Learning with Multimodal Self-Supervision

Honglie Chen

Oriel College

University of Oxford

A thesis submitted for the degree of

Doctor of Philosophy

Michaelmas 2021

This thesis is submitted to the Department of Engineering Science, University of Oxford, in fulfillment of the requirements for the degree of Doctor of Philosophy. This thesis is entirely my own work, and except where otherwise stated, describes my own research.

Honglie Chen, Oriel College

Acknowledgements

I am indebted and grateful to my advisors Andrew Zisserman and Andrea Vedaldi, for their continuous support, encouragement, and trust over the years. This thesis would not have been possible without their mentorship.

I appreciate Weidi Xie, for his kind help on my first conference paper. His valuable feedback and insightful comments have helped me enrich myself for the past four years. To my amazing collaborators: Weidi Xie, Arsha Nagrani and Triantafyllos Afouras. And to all the members of Visual Geometry Group (VGG), for all the countless thoughtful discussions and all the enjoyable time.

I gratefully acknowledge my funding from Oxford University. Thanks to Oriel college for providing a friendly environment through both my undergraduate and PhD studies.

I shall always be thankful to my family for their endless love, support and understanding.

Abstract

Deep learning has fueled an explosion of applications, yet training deep neural networks usually requires expensive human annotations. In this thesis we explore alternatives to avoid the substantial reliance on manual annotated examples when training deep neural networks. Specifically, we do so by either adapting self-supervised methods to automatically correct freely obtained data labels, or by completely abandoning the use of human labels and instead utilizing the natural co-occurrence of audio and visual information to learn object representations in videos.

Growing collections of digital data often provide noisy labels that can be exploited to supervise the learning process. Conventional data pre-processing includes correcting/cleaning them before training recognition models, but this can require infeasible amounts of manual effort. We consider correcting the annotation noise automatically, and hence eschew the need for costly manual annotation. We build and extend recent breakthroughs with a consistency loss which enables training even without ground truth, and an spatial memory map that provides flexible instance-level registration, leading to greater generalization.

We further explore multimodal sensory streams to provide self-supervision to the model by utilizing modality redundancy, *i.e.* the overlapping information between modalities. Representations are learned by harnessing different modalities without using any human-annotated labels. We demonstrate this technique using three different applications. First, we automatically curate a large-scale audio dataset, VGG-Sound, with more than 200k videos collected

using visual guidance, training on which yields state-of-the-art models for audio recognition. Second, we present a method to improve and extend recent sound source localization techniques by introducing a mechanism to mine hard samples and add them to a contrastive learning formulation automatically. Finally, unlike existing audio-visual synchronization tasks performed on one specific domain, we propose to solve the synchronization problem in open world settings by exploring the use of several transformer-based architectures. With these models, we achieve state-of-the-art results in challenging speech datasets and show excellent generalization in a general sound dataset.

Contents

1 Introduction **1**

1.1 Motivation . . . 2

1.1.1 Label correction via cycle-consistency . . . 2

1.1.2 Audio-visual learning . . . 3

1.2 Thesis outline and contributions . . . 5

1.3 Publications . . . 7

2 Literature Review **9**

2.1 Large-scale datasets . . . 9

2.1.1 Image datasets . . . 11

2.1.2 Video and audio datasets . . . 12

2.2 Self-supervised learning . . . 14

2.2.1 Temporal sequences . . . 14

2.2.2 Spatial context . . . 16

2.2.3 Context similarity . . . 17

2.2.4 Generation methods . . . 17

2.3 Audio-visual learning . . . 18

2.3.1 Correspondence . . . 18

2.3.2 Synchronization . . . 21

2.3.3 Generation . . . 22

2.4 Transformers . . . 22

3 AutoCorrect: Deep Inductive Alignment of Noisy Geometric Annotations 26

3.1 Introduction . . . 27

3.2 Related work . . . 29

3.3 Approach . . . 30

3.3.1 Single instance alignment . . . 31

3.3.2 Inductive alignment . . . 34

3.3.3 Implementation details . . . 35

3.4 Experiments . . . 36

3.4.1 Datasets and evaluation . . . 36

3.4.2 Railway tracks results . . . 37

3.4.3 INRIA buildings dataset results . . . 38

3.4.4 Qualitative results . . . 39

3.5 Conclusion . . . 41

3.A Architecture details . . . 42

3.B More results from the *AutoCorrect* approach . . . 42

4 VGG-Sound: A Large-scale Audio-Visual Dataset 47

4.1 Introduction . . . 48

4.2 Related Work . . . 49

4.3 The VGG-Sound dataset . . . 51

4.3.1 Stage 1: Obtaining the class list and candidate videos. . . . 52

4.3.2 Stage 2: Visual verification. . . . 53

4.3.3 Stage 3. Audio verification to remove negative clips. . . . 54

4.3.4 Stage 4: Iterative noise filtering. . . . 55

4.4 Experiments . . . 56

4.4.1 Experimental setup and Evaluation . . . 56

4.4.2 Implementation details . . . 56

4.4.3 Results . . . 57

4.5 Conclusion . . . 58

5 Localizing Visual Sounds the Hard Way **59**

5.1 Introduction . . . 60

5.2 Related Work . . . 62

5.2.1 Audio-Visual Sound Source Localization . . . 62

5.2.2 Audio-Visual Localization Benchmarks . . . 63

5.3 Method . . . 64

5.3.1 Audio-Visual Feature Representation . . . 65

5.3.2 Audio-Visual Correspondence . . . 66

5.3.3 Audio-Visual Localization with an Oracle . . . 66

5.3.4 Self-supervised Audio-Visual Localization . . . 68

5.4 The VGG-Sound Source Benchmark . . . 69

5.4.1 Test Set Annotation Pipeline . . . 69

5.5 Experiments . . . 72

5.5.1 Training Data . . . 72

5.5.2 Evaluation protocol . . . 73

5.5.3 Implementation details . . . 73

5.6 Results . . . 74

5.6.1 Comparison on the Flickr SoundNet Test Set . . . 75

5.6.2 Ablation Analysis . . . 75

5.6.3 Comparison on VGG-Sound Source . . . 77

5.6.4 Qualitative results . . . 77

5.6.5 Open Set Audio-visual Localization . . . 77

5.7 Conclusion . . . 79

5.A Evaluation metric . . . 81

5.A.1 Tri-map visualization . . . 82

5.B VGG-Sound Source (VGG-SS) . . . 83

5.B.1 VGG-SS annotation interface . . . 83

5.B.2 VGG-SS examples . . . 83

6 Audio-Visual synchronization in the wild **86**

6.1 Introduction . . . 87

6.2 Related Work . . . 89

6.3 Method . . . 91

6.3.1 Architecture . . . 91

6.3.1.1 Audio and visual representations . . . 91

6.3.1.2 synchronization module . . . 92

6.3.1.3 Output head . . . 95

6.3.2 Training objectives . . . 95

6.4 Experiments . . . 95

6.4.1 Datasets . . . 96

6.4.2 Evaluation protocol . . . 96

6.4.2.1 Audio-visual synchronization on speech . . . 96

6.4.2.2 Audio-visual synchronization on general classes . . . 97

6.4.3 Implementation details . . . 99

6.4.4 Results on speech datasets . . . 99

6.4.5 Results on general sound classes . . . 101

6.4.5.1 Visualisation of attention heatmaps . . . 102

6.5 Conclusion . . . 102

6.A Implementation details . . . 104

6.B Robustness test . . . 104

6.C synchronization on general sound classes . . . 104

6.D Attention heatmaps visualisation . . . 109

7 Conclusion **116**

7.1 Achievements and Impact . . . 116

7.2 Future work . . . 118

A Statements of Authorship **156**

List of Figures

2.1 Selection of modern datasets. 10
2.2 Overview of contrastive predictive coding. 15
2.3 The vanilla transformer model and positional embeddings. 24

3.1 Example aerial images with noisy labels. 28
3.2 AutoCorrect architecture. 30
3.3 Correcting annotations sequentially using a memory map. 33
3.4 INRIA buildings dataset results. 39
3.5 AutoCorrect correction progression. 40
3.6 Alignment results for the *Railway tracks* dataset, and *INRIA buildings* dataset. 40
3.7 Railway tracks alignment. 43
3.8 Railway tracks alignment. 44
3.9 INRIA buildings alignment. 45
3.10 INRIA buildings alignment. 46

4.1 VGG-Sound example. 51

5.1 Visual sound source localization. 60
5.2 Audio-visual localization architecture overview. 64
5.3 VGG-SS Statistics. 71
5.4 Example Tri-map visualizations. 76
5.5 Qualitative results for Audio-Visual localization. 78
5.6 Example predictions with calculated cIoU. 81
5.7 Tri-map visualization examples. 82
5.8 LISA Annotation Interface. 83

5.9 VGG-SS examples. 84
5.10 VGG-SS benchmark per class statistics. 85

6.1 Audio-visual synchronization in the wild. 87
6.2 The AVST model architecture and variants. 93
6.3 Per-class accuracy on VGG-Sound Sync. 102
6.4 Attention heatmaps on VGG-Sound Sync. 103
6.5 Proportions of videos considered to be syncd by a manual observer. 105
6.6 Per-class accuracy on VGG-Sound Sync. 106
6.7 Synchronization scores along temporal axis. 108
6.8 Attention heatmap visualisations on LRS3 dataset. 110
6.9 Attention heatmap visualisations on VGG-Sound Sync dataset. 111

List of Tables

3.1 Railway tracks dataset results. 37
3.2 Architecture for Railway tracks dataset. 42
3.3 Architecture for INRIA buildings dataset. 42

4.1 VGG-Sound Dataset Statistics. 48
4.2 Stats for recent audio datasets. 50
4.3 Stats after each stage of the dataset generation pipeline 52
4.4 Audio classification results. 57

5.1 Comparison with the existing sound-source localization benchmrks. 64
5.2 The number of classes and videos in VGG-SS after each annotation stage. . 72
5.3 Quantitative results on Flickr SoundNet testset. 74
5.4 Audio-Visual localization ablation study. 74
5.5 Quantitative results on the VGG-SS testset. 77
5.6 Quantitative results on VGG-SS for unheard classes. 79

6.1 Categorisation of video clips as duration of video varies. 97
6.2 Architecture comparison on LRS3 and LRS2. 101
6.3 Ablation on Transformer depth (LRS2). 101
6.4 Robustness test on LRS2. 101
6.5 Audio-visual synchronization results on VGG-Sound Sync. 102
6.6 Robustness test on LRS2. 105

Chapter 1

Introduction

Over the past decade, deep learning [LeCun et al., 2015] has gained immense interest and has become a compelling approach for various domains such as images [Krizhevsky et al., 2012; Simonyan and Zisserman, 2015; He et al., 2016; Hu et al., 2018; Dosovitskiy et al., 2021], audio [Hinton et al., 2012; Maas et al., 2013; Gong et al., 2021] and text [Sutskever et al., 2014; Vaswani et al., 2017; Katharopoulos et al., 2020]. In the visual domain, Convolutional Neural Networks (CNNs) have led to ground-breaking results across many tasks such as object recognition [Simonyan and Zisserman, 2015], detection [Girshick, 2015] and segmentation [Shelhamer et al., 2015]. However, large-scale, well-labeled data is generally required to train CNNs in order to obtain these convincing performances. Collecting and annotating such datasets are prohibitively expensive, and it is tedious/impractical for human experts to label the tremendous collections of digital data. The quality and scale of labels are often considered as the bottleneck in the adoption and wide-spread use of modern deep CNNs. Are there any alternatives to compensate for this shortage?

In this thesis, we explore the training of CNNs by correcting existing noisy examples using cycle-consistency, or by entirely alleviating any reliance on manually-annotated data through audio-visual self-supervision. The unifying objective is to remove the need for manual supervision. This would make training deep neural network more scalable and cost-efficient, hence evading this annotation bottleneck for future research.

We exploit geometric consistency as an alternative supervision and correct existing noisy labels in Chapter 3. The dominant application domain in this thesis (Chapter 4 - 6) is audio-visual learning. The goal is to make use of cross-modal self-supervised learning

to map high dimensional sensory inputs into meaningful representation vectors that can be used for downstream tasks such as localization and synchronization. In such a way, the need for expensive annotations is eliminated, and the feature representations are instead learned by exploiting the redundancy and complementarity in multimodal data.

