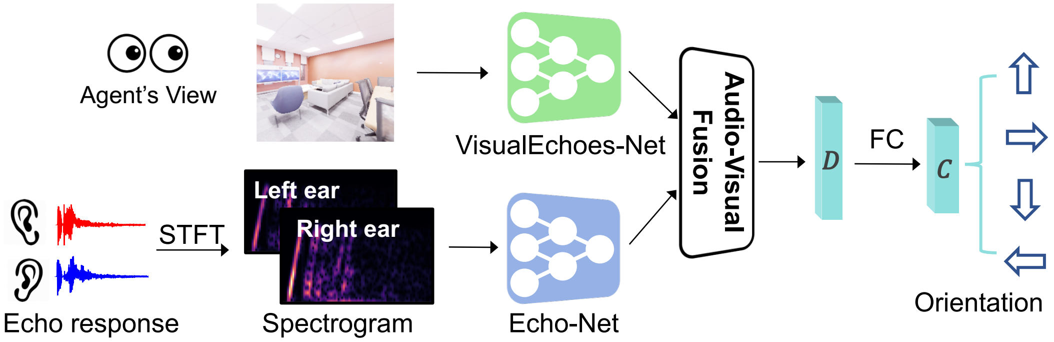


Figure 8.4: Our VisualEchoes network takes the agent's current RGB view as visual input, and the echo responses from one of the four orientations as audio input. The goal is to predict the orientation at which the agent would receive the input echoes based on analyzing the spatial layout in the image. After training with RGB and echoes, the VisualEchoes-Net is a pre-trained encoder ready to extract spatially enriched features from novel RGB images, as we validate with multiple downstream tasks (cf. Sec. 8.1.4).

different orientation. If the agent correctly interprets the spatial layout of the current view from *visual* information, it should be able to tell whether that visual input is congruous with the echo response it hears. Furthermore, and more subtly, to the extent the agent implicitly learns about probable views surrounding its current egocentric field of view (e.g., what the view just to its right may look like given the context of what it sees in front of it), it should be able to tell which direction the received echo *would* be congruous with, if not the current view.

We introduce a representation learning network to capture this insight. See Fig. 8.4. The visual stream takes the agent's current RGB view as input, and the audio stream takes the echo response received from one of the four orientations—not necessarily the one that coincides with the visual stream orientation. The fusion layer fuses the audio and visual information to generate an audio-visual feature of dimension D. A final fully-connected layer is used to make the final prediction among four classes. See [80] and Sec. 8.2 for architecture details.

The four classes are defined as follows:

$\uparrow$: The echo is received from the same orientation as the agent's current view.

$\rightarrow$: The echo is received from the orientation if the agent turns right by 90°.

$\downarrow$: The echo is received from the orientation opposite the agent's current view.

$\leftarrow$: The echo is received from the orientation if the agent turns left by 90°.

The network is trained with cross-entropy loss. Note that although the emitted source signal is always the same (3 ms *omnidirectional* sweep signal, cf. Sec. 8.1.1), the agent hears different echoes when facing the four directions because of the shape of the ears and the head shadowing effect modeled in the binaural head-related transfer function (HRTF). Since the classes above are defined relative to the agent's current view, it can only tell the orientation for which it is receiving the echoes if it can correctly interpret the 3D spatial layout within the RGB input. In this way, the agent's aural interaction with the scene enhances spatial feature learning for the visual stream.

The proposed idea generalizes trivially to use more than four discrete orientations—and even arbitrary orientations if we were to use regression rather than classification. The choice of four is simply based on the sound simulations available in existing data [28], though we anticipate it is a good granularity to capture the major directions around the agent. Our training paradigm requires the representation to discern mismatches between the image and echo using echoes generated from the same physical position on the ground plane but different orientations. This is in line with our interactive embodied agent motivation, where an agent can look ahead, then turn and hear echoes from another orientation at the same place in the environment, and learn their (dis)association. In fact, ecological psychologists report that humans can perform more accurate echolocation when moving, supporting the rationale of our design [185, 203]. Furthermore, our design ensures the mismatches are "hard" examples useful for learning spatial features because the audio-visual data at offset views will naturally be related to one another (as opposed to views or echoes from an unrelated environment).

8.1.4 Downstream Tasks for the Learned Spatial Representation

Having introduced our VisualEchoes feature learning framework, next we describe how we repurpose the learned visual representation for three fundamental downstream tasks that require spatial reasoning: monocular depth prediction, surface normal estimation, and visual navigation. See Fig. 8.5 for an illustration of the three tasks. For each task, we adopt strong models from

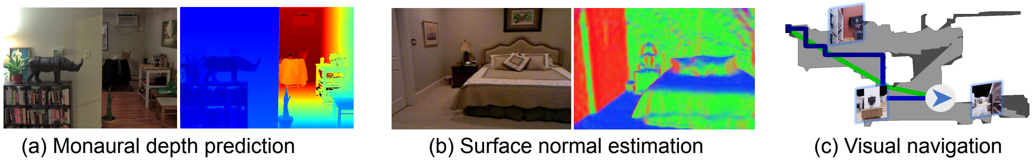


Figure 8.5: Illustration of the three downstream tasks that require spatial reasoning.

the literature and swap in our pre-trained encoder VisualEchoes-Net for the RGB input.

Monocular Depth Prediction: We explore how our echo-based pre-training can benefit performance for traditional monocular depth prediction. Note that unlike the case study in Sec. 8.1.2, in this case there are no echo inputs at test time, only RGB. To evaluate the quality of our learned representation, we adopt a strong recent approach for monocular depth prediction [106] consisting of several novel loss functions and a multi-scale network architecture that is based on a backbone network. We pre-train ResNet-50 [99] using VisualEchoes and use it as the backbone for comparison with [106].

Surface Normal Estimation: We also evaluate the learned spatial representation to predict surface normals from a single image, another fundamental mid-level vision task that requires spatial understanding of the geometry of the surfaces [66]. We adopt the the state-of-the-art pyramid scene parsing network PSPNet architecture [259] for surface normal prediction, again swapping in our pre-trained VisualEchoes network for the RGB feature backbone.

Visual Navigation: Finally, we validate on an embodied visual navigation task. In this task, the agent receives a sequence of RGB images as input and a point goal defined by a displacement vector relative to the starting position of the agent [11]. The agent is spawned at random locations and must navigate to the target location quickly and accurately. This entails reasoning about 3D spatial configurations to avoid obstacles and find the shortest path. We adopt a state-of-the-art reinforcement learning-based PointGoal visual navigation model [189]. It consists of a three-layer convolutional network and a fully-connected layer to extract visual feature from the RGB images. We pre-train its visual network using VisualEchoes, then train the full network end to end.

While other architectures are certainly possible for each task, our choices are based on both on the methods' effectiveness in practice, their wide use in the literature, and code availability. Our contribution is feature learning from echoes as a pre-training mechanism for spatial tasks, which is orthogonal to advances on architectures for each individual task. In fact, a key message of our results is that the VisualEchoes-Net encoder boosts multiple spatial tasks, under multiple different architectures, and on multiple datasets.

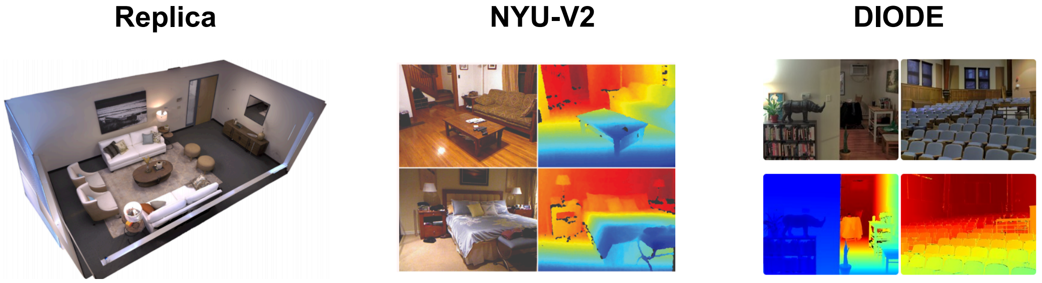


Figure 8.6: We evaluate on three detasets: Replica [202], NYU-V2 [193], and DIODE [215].

8.2 Experiments

We present experiments to validate VisualEchoes for three tasks and three datasets as shown in Fig. 8.6 (Replica [202], NYU-V2 [193], and DIODE [215]). The goal is to examine the impact of our features compared to either learning features for that task from scratch or learning features with manual semantic supervision. See [80] for details of the three datasets.

Implementation Details: All networks are implemented in PyTorch. For the echoes, we use the first 60 ms, which allows most of the room echo responses following the 3 ms chirp to be received. We use an audio sampling rate of 44.1 kHz. STFT is computed using a Hann window of length 64, hop length of 16, and FFT size of 512. The audio-visual fusion layer (see Fig. 8.4) concatenates the visual and audio feature, and then uses a fully-connected layer to reduce the feature dimension to $D = 128$. See [74] for details of the network architectures and optimization hyperparameters.

Evaluation Metrics: We report standard metrics for the downstream tasks. 1) *Monocular Depth Prediction:* RMS, REL, and others as defined above, following [52, 106]. 2) *Surface Normal Estimation:* mean and median of the angle distance and the percentage of good pixels (i.e., the fraction of pixels with cosine distance to ground-truth less than t) with $t = 11.25°, 22.5°, 30°$, following [66]. 3) *Visual Navigation:* success rate normalized by inverse path length (SPL), the distance to the goal at the end of the episode, and the distance to the goal normalized by the trajectory length, following [11].

8.2.1 Transferring VisualEchoes Features for RGB2Depth

Having confirmed echoes reveal spatial cues in Sec. 8.1.2, we now examine the effectiveness of VisualEchoes, our learned representation. Our model achieves 66% test accuracy on the orientation prediction pretext task, while chance performance is only 25%; this shows learning the visual-echo consistency task itself is possible.