1.1 Motivation

1.1.1 Label correction via cycle-consistency

With the recent emergence of large-scale datasets, deep neural networks have demonstrated impressive performances on many machine learning tasks [Krizhevsky et al., 2012; He et al., 2016]. The quality of the data labels however plays a crucial role, and model performance drops dramatically as label noise increases, regardless of the learning algorithm used [Elsayed et al., 2018]. However, high quality labels for large-scale datasets are often extremely expensive to obtain, especially for dense pixel-level tasks. For example, annotation and quality control for image segmentation require more than 1.5h on average for a single image in the Cityscapes dataset [Cordts et al., 2016], though only 7min is needed when annotating partial images. The annotation cost can be substantially reduced if the labels need not be accurate. In the satellite image domain, there are publicly available maps which can provide segmentation labels for free, but often with noise. The pixel-level annotations often fail to match the objects (*e.g.*, road, buildings, vegetation etc.) on satellite images due to several issues, such as viewpoint variation as maps do not capture 3D structure feature, or invisibility caused by occlusions, resulting in geometric label noise. A model that automatically fixes these noisy labels obtained from open sources online or via minimal human annotations can have several useful applications. It can provide higher quality, clean annotations without requiring additional human effort, and these clean labels can be used for a range of computer vision applications such as image segmentation and tracking.

Learning with cycle consistency. Learning correct labels automatically with access only to the noisy labels is an ambiguously defined task. One popular alternative when manual labels are not available is to obtain supervision using cycle-consistency. As early as the 1970s, a technique called "back translation and reconciliation" was used to verify and improve translations by human translators [Brislin, 1970]. Inspired by these human studies, cycling between two or more samples has become a commonly used technique for assessing performance in machine vision. Zhu et al. [2017] exploit the property that translation should be consistency in the sense that the inverse transformation of the transformed input should arrive back to the original input. With a cycle-consistency loss, the learning process would be valid even if the ground-truth annotations are unknown.

1.1.2 Audio-visual learning

In addition to using geometric consistency to correct noisy labels, another trend to reduce human effort is to follow psychology studies on human perception and exploit self-labels using multiple sensory cues such as vision, audition, touch, smell, proprioception and balance [Smith and Gasser, 2005]. Multidimensional consciousness, an innate ability which allows humans to interact with the world through multiple sensory inputs, was extensively studied in the psychology literature [Edelman, 1987]. Humans interact with the world through many sensory streams via the concept of redundancy. One specific aspect of redundancy is called `degeneracy`, which enables the system functionality even when one modality is lost. For example, spatial concepts can be developed by blind children by learning from other modalities [Landau and Gleitman, 1985].

Another aspect of redundancy is that different modalities can educate each other, namely, `reentry`, which is the ongoing, parallel and recursive signalling between multiple simultaneous representations across modalities [Edelman and Gally, 2013]. It allows humans to form explicit knowledge representations of the same information across two or more sensory modalities. For example, early works [Knight and Johnston, 1997; O'Toole et al., 2002] from the face recognition community suggested that the movement of the face is associated with speech and is useful for face recognition. This indicates the existence of overlapping information in multimodality, as those critical cues are often available bi-modally.

Therefore, the unique, shared, high-level semantics across different sensory inputs can lead to a powerful learning mechanism where different modalities can be used to supervise one another.

In contrast to utilizing the overlapping information in multimodality to form a common perception, humans also bind cross-modal sensory features to complement each other. The mechanisms of multiple, functionally cortical areas are coordinated and integrated to yield a unified perceptual response. For example, visual sensory input can affect auditory perception, as shown by the McGurk effect [McGurk and MacDonald, 1976], in which fusion of auditory "ba" and visual "fa" or "da" results in a perception "fa" or "da" which are dominated by the visual inputs, *i.e.* the mouth movements. Another example is the ventriloquist illusion [Driver, 1996], where a voice appears to come from the moving mouth of a puppet rather than from the actual speaker. Such effects show that the perception can be determined by visual modality, which provides complementary information on the place of articulation and muscle movements.

Taking inspiration from human perception studies, we particularly focus on audio-visual learning, which has gained tremendous interest recently due to two additional reasons: First, a vast supply of audio-visual information content is readily present in videos online; Second, exploiting cross-modal redundancy as a source of self-supervision from multi-modal inputs has led a number of successful applications in machine perception and learning.

Proliferation of multimodal content. Recently, the amount of multimodal data has exploded due to the widespread use of platforms like Instagram and TicToc. Images uploaded to those social media websites are frequently accompanied by contextual text in the form of captions or hashtags. Moreover, more than 500 hours of video are uploaded to YouTube every minute and we watch over 1 billion hours of YouTube videos a day. Online videos are naturally multimodal, often containing an audio track accompanying visual content with high correlation. Benefiting from the abundant videos online, a range of existing audio-visual datasets such as Kinetics [Kay et al., 2017] and AudioSet [Gemmeke et al., 2017]

are curated from the open world. Such datasets have been considered as the key benchmarks to develop modern learning algorithms.

Success in self-supervised learning. Cross-modal redundancy has been exploited by a number of audio-visual methods [Arandjelovic and Zisserman, 2017; Owens and Efros, 2018], where the goal is to learn representations by exploiting audio-visual co-occurrence, *i.e.* the overlapping information between modalities. Rather than explicitly using human annotations to supervise the learning process, these works focus on matching visual and audio components extracted from the same video by minimizing the disparity between the cross-model features from two separate networks. In such a way, high level representations are learnt as a pretext task and can be used in a range of downstream applications such as action recognition [Korbar et al., 2018], audio-visual event detection [Tian et al., 2018; Lin et al., 2019], audio separation [Hershey and Casey, 2002; Zhao et al., 2018a, 2019] and audio localization [Arandjelovic and Zisserman, 2018; Qian et al., 2020; Afouras et al., 2020b]. Notably, Zhu and Rahtu [2020] tackle both audio separation and localization at the same time by using a Cascaded Opponent Filter and a Sound Source Location Masking technique.

1.2 Thesis outline and contributions

In this section, we summarize the contributions of this thesis, and provide an outline of the chapters.

Chapter 3 - AutoCorrect: Deep Inductive Alignment of Noisy Geometric Annotations. In this chapter, we propose a self-supervised method to correct noisy labels which effectively relaxes the human labeling efforts. As we only have access to the noisy labels, we design a cycle-consistency loss which forces the learned transformation parameters to correct the misaligned labels to a unique location, *i.e.* the true location aligning the images. Furthermore, since image patches often contain more than one misaligned object, a universal transformation often leads to poor correction results. We therefore condition each instance prediction explicitly based on a spatial memory map consisting of all previous

iterations of corrections. In such a way, the resulting auto-regressive model corrects multiple objects sequentially. Finally, we show state-of-the-art results on the public INRIA Buildings benchmarks and more importantly, release and give baseline results on a new challenging railway tracks dataset for future research.

Chapter 4 - VGG-Sound: A Large-scale Audio-Visual Dataset. Gemmeke et al. [2017] proposed a audio dataset for training audio recognition networks, however, human annotations are extremely expensive for collecting such a large-scale dataset. This Chapter presents, to our knowledge, the first work to curate a large-scale audio-visual dataset in the wild automatically. Our automatic pipeline involves obtaining class list and candidate videos; visual verification; audio verification and a final iterative noise filtering. By running through the steps above, we guarantee that the sound is visually evident. We use this pipeline to build the VGG-Sound dataset consisting of more than 200k video clips spanning 309 classes. Furthermore, we compare audio recognition models trained on both VGG-Sound and AudioSet [Gemmeke et al., 2017] and demonstrate superior results given a lower amount of training data. Our dataset, VGG-Sound, has become one of the most widely used dataset for audio recognition as well as many audio-visual tasks.

Chapter 5 - Localizing Visual Sounds the Hard Way. In this chapter, we develop a method for sound localization in videos without requiring any labels. Most prior works address this problem by finding the correlation between the visual and audio modalities, forming a spatial attention map. We extend the model localization capabilities by automatically mining hard samples and adding them to a contrastive learning formulation. We show that this method significantly boosts sound localization performance on standard benchmarks, such as Flickr SoundNet [Senocak et al., 2018]. Furthermore, we collect a new benchmark called VGG-Sound Source (VGG-SS) for this task. We provide high-quality bounding box annotations for objects that produce sounds, this test set consists of more than 5k videos spanning 200 different classes. Lastly, we benchmark this task on VGG-SS, showing our method surpassing several baseline methods.

Chapter 6 - Audio-Visual synchronization in the wild. We address the problem of multi-class audio-visual synchronization in this chapter. The aim is to predict whether a given visual and audio pair is in-sync. Most prior arts [Chung and Zisserman, 2016a,b; Chung et al., 2019] focus on specific classes such as speech or musical instruments. To our best knowledge, this is the first work trying to solve audio-visual synchronization for general classes. To solve this problem, we identify and curate a dataset with classes containing high audio-visual evidence. In addition, we introduce a new transformer based architecture which is specifically designed to learn and predict using variable length video sequences. Several architecture variants are extensively ablated in various aspects. We further conduct an in-depth analysis on the curated dataset and define an evaluation metric for open domain audio-visual synchronization. Finally, we demonstrate state-of-the-art performances on standard lip reading speech benchmarks LRS2, LRS3, and set the first benchmark on general sound classes audio-visual synchronization.

1.3 Publications

The subsequent chapters in this thesis will describe the works that have been published in the following conferences;

- Chapter 3: AutoCorrect: Deep Inductive Alignment of Noisy Geometric Annotations:
  Honglie Chen, Weidi Xie, Andrea Vedaldi, Andrew Zisserman.
  Published in the proceedings of the British Machine Vision Conference (BMVC), 2019.

- Chapter 4: VGG-Sound: A Large-scale Audio-Visual Dataset:
  Honglie Chen, Weidi Xie, Andrea Vedaldi, Andrew Zisserman.
  Published in IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2020.

- Chapter 5: Localizing Visual Sounds the Hard Way:
  Honglie Chen, Weidi Xie, Triantafyllos Afouras, Arsha Nagrani, Andrea Vedaldi, Andrew Zisserman.

Published in the proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR), 2021.

- Chapter 6: Audio-Visual synchronization in the wild:

  Honglie Chen, Weidi Xie, Triantafyllos Afouras, Arsha Nagrani, Andrea Vedaldi, Andrew Zisserman.

  Published in the proceedings of the British Machine Vision Conference (BMVC), 2021.

Chapter 2

Literature Review

This thesis primarily focuses on how to train CNNs to learn a good representation without any manual supervision through the use of self-supervised learning. Self-supervised learning has been heavily researched in the past decade, with many successful approaches and applications.

In this chapter, large-scale audio and visual datasets are firstly reviewed in Section 2.1, as high quality datasets are extremely important when developing advanced learning algorithms. Next, in Section 2.2, we thoroughly explore self-supervised learning methods which learn invariant representations by using transformation/sequence correspondences. The dominant application domain in this thesis is audio-visual representation learning, we review a variety of audio-visual applications most pertinent to our work in Section 2.3. Finally, the recent proposed transformer, a new attention mechanism is discussed in Section 2.4.

2.1 Large-scale datasets

Throughout the history of Deep Learning research, the significant advancements not only rely on the development of new learning methods and utilization of powerful hardwares, datasets have also played a critical role. Large-scale datasets provide challenging benchmarks to train and evaluate, and more importantly, drive Artificial Intelligence research into many interesting directions. A collection of visual and audio milestone datasets are shown in Figure 2.1.

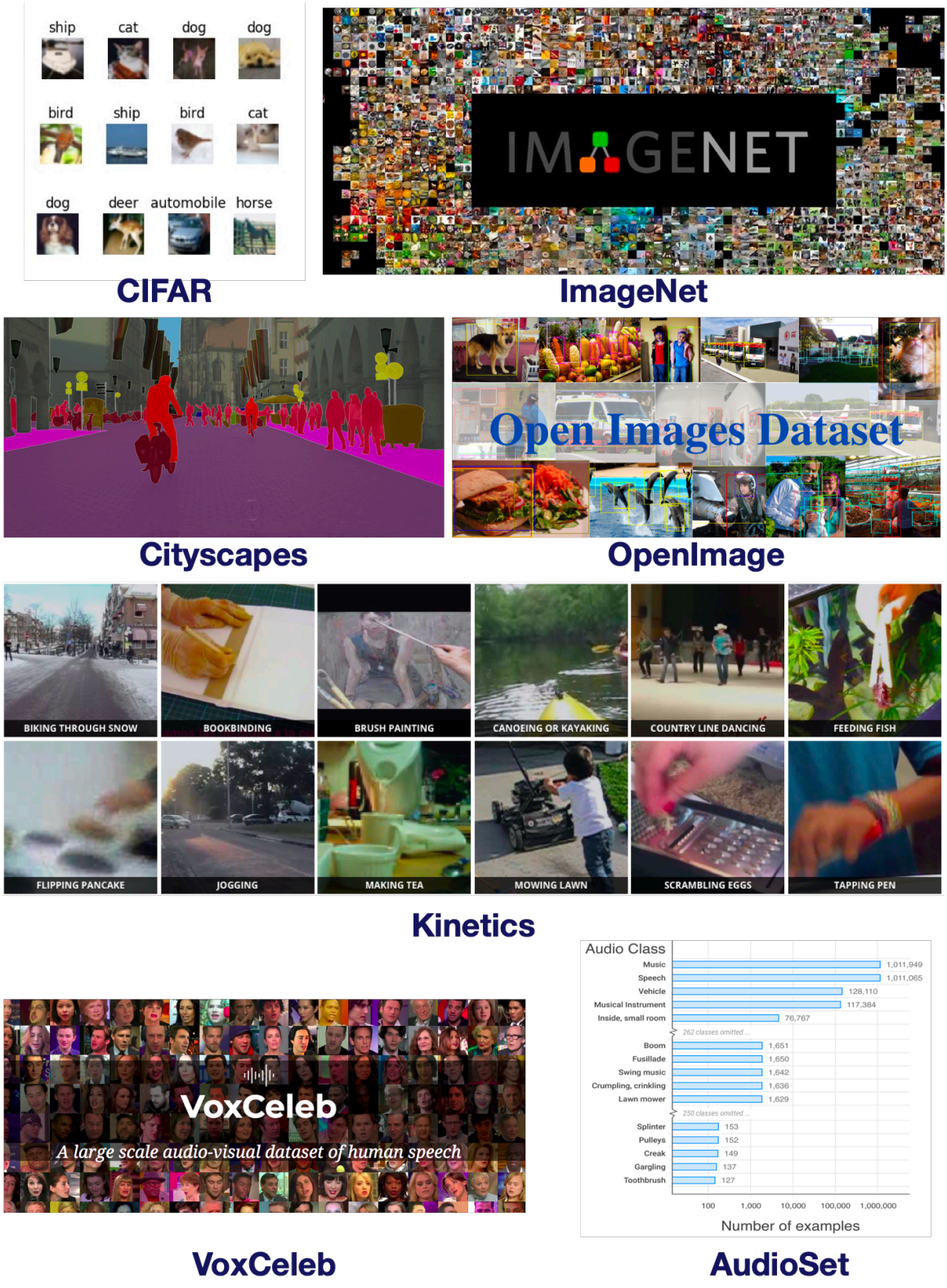


Figure 2.1: Selection of modern datasets.

2.1.1 Image datasets

We group image datasets roughly according to the object recognition type, *e.g.*, image classification, object detection and semantic segmentation datasets.

Image classification. The aim of image classification is to identify whether certain objects are present in the image. Early datasets of this type comprised images of a single object with empty background, *e.g.*, MNIST handwritten digits [LeCun] or COIL household objects [Nene et al., 1996]. Krizhevsky [2009]; Krizhevsky et al. [2012] propose CIFAR10 and CIFAR100 using small patches of images (32×32) with 6000 images per class. While the datasets above are actively utilized in many research studies, they only contain a small part of visual world. In 2009, Deng et al. [2009] revolutionized the field by building a large-scale image database called ImageNet, which consists of 22k categories with 500-1000 images each. Unlike previous datasets which are labeled with low-level classes such as "dog" or "chair", ImageNet is organized according to the WordNet hierarchy with more fine-grained categories. This is one of the most important and influential dataset in computer vision community, and has significantly advanced image classification domain. To further increase the number of complex scenes, the Open Image dataset [Kuznetsova et al., 2018] was released in 2016 with around 9 million images.

Object detection. Object detection refers to the capability of computer to locate the presence of objects with a bounding box and to predict classes of the located objects in an image. As a fundamental research topic in computer vision community, the classic works of [Wah et al., 2011] and [Dollar et al., 2009] play an important role in object detection research. While above works focus on particular classes, such as bird or pedestrian environments, PASCOL VOC [Everingham et al., 2010, 2015] evaluates 20 object categories including vehicles, household, animals, etc with $2,913$ images. This dataset has been widely used as a benchmark for object detection. More recently, Lin et al. [2014] proposed a dataset encompassing 80 categories of objects and $328,000$ images, namely Microsoft Common Objects in Context (MS COCO). In addition, Objects365 [Shao et al.,

2019] was released in 2019, which is 5 times larger than MS COCO [Lin et al., 2014], with 365 categories, 638k images, and 10,101k bounding boxes.

Semantic segmentation. The goal of semantic segmentation is to assign a category label to each pixel of an image. One of the hardest problems for any deep learning segmentation engines is the collection of data in order to construct high-quality image-label pairs. Early stage datasets such as Brostow et al. [2008] and Silberman et al. [2012] contain high quality annotations, but are often too small to train the data-hungry deep neural networks. Cordts et al. [2016] collected the Cityscapes dataset from 50 different European cities, which is widely used in semantic segmentation. To trade off number of annotations and speed, 20,000 coarse-annotated images and 5,000 fine-annotated images were released, where the coarse-annotated images are usually used in the pre-training stage to promote the models generalization.

In satellite imagery domain, semantic segmentation is useful for a number of applications such as urban planning, crop and forest management, etc. Several existing satellite image datasets [Mnih, 2013; Girard et al., 2018] are created via open source maps with semantic labels such as road and building footprint. However, those datasets often suffer from misalignment noise due to the mismatch between 3D structure and 2D image or labels not being temporally synchronized. In Chapter 3, we propose AutoCorrect to automatically fix geometric misaligned labels, so that high quality segmentation labels can be transformed given noisy labels. We then use this method to build a large-scale satellite image dataset, namely Railway Tracks dataset.

2.1.2 Video and audio datasets

Over the last decade, there has been growing research interests in video and audio based tasks. A number of challenging datasets were proposed and experimented, we will briefly review them in the following paragraphs.

Action recognition. Video datasets are often built through the following procedures: (1) Define a class list and obtain candidate videos; (2) Provide video-level/temporal annotations; (3) Clean the dataset by de-duplication and removing noisy clips. HMDB51 [Kuehne et al., 2011] was introduced in 2011, and collected mainly from movies. This dataset contains $6,849$ clips spanning 51 action categories. Similar in spirit, is the UCF101 [Soomro et al., 2012], but is larger to facilitate training of deep ConvNets. In 2017, Kay et al. [2017] curated a large-scale, high quality video benchmark obtained from YouTube videos called the Kinetics dataset. More than 240k training and 20k validation videos are available from 400 human action categories. Each video clip is trimmed into 10 seconds and is labeled with a single action class from a flatten list. More recently, Moment in Time [Monfort et al., 2019] was released with 1m video clips divided into 339 classes. Comparing to other datasets which focus on human actions, Moment in Time extends action recognition to a wider range of classes such as people, animals, objects and natural phenomena.

Audio recognition. The goal of audio recognition is to determine the semantic content of an acoustic signal, *e.g.*, recognizing the sound of a car engine, or a dog barking, etc. The early audio dataset was constructed using synthetic sound effects to convey different actions and materials [Gaver, 1993]. Burger et al. [2012] extends the former and introduce a dataset of 42 distinct labels called the noisemes, where the 5.6 hours of manually labeled data describe the distinct noise units based on audio concepts. Salamon et al. [2014] instead focus on sound classes chosen from the urban sound taxonomy and create an audio dataset with more than 18.5 hours. Recently, a large-scale dataset, namely AudioSet [Gemmeke et al., 2017; Hershey et al., 2021], was released with more than 2M video clips spanning more than 560 classes. This is one of the most influential datasets in audio research community, but such huge number of manual annotations can be very expensive to obtain. In contrast, automatic data collection is an economical and scalable alternative to manual supervision, we introduce an scalable audio dataset collection pipeline in Chapter 4. We curate VGG-Sound automatically and show state-of-the-art audio recognition results using this new dataset.

Audio-visual localization. The Flickr SoundNet sound source localization benchmark [Senocak et al., 2018] is an annotated collection of single frames randomly sampled from videos of the Flickr SoundNet dataset [Aytar et al., 2016]. This is the standard benchmark for sound source localization with 250 testing image-audio pair spanning around 50 classes. This testset is constructed with high quality bounding box annotation, but is in relatively small scale and only provides image-audio pairs. To address these problems, in Chapter 5, we build on VGG-Sound dataset and create VGG Sound Source, a new challenging benchmark with over 5k video clips spanning 220 classes.

2.2 Self-supervised learning

Recently, many self-supervised representation learning methods have gained popularity because of its ability to adopt self-defined pseudo labels as an alternative supervision to manual labels. We group these methods according to the supervision cues and discuss them in the following sections.

2.2.1 Temporal sequences

Temporal consistency has been considered as one of the most popular supervision sources and has been used in wide range of studies. Early works include learning invariant features from transformation sequences [Fldik, 1991], or using a construction loss to learn unsupervised features based on the fact that things typically cannot change too quickly from frame to frame in slow feature analysis (SFA) [Wiskott and Sejnowski, 2002; Hurri and Hyvrinen, 2003]. The authors of Mobahi et al. [2009] propose to add a temporal coherence regularizer to a traditional loss objective. Bengio and Bergstra [2009] extend on Mobahi et al. [2009] and use decorrelation as a mechanism for preventing trivial solutions. Furthermore, temporal coherence is defined to set hyper-parameters in a principled and automated manner [Goroshin et al., 2015]. While all above studies focus on preserving feature slowness, Jayaraman and Grauman [2016] propose preserving higher order feature steadiness, where the changes should be small in adjacent time intervals.

Temporal context can be also used as supervision for verifying or recognizing the temporal orders. Fernando et al. [2017] propose to identify the odd sequence from a set of sequences with correct temporal orders. Lee et al. [2017] formulate representation learning as a sequence sorting task and predict the correct frame orders. In Kim et al. [2019], 3D CNN-based methods are used to learn both spatial appearance and temporal relation of the video frames.

Another set of methods learn future frames sequences based on a limited number of frames in a video using a Long-term short-term memory based recurrent neural networks [Srivastava et al., 2015], or more generally, recurrent encoder-decoder architectures [Cho et al., 2014]. In Villegas et al. [2017], both motion and content information are decomposed in a end-to-end learning framework. Mathieu et al. [2016] explore a multi-scale architecture and an image gradient difference loss function to avoid blurry predictions obtained from the standard Mean Squared Error (MSE) loss function. Finn et al. [2016] train flexible parametric models to effectively predict interactions with objects.

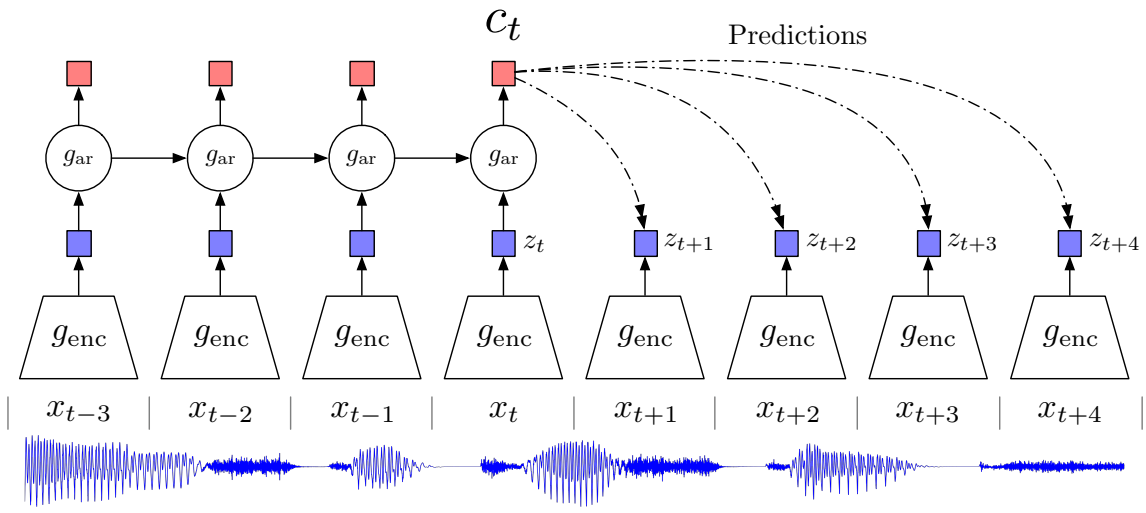


Figure 2.2: Overview of Contrastive Predictive Coding, adapted from Oord et al. [2018]. This approach is suitable for audio, images, text and reinforcement learning.

Contrastive Predictive Coding (CPC) [Oord et al., 2018] is a method that predicts future observations with a probabilistic contrastive loss [Chopra et al., 2005; Schroff et al., 2015]. The architecture of CPC models is shown in Figure 2.2, autoregressive models are used in

the latent space to make predictions many steps in the future. During training, both the encoder and autoregressive model are jointly optimized using the InfoNCE loss. In practice, given a training set $X = \{x_1, \dots x_N\}$ of N random samples containing one positive sample from $p(x_{t+k}|c_t)$ and $N-1$ negative samples from the 'proposal' distribution $p(x_{t+k})$, the loss is defined as:

$$\mathcal{L}_{\mathrm{N}} = -\mathop{\mathbb{E}}_{X}\left[\log \frac{f_k(x_{t+k}, c_t)}{\sum_{x_j \in X} f_k(x_j, c_t)}\right] \tag{2.1}$$

More recent approaches involve training with predictive attention mechanism over a set of compressed memories [Han et al., 2020a], jointly exploiting the information between RGB streams and optical flow [Han et al., 2020b], or matching several views of the same scene in Tian et al. [2019].