First, we use the same RGB2Depth network from our case study in Sec. 8.1.2 as a testbed to demonstrate the learned spatial features can be successfully transferred to other domains. Instead of randomly initializing the RGB2Depth UNet encoder, we initialize with an encoder 1) pre-trained for our visual-echo consistency task, 2) pre-trained for image classification using ImageNet [41], or 3) pre-trained for scene classification using the MIT Indoor Scene dataset [174]. Throughout, aside from the standard ImageNet pre-training baseline, we also include MIT Indoor Scenes pre-training, in case

		RMS ↓	REL ↓	log 10 ↓	$\delta < 1.25$ ↑	$\delta < 1.25^2$ ↑	$\delta < 1.25^3$ ↑
Sup	ImageNet Pre-trained	0.356	0.203	0.076	0.748	0.891	0.948
	MIT Indoor Scene Pre-trained	0.334	0.196	0.072	0.770	0.897	0.950
Unsup	Scratch	0.360	0.214	0.078	0.747	0.879	0.940
	VisualEchoes (Ours)	**0.332**	**0.195**	**0.070**	**0.773**	**0.899**	**0.951**

(a) Replica

		RMS ↓	REL ↓	log 10 ↓	$\delta < 1.25$ ↑	$\delta < 1.25^2$ ↑	$\delta < 1.25^3$ ↑
Sup	ImageNet Pre-trained	0.812	0.249	0.102	0.589	0.855	0.955
	MIT Indoor Scene Pre-trained	0.776	0.239	0.098	0.610	0.869	0.959
Unsup	Scratch	0.818	0.252	0.103	0.586	0.853	0.950
	VisualEchoes (Ours)	**0.797**	**0.246**	**0.100**	**0.600**	**0.863**	**0.956**

(b) NYU-V2

		RMS ↓	REL ↓	log 10 ↓	$\delta < 1.25$ ↑	$\delta < 1.25^2$ ↑	$\delta < 1.25^3$ ↑
Sup	ImageNet Pre-trained	2.250	0.453	0.199	0.336	0.591	0.766
	MIT Indoor Scene Pre-trained	2.218	0.424	0.198	0.363	0.632	0.776
Unsup	Scratch	2.352	0.481	0.214	0.321	0.581	0.742
	VisualEchoes (Ours)	**2.223**	**0.430**	**0.198**	**0.340**	**0.610**	**0.769**

(c) DIODE

Table 8.2: Depth prediction results on the Replica, NYU-V2, and DIODE datasets. We use the RGB2Depth network from Sec. 8.1.2 for all models. Our VisualEchoes pre-training transfers well, consistently predicting depth better than the model trained from scratch. Furthermore, it is even competitive with the supervised models, whether they are pre-trained for ImageNet or MIT Indoor Scenes (1M/16K manually labeled images). ↓ lower better, ↑ higher better. (Un)sup = (un)supervised. We boldface the best unsupervised method.

it strengthens the baseline due to its domain alignment with the indoor scenes in Replica, DIODE, and NYU-2.[3]

Table 8.2 shows the results on all three datasets: Replica, NYU-V2, and DIODE. The model initialized with our pre-trained VisualEchoes network achieves much better performance compared to the model trained from

[3] Like the test datasets, MIT Indoor Scenes contains indoor scenes. Performance is similar when pre-training on Places [261], which is larger but contains diverse indoor and outdoor scenes.

	RMS ↓	REL ↓	log 10 ↓	$\delta < 1.25$ ↑	$\delta < 1.25^2$ ↑	$\delta < 1.25^3$ ↑
Scratch	0.360	0.214	0.078	0.747	0.879	0.940
SimpleVisualEchoes	0.340	0.198	0.073	0.763	0.892	0.948
BinaryMatching	0.345	0.199	0.074	0.760	0.889	0.944
VisualEchoes (Ours)	**0.332**	**0.195**	**0.070**	**0.773**	**0.899**	**0.951**

Table 8.3: Ablation study on Replica. See [74] for results on NYU-V2 and Diode.

scratch. Moreover, it even outperforms the supervised model pre-trained on scene classification in some cases. The ImageNet pre-trained model performs much worse; we suspect that the UNet encoder does not have sufficient capacity to handle ImageNet classification, and also the ImageNet domain is much different than indoor scene environments. This result accentuates that task similarity promotes positive transfer [254]: our unsupervised spatial pre-training task is more powerful for depth inference than a supervised semantic category pre-training task. See [74] for low-shot experiments varying the amount of training data.

We also perform an ablation study to demonstrate that the design of our spatial representation learning framework is essential and effective. We compare with the following two variants: SimpleVisualEchoes, which simplifies our orientation prediction task to two classes; and BinaryMatching, which mimics prior work [13] that leverages the correspondence between images and audio as supervision by training a network to decide if the echo and RGB are from the same environment. As shown in Table 8.3, our method performs much better than both baselines. See [74] for details.

		RMS ↓	REL ↓	log 10 ↓	$\delta < 1.25$ ↑	$\delta < 1.25^2$ ↑	$\delta < 1.25^3$ ↑
Sup	ImageNet Pre-trained [106]	**0.555**	**0.126**	**0.054**	**0.843**	**0.968**	**0.991**
	MIT Indoor Scene Pre-trained	0.711	0.180	0.075	0.730	0.925	0.979
Unsup	Scratch	0.804	0.209	0.086	0.676	0.897	0.967
	VisualEchoes (Ours)	0.683	0.165	0.069	0.762	0.934	0.981

(a) Depth prediction results on NYU-V2.

		Mean Dist. ↓	Median Dist. ↓	$t < 11.25°$ ↑	$t < 22.5°$ ↑	$t < 30°$ ↑
Sup	ImageNet Pre-trained	26.4	17.1	36.1	59.2	68.5
	MIT Indoor Scene Pre-trained	25.2	17.5	36.5	57.8	67.2
Unsup	Scratch	26.3	16.1	37.9	60.6	69.0
	VisualEchoes (Ours)	**22.9**	**14.1**	**42.7**	**64.1**	**72.4**

(b) Surface normal estimation results on NYU-V2. The results for the ImageNet Pre-trained baseline and the Scratch baseline are directly quoted from [91].

		SPL ↑	Distance to Goal ↓	Normalized Distance to Goal ↓
Sup	ImageNet Pre-trained	0.833	0.663	0.081
	MIT Indoor Scene Pre-trained	0.798	1.05	0.124
Unsup	Scratch	0.830	0.728	0.096
	VisualEchoes (Ours)	**0.856**	**0.476**	**0.061**

(c) Visual navigation performance in unseen Replica environments.

Table 8.4: Results for three downstream tasks. ↓ lower better, ↑ higher better.

8.2.2 Evaluating on Downstream Tasks

Next we evaluate the impact of our learned VisualEchoes representation on all three downstream tasks introduced in Sec. 8.1.4.

Monocular Depth Prediction: Table 8.4a shows the results.[4] All methods use the same settings as [106], where they evaluate and report results on NYU-V2. We use the authors' publicly available code[5] and use ResNet-50 as the encoder. See [74] for details. With this apples-to-apples comparison, the

[4] We evaluate on NYU-V2, the most widely used dataset for the task of single view depth prediction and surface normal estimation. The authors's code [91, 106] is tailored to this dataset.

[5] `https://github.com/JunjH/Revisiting_Single_Depth_Estimation`

difference in performance can be attributed to whether/how the encoder is pre-trained. Although our VisualEchoes features are learned from Replica, they transfer reasonably well to NYU-V2, outperforming models trained from scratch by a large margin. This result is important because it shows that despite training with simulated audio, our model generalizes to real-world test images. Our features also compare favorably to supervised models trained with heavy supervision.

Surface Normal Estimation: Table 8.4b shows the results. We follow the same setting as [91] and we use the authors' publicly available code.[6] Our model performs much better even compared to the ImageNet-supervised pre-trained model, demonstrating that our interaction-based feature learning framework via echoes makes the learned features more useful for 3D geometric tasks.

Visual Navigation: Table 8.4c shows the results. By pre-training the visual network, VisualEchoes equips the embodied agents with a better sense of room geometry and allows them to learn faster. Notably, the agent also ends much closer to the goal. We suspect it can better gauge the distance because of our VisualEchoes pre-training. Models pre-trained for classification on MIT Indoor Scenes perform more poorly than Scratch; again, this suggests features useful for recognition may not be optimal for a spatial task like point goal navigation.

[6] `https://github.com/facebookresearch/fair_self_supervision_benchmark`

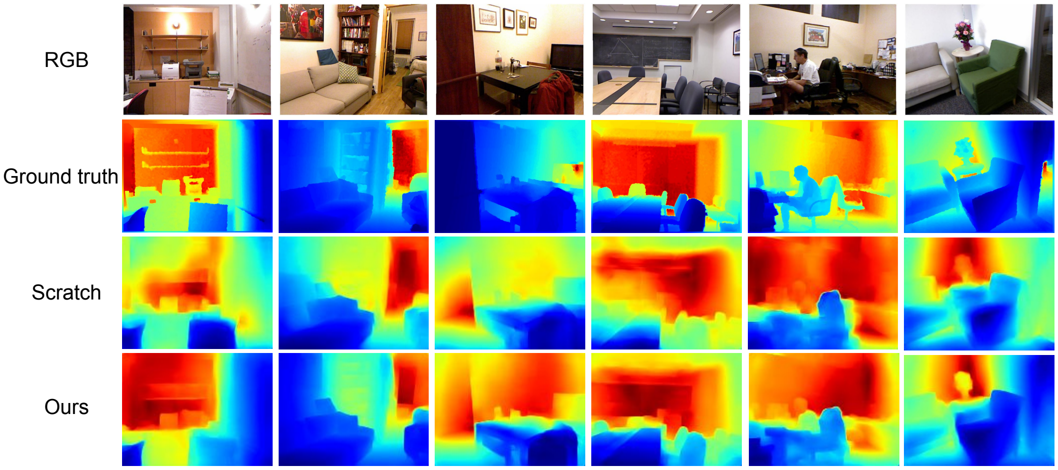


Figure 8.7: Qualitative results of monocular depth prediction on the NYU-V2 dataset.