2.2.2 Spatial context

Solving jigsaw puzzles involves predicting the relative position of the different patches in an image [Doersch et al., 2015], or in a similar spirit, identifying the order of the shuffled sequence of patches from the same image [Noroozi and Favaro, 2016; Dahun et al., 2018; Noroozi et al., 2018]. Mundhenk et al. [2017] combine a set of methods to improve self-supervised learning results using context.

Transformation based self-supervised learning aims to learn invariant features by applying a set of transformations to each image, such as predicting rotation [Gidaris et al., 2018; Jing et al., 2018]. Novotny et al. [2018] propose to learn representation via correspondences obtained from synthetic warps. Invariant Information Clustering [Ji et al., 2019] is introduced to maximize mutual information between the function's classifications for paired data samples. Chen et al. [2020b] propose to use a stochastic data augmentation module to create two correlated views of the same example considered as positive training data, and all other samples in the minibatch are negative training data. Building on Chen et al. [2020b], He et al. [2020] create a dynamic dictionary containing large amount of negative data during training, showing superior results. In Chapter 3, we extend this idea of

transformation based self-supervised learning on satellite image domain that the location of railway tracks or buildings in both satellite image and semantic labels should be consistent. Therefore, any perturbed or noisy labels should arrive back to the unique location, *i.e.* the ground truth location.

2.2.3 Context similarity

One of the pretext tasks in the self-supervised scenario is clustering, where the cluster assignment is often used as an pseudo label to supervise the trainings. Traditional methods cluster images based on hand-crafted features such as HOG [Dalal and Triggs, 2005], SIFT [Lowe, 2004] and fisher vector [Sánchez et al., 2013]. Recently, instead of obtaining hand-designed features in advance, Li et al. [2016] train CNNs to recognize whether two images are from the same cluster. Noroozi et al. [2018]; Caron et al. [2018] propose to learn the cluster assignments using CNNs by iteratively clustering images and updating the weights of the network. Asano et al. [2020b] automatically estimate the data labels by combining clustering and representation learning.

2.2.4 Generation methods

Generation-based self-supervised learning aims to use pseudo labels generated from images themselves to supervise the training processes, several tasks including colorization, super-resolution and inpainting have been broadly researched in the vision community.

The objective of colorization is to predict a plausible color version of the photograph given a grayscale photograph as input. Larsson et al. [2016, 2017]; Zhang et al. [2016] consider colorization as a proxy task for visual understanding and show competitive results on several classification and segmentation benchmarks. While the works above focus on image domain, video colorization also gains tremendous interests in recent years, Tran et al. [2016] introduce an encoder-decoder based 3D ConvNet which can achieve competitive results for multiple tasks including optical flow estimation, semantic segmentation and video colorization. Vondrick et al. [2018] propose to solve video colorization by learning to copy colors from a reference frame. Another line of works focuses on super-resolution [Rudin et al., 1999; Tipping and Bishop, 2003; Kim et al., 2016; Dong et al., 2014], where the

goal is to enhance the resolution of images/videos. SRGAN [Ledig et al., 2017] casts the problem of super-resolution in a generative model with an adversarial loss and a content loss, photo-realistic textures were recovered from heavily downsampled images on public benchmarks. Image inpainting [Bertalmio et al., 2000; Criminisi et al., 2004; Patwardhan et al., 2007] is a task aimed at regressing missing pixel values based on the rest of an image. Pathak et al. [2016] train context encoders to learn the common knowledge including the color and structure of the objects, the learned features are then evaluated on classification, detection, and segmentation tasks.

2.3 Audio-visual learning

Human are surrounded by a world with multiple modalities, such as vision, audition, touch, taste, and smell. Following human multi-modal perception, it is natural to train intelligence models to interpret and reason about multi-modal messages. The aim of audio-visual learning is to build models that can process information from multiple modalities. There are a wide range of applications using multi-modal machine learning including image captioning [Farhadi et al., 2010; Donahue et al., 2014; Xu et al., 2015; Laina et al., 2019; Vinyals et al., 2015], visual question-answering [Antol et al., 2015; Selvaraju et al., 2017; Gao et al., 2015] and cross modal retrieval [Lee et al., 2018; Liu et al., 2019, 2021a]. In the following sections, we will review audio-visual learning in details as it is most pertinent to our works in Chapter 5 and Chapter 6.

2.3.1 Correspondence

Audio-visual concurrence or correlation has been considered for supervision for long time. In pre deep-learning era, Kidron et al. [2005] detect pixels that are associated with the sound (*e.g.* for a guitar) using canonical correlation analysis (CCA). Recent works [Aytar et al., 2016; Harwath et al., 2016] pre-train visual networks as a teacher and distill knowledge to the audio networks. Arandjelovic and Zisserman [2017] proposed a novel audio-visual correspondence task, the aim of this task is to decide whether a video frame and a short audio clip correspond to each other. They train a classifier in a completely unsupervised

manner by learning relevant semantic concepts in both modalities. In addition, semantic sound localization is investigated in Arandjelovic and Zisserman [2018]. While the above studies mainly focus on image domain, Owens and Efros [2018] extend this to a video-audio correspondence using a 3D convolution network which can also capture the motion information. Furthermore, a deep clustering model that extracted a set of distinct components from each modality was proposed in Hu et al. [2019]. The representation learned through audio-visual correspondence can indeed lead to numerous downstream tasks such as audio-visual matching, retrieval, separation and localization.

Matching and retrieval. Given an audio clip of a voice or an image of a face, the aim of Voice-Facial Matching is to determine the corresponding face image /video (V-F) or voice (F-V) respectively. Nagrani et al. [2018a] propose various different models to add temporal information and form a N-way network to deal with arbitrary number of face inputs at test time. Following work [Wen et al., 2019] considered using a triplet loss [Kim et al., 2018] or covariates (*e.g.* gender and nationality) to bridge the relation between voice and face information. While voice-facial matching focuses on human-centric performance, audio-visual retrieval comprises more general categories. Surs et al. [2018] proposed a joint embedding model and calculate the Euclidean distance between the audio and visual domains. Hong et al. [2018] investigated content-based music-video retrieval using pre-trained features and a intermodal ranking loss. Moreover, a curriculum learning [Bengio et al., 2009] schedule is employed during data sampling in Nagrani et al. [2018b] to further improve the performance.

Separation. Audio source separation, also known as the "cocktail party problem" [Haykin and Chen, 2005] has been extensively researched in the signal processing literature, traditional approaches include Non-negative Matrix Factorization [Smaragdis and Brown, 2003; Fvotte et al., 2009] and sparse decomposition [Zibulevsky and Pearlmutter, 2001]. While the above approaches only consider pure audio input, another class of methods solves separation with the aid of visual information. The classic audio-visual separation methods can be traced back to 2000s [Hershey and Movellan, 2000; Fisher III et al., 2000], where

a joint distribution of visual and audio signals is modeled and projected to a learned subspace. Recent works propose to separate a speakers voice given lip or human body regions in the corresponding video [Ephrat et al., 2018; Afouras et al., 2018a; Owens and Efros, 2018], and can achieve competitive performance even when lip region is occluded [Afouras et al., 2019b]. While the above works focus on human speech, Gao et al. [2018] extend this task to more general classes (*e.g.*, instruments, animals, and vehicles.) in large-scale in-the-wild videos containing multiple audio sources. Zhao et al. [2018b] propose to separate the input sounds into a set of components that represent the sound from each pixel in a range of instruments. In Zhao et al. [2019], instead of relying on image semantics, temporal information in the video is used to learn motion cues. Finally, the authors of Zhu and Rahtu [2021a] propose to use a light and efficient appearance attention module to achieve comparable or better audio-visual separation performance on the public MUSIC dataset [Zhao et al., 2018a].

Localization. The sound localization problem entails localizing the sound source by observing sound and visual scene pairs. The past efforts in this domain explored correlating image with sounds using a shallow probabilistic model [Hershey and Movellan, 1999; Fisher III et al., 2000; Kidron et al., 2005] or learning multimodal representations using canonical correlation analysis [Kidron et al., 2005; Izadinia et al., 2012]. As the number of unlabeled videos on the Internet has increased dramatically, recent methods mainly focus on self-supervised learning under cross-modal supervision. Arandjelovic and Zisserman [2018]; Senocak et al. [2018] propose to learn audio-visual representation, and visualize the learnt visual feature map. Rouditchenko et al. [2019] perform localization and separation tasks using only video frames or sound by disentangling concepts learned by CNNs. Hu et al. [2019] cluster audio and visual representations within each modality, followed by associating the resulting centroids with contrastive learning. Qian et al. [2020] propose to localize the sound source via a bootstrap approach, where the approximate locations of the objects are obtained from CAMs [Zhou et al., 2016a]. In Chapter 5, we propose a new learning algorithm which automatically mines hard examples during training and show significant boosts on audio-visual localization performance.

2.3.2 Synchronization

In contrast to previous audio-visual correspondence learning, the objective of audio-visual synchronization is to decide whether a given audio sample and a visual sequence are in-sync or out-of-sync. The classic works of Hershey and Movellan [1999]; Slaney et al. [2000] evaluate several statistical models in talking faces synchronization. Casanovas and Cavallaro [2014] propose a method for synchronizing audio-visual recordings of the same events from different cameras. Chung and Zisserman [2016a] employ a model for synchronizing lip movements to audio speech, based on a dual-encoder architecture trained with contrasting learning. Follow-up works improved this pipeline by moving to noise-contrastive objectives [Chung et al., 2019], or directly inferring the audio-visual offset conditional on cross-similarity patterns [Kim et al., 2021]. Inspired by human perception, which ignores large portions of the video in which no discriminative sounds exist, attention was investigated in [Khosravan et al., 2019] for audio-visual synchronization on speech data. Lip synchronization models are extremely useful for various visual speech related tasks, such as lipreading [Chung and Zisserman, 2016c; Afouras et al., 2019a], lip-syncing [Halperin et al., 2019], active speaker detection [Chung and Zisserman, 2016a] and sign language recognition [Albanie et al., 2020].

While the works above demonstrate strong synchronization performance, they are limited in terms of deployment as they are applicable only on videos that belong to human speech. Recently, Korbar et al. [2018] defined a binary classification problem called "Audio-Visual Temporal Synchronization" and adopted a curriculum learning strategy to improve learned feature quality on multiple classes. Similar in spirit is the Audio-Visual Scene Analysis by Owens and Efros [2018], but evaluates on more audio-visual tasks such as audio-visual localization and audio-visual separation. Our method in Chapter 6 aims to synchronize even broader sound classes (160 classes), while also outperform prior works in the speech domain.

2.3.3 Generation

Audio-to-Vision. Audio-to-visual generation aims to synthesize natural and intelligible images/videos given audio input. Traditional approaches [Garrido et al., 2015; Fan et al., 2015] are mainly limited to synthesize a talking face from speech audio of a specific person. More recently, a encoder-decoder architecture is proposed in Chung et al. [2017a] to generate a video of a talking face for more identities. In Chen et al. [2018], they extend the former work to fuse audio and image embedding to generate multiple lip images at once by designing a correlation loss to synchronize lip changes and speech changes. Furthermore, a cascade GAN is introduced to generate talking face videos robust to different face shapes, view angles, facial characteristics, and noisy audio conditions [Chen et al., 2019].

Vision-to-Audio. Many methods have been explored to extract audio information from visual information, popular applications include generating speech using lip motion [Ephrat et al., 2017; Le Cornu and Milner, 2017] or general video-to-audio translation [Owens et al., 2016; Zhou et al., 2017]. Owens et al. [2016] predict hitting sounds based on the different materials of objects and physical interactions. On the other hand, Zhou et al. [2017] directly synthesize waveforms using visual frames with a SampleRNN-style model [Mehri et al., 2016]. Chen et al. [2017] solve two generation tasks, Sound-to-Image and Image-to-Sound, using conditional generative adversarial networks. Finally, a generative model integrating physics, audio, and graphics engines was proposed to construct a synthetic audio-visual dataset [Zhang et al., 2017].

2.4 Transformers

The vanilla Transformer [Vaswani et al., 2017], a sequence-to-sequence model (Figure 2.3), was originally introduced for NLP tasks in 2017. Lately, transformer variants have garnered remarkable interests due to their strong performance across a range of domains like natural language processing [Devlin et al., 2019; Dai et al., 2019; So et al., 2019], computer vision [Parmar et al., 2018a; Sun et al., 2019; Dosovitskiy et al., 2021; Carion et al., 2020;

Arnab et al., 2021; Liu et al., 2021b] and speech processing [Dong et al., 2018; Gulati et al., 2020a; Chen et al., 2021b].