This series of results on three tasks consistently shows the promise of our VisualEchoes features. We see that learning from echoes translates into a strengthened *visual* encoding. Importantly, while it is always an option to train multiple representations entirely from scratch to support each given task, our results are encouraging since they show the *same* fundamental interaction-based pre-training is versatile across multiple tasks.

8.2.3 Qualitative Results

Next, we show some qualitative results for the downstream tasks described in the last section. Fig. 8.7 and Fig. 8.8 show example results on monocular depth prediction and surface normal estimation, respectively. Using our pre-trained VisualVoice network as initialization leads to much more accurate depth prediction and surface normal estimation results compared to

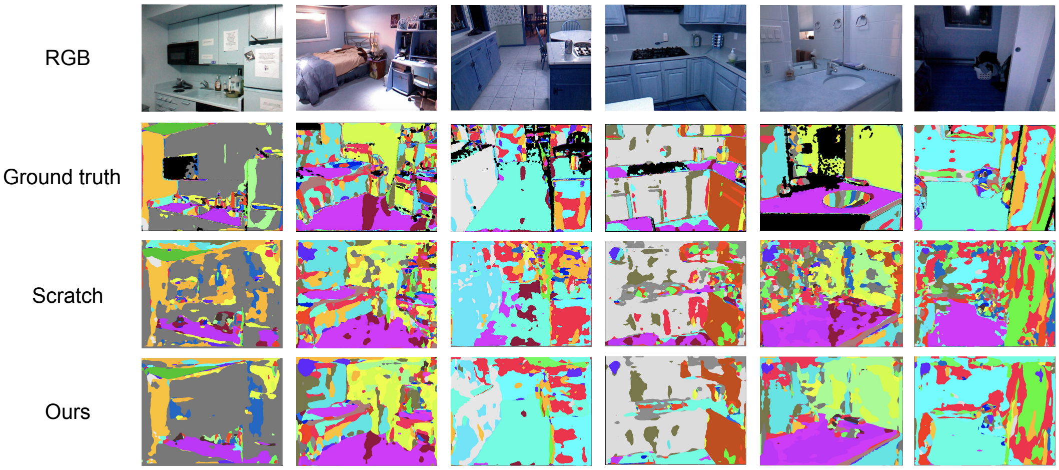


Figure 8.8: Qualitative results of surface normal estimation on the NYU-V2 dataset.

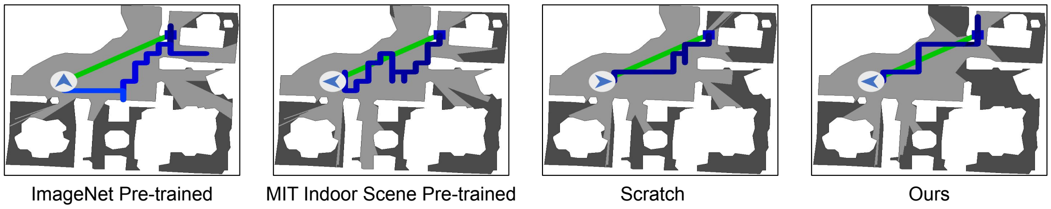


Figure 8.9: Qualitative examples of visual navigation trajectories on top-down maps. Blue square and arrow denote agent's starting and ending positions, respectively. The green path indicates the shortest geodesic path to the goal, and the agent's path is in dark blue. Agent path color fades from dark blue to light blue as time goes by. Note, the agent sees a sequence of egocentric views, not the map.

no pre-training, demonstrating the usefulness of the learned spatial features. Fig. 8.9 shows example navigation trajectories on top-down maps. Our visual-echo consistency pre-training task allows the agent to better interpret the room's spatial layout to find the goal more quickly than the baselines.

8.3 Conclusions

In this chapter, I presented an approach to learn spatial image representations via echolocation. We performed an in-depth study on the spatial cues contained in echoes and how they can inform single-view depth estimation. We showed that the learned spatial features can benefit three downstream vision tasks. Our work opens a new path for interaction-based representation learning for embodied agents and demonstrates the potential of learning spatial visual representations even with a limited amount of multisensory data.

While our current implementation learns from audio rendered in a simulator, the results show that the learned spatial features already benefit transfer to vision-only tasks in real photos outside of the scanned environments (e.g., the NYU-V2 [193] and DIODE [215] images), indicating the realism of what our system learned. Nonetheless, it will be interesting future work to capture the echoes on a real robot. I am also interested in pursuing these ideas within a sequential model, such that the agent could actively decide when to emit chirps and what type of chirps to emit to get the most informative echo responses.

Chapter 9

Conclusion and Future Work

In the preceding chapters, I have presented my thesis research on audio-visual learning with videos and embodied agents, leveraging audio itself as a supervision signal both semantically and spatially, in six stages:

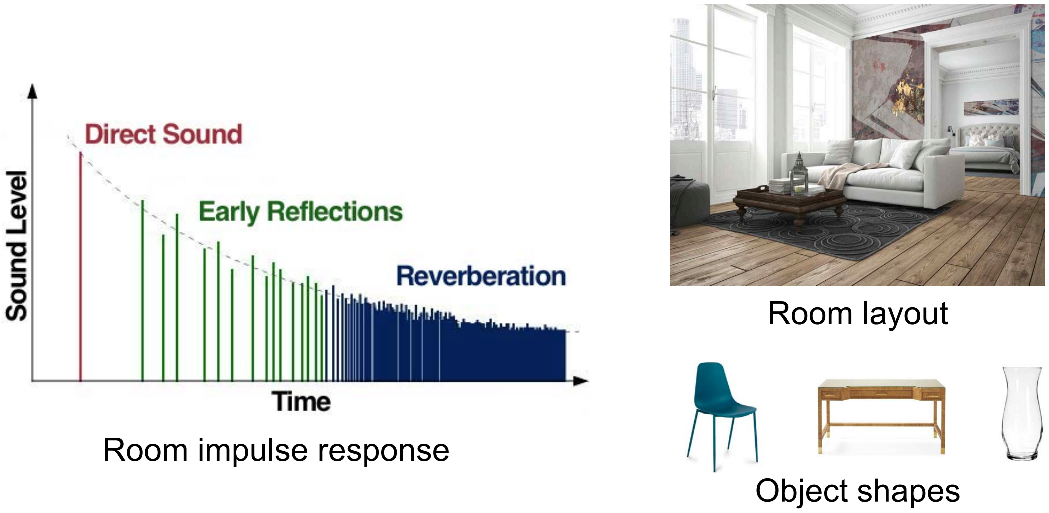
- *Learning to separate object sounds from unlabeled video*, in Chapter 3 [75].
- *Co-separating sounds of visual objects*, in Chapter 4 [77].
- VisualVoice*: Audio-visual speech separation with cross-modal consistency*, in Chapter 5 [78].
- *Listen to look: Action recognition by previewing audio*, in Chapter 6 [80].
- *2.5D visual sound*, in Chapter 7 [76].
- VisualEchoes*: Learning spatial image representations via echolocation*, in Chapter 8 [74].

My thesis research has focused on addressing the question: How can algorithms learn the "*What*" and "*Where*" about sound-making objects when multiple sound sources are present? Leveraging audio as a semantic signal, I have studied how to disentangle object sounds from unlabeled videos [75, 77, 78], and how to use audio as an efficient preview for action recognition in untrimmed videos [80]; Using audio as a spatial signal, I have studied how to infer binaural audio from monaural audio by leveraging visual cues in videos [76], and how to use echoes to learn spatial image representation [74].

Throughout my thesis research on audio-visual learning, I believe that unsupervised or self-supervised learning from multisensory data play a positive and crucial role in the future progress of Artificial Intelligence. Learning from how we humans perceive and act in the world by making use of all our senses, the long-term goal of my research is to build systems that can perceive as well as we do by combining all the multisensory inputs. My research presented in this dissertation has been mainly focusing on two paramount sensory streams: vision and audition. While the progress is encouraging, there is still a long way to go before I reach my ultimate goal. Below, I outline three main topics that I plan to pursue next beyond this Ph.D. thesis.

9.1 Audio and Geometry

Building on my work on leveraging the spatial signal in audio presented in Chapter 7 and Chapter 8, I would like to explore how audio can further reveal the geometry of the environment. The room impulse response (Fig. 9.1) recorded in an environment is influenced by the room geometry, materials, and the sound source locations. The direct sound reveals the position of the sound source; the early reflection conveys a sense of the environmental geometry;



Room impulse response

Room layout

Object shapes

Figure 9.1: Audio is a rich source of information to understand the geometry of the physical world such as the room layout and object shapes.

and the late reverberations can indicate the size of the environment. These are rich source of information to understand the geometry of the visual world such as the room layout and object shapes.

Many prior work in the literature addresses 3D shape reconstruction from visual signals. In particular, it would be interesting future work to combine audio and visual signals for 3D reconstruction. Moreover, I would also like to explore how audio sensing can enable embodied agents to efficiently map complex 3D environments, and “see” beyond visible regions. Audio can complement visual sensing and provides strong spatial and semantic signals for visual navigation or exploration.

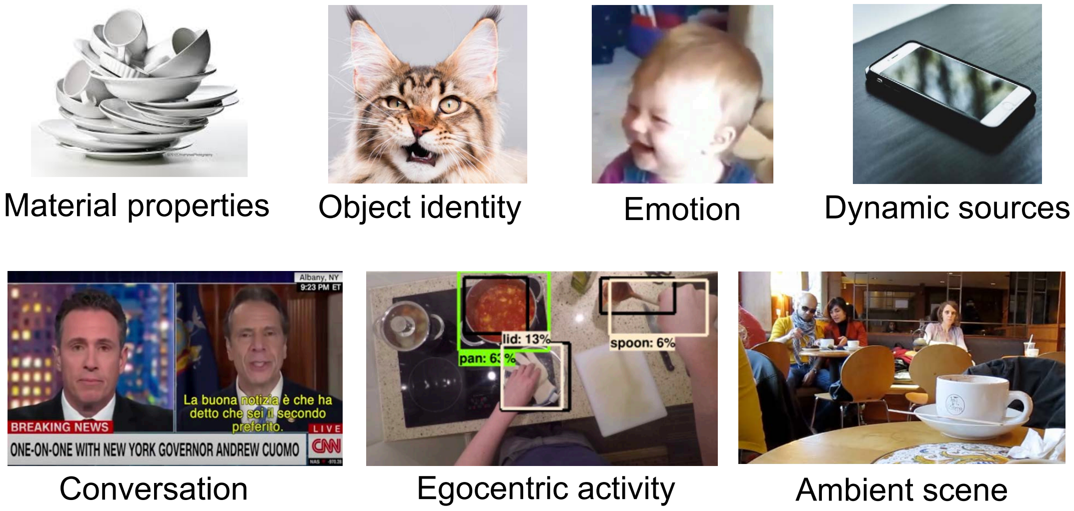


Figure 9.2: Audio-visual video analysis.

9.2 Audio-Visual Video Analysis

As shown in Fig. 9.2, there are many aspects that we can capitalize on once we are able to "listen" from video, e.g., the sounds of the natural sound makers can tell us the object identity based on the sound they emit; the material properties of objects can be revealed when they bang against other objects; something that draws our attention (e.g., a phone ringing) might not be the object itself, but an event that happens to it; and understanding conversations, recognizing egocentric activities, inferring the space and location from ambient sound, etc. These are all exciting directions to get a more comprehensive understanding of our visual world.

My thesis research presented in Chapter 3 through 7 has studied some of the aspects for learning from videos of audio-visual streams. My future

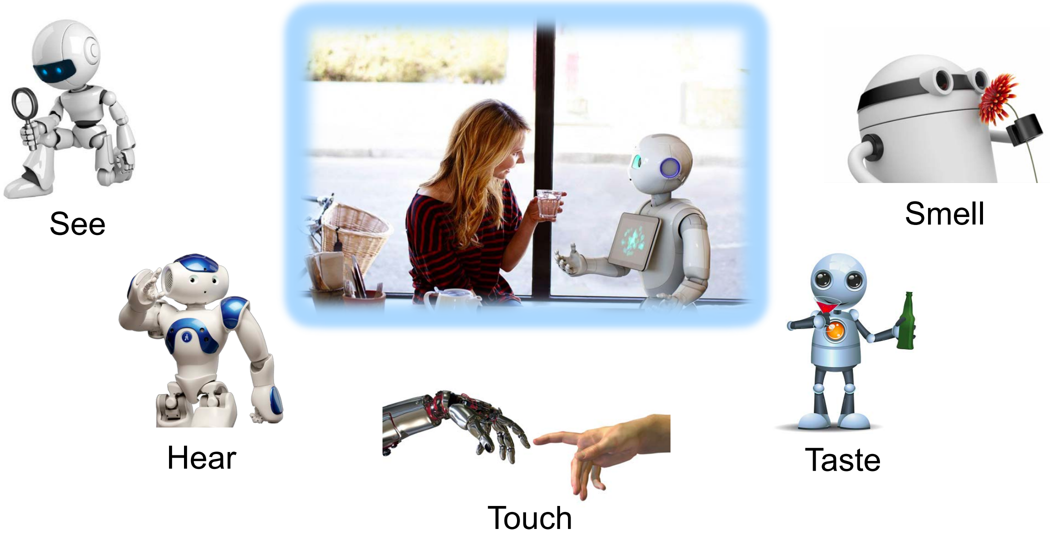


Figure 9.3: Multimodal embodied learning to perceive the world by looking, listening, touching, smelling, and tasting.

research aims at further pushing the boundaries of audio-visual learning from video. Particularly, despite the encouraging progress presented in this dissertation, it remains challenging to perform audio-visual source separation for general objects in the wild. There is a long-tail distribution of natural sound makers. How to separate sounds for objects sporadically making sounds is a challenging and important research problem. Furthermore, I would also like to design better modality fusion mechanisms and network architectures for learning audio-visual sound models from videos.

9.3 Multimodal Embodied Learning

Beyond audio-visual learning, more broadly I am interested in exploring other modalities for embodied learning. My ultimate goal is to build systems that can see, hear, touch, smell, taste, and act in the world by analyzing all the sensory inputs. In the future, I hope to have embodied agents naturally interact with humans and the environment using all the senses (Fig. 9.3).

Bibliography

[1] Cisco visual networking index: Forecast and trends, 2017–2022 white paper.

[2] Triantafyllos Afouras, Joon Son Chung, Andrew Senior, Oriol Vinyals, and Andrew Zisserman. Deep audio-visual speech recognition. *TPAMI*, 2018.

[3] Triantafyllos Afouras, Joon Son Chung, and Andrew Zisserman. The conversation: Deep audio-visual speech enhancement. In *Interspeech*, 2018.

[4] Triantafyllos Afouras, Joon Son Chung, and Andrew Zisserman. My lips are concealed: Audio-visual speech enhancement through obstructions. In *ICASSP*, 2019.

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

[6] Pulkit Agrawal, Joao Carreira, and Jitendra Malik. Learning to see by moving. In *ICCV*, 2015.

[7] Pulkit Agrawal, Ashvin V Nair, Pieter Abbeel, Jitendra Malik, and Sergey Levine. Learning to poke by poking: Experiential learning of intuitive physics. In *NeurIPS*, 2016.

[8] Samuel Albanie, Arsha Nagrani, Andrea Vedaldi, and Andrew Zisserman. Emotion recognition in speech using cross-modal transfer in the wild. In *ACMMM*, 2018.

[9] S. Ali and M. Shah. Human action recognition in videos using kinematic features and multiple instance learning. *PAMI*, 2010.

[10] Humam Alwassel, Fabian Caba Heilbron, and Bernard Ghanem. Action search: Spotting actions in videos and its application to temporal action localization. In *ECCV*, 2018.

[11] Peter Anderson, Angel Chang, Devendra Singh Chaplot, Alexey Dosovitskiy, Saurabh Gupta, Vladlen Koltun, Jana Kosecka, Jitendra Malik, Roozbeh Mottaghi, Manolis Savva, et al. On evaluation of embodied navigation agents. *arXiv preprint arXiv:1807.06757*, 2018.

[12] Fabio Antonacci, Jason Filos, Mark RP Thomas, Emanuël AP Habets, Augusto Sarti, Patrick A Naylor, and Stefano Tubaro. Inference of room geometry from acoustic impulse responses. *IEEE Transactions on Audio, Speech, and Language Processing*, 2012.

[13] Relja Arandjelovic and Andrew Zisserman. Look, listen and learn. In *ICCV*, 2017.

[14] Relja Arandjelović and Andrew Zisserman. Objects that sound. In *ECCV*, 2018.

[15] Yusuf Aytar, Carl Vondrick, and Antonio Torralba. Soundnet: Learning sound representations from unlabeled video. In *NeurIPS*, 2016.

[16] Yusuf Aytar, Carl Vondrick, and Antonio Torralba. See, hear, and read: Deep aligned representations. *arXiv preprint arXiv:1706.00932*, 2017.

[17] K. Barnard, P. Duygulu, N. de Freitas, D. Blei, and M. Jordan. Matching words and pictures. *JMLR*, 2003.

[18] B Barsties, R Verfaillie, N Roy, and Y Maryn. Do body mass index and fat volume influence vocal quality, phonatory range, and aerodynamics in females? In *CoDAS*, 2013.

[19] Zohar Barzelay and Yoav Y Schechner. Harmony in motion. In *CVPR*, 2007.

[20] T. Berg, A. Berg, J. Edwards, M. Maire, R. White, Y. Teh, E. Learned-Miller, and D. Forsyth. Names and faces in the news. In *CVPR*, 2004.

[21] H. Bilen and A. Vedaldi. Weakly supervised deep detection networks. In *CVPR*, 2016.

[22] Vicki Bruce and Andy Young. Understanding face recognition. *British journal of psychology*, 1986.

[23] Nicholas Bryan. *Interactive Sound Source Separation.* PhD thesis, Stanford University, 2014.

[24] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *CVPR*, 2015.

[25] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, 2017.

[26] Anna Llagostera Casanovas, Gianluca Monaci, Pierre Vandergheynst, and Rémi Gribonval. Blind audiovisual source separation based on sparse redundant representations. *IEEE Transactions on Multimedia*, 2010.

[27] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. *3DV*, 2017.

[28] Changan Chen, Unnat Jain, Carl Schissler, Sebastia Vicenc Amengual Gari, Ziad Al-Halah, Vamsi Krishna Ithapu, Philip Robinson, and Kristen Grauman. SoundSpaces: Audio-visual navigation in 3d environments. In *ECCV*, 2020.

[29] Changan Chen, Sagnik Majumder, Ziad Al-Halah, Ruohan Gao, Santhosh K Ramakrishnan, and Kristen Grauman. Learning to set way-

points for audio-visual navigation. In *ICLR*, 2021.

[30] Lele Chen, Sudhanshu Srivastava, Zhiyao Duan, and Chenliang Xu. Deep cross-modal audio-visual generation. In *on Thematic Workshops of ACM Multimedia*, 2017.

[31] Yunpeng Chen, Yannis Kalantidis, Jianshu Li, Shuicheng Yan, and Jiashi Feng. Multi-fiber networks for video recognition. In *ECCV*, 2018.