Transformer architectures can be roughly grouped into three different models, Encoder-only, Decoder-only and Encoder-Decoder. Among these models, the most popular Encoder-only ones include BERT [Devlin et al., 2019] that is typically used for natural language understanding tasks, or RoBERTa [Liu et al., 2020] which removes the NSP objective during training. The Decoder-only model, on the other hand, focus on language modeling such as GPT V1-3 [Radford and Narasimhan, 2018; Radford et al., 2018; Brown et al., 2020]. BART [Lewis et al., 2020] extend BERT to an encoder-decoder architecture so that both natural language understanding and generation can be performed. In Chapter 6, we use the Encoder-only and Decoder-only Transformer models extensively for uni-modal sequence modelling as well as for multi-modal fusion through attention, demonstrating state-of-the-art results on audio-visual synchronization.

Regardless of transformer architectures, the self-attention modules play an important role. Recent works improve the attention mechanism in various ways, including Sparse Attention [Child et al., 2019; Ho et al., 2020], Linearized Attention [Katharopoulos et al., 2020; Parmar et al., 2018b], Prototype and Memory Compression [Vyas et al., 2020], Low-rank Self-Attention [Guo et al., 2019], Attention with Prior [Yang et al., 2018] and improved Multi-Head Mechanism [Li et al., 2018a]. Inspired by dilated convolutions [van den Oord et al., 2016], Child et al. [2019] introduce sparse factorizations of the attention matrix, which can effectively relax memory and computational requirements while maintaining the performance on several benchmarks. Ho et al. [2020] employ independent attention modules over each axis of the image. The original dot-product attention was replaced by one that uses locality-sensitive hashing to select key-value pairs for each query, changing the complexity from $O(L^2)$ to $O(L \log L)$ in Kitaev et al. [2020]. Furthermore, linearized transformers were introduced in Katharopoulos et al. [2020], where a linear formulation is used to calculate self-attention weight, significantly reducing the memory. Similarly, performer [Choromanski et al., 2021] uses random feature maps that approximate the scoring function of a Transformer. In contrast to reducing memory in the attention matrix, clustered attention is proposed to only compute the attention for the centroids by firstly grouping the

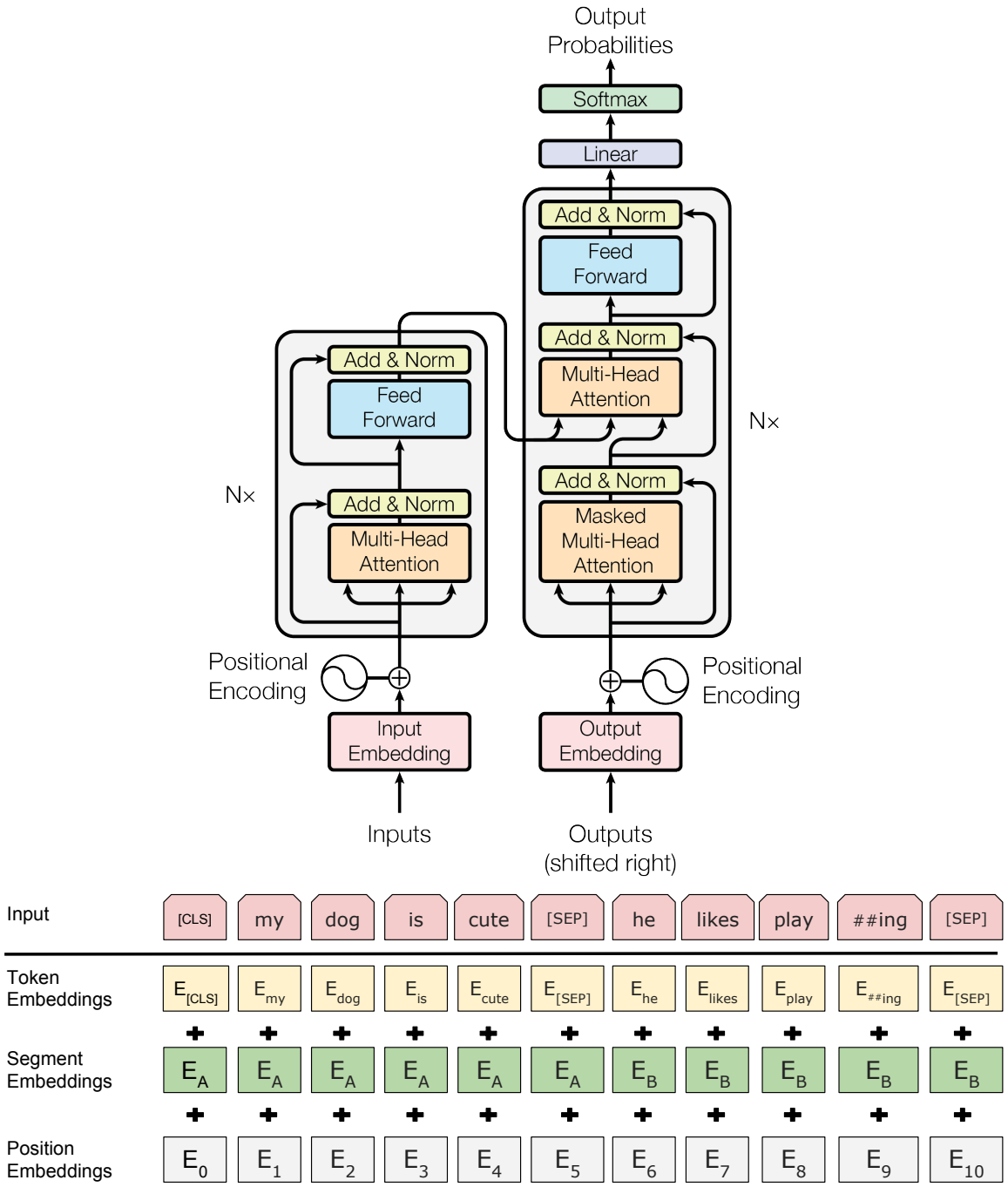


Figure 2.3: The vanilla transformer model and positional embeddings.

queries into clusters, hence reducing the memory usage [Vyas et al., 2020]. The authors of Liu* et al. [2018] propose Memory Compressed Attention (MCA) that reduces the number of keys and values using strided convolution. Another line of work exhibits a strong preference for the locality as a prior attention, which was modeled in Yang et al. [2018] to enhance the ability of capturing useful local context. Finally, Li et al. [2018a] introduce an auxiliary disagreement regularization term into the loss function to encourage diversity among different attention heads.

Chapter 3

AutoCorrect: Deep Inductive Alignment of Noisy Geometric Annotations

Honglie Chen **Weidi Xie**

Andrea Vedaldi **Andrew Zisserman**

Visual Geometry Group, University of Oxford

Abstract

We propose AutoCorrect, a method to automatically learn object-annotation alignments from a dataset with annotations affected by geometric noise. The method is based on a consistency loss that enables deep neural networks to be trained, given only noisy annotations as input, to correct the annotations. When some noise-free annotations are available, we show that the consistency loss reduces to a stricter self-supervised loss. We also show that the method can implicitly leverage object symmetries to reduce the ambiguity arising in correcting noisy annotations. When multiple object-annotation pairs are present in an image, we introduce a spatial memory map that allows the network to correct annotations sequentially, one at a time, while accounting for all other annotations in the image and corrections performed so far. Through ablation, we show the benefit of these contributions, demonstrating excellent results on geo-spatial imagery. Specifically, we show results using a new Railway tracks dataset as well as the public INRIA Buildings benchmarks, achieving new state-of-the-art results for the latter.

Published in the Proceedings of the British Machine Vision Conference (BMVC), 2019.

3.1 Introduction

Digital images are nowadays collected in enormous quantities. An important example is geo-spatial data, collected continuously by satellites, and containing a wealth of information useful for urban planning, crop and forest management, disaster relief, climate modelling, and many other applications. However, the scale of such datasets requires automated processing via machine learning and, while machine learning methods are increasingly powerful, providing annotations manually to train them can be prohibitively expensive.

The annotation costs may be substantially reduced if labels need not be very accurate. In this case, it is sometimes possible to *recycle* annotations that were not collected specifically for the images at hand. With geo-spatial data, for instance, there are publicly available maps (e.g. OpenStreetMap, Google Maps) that can provide annotations for large areas of the planet for free. However, while maps are generally accurate, they usually fail to match satellite images exactly due to various issues. To list a few: 1) maps do not capture the 3D structure of features such as buildings or vegetation, leading to misaligned annotations due to viewpoint variations; 2) maps may not be temporally synchronized with the satellite data, thus failing to account for variations in buildings, roads and vegetation; 3) features recorded in a map (e.g. subways) may not necessarily be visible in images and vice-versa. Figure 3.1 shows examples of noisy geometric labels obtained from these data sources in the *INRIA buildings* and our new *Railway tracks* datasets, and compares them with the manually-corrected versions.

Noisy labels can severely impact the quality of learned object detectors, as shown in satellite/aerial segmentation [Mnih and Hinton, 2012; Saito et al., 2016; Alshehhi et al., 2017] and detection [Laptev et al., 2000; Hu et al., 2007]. Hence, in this paper, we consider the problem of improving noisy labels to reduce or eliminate the impact of such noise on learned models. Our method, *AutoCorrect*, is mostly concerned with registration noise, which is usually the predominant noise type in geo-spatial data (Figure 3.1). We build a model that takes a set of images and misaligned object annotations as input and shifts the annotations to their correct image locations.

(a) (b) (c) (d)

Figure 3.1: Example aerial images with noisy labels (Red) and accurate labels (Green). (a) and (b) are extracted from the *INRIA buildings* dataset. (c) and (d) are examples in the *Railway tracks* dataset. The original labels (Red) demonstrate the clear registration noise. The cleaned labels (Green) show the corrections we aim to achieve (Human corrected).

There are several challenges. Satellite images usually contain multiple occurrences of the same object types, which may lead to association errors. Geo-spatial images capture the top of tall objects such as building and trees, whereas maps annotate their base. Finally, tall objects (e.g. trees in Figure 3.1(c) or buildings) can occlude other objects or cast significant shadows, so that some objects annotated in the map may effectively be invisible.

Given an image and a set of object annotations, *AutoCorrect* sequentially registers each annotation to its corresponding object occurrence by estimating an instance-level transformation. This is much more flexible than existing works that seek a single image-level transformation and allows us to obtain substantial improvements compared to these (indeed, as will be seen in the results, the annotations are displaced independently per object, and a single image-level correction will not suffice). However, this comes with several challenges. First, the model may not have access to any noise-free annotation, or at least not be aware of which ones are noise-free, making the correction process ambiguous. Second, there usually are several objects in each image, which means that the model must generalise to an arbitrary number of object occurrences whilst avoiding errors due to duplicate associations.

We solve the first problem by combining a geometric consistency loss, which is valid even if the ground-truth annotations are unknown, with a self-supervised loss, which is reliable for annotations with a small amount of noise. We also show that the symmetry of

certain objects such as roads provides an implicit constraint that makes registering annotations much less ambiguous. We solve the second problem by introducing a *spatial memory map* which represents all image annotations and reflects all previously-applied corrections.

3.2 Related work

Image alignment. Two very related works [Girard et al., 2018; Zampieri et al., 2018] have shown good alignment performance by training a CNN to predict a displacement field between a map and an image. Zampieri et al. [2018] uses a multi-scale CNN, and Girard et al. [2018] improves performance by training jointly for both alignment and segmentation. We compare to their results (and improve over them) in Section 3.4.

Inductive models and spatial memory. Explicit decomposition into repeated sub-tasks and recursively solving the problem have been applied in neural programming [Zaremba et al., 2016; Cai et al., 2017; Reed and de Freitas, 2016] and many visual tasks [Li et al., 2018b; Romera-Paredes and Torr, 2015; Kowalski et al., 2017; Carreira et al., 2016; Oberweger et al., 2015; Gupta et al., 2018]. In Kowalski et al. [2017], each stage predicts a landmark transformation that updates the keypoints iteratively. Similarly, an updater function is formulated in Oberweger et al. [2015] for hand pose alignment. Gupta et al. [2018] proposes an inductive RNN to localize visual objects which can generalise to an arbitrary number of inputs. Many of these methods use a form of spatial memory, though this isn't always made explicit. Others have used spatial memory for interactive image segmentation [Li et al., 2018b], and context reasoning in object detection [Chen and Gupta, 2017].