[32] Jesper Christensen, Sascha Hornauer, and Stella Yu. Batvision - learning to see 3d spatial layout with two ears. In *ICRA*, 2020.

[33] J. S. Chung, A. Nagrani, and A. Zisserman. Voxceleb2: Deep speaker recognition. In *INTERSPEECH*, 2018.

[34] Joon Son Chung, Jaesung Huh, Arsha Nagrani, Triantafyllos Afouras, and Andrew Zisserman. Spot the conversation: speaker diarisation in the wild. In *INTERSPEECH*, 2020.

[35] Soo-Whan Chung, Soyeon Choe, Joon Son Chung, and Hong-Goo Kang. Facefilter: Audio-visual speech separation using still images. In *INTERSPEECH*, 2020.

[36] Soo-Whan Chung, Joon Son Chung, and Hong-Goo Kang. Perfect match: Improved cross-modal embeddings for audio-visual synchronisation. In *ICASSP*, 2019.

[37] Soo-Whan Chung, Hong Goo Kang, and Joon Son Chung. Seeing voices and hearing voices: learning discriminative embeddings using cross-modal self-supervision. In *INTERSPEECH*, 2020.

[38] R. Cinbis, J. Verbeek, and C. Schmid. Weakly supervised object localization with multi-fold multiple instance learning. *PAMI*, 2017.

[39] T. Darrell, J. Fisher, P. Viola, and W. Freeman. Audio-visual segmentation and the cocktail party effect. In *ICMI*, 2000.

[40] Virginia R de Sa. Learning classification with unlabeled data. In *NeurIPS*, 1994.

[41] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.

[42] T. Deselaers, B. Alexe, and V. Ferrari. Weakly supervised localization and learning with generic knowledge. *IJCV*, 2012.

[43] Thomas G Dietterich, Richard H Lathrop, and Tomás Lozano-Pérez. Solving the multiple instance problem with axis-parallel rectangles. *Artificial intelligence*, 1997.

[44] Ivan Dokmanić, Reza Parhizkar, Andreas Walther, Yue M Lu, and Martin Vetterli. Acoustic echoes reveal room shape. *Proceedings of the National Academy of Sciences*, 2013.

[45] J. Donahue, L. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *CVPR*, 2015.

[46] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *CVPR*, 2015.

[47] Ngoc QK Duong, Alexey Ozerov, Louis Chevallier, and Joël Sirot. An interactive audio source separation framework based on non-negative matrix factorization. In *ICASSP*, 2014.

[48] Ngoc QK Duong, Emmanuel Vincent, and Rémi Gribonval. Under-determined reverberant audio source separation using a full-rank spatial covariance model. *IEEE Transactions on Audio, Speech, and Language Processing*, 2010.

[49] P. Duygulu, K. Barnard, N. de Freitas, and D. Forsyth. Object recognition as machine translation: learning a lexicon for a fixed image vocabulary. In *ECCV*, 2002.

[50] Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. Temporal cycle-consistency learning. In *CVPR*, 2019.

[51] David Eigen and Rob Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *ICCV*, 2015.

[52] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. In *NeurIPS*, 2014.

[53] Itamar Eliakim, Zahi Cohen, Gabor Kosa, and Yossi Yovel. A fully autonomous terrestrial bat-like acoustic robot. *PLoS computational biology*, 2018.

[54] Daniel Patrick Whittlesey Ellis. *Prediction-driven computational auditory scene analysis*. PhD thesis, Massachusetts Institute of Technology, 1996.

[55] 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. In *SIGGRAPH*, 2018.

[56] H Fan, Z Xu, L Zhu, C Yan, J Ge, and Y Yang. Watching a small portion could be as good as watching all: Towards efficient video classification. In *IJCAI*, 2018.

[57] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *ICCV*, 2019.

[58] Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman. Convolutional two-stream network fusion for video action recognition. In *CVPR*, 2016.

[59] Ji Feng and Zhi-Hua Zhou. Deep miml network. In *AAAI*, 2017.

[60] Zeyu Feng, Chang Xu, and Dacheng Tao. Self-supervised representation learning by rotation feature decoupling. In *CVPR*, 2019.

[61] Basura Fernando, Hakan Bilen, Efstratios Gavves, and Stephen Gould. Self-supervised video representation learning with odd-one-out networks. In *CVPR*, 2017.

[62] Basura Fernando, Efstratios Gavves, Jose M Oramas, Amir Ghodrati, and Tinne Tuytelaars. Modeling video evolution for action recognition. In *CVPR*, 2015.

[63] Cédric Févotte, Nancy Bertin, and Jean-Louis Durrieu. Nonnegative matrix factorization with the itakura-saito divergence: With application to music analysis. *Neural computation*, 2009.

[64] Cédric Févotte and Jérôme Idier. Algorithms for nonnegative matrix factorization with the β-divergence. *Neural computation*, 2011.

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

[66] David F Fouhey, Abhinav Gupta, and Martial Hebert. Data-driven 3d primitives for single image understanding. In *ICCV*, 2013.

[67] Netanel Frank, L. Wolf, Danny Olshansky, Arjan Boonman, and Yossi Yovel. Comparing vision-based to sonar-based 3d reconstruction. *ICCP*, 2020.

[68] Huan Fu, Mingming Gong, Chaohui Wang, Kayhan Batmanghelich, and Dacheng Tao. Deep ordinal regression network for monocular depth estimation. In *CVPR*, 2018.

[69] Aviv Gabbay, Asaph Shamir, and Shmuel Peleg. Visual speech enhancement. In *Interspeech*, 2018.

[70] Chuang Gan, Deng Huang, Hang Zhao, Joshua B Tenenbaum, and Antonio Torralba. Music gesture for visual sound separation. In *CVPR*, 2020.

[71] Chuang Gan, Yiwei Zhang, Jiajun Wu, Boqing Gong, and Joshua B Tenenbaum. Look, listen, and act: Towards audio-visual embodied navigation. In *ICRA*, 2020.

[72] Chuang Gan, Hang Zhao, Peihao Chen, David Cox, and Antonio Torralba. Self-supervised moving vehicle tracking with stereo sound. In *ICCV*, 2019.

[73] Dhiraj Gandhi, Lerrel Pinto, and Abhinav Gupta. Learning to fly by crashing. In *IROS*, 2017.

[74] Ruohan Gao, Changan Chen, Ziad Al-Halah, Carl Schissler, and Kristen Grauman. Visualechoes: Spatial visual representation learning through echolocation. In *ECCV*, 2020.

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

[76] Ruohan Gao and Kristen Grauman. 2.5d visual sound. In *CVPR*, 2019.

[77] Ruohan Gao and Kristen Grauman. Co-separating sounds of visual objects. In *ICCV*, 2019.

[78] Ruohan Gao and Kristen Grauman. Visualvoice: Audio-visual speech separation with cross-modal consistency. In *CVPR*, 2021.

[79] Ruohan Gao, Dinesh Jayaraman, and Kristen Grauman. Object-centric representation learning from unlabeled videos. In *ACCV*, 2016.

[80] Ruohan Gao, Tae-Hyun Oh, Kristen Grauman, and Lorenzo Torresani. Listen to look: Action recognition by previewing audio. In *CVPR*, 2020.

[81] Ruohan Gao, Bo Xiong, and Kristen Grauman. Im2flow: Motion hallucination from static images for action recognition. In *CVPR*, 2018.

[82] Ravi Garg, Vijay Kumar BG, Gustavo Carneiro, and Ian Reid. Unsupervised cnn for single view depth estimation: Geometry to the rescue. In *ECCV*, 2016.

[83] William W Gaver. What in the world do we hear?: An ecological approach to auditory event perception. *Ecological psychology*, 1993.

[84] 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 *ICASSP*, 2017.

[85] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. In *ICLR*, 2018.

[86] Rohit Girdhar, Deva Ramanan, Abhinav Gupta, Josef Sivic, and Bryan Russell. ActionVLAD: Learning spatio-temporal aggregation for action classification. In *CVPR*, 2017.

[87] Rohit Girdhar, Du Tran, Lorenzo Torresani, and Deva Ramanan. Distinit: Learning video representations without a single labeled video. In *ICCV*, 2019.

[88] Clément Godard, Oisin Mac Aodha, and Gabriel J Brostow. Unsupervised monocular depth estimation with left-right consistency. In *CVPR*, 2017.

[89] Clément Godard, Oisin Mac Aodha, Michael Firman, and Gabriel J Brostow. Digging into self-supervised monocular depth estimation. In *CVPR*, 2019.

[90] Elana Zion Golumbic, Gregory B Cogan, Charles E Schroeder, and David Poeppel. Visual input enhances selective speech envelope tracking in auditory cortex at a “cocktail party”. *Journal of Neuroscience*, 2013.

[91] Priya Goyal, Dhruv Mahajan, Abhinav Gupta, and Ishan Misra. Scaling and benchmarking self-supervised visual representation learning. In *ICCV*, 2019.

[92] Alex Graves, Greg Wayne, and Ivo Danihelka. Neural turing machines. *arXiv preprint arXiv:1410.5401*, 2014.

[93] Daniel Griffin and Jae Lim. Signal estimation from modified short-time fourier transform. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 1984.

[94] Xin Guo, Stefan Uhlich, and Yuki Mitsufuji. Nmf-based blind source separation using a linear predictive coding error clustering criterion. In *ICASSP*, 2015.

[95] Saurabh Gupta, Judy Hoffman, and Jitendra Malik. Cross modal distillation for supervision transfer. In *CVPR*, 2016.

[96] Naomi Harte and Eoin Gillen. Tcd-timit: An audio-visual corpus of continuous speech. *IEEE Transactions on Multimedia*, 2015.

[97] David Harwath and James Glass. Learning word-like units from joint audio-visual analysis. In *ACL*, 2017.

[98] David Harwath, Adrià Recasens, Dídac Surís, Galen Chuang, Antonio Torralba, and James Glass. Jointly discovering visual objects and spoken words from raw sensory input. In *ECCV*, 2018.

[99] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[100] Romain Hennequin, Bertrand David, and Roland Badeau. Score informed audio source separation using a parametric model of non-negative spectrogram. In *ICASSP*, 2011.

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

[102] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[103] Thomas Hofmann. Probabilistic latent semantic indexing. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1999.

[104] Jen-Cheng Hou, Syu-Siang Wang, Ying-Hui Lai, Yu Tsao, Hsiu-Wen Chang, and Hsin-Min Wang. Audio-visual speech enhancement using multimodal deep convolutional neural networks. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 2018.

[105] Di Hu, Rui Qian, Minyue Jiang, Xiao Tan, Shilei Wen, Errui Ding, Weiyao Lin, and Dejing Dou. Discriminative sounding objects localization via self-supervised audiovisual matching. In *NeurIPS*, 2020.

[106] Junjie Hu, Mete Ozay, Yan Zhang, and Takayuki Okatani. Revisiting single image depth estimation: Toward higher resolution maps with accurate object boundaries. In *WACV*, 2019.

[107] Po-Sen Huang, Minje Kim, Mark Hasegawa-Johnson, and Paris Smaragdis. Deep learning for monaural speech separation. In *ICASSP*, 2014.

[108] Po-Sen Huang, Minje Kim, Mark Hasegawa-Johnson, and Paris Smaragdis. Joint optimization of masks and deep recurrent neural networks for monaural source separation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2015.

[109] Aapo Hyvärinen and Erkki Oja. Independent component analysis: algorithms and applications. *Neural networks*, 2000.

[110] Kazuhiro Iida, Yohji Ishii, and Shinsuke Nishioka. Personalization of head-related transfer functions in the median plane based on the anthropometry of the listener's pinnae. *The Journal of the Acoustical Society of America*, 2014.

[111] Satoshi Innami and Hiroyuki Kasai. Nmf-based environmental sound source separation using time-variant gain features. *Computers & Mathematics with Applications*, 2012.

[112] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 2015.

[113] Hamid Izadinia, Imran Saleemi, and Mubarak Shah. Multimodal analysis for identification and segmentation of moving-sounding objects. *IEEE Transactions on Multimedia*, 2013.

[114] Arpit Jain, Abhinav Gupta, Mikel Rodriguez, and Larry S Davis. Representing videos using mid-level discriminative patches. In *CVPR*, 2013.

[115] Rajesh Jaiswal, Derry FitzGerald, Dan Barry, Eugene Coyle, and Scott Rickard. Clustering nmf basis functions using shifted nmf for monaural sound source separation. In *ICASSP*, 2011.

[116] D. Jayaraman and K. Grauman. Slow and steady feature analysis: Higher order temporal coherence in video. In *CVPR*, 2016.

[117] Dinesh Jayaraman, Ruohan Gao, and Kristen Grauman. Shapecodes: self-supervised feature learning by lifting views to viewgrids. In *ECCV*, 2018.

[118] Dinesh Jayaraman and Kristen Grauman. Learning image representations equivariant to ego-motion. In *ICCV*, 2015.

[119] Huaizu Jiang, Gustav Larsson, Michael Maire Greg Shakhnarovich, and Erik Learned-Miller. Self-supervised relative depth learning for urban scene understanding. In *ECCV*, 2018.

[120] A. Karpathy and L. Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *CVPR*, 2015.

[121] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. Large-scale video classification with convolutional neural networks. In *CVPR*, 2014.

[122] Kevin Karsch, Ce Liu, and Sing Bing Kang. Depth transfer: Depth extraction from video using non-parametric sampling. *TPAMI*, 2014.

[123] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017.

[124] Evangelos Kazakos, Arsha Nagrani, Andrew Zisserman, and Dima Damen. Epic-fusion: Audio-visual temporal binding for egocentric action recognition. In *ICCV*, 2019.

[125] Einat Kidron, Yoav Y Schechner, and Michael Elad. Pixels that sound. In *CVPR*, 2005.

[126] Changil Kim, Hijung Valentina Shin, Tae-Hyun Oh, Alexandre Kaspar, Mohamed Elgharib, and Wojciech Matusik. On learning associations of faces and voices. In *ACCV*, 2018.

[127] Hansung Kim, Luca Remaggi, Philip JB Jackson, Filippo Maria Fazi, and Adrian Hilton. 3d room geometry reconstruction using audio-visual sensors. In *3DV*, 2017.

[128] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

[129] W Koenig. Subjective effects in binaural hearing. *The Journal of the Acoustical Society of America*, 1950.

[130] Bruno Korbar, Du Tran, and Lorenzo Torresani. Co-training of audio and video representations from self-supervised temporal synchronization. In *NeurIPS*, 2018.

[131] Bruno Korbar, Du Tran, and Lorenzo Torresani. Scsampler: Sampling salient clips from video for efficient action recognition. In *ICCV*, 2019.

[132] Ivan Krasin, Tom Duerig, Neil Alldrin, Vittorio Ferrari, Sami Abu-El-Haija, Alina Kuznetsova, Hassan Rom, Jasper Uijlings, Stefan Popov, Shahab Kamali, Matteo Malloci, Jordi Pont-Tuset, Andreas Veit, Serge Belongie, Victor Gomes, Abhinav Gupta, Chen Sun, Gal Chechik, David Cai, Zheyun Feng, Dhyanesh Narayanan, and Kevin Murphy. Openimages: A public dataset for large-scale multi-label and multi-class image classification. 2017.

[133] Matthias Kronlachner. Spatial transformations for the alteration of ambisonic recordings. *M. Thesis, University of Music and Performing*

Arts, Graz, Institute of Electronic Music and Acoustics, 2014.

[134] Heinrich Kuttruff. *Room Acoustics*. CRC Press, 2017.

[135] Ivan Laptev and Tony Lindeberg. Space-time interest points. In *ICCV*, 2003.

[136] Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. Colorization as a proxy task for visual understanding. In *CVPR*, 2017.

[137] Luc Le Magoarou, Alexey Ozerov, and Ngoc QK Duong. Text-informed audio source separation. example-based approach using non-negative matrix partial co-factorization. *Journal of Signal Processing Systems*, 2015.

[138] Daniel D Lee and H Sebastian Seung. Algorithms for non-negative matrix factorization. In *NeurIPS*, 2001.

[139] Bochen Li, Karthik Dinesh, Zhiyao Duan, and Gaurav Sharma. See and listen: Score-informed association of sound tracks to players in chamber music performance videos. In *ICASSP*, 2017.

[140] Kai Li, Jun Ye, and Kien A Hua. What's making that sound? In *ACMMM*, 2014.

[141] Ji Lin, Chuang Gan, and Song Han. Tsm: Temporal shift module for efficient video understanding. In *ICCV*, 2019.

[142] Beyang Liu, Stephen Gould, and Daphne Koller. Single image depth estimation from predicted semantic labels. In *CVPR*, 2010.

[143] Fayao Liu, Chunhua Shen, and Guosheng Lin. Deep convolutional neural fields for depth estimation from a single image. In *CVPR*, 2015.

[144] Antoine Liutkus, Derry Fitzgerald, Zafar Rafii, Bryan Pardo, and Laurent Daudet. Kernel additive models for source separation. *IEEE Transactions on Signal Processing*, 2014.

[145] Eric F Lock, Katherine A Hoadley, James Stephen Marron, and Andrew B Nobel. Joint and individual variation explained (jive) for integrated analysis of multiple data types. *The annals of applied statistics*, 2013.

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

[147] Ningning Ma, Xiangyu Zhang, Hai-Tao Zheng, and Jian Sun. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In *ECCV*, 2018.

[148] Pingchuan Ma, Brais Martinez, Stavros Petridis, and Maja Pantic. Towards practical lipreading with distilled and efficient models. *arXiv preprint arXiv:2007.06504*, 2020.

[149] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *JMLR*, 2008.

[150] Michael I Mandel. *Binaural model-based source separation and localization*. Citeseer, 2010.

[151] Brais Martinez, Pingchuan Ma, Stavros Petridis, and Maja Pantic. Lipreading using temporal convolutional networks. In *ICASSP*, 2020.

[152] KN McGuire, C De Wagter, K Tuyls, HJ Kappen, and GCHE de Croon. Minimal navigation solution for a swarm of tiny flying robots to explore an unknown environment. *Science Robotics*, 2019.

[153] Ishan Misra, C Lawrence Zitnick, and Martial Hebert. Shuffle and learn: unsupervised learning using temporal order verification. In *ECCV*, 2016.

[154] Volodymyr Mnih, Nicolas Heess, Alex Graves, et al. Recurrent models of visual attention. In *NeurIPS*, 2014.

[155] Pedro Morgado, Nono Vasconcelos, Timothy Langlois, and Oliver Wang. Self-supervised generation of spatial audio for 360° video. In *NeurIPS*, 2018.

[156] A. Nagrani, J. S. Chung, and A. Zisserman. Voxceleb: a large-scale speaker identification dataset. In *INTERSPEECH*, 2017.

[157] Arsha Nagrani, Samuel Albanie, and Andrew Zisserman. Learnable pins: Cross-modal embeddings for person identity. In *ECCV*, 2018.

[158] Arsha Nagrani, Samuel Albanie, and Andrew Zisserman. Seeing voices and hearing faces: Cross-modal biometric matching. In *CVPR*, 2018.

[159] Kazuhiro Nakadai, Ken-ichi Hidai, Hiroshi G Okuno, and Hiroaki Kitano. Real-time speaker localization and speech separation by audio-visual integration. In *IEEE International Conference on Robotics and Automation*, 2002.