Cycle consistency. Assessing performance via cycling between two or more samples is a commonly used technique in computer vision. Many successful tasks like optical flow (with forward-backward consistency) [Sundaram et al., 2010], co-segmentation [Wang et al., 2014], image matching [Zhou et al., 2015, 2016b], image translation [Zhu et al., 2017] and domain adaptation [Hoffman et al., 2018] have shown its effectiveness. We introduce here a geometric-consistency loss: that within an image, misaligned annotations should be able to transform back to a single unique position.

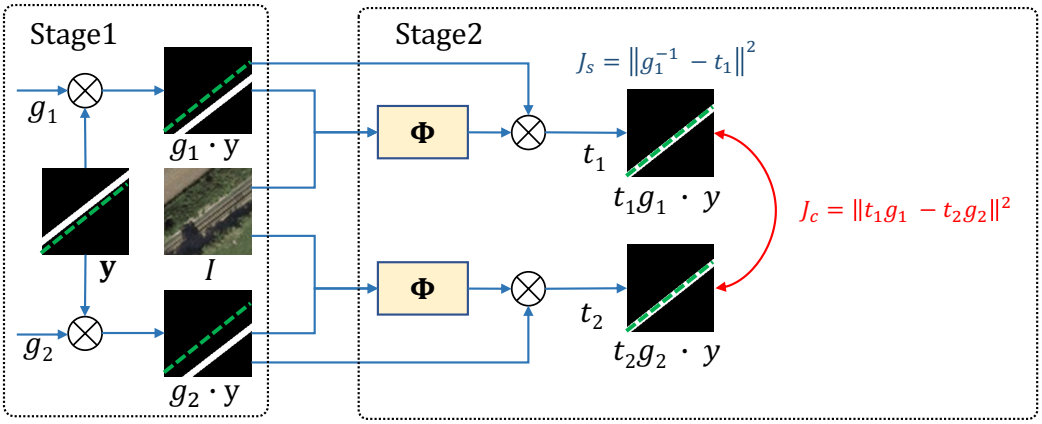


Figure 3.2: AutoCorrect architecture. The green dotted line shows the ground-truth label for the example image of a railway track. In Stage 1, given an image-label pair (I, y), the noisy annotation y is further perturbed by applying random transformations g_1 and g_2. In Stage 2, the network Φ computes corrections $t_i = \Phi(I, g_i \cdot y)$, $i = 1, 2$, producing corrected labels $(t_i g_i) \cdot y$ which must satisfy the consistency equation $J_c = 0$ (see text). If y is known to be a noise-free, then we can set $g_2 = t_2 = 1$ reducing J_c to the stricter constraint $J_s = 0$.

Learning with imperfect annotation. Most works on learning with imperfect annotations have considered classification, rather than registration. Examples include having a small set of clean samples (as well as many noisy) [Xiao et al., 2015; Veit et al., 2017], using robust loss functions [Ghosh et al., 2017; Patrini et al., 2017], or using a top-k loss [Berrada et al., 2018].

3.3 Approach

Our goal is to train deep networks for the detection of visual objects while relying on noisy annotations. While the approach is fairly general, we apply it to the detection of objects such as building and roads in geo-spatial images, where noisy annotations can be extracted from on-line data repositories such as mapping services. The mismatch between annotations and images is sometimes large, as shown in Figure 3.1. Naïvely training a model with these annotations leads to inaccurate predictions.

There are two main challenges. First, all annotations are potentially noisy and thus it is not clear how the noise can be identified and removed. Second, as different objects in the image may be misaligned in different ways, we must enable instance-level corrections

while handling an arbitrary number of object instances per image. We address these challenges in three ways. First, we use a self-supervised consistency loss based on the fact that multiple perturbations of the same label must always map to the same noiseless label. Second, we show that the intrinsic symmetry of certain visual objects provides a powerful implicit constraint that can reduce the ambiguity in the annotation clean-up process. Third, we introduce the idea of inductive alignment, adjusting annotations one instance at a time, sequentially, keeping track of the algorithm state by means of a *spatial memory map*. This is implemented by a recurrent neural network (RNN), which applies the same alignment logic to each annotation, but accounting for annotations already processed.

3.3.1 Single instance alignment

We start by describing a neural network architecture that can predict a translation and rotation for an individual object annotation in order to better align it to the image content. Note that, while this task may sound similar to object detection, it is in fact much easier as the annotation cues us to the *existence* and rough location of an object.

At each step, the input to the model is a concatenation of the RGB image $I \in \mathbb{R}^{3\times H\times W}$ with a scalar label map $y \in \{0,1\}^{H\times W}$ which encodes the annotation as a binary image. We know that the annotations can potentially be noisy, so we wish to learn a predictor function that outputs the transformation (*i.e.*2 scalars for translation and 1 for rotation) to align the image and annotation. This is implemented using a CNN that takes as input I and y and outputs a transformation t:

$$t = \Phi(I, y). \tag{3.1}$$

The corrected annotation $\hat{y} = t \cdot y$ is expressed as the transformed version of the annotation y by the predicted transformation $t \in G$, where G is a group of transformations $\mathbb{R}^2 \to \mathbb{R}^2$ such as 2D similarities. The symbol $\cdot$ denotes warping an image by a transformation. If the annotation is noise-free, t is expected to be an identity matrix and $\hat{y} = y$. If the annotation is noisy, the corrected annotation $\hat{\mathbf{y}}$ should approximate the underlying noise-free annotation $\mathbf{y}_{\text{gt}}$, which however is never observed during training.

Model (3.1) has several useful geometric properties:

Lemma 1. *If y_{gt} is the ground-truth annotation for image I and a perfect Φ is available, then $\Phi(I, y_{gt}) = \mathbf{1}$ is the identity transformation. Furthermore, for all invertible transformations $g \in G$, we have $\Phi(g \cdot I, y) = g\Phi(I, y)$ and $\Phi(I, g \cdot y) = \Phi(I, y)g^{-1}$.*

The lemma is easy to prove once we note that, if y_{gt} is the ground-truth label of image I, then $g \cdot y_{\text{gt}}$ is the ground-truth label of image $g \cdot I$. From this lemma, we can also see that any annotation that can be recovered from an image must have the same *symmetries* as the image itself.

Lemma 2. *Let $\hat{y} = \Phi(I, y) \cdot y$ be the annotation reconstructed from image I using model (3.1) and assume that $m \in G$ is a symmetry of the image,* i.e.*$I = m \cdot I$. Then the reconstructed annotation has the same symmetry, in the sense that $\hat{y} = m \cdot \hat{y}$.*

Proof. $$\hat{y} = \Phi(I, y) \cdot y = \Phi(m \cdot I, y) \cdot y = m\Phi(I, y) \cdot y = m \cdot \hat{y}.$$ □

This lemma shows that annotations can be predicted from images only if they have the same symmetries as the images. For example, if the model labels a straight road with a line, then the line must coincide to the road axis of symmetry. Hence image symmetries implicitly constrain the predictor (3.1) (in the example of the road, the correction must move the line onto the visual axis of symmetry of the road), reducing the ambiguity in registering the annotation. Note that this effect does not require specific images to be exactly symmetric; rather, it suffices that the object category is *statistically* symmetric (for example it is not possible to tell the direction of a road even if there are a few trees on one side, making the image asymmetric).

If we assume all annotations are correct, *i.e.*$y = y_{gt}$, then Lemma 1 can be used to train model (3.1) via self-supervised learning. The idea is to perturb the noise-free annotations synthetically by applying a random transformation $g \in G$ to the annotation $y = y_{\text{gt}}$. From Lemma 1, and using the assumption $y = y_{\text{gt}}$, we have $\Phi(I, g \cdot y) = \Phi(I, y)g^{-1} = 1g^{-1} = g^{-1}$. We may capture this constrain in the *self-supervised loss*:

$$J_s = \|g^{-1} - \Phi(I, g \cdot y)\|^2 \tag{3.2}$$

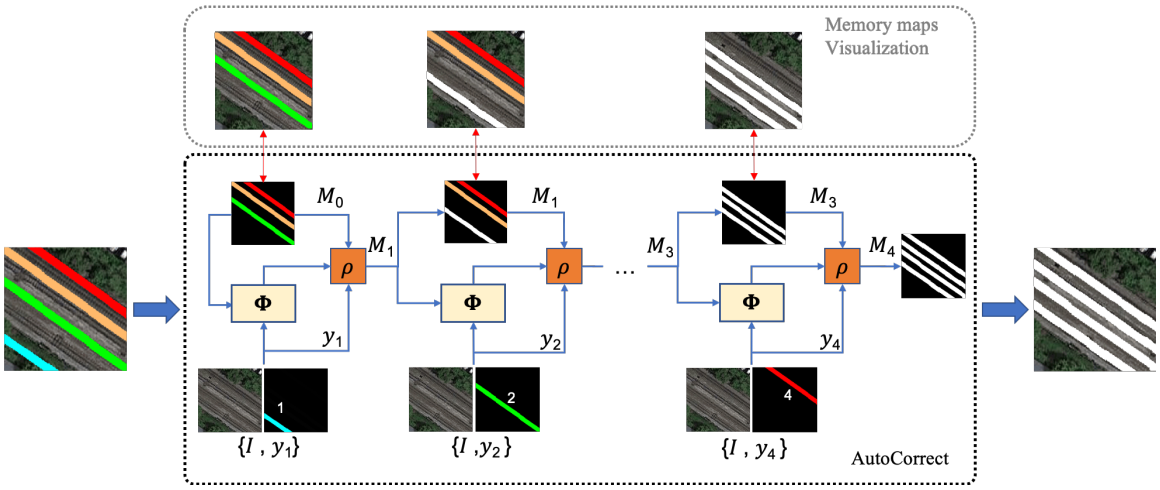


Figure 3.3: Correcting annotations sequentially using a memory map. Note, we demonstrate our correction process in the *AutoCorrect* box, whereas the top *Memory maps Visualization* box highlights the corrected annotation (white) on the satellite image. In detail, at each step, the input to the network is the concatenation of the RGB image I, the image of the annotation y_i to be corrected, and a memory M_{i-1} representing all other annotations, part of which have already been corrected (we colour-code annotations not yet corrected). An update function ρ, is applied to obtain the correction $y_i \mapsto t_i \cdot y_i$ and the latter is used to update M_i.

However, in our case y_{gt} is unknown so this loss can be used only as an approximation. In this case, the constraint can be written in term of *relative* transformations. To this end, consider applying two random transformations $y_1 = g_1 \cdot y$ and $y_2 = g_2 \cdot y$ to the annotation y. From Lemma 1, we have $\Phi(I, g_1 \cdot y)g_1 = \Phi(I, y) = \Phi(I, g_2 \cdot y)g_2$. This can be written as a *consistency loss*:

$$J_c = \|t_1 g_1 - t_2 g_2\|^2, \quad t_1 = \Phi(I, g_1 \cdot y), \quad t_2 = \Phi(I, g_2 \cdot y). \tag{3.3}$$

Intuitively, when two random transformations operate on one annotation, an ideal alignment model should be able to transform the annotation back to the same position, as y_{gt} is unique.

Overall, to train models on noisy data, we therefore consider a weighted combination $J = \alpha_s \cdot J_s + \alpha_c \cdot J_c$ (details in Section 3.3.3).

3.3.2 Inductive alignment

A naïve implementation of model (3.1) may align single object instances well, but it would fail when an image contains multiple object occurrences, especially when, as in satellite images, these are spatially close and similar in appearance. In particular, independent alignment may cause different noisy annotations to be incorrectly associated to the same object occurrence. To tackle this challenge, we introduce an inductive alignment model which uses an external *spatial memory map* to make the algorithm aware of all annotations present in the image as well as to keep track of all correction processed so far. Formally, given a training image I with n object annotations $\mathbf{y} = (y_1, \dots, y_n)$, our goal can be seen as estimating the joint posterior density of the transformation matrix for all the noisy object annotations $p(\{t_i\}_{i=1}^n | I, \mathbf{y})$. Rather than modelling multiple object annotations simultaneously, we break this down as *sequence* of simpler steps, in which a single transformation is predicted at a time, conditioned on the previous decisions, resembling an autoregressive model. Formally, this autoregressive model can be written as: $P(t_1, t_2, \dots, t_n | I, \mathbf{y}) = P(t_1 | I, \mathbf{y}) P(t_2 | I, t_1, \mathbf{y}) \cdots P(t_n | I, \{t_i\}_{i=1}^{n-1}, \mathbf{y})$. Note that this process requires learning a sequence of models $P(t_i | I, t_1, \dots, t_{i-1}, \mathbf{y})$. Directly parameterising the relations among transformations is difficult and results in a model which is rather opaque; instead, we propose to summarise the effect of conditioning on the previous corrections $t_1, \dots, t_{i-1}$ via a *spatial memory map* M_{i-1}, ideally, the memory map should represent all annotations and corrections performed so far *except* the annotation y_i that is currently being processed, formally:

$$P(t_i | I, t_1, \dots, t_{i-1}, \mathbf{y}) = P(t_i | I, M_{i-1}, y_i), \qquad M_i = M_{i-1} + t_i \cdot y_i - y_{i+1}. \tag{3.4}$$

An explicit example is illustrated in Figure 3.3, showing four railway track annotations to be corrected. The algorithm starts with four binary masks, each coding one of the noisy railway annotations, at the very *first* step, the memory M_0 is composed of three annotations (y_2, y_3, y_4), and y_1 is concatenated as additional input to the network Φ. Then, the first annotation y_1 is corrected by predicting the rigid transformation t_1, and the memory M_1 is updated by adding the image of $t_1 \cdot y_1$ and removing the image of annotation y_2,

readying for the next cycle. The induction process ends at M_4, where all tracks have been effectively corrected by the model. Note that the order we align instances is from left to right and bottom to top.