[160] Markus Noisternig, Alois Sontacchi, Thomas Musil, and Robert Holdrich. A 3d ambisonic based binaural sound reproduction system. In *Audio Engineering Society Conference: 24th International Conference: Multichannel Audio, The New Reality.* Audio Engineering Society, 2003.

[161] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *ECCV*, 2016.

[162] Tae-Hyun Oh, Tali Dekel, Changil Kim, Inbar Mosseri, William T Freeman, Michael Rubinstein, and Wojciech Matusik. Speech2face: Learning the face behind a voice. In *CVPR*, 2019.

[163] Andrew Owens and Alexei A Efros. Audio-visual scene analysis with self-supervised multisensory features. In *ECCV*, 2018.

[164] Andrew Owens, Phillip Isola, Josh McDermott, Antonio Torralba, Edward H Adelson, and William T Freeman. Visually indicated sounds. In *CVPR*, 2016.

[165] Andrew Owens, Jiajun Wu, Josh H McDermott, William T Freeman, and Antonio Torralba. Ambient sound provides supervision for visual learning. In *ECCV*, 2016.

[166] Daniele Palossi, Antonio Loquercio, Francesco Conti, Eric Flamand, Davide Scaramuzza, and Luca Benini. A 64-mw dnn-based visual navigation engine for autonomous nano-drones. *IEEE Internet of Things Journal*, 2019.

[167] Sanjeel Parekh, Slim Essid, Alexey Ozerov, Ngoc QK Duong, Patrick Pérez, and Gaël Richard. Motion informed audio source separation. In *ICASSP*, 2017.

[168] Eric K Patterson, Sabri Gurbuz, Zekeriya Tufekci, and John N Gowdy. Cuave: A new audio-visual database for multimodal human-computer interface research. In *IEEE International conference on acoustics, speech, and signal processing*, 2002.

[169] L. Pinto and A. Gupta. Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours. In *ICRA*, 2016.

[170] Hamed Pirsiavash and Deva Ramanan. Parsing videos of actions with segmental grammars. In *CVPR*, 2014.

[171] Jie Pu, Yannis Panagakis, Stavros Petridis, and Maja Pantic. Audio-visual object localization and separation using low-rank and sparsity. In *ICASSP*, 2017.

[172] Senthil Purushwalkam, Abhinav Gupta, Danny M Kaufman, and Bryan Russell. Bounce and learn: Modeling scene dynamics with real-world bounces. In *ICLR*, 2019.

[173] Zhaofan Qiu, Ting Yao, and Tao Mei. Learning spatio-temporal representation with pseudo-3d residual networks. In *ICCV*, 2017.

[174] Ariadna Quattoni and Antonio Torralba. Recognizing indoor scenes. In *CVPR*, 2009.

[175] Colin Raffel, Brian McFee, Eric J Humphrey, Justin Salamon, Oriol Nieto, Dawen Liang, Daniel PW Ellis, and C Colin Raffel. mir_eval: A transparent implementation of common mir metrics. In *ISMIR*, 2014.

[176] Torsten Rahne, Martin Böckmann, Hellmut von Specht, and Elyse S Sussman. Visual cues can modulate integration and segregation of objects in auditory scene analysis. *Brain research*, 2007.

[177] Anurag Ranjan, Varun Jampani, Lukas Balles, Kihwan Kim, Deqing Sun, Jonas Wulff, and Michael J. Black. Competitive collaboration: Joint unsupervised learning of depth, camera motion, optical flow and motion segmentation. In *CVPR*, 2019.

[178] Michalis Raptis, Iasonas Kokkinos, and Stefano Soatto. Discovering discriminative action parts from mid-level video representations. In *CVPR*, 2012.

[179] Lord Rayleigh. On our perception of the direction of a source of sound. *Proceedings of the Musical Association*, 1875.

[180] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NeurIPS*, 2015.

[181] Zhongzheng Ren and Yong Jae Lee. Cross-domain self-supervised multi-task feature learning using synthetic imagery. In *CVPR*, 2018.

[182] Bertrand Rivet, Laurent Girin, and Christian Jutten. Mixing audiovisual speech processing and blind source separation for the extraction of speech signals from convolutive mixtures. *IEEE transactions on audio, speech, and language processing*, 2007.

[183] Antony W Rix, John G Beerends, Michael P Hollier, and Andries P Hekstra. Perceptual evaluation of speech quality (pesq)-a new method for speech quality assessment of telephone networks and codecs. In *ICASSP*, 2001.

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

[185] Lawrence D Rosenblum, Michael S Gordon, and Luis Jarquin. Echolocating distance by moving and stationary listeners. *Ecological Psychol-*

ogy, 2000.

[186] Carsten Rother, Tom Minka, Andrew Blake, and Vladimir Kolmogorov. Cosegmentation of image pairs by histogram matching-incorporating a global constraint into mrfs. In *CVPR*, 2006.

[187] Andrew Rouditchenko, Hang Zhao, Chuang Gan, Josh McDermott, and Antonio Torralba. Self-supervised audio-visual co-segmentation. In *ICASSP*, 2019.

[188] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *CVPR*, 2018.

[189] Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, Devi Parikh, and Dhruv Batra. Habitat: A Platform for Embodied AI Research. In *ICCV*, 2019.

[190] Farnaz Sedighin, Massoud Babaie-Zadeh, Bertrand Rivet, and Christian Jutten. Two multimodal approaches for single microphone source separation. In *24th European Signal Processing Conference*, 2016.

[191] Arda Senocak, Tae-Hyun Oh, Junsik Kim, Ming-Hsuan Yang, and In So Kweon. Learning to localize sound source in visual scenes. In *CVPR*, 2018.

[192] Zheng Shou, Xudong Lin, Yannis Kalantidis, Laura Sevilla-Lara, Marcus Rohrbach, Shih-Fu Chang, and Zhicheng Yan. Dmc-net: Generating

discriminative motion cues for fast compressed video action recognition. In *CVPR*, 2019.

[193] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *ECCV*, 2012.

[194] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *NeurIPS*, 2014.

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

[196] Paris Smaragdis and Michael Casey. Audio/visual independent components. In *International Conference on Independent Component Analysis and Signal Separation*, 2003.

[197] Paris Smaragdis, Bhiksha Raj, and Madhusudana Shashanka. A probabilistic latent variable model for acoustic modeling. In *NeurIPS*, 2006.

[198] Paris Smaragdis, Bhiksha Raj, and Madhusudana Shashanka. Supervised and semi-supervised separation of sounds from single-channel mixtures. In *International Conference on Independent Component Analysis and Signal Separation*, 2007.

[199] Linda Smith and Michael Gasser. The development of embodied cognition: Six lessons from babies. *Artificial life*, 2005.

[200] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012.

[201] Martin Spiertz and Volker Gnann. Source-filter based clustering for monaural blind source separation. In *12th International Conference on Digital Audio Effects*, 2009.

[202] Julian Straub, Thomas Whelan, Lingni Ma, Yufan Chen, Erik Wijmans, Simon Green, Jakob J Engel, Raul Mur-Artal, Carl Ren, Shobhit Verma, et al. The replica dataset: A digital replica of indoor spaces. *arXiv preprint arXiv:1906.05797*, 2019.

[203] Thomas A Stroffregen and John B Pittenger. Human echolocation as a basic form of perception and action. *Ecological psychology*, 1995.

[204] Yu-Chuan Su and Kristen Grauman. Leaving some stones unturned: dynamic feature prioritization for activity detection in streaming video. In *ECCV*, 2016.

[205] Sainbayar Sukhbaatar, Jason Weston, Rob Fergus, et al. End-to-end memory networks. In *NeurIPS*, 2015.

[206] Cees H Taal, Richard C Hendriks, Richard Heusdens, and Jesper Jensen. An algorithm for intelligibility prediction of time–frequency weighted

noisy speech. *IEEE Transactions on Audio, Speech, and Language Processing*, 2011.

[207] Y. Tian, J. Shi, B. Li, Z. Duan, and C. Xu. Audio-visual event localization in unconstrained videos. In *ECCV*, 2018.

[208] Edgar A Torres-Gallegos, Felipe Orduna-Bustamante, and Fernando Arámbula-Cosío. Personalization of head-related transfer functions (hrtf) based on automatic photo-anthropometry and inference from a database. *Applied Acoustics*, 2015.

[209] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *CVPR*, 2015.

[210] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *CVPR*, 2018.

[211] Jasper RR Uijlings, Koen EA van de Sande, Theo Gevers, and Arnold WM Smeulders. Selective search for object recognition. *IJCV*, 2013.

[212] Benjamin Ummenhofer, Huizhong Zhou, Jonas Uhrig, Nikolaus Mayer, Eddy Ilg, Alexey Dosovitskiy, and Thomas Brox. Demon: Depth and motion network for learning monocular stereo. In *CVPR*, 2017.

[213] Dieter Vanderelst, Marc W Holderied, and Herbert Peremans. Sensorimotor model of obstacle avoidance in echolocating bats. *PLoS computational biology*, 2015.

[214] Gül Varol, Ivan Laptev, and Cordelia Schmid. Long-term temporal convolutions for action recognition. *PAMI*, 2017.

[215] Igor Vasiljevic, Nick Kolkin, Shanyi Zhang, Ruotian Luo, Haochen Wang, Falcon Z. Dai, Andrea F. Daniele, Mohammadreza Mostajabi, Steven Basart, Matthew R. Walter, and Gregory Shakhnarovich. DIODE: A Dense Indoor and Outdoor DEpth Dataset. *arXiv preprint arXiv:1908.00463*, 2019.

[216] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017.

[217] Eric Veach and Leonidas Guibas. Bidirectional estimators for light transport. In *Photorealistic Rendering Techniques*. 1995.

[218] S. Vijayanarasimhan and K. Grauman. Keywords to visual categories: Multiple-instance learning for weakly supervised object categorization. In *CVPR*, 2008.