3.3.3 Implementation details

Consider a training image I with n noisy object annotations $\mathbf{y} = (y_1, \ldots, y_n)$. Annotations are perturbed by applying random transformations $g \cdot y_i$ where g is the composition of a translation of up to 25px in each direction and a rotation of up to 5 degrees (clockwise or anticlockwise) as this was found to be commensurate to the maximum amount of noise in the geo-spatial datasets we used for assessment.

During early training, we set the gating parameters in the joint objective function J as $\alpha_s = \alpha_c = 1$. This ensures the model converges quickly to an approximate solution within a few pixels of the ground-truth annotation, despite the fact that annotations are noisy so that term J_s in the objective function is not exactly valid. In a second phase, when the model is close to the final solution, the terms J_s and J_c start to be in conflict for the annotations that contain the largest amount of noise. Hence the coefficient α_s and α_c are adjusted as follows:

$$\begin{cases} \alpha_s = 0,\ \alpha_c = 1 & \text{if } \min(\mathrm{IoU}(t_1 g_1 \cdot y, y), \mathrm{IoU}(t_2 g_2 \cdot y, y)) < 0.2\ , \\ \alpha_s = 1,\ \alpha_c = 0 & \text{otherwise,} \end{cases} \tag{3.5}$$

where IoU denotes the standard *Intersection over Union* measure. This states that when any of the predicted corrections is far away from the given label, the label is expected to contain a large amount of noise, only the consistency loss J_c is applied; otherwise only the stricter self-supervised loss J_s is used.

Architecture and optimisation. The proposed *AutoCorrect* model uses as backbone architecture the VGG-M network [Chatfield et al., 2014] with minor modifications (details in the supplementary materiel). The network is trained using the Adam optimiser at an initial learning rate of 10^{-4}, which is divided by 10 after the training error plateaus.

3.4 Experiments

The experiments thoroughly assess our *AutoCorrect* method on two benchmark datasets: our own Railway tracks dataset and the INRIA buildings dataset. Project page: http://www.robots.ox.ac.uk/~vgg/research/autocorrect/.

3.4.1 Datasets and evaluation

Railway tracks dataset. The *Railway tracks* dataset was obtained by extracting views of railways in the UK region from Google Maps. We used zoom level 19, which corresponds to approx. 0.5 meter/pixel (this is the minimum zoom level at which railway tracks can be resolved) and results in images with a 640×640 pixel resolution. The dataset contains approximately 35k overhead images of the tracks. Binary mask annotations are provided by Google Maps to indicate the position of the railway tracks; however, the annotations are not perfectly aligned with the images (shown in Figure 3.1). In order to evaluate the effectiveness of the self-supervised learning loss, the consistency loss, and the spatial memory map, we manually identify 4,000 images for which railway annotations are accurate. We use these in the experiments by synthetically adding noise to these ground-truth annotations.

INRIA buildings dataset. The *INRIA buildings* dataset contains 360 images of $5{,}000 \times 5{,}000$ pixels. This dataset may seem small compared to other deep learning datasets, but as each image has a large spatial footprint, it contains a large number of buildings (13,614 buildings just in the test split). In order to directly compare with prior work, we adopt the same data and evaluation protocol of Girard et al. [2018].

Evaluation metrics. In order to evaluate the effectiveness of the proposed *AutoCorrect* model, For *Railway tracks* dataset, we assess railway alignment using the standard IoU measure between the image of a noise-free label and the predicted correction of a noisy label. For the *INRIA buildings* dataset, in order to compare with existing work, we adopt the standard protocol and report results using the *Percentage of Correct Keypoints* (PCKs) metric. The reason we apply the IoU measure on railway tracks is that railway tracks tend

to be straight and long, so that, unlike for buildings, it is difficult to define keypoints. Note that IoU is very sensitive for thin structures such as railroads.

Model	Data	Noise		SMM	Consist.	IoU
A	3k	0%	—	×	×	0.321
B	3k	0%	—	✓	×	0.425
C	3k	0%	—	✓	✓	0.436
D	3k	20%	Synth.	✓	×	0.404
E	3k	20%	Synth.	✓	✓	0.429
F	3k	40%	Synth.	✓	×	0.369
G	3k	40%	Synth.	✓	✓	0.381
H	20k	∼40%	Natural	✓	×	0.417
I	20k	∼40%	Natural	✓	✓	0.435
J	35k	∼40%	Natural	✓	✓	0.445

Table 3.1: Railway tracks dataset results. The SMM and Consist. refers to the spatial memory map and consistency loss respectively.

3.4.2 Railway tracks results

Synthetic annotation noise. In the following, we use the $4{,}000$ images with ground-truth (i.e. correct) annotations, split as $3{,}000$ for training the *AutoCorrect* network and $1{,}000$ for testing. With these image-annotation pairs, we aim to perform controlled experiments on evaluating the effectiveness of the proposed components. First, we assess the effectiveness of the spatial memory map by training our model using only the $3{,}000$ noise-free annotations and the self-supervised loss. Then, to evaluate the robustness of the consistency loss against different levels of noise, we intentionally replace the noise-free annotations with perturbed ones in training set, and train three sets of models, with resp. $0\%, 20\%, 40\%$ noisy annotations, and using or not using the consistency loss. During the testing stage, we artificially perturb the $1{,}000$ testing images three times, and apply our models to correct the perturbed testing annotations. All artificial perturbations are composed of a random translation up to 25px in each direction and a random rotation up to 5 degrees (clockwise or anti-clockwise).

As shown in Table 3.1, models A-G are trained on only 3k images with noise-free labels or with the injection of synthetic noise in part of those. H, I, J are trained on real annotation noise. First, to show that our *spatial memory map* plays an important role in the instance alignment, we compare models A and B: the performance gap is significant (0.321 vs. 0.425 IoU), as the spatial memory map gives important contextual information. Second, comparing models B and C shows that the consistency loss is beneficial even when training on the noise-free subset of the data. We conjecture that this is because the consistency loss acts as a regularizer. Third, to verify the effectiveness of the consistency loss in dealing with noisy data, we note that as the noise ratio is increased (models D and F), the performance of the model that uses only the self-supervised loss starts to drop dramatically (0.404 and 0.369 IoU); however, the transformation consistency loss improves the robustness to noise significantly (models E and G, 0.429 and 0.381 IoU). Note that, when the noise ratio is around 20%, model E actually performs about as well as model C, which learned on noise-free annotations. This shows that models trained with transformation consistency can discount almost entirely moderate amounts of noise.

Natural annotation noise. After demonstrating the concept in these controlled experiments, we now train the network using the entire dataset (which we estimate to contain 40% of labels with significant geometric distortion), using either 20k or 35k images and switching the consistency loss on and off to test its effectiveness once more. Similar to synthetic annotation noise, we artificially perturb the 1,000 testing images to evaluate models trained on natural annotation noise. The models I and J (20k/35k images, 0.435/0.445 IoU) show that, even with substantial real annotation noise ($\sim$40%), our model reaches similar or superior performance to using a manually filtered dataset (C, 3k images, 0.436 IoU) with no annotation noise. The advantage is that, while datasets I and J are large, they are obtained "for free" without any manual filtering.

3.4.3 INRIA buildings dataset results

To evaluate our alignment method on the *INRIA buildings* dataset, we follow the standard testing protocol introduced in [Girard et al., 2018; Zampieri et al., 2018] by randomly and

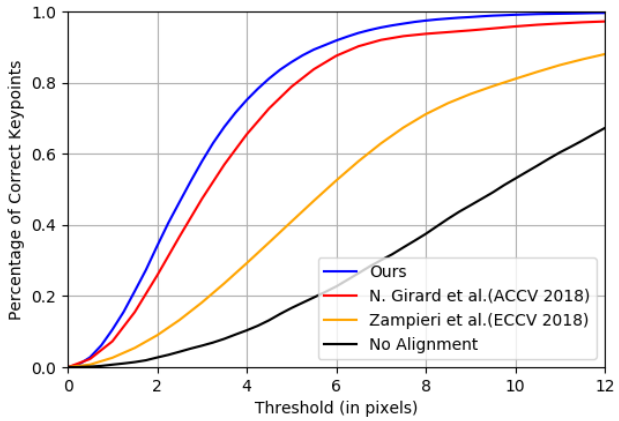


Figure 3.4: INRIA buildings dataset results. We outperform all recent works; from around 10 pixels threshold, our result is 100% (i.e. it cannot be improved further).

independently perturbing the accurate annotations on 3 images of the city of San Francisco. In contrast to generating displacement maps in [Girard et al., 2018; Zampieri et al., 2018], we consider instance-level transformations. The testing labels are generated by randomly and independently perturbing the accurate annotation instances to achieve an error comparable to that of [Girard et al., 2018; Zampieri et al., 2018]. As shown in Figure 3.4, the *AutoCorrect* approach outperforms all previous methods at all thresholds (in pixels). This is because our method outputs transformation parameters for each instance independently, whereas prior works outputs a displacement field map, which is less expressive. Furthermore, our consistency loss also works as a form of data augmentation which counters the small size of the *INRIA buildings* dataset, further improving the performance.

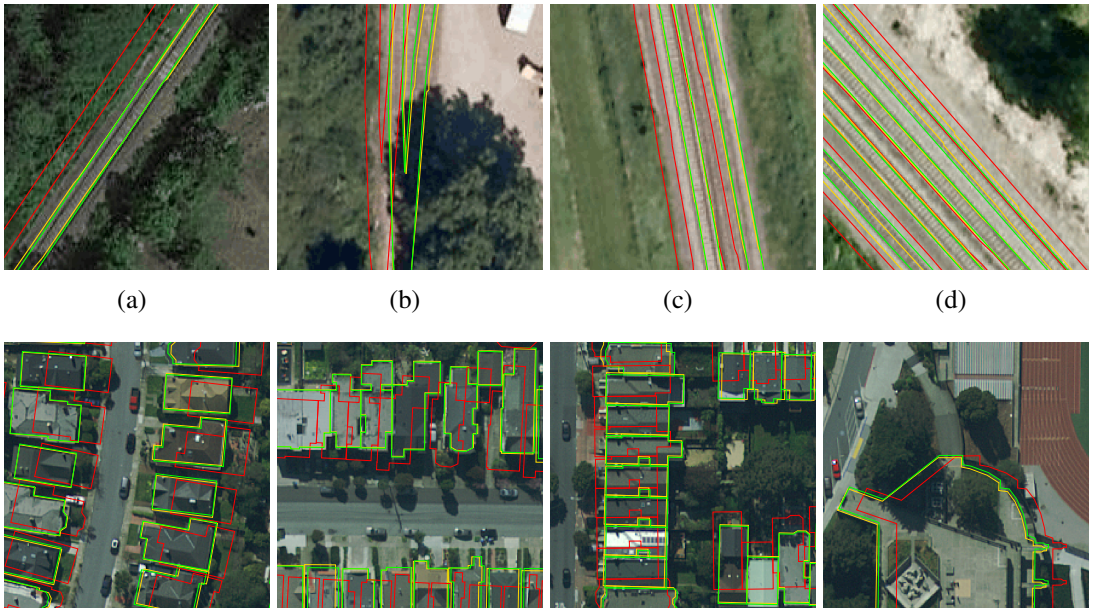
Note that we learn to correct random and *different* perturbations of objects that co-occur in the same image, therefore, our proposed local (per-object) correction is a better match to the type of errors observed in practice in aerial datasets as the location of the shifted annotations can be random and uncorrelated.

3.4.4 Qualitative results

As label noise of Satellite imagery is random, each instance label must be considered and corrected individually. Our *AutoCorrect* models deal with geometric alignments on an arbitrary number of instances, by aligning each instance sequentially. Figure 3.5 shows the

AutoCorrect correction progression on testing data. Since the noise of each label is random, our model handles it by iteratively correcting each semantic label individually. Figure 3.6 shows the *AutoCorrect* final predictions on a number of examples from the test data.