[219] Sudheendra Vijayanarasimhan, Susanna Ricco, Cordelia Schmid, Rahul Sukthankar, and Katerina Fragkiadaki. Sfm-net: Learning of structure and motion from video. *arXiv preprint arXiv:1704.07804*, 2017.

[220] Emmanuel Vincent, Rémi Gribonval, and Cédric Févotte. Performance measurement in blind audio source separation. *IEEE transactions on audio, speech, and language processing*, 2006.

[221] Oriol Vinyals, Meire Fortunato, and Navdeep Jaitly. Pointer networks. In *NeurIPS*, 2015.

[222] Tuomas Virtanen. Sound source separation using sparse coding with temporal continuity objective. In *International Computer Music Conference*, 2003.

[223] Tuomas Virtanen. Monaural sound source separation by nonnegative matrix factorization with temporal continuity and sparseness criteria. *IEEE transactions on audio, speech, and language processing*, 2007.

[224] Beiming Wang and Mark D Plumbley. Investigating single-channel audio source separation methods based on non-negative matrix factorization. In *ICA Research Network International Workshop*, 2006.

[225] DeLiang Wang and Jitong Chen. Supervised speech separation based on deep learning: An overview. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2018.

[226] Heng Wang and Cordelia Schmid. Action recognition with improved trajectories. In *CVPR*, 2013.

[227] L. Wang, Y. Xiong, D. Lin, and L. Van Gool. Untrimmednets for weakly supervised action recognition and detection. In *CVPR*, 2017.

[228] LiMin Wang, Yu Qiao, and Xiaoou Tang. Motionlets: Mid-level 3d parts for human motion recognition. In *CVPR*, 2013.

[229] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *ECCV*, 2016.

[230] Peng Wang, Xiaohui Shen, Zhe Lin, Scott Cohen, Brian Price, and Alan L Yuille. Towards unified depth and semantic prediction from a single image. In *CVPR*, 2015.

[231] Weiyao Wang, Du Tran, and Matt Feiszli. What makes training multimodal networks hard? *arXiv preprint arXiv:1905.12681*, 2019.

[232] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Nonlocal neural networks. In *CVPR*, 2018.

[233] Xiaolong Wang and Abhinav Gupta. Unsupervised learning of visual representations using videos. In *ICCV*, 2015.

[234] Yandong Wen, Mahmoud Al Ismail, Weiyang Liu, Bhiksha Raj, and Rita Singh. Disjoint mapping network for cross-modal matching of voices and faces. In *ICLR*, 2019.

[235] Geert Willems, Tinne Tuytelaars, and Luc Van Gool. An efficient dense and scale-invariant spatio-temporal interest point detector. In *ECCV*, 2008.

[236] Donald S Williamson, Yuxuan Wang, and DeLiang Wang. Complex ratio masking for monaural speech separation. *IEEE/ACM transactions on audio, speech, and language processing*, 2015.

[237] Chao-Yuan Wu, Christoph Feichtenhofer, Haoqi Fan, Kaiming He, Philipp Krahenbuhl, and Ross Girshick. Long-term feature banks for detailed video understanding. In *CVPR*, 2019.

[238] Chao-Yuan Wu, Manzil Zaheer, Hexiang Hu, R Manmatha, Alexander J Smola, and Philipp Krähenbühl. Compressed video action recognition. In *CVPR*, 2018.

[239] Jiajun Wu, Yinan Yu, Chang Huang, and Kai Yu. Deep multiple instance learning for image classification and auto-annotation. In *CVPR*, 2015.

[240] Wenhao Wu, Dongliang He, Xiao Tan, Shifeng Chen, and Shilei Wen. Multi-agent reinforcement learning based frame sampling for effective untrimmed video recognition. In *ICCV*, 2019.

[241] Zuxuan Wu, Yu-Gang Jiang, Xi Wang, Hao Ye, and Xiangyang Xue. Multi-stream multi-class fusion of deep networks for video classification. In *ACM-MM*, 2016.

[242] Zuxuan Wu, Caiming Xiong, Chih-Yao Ma, Richard Socher, and Larry S Davis. Adaframe: Adaptive frame selection for fast video recognition. In *CVPR*, 2019.

[243] Fei Xia, Amir R Zamir, Zhiyang He, Alexander Sax, Jitendra Malik, and Silvio Savarese. Gibson env: Real-world perception for embodied agents. In *CVPR*, 2018.

[244] Saining Xie, Chen Sun, Jonathan Huang, Zhuowen Tu, and Kevin Murphy. Rethinking spatiotemporal feature learning: Speed-accuracy tradeoffs in video classification. In *ECCV*, 2018.

[245] Dan Xu, Elisa Ricci, Wanli Ouyang, Xiaogang Wang, and Nicu Sebe. Multi-scale continuous crfs as sequential deep networks for monocular depth estimation. In *CVPR*, 2017.

[246] Xudong Xu, Bo Dai, and Dahua Lin. Recursive visual sound separation using minus-plus net. In *ICCV*, 2019.

[247] Hao Yang, Joey Tianyi Zhou, Jianfei Cai, and Yew Soon Ong. Miml-fcn+: Multi-instance multi-label learning via fully convolutional networks with privileged information. In *CVPR*, 2017.

[248] Zhenheng Yang, Peng Wang, Wei Xu, Liang Zhao, and Ramakant Nevatia. Unsupervised learning of geometry with edge-aware depth-normal consistency. In *AAAI*, 2018.

[249] Mao Ye, Yu Zhang, Ruigang Yang, and Dinesh Manocha. 3d reconstruction in the presence of glasses by acoustic and stereo fusion. In *ICCV*, 2015.

[250] Serena Yeung, Olga Russakovsky, Greg Mori, and Li Fei-Fei. End-to-end learning of action detection from frame glimpses in videos. In *CVPR*, 2016.

[251] Ozgur Yilmaz and Scott Rickard. Blind separation of speech mixtures via time-frequency masking. *IEEE Transactions on signal processing*, 2004.

[252] Dong Yu, Morten Kolbæk, Zheng-Hua Tan, and Jesper Jensen. Permutation invariant training of deep models for speaker-independent multi-talker speech separation. In *ICASSP*, 2017.

[253] Joe Yue-Hei Ng, Matthew Hausknecht, Sudheendra Vijayanarasimhan, Oriol Vinyals, Rajat Monga, and George Toderici. Beyond short snippets: Deep networks for video classification. In *CVPR*, 2015.

[254] Amir R. Zamir, Alexander Sax, William Shen, Leonidas J. Guibas, Jitendra Malik, and Silvio Savarese. Taskonomy: Disentangling task transfer learning. In *CVPR*, 2018.

[255] Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *ECCV*, 2016.

[256] Zhoutong Zhang, Jiajun Wu, Qiujia Li, Zhengjia Huang, James Traer, Josh H. McDermott, Joshua B. Tenenbaum, and William T. Freeman. Generative modeling of audible shapes for object perception. In *ICCV*, 2017.

[257] Hang Zhao, Chuang Gan, Wei-Chiu Ma, and Antonio Torralba. The sound of motions. In *ICCV*, 2019.

[258] Hang Zhao, Chuang Gan, Andrew Rouditchenko, Carl Vondrick, Josh McDermott, and Antonio Torralba. The sound of pixels. In *ECCV*, 2018.

[259] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *CVPR*, 2017.

[260] Bolei Zhou, Alex Andonian, Aude Oliva, and Antonio Torralba. Temporal relational reasoning in videos. In *ECCV*, 2018.

[261] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *TPAMI*, 2017.

[262] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *CVPR*, 2017.

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

[264] Hang Zhou, Xudong Xu, Dahua Lin, Xiaogang Wang, and Ziwei Liu. Sep-stereo: Visually guided stereophonic audio generation by associating source separation. In *ECCV*, 2020.

[265] Tinghui Zhou, Matthew Brown, Noah Snavely, and David G Lowe. Unsupervised learning of depth and ego-motion from video. In *CVPR*, 2017.

[266] Yipin Zhou, Zhaowen Wang, Chen Fang, Trung Bui, and Tamara L Berg. Visual to sound: Generating natural sound for videos in the wild. In *CVPR*, 2018.

[267] Chen Zhu, Xiao Tan, Feng Zhou, Xiao Liu, Kaiyu Yue, Errui Ding, and Yi Ma. Fine-grained video categorization with redundancy reduction attention. In *ECCV*, 2018.

[268] Michael Zibulevsky and Barak A Pearlmutter. Blind source separation by sparse decomposition in a signal dictionary. *Neural computation*, 2001.

[269] Mohammadreza Zolfaghari, Kamaljeet Singh, and Thomas Brox. Eco: Efficient convolutional network for online video understanding. In *ECCV*, 2018.

Vita

Ruohan Gao was born in Sichuan Province, China on September 5th, 1993. He received his high school diploma from Chengdu Foreign Languages School in 2011, and B.Eng. degree from the Department of Information Engineering at The Chinese University of Hong Kong in 2015. In 2015 fall, he began his graduate studies in the Department of Computer Science at The University of Texas at Austin, and he has been studying computer vision under the supervision of Prof. Kristen Grauman since January 2016. During his PhD, he has received the 2021 Michael H. Granof University's Top Dissertation Award, the Google PhD Fellowship for 2019-2021, the Graduate Dean's Prestigious Fellowship Supplement Award in 2019 and 2020, the Outstanding Reviewer Award at Conference on Computer Vision and Pattern Recognition 2020, the Adobe Research Fellowship in 2019, and a Best Paper Award Finalist at Conference on Computer Vision and Pattern Recognition 2019 for his work on 2.5D visual sound. Starting from February 2021, he began a postdoc at Stanford University.

Permanent email address: rhgao@utexas.edu

This dissertation was typeset with LaTeX[†] by the author.

[†] LaTeX is a document preparation system developed by Leslie Lamport as a special version of Donald Knuth's TeX Program.