Figure 3.5: Correction progression. Red polygons refer to noisy labels, green to noise-free labels, and yellow are our predictions. Noisy annotations are cleaned inductively.



(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 3.6: Alignment results for the *Railway tracks* dataset examples (top row), and *INRIA buildings* dataset (bottom row). The label noise of each instance (denoted in red) is random *i.e.*local transformation of each instance is needed. Our predictions (denoted in yellow) achieve accurate correction comparing to ground truth (Green), by predicting a transformation on each instance. In reality, *AutoCorrect* can correct both noisy instance with regular shape (a), as well as instances with complex shapes ((b) & (h)). Figures (c) and (d) illustrate the capability of correcting an arbitrary number of instances.

3.5 Conclusion

The *AutoCorrect* method is based on three ideas: a spatial memory map that enables annotations to be adjusted sequentially while taking into account the other annotations and their corrections, a consistency loss that enables the model to be trained without the knowledge of any noise-free annotation, and a self-supervised loss that generates training data automatically. *AutoCorrect* outperforms previously-published works and can learn to correct almost for free from a large dataset where 40% of the annotations are heavily distorted, and obtain results that are comparable to approaches that require noise-free annotations. Finally, we have introduced the new *Railway tracks* benchmark.

Acknowledgement. We thank Kai Han, Erika Lu and Tengda Han for proofreading. Financial support was provided by the EPSRC Programme Grant Seebibyte EP/M013774/1.

Appendices

3.A Architecture details

In table 3.2 and table 3.3, we describe the architecture details used in the paper. Due to input resolutions for the two datasets, the network architectures are slightly different. The network layer name is denoted as "⟨type⟩⟨kernel size⟩-⟨number of channels⟩".

Input (640 x 640 x 5)
conv7-96
maxpool
conv5-256
maxpool
conv5-512
maxpool
conv5-512
maxpool
conv5-512
maxpool
FC-4096
FC-4096
FC-3

Table 3.2: Architecture for Railway tracks dataset.

Input (384 x 384 x 5)
conv7-96
maxpool
conv5-256
maxpool
conv5-512
maxpool
conv5-512
conv5-512
maxpool
FC-4096
FC-4096
FC-3

Table 3.3: Architecture for INRIA buildings dataset.

3.B More results from the *AutoCorrect* approach

We show additional railway alignment results in Figure 3.7 and Figure 3.8, as well as additional building alignment results in Figure 3.9 and Figure 3.10. In all figures, the original noisy labels are denoted as red, our predictions are in yellow, and ground truth in green. For the railway tracks, the noisy annotations are from actual samples, no synthetic perturbations are added. For the INRIA buildings, we follow previous work [Girard et al., 2018], and perturb the annotations with artificial transformation using their specified noise level. As shown in the figures, our predictions achieve accurate alignments for both datasets.

Figure 3.7: Railway tracks alignment.

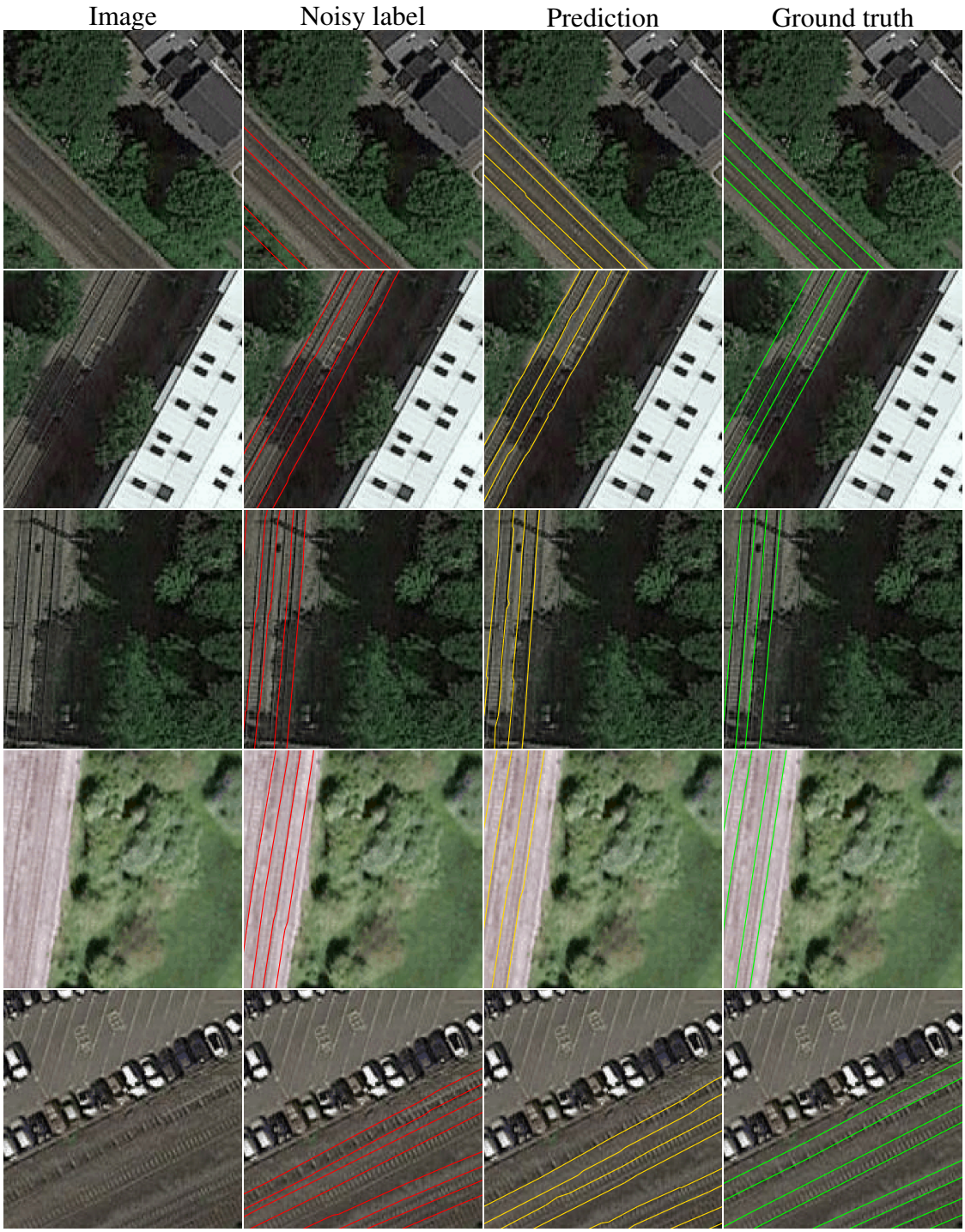


Figure 3.8: Railway tracks alignment.

Figure 3.9: INRIA buildings alignment.

Figure 3.10: INRIA buildings alignment.

Chapter 4

VGG-Sound: A Large-scale Audio-Visual Dataset

Honglie Chen **Weidi Xie**

Andrea Vedaldi **Andrew Zisserman**

Visual Geometry Group, University of Oxford

Abstract

Our goal is to collect a large-scale audio-visual dataset with low label noise from videos 'in the wild' using computer vision techniques. The resulting dataset can be used for training and evaluating audio recognition models. We make three contributions. First, we propose a scalable pipeline based on computer vision techniques to create an audio dataset from open-source media. Our pipeline involves obtaining videos from YouTube; using image classification algorithms to localize audio-visual correspondence; and filtering out ambient noise using audio verification. Second, we use this pipeline to curate the VGG-Sound dataset consisting of more than 200k videos for 309 audio classes. Third, we investigate various Convolutional Neural Network (CNN) architectures and aggregation approaches to establish audio recognition baselines for our new dataset. Compared to existing audio datasets, VGG-Sound ensures audio-visual correspondence and is collected under unconstrained conditions. Code and the dataset are available at http://www.robots.ox.ac.uk/~vgg/data/vggsound/.

Published in IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2020.

4.1 Introduction

Large-scale datasets [Everingham et al., 2010; Deng et al., 2009] have played a crucial role. in many deep learning recognition tasks [Simonyan and Zisserman, 2015; He et al., 2016; Hershey et al., 2017]. However in the audio domain, while several audio datasets have been released in the past few years [Salamon et al., 2014; Foggia et al., 2015; Fonseca et al., 2017; Mesaros et al., 2016], the data collection process usually requires extensive human efforts, making it unscalable and often limited to narrow domains. *AudioSet* [Gemmeke et al., 2017], is a large-scale audio-visual dataset containing over 2 million clips in unconstrained conditions. This is a valuable dataset, but it required extensive human verification in order to construct it. In contrast to these manually curated datasets, recent papers have demonstrated the possibility of collecting high-quality human speech datasets in an automated and scalable manner by using computer vision algorithms [Nagrani et al., 2017a, 2020; Chung et al., 2018].

In this paper, our objective is to collect a large-scale audio dataset, similar to *AudioSet*, containing various sounds in the natural world and obtained 'in the wild' from unconstrained open-source media. We do this using a pipeline based on computer vision techniques that guarantees audio-visual correspondence (*i.e.* the sound source is visually evident) and low label noise, but requires only minimal manual effort.

Train	Val	Test	Total train	Total	Classes
130-900	20	50	177,837	199,467	309

Table 4.1: `VGG-Sound` Dataset Statistics. The number of clips for each class in the train/-val/test partitions and the total numbers of train and classes.

Our contributions are three-fold: The first is to propose an automated and scalable pipeline for creating an 'in the wild' audio-visual dataset with low label noise. By using existing image classification algorithms, our method can generate accurate annotations, circumventing the need for human annotation. Second, we use this method to curate

VGG-Sound, a large-scale dataset with over 200k video clips (visual frames and audio sound) for 309 audio classes, from YouTube videos. Each 10s clip contains frames that *show* the object making the sound, and the audio track contains the the *sound* of the object. There are at least 200 clips for each audio class. Our third contribution is to establish several baselines for audio recognition on the new dataset. To this end, we investigate different architectures, VGGish [Simonyan and Zisserman, 2015; Gemmeke et al., 2017] and ResNet [He et al., 2016] networks, as well as different aggregation approaches, global average pooling and NetVLAD [Arandjelovic et al., 2016; Xie et al., 2019], for training deep CNNs on spectrograms extracted directly from the audio files with little pre-processing.

We expect VGG-Sound to be useful for both audio recognition and audio-visual prediction tasks. The goal of audio recognition is to determine the semantic content of an acoustic signal, *e.g.* recognizing the sound of a car engine, or a dog barking, *etc*. In addition, VGG-Sound is equally well suited for studying multi-modal audio-visual analysis tasks, for example, *audio grounding* aims to localize a sound spatially, by identifying in an image the object(s) emitting it [Arandjelovic and Zisserman, 2017; Kidron et al., 2005]. Another important task is to separate the sound of specific objects as they appear in a given frame or video clip [Owens et al., 2016; Zhao et al., 2018b].

4.2 Related Work

Audio and audio-visual datasets. Several audio datasets exist, as shown in Table 4.2. The UrbanSound dataset [Salamon et al., 2014] contains more than 8k urban sound recordings for 10 classes drawn from the urban sound taxonomy. The Mivia Audio Events Dataset [Foggia et al., 2015] focuses on surveillance applications and contains 6k audio clips for 3 classes. The Detection and Classification of Acoustic Scenes and Events (DCASE) community organizes audio challenges annually, for example, the authors of [Mesaros et al., 2019] released a dataset containing 17 classes with more than 56k audio clips. These datasets are relatively clean, but the scale is often too small to train the data-hungry Deep Neural Networks (DNNs).

To remedy this shortcoming, a large-scale dataset of video clips was released by Google. This dataset, called *AudioSet*, contains more than 2 million clips drawn from YouTube and is helpful not only for audio research, but audio-visual research as well, where the audio and visual modalities are analysed jointly. This dataset is a significant milestone, however, the process used to curate *AudioSet* requires extensive human rating and filtering. In addition, the authors of Tian et al. [2018] manually curated a high-quality, but small dataset that guarantees audio-visual correspondence for multi-modal learning, where the objects or events that are the cause of a sound must also be observable in the visual stream.

Datasets	# Clips	Length	# Class	Video	AV-C
UrbanSound [Salamon et al., 2014]	8k	8.75h	10	×	×
MIVIA [Foggia et al., 2015]	6k	29h	3	×	×
DCASE2017 [Mesaros et al., 2019]	57k	89h	17	×	×
FSD [Fonseca et al., 2017]	24k	119h	398	×	×
AudioSet [Gemmeke et al., 2017]	2.1m	5833h	527	✓	×
AVE [Tian et al., 2018]	4k	11.5h	28	✓	✓
VGG-Sound (Ours)	200k	550h	309	✓	✓

Table 4.2: Statistics for recent audio datasets. "# Clips", the number of clips in the dataset; "Length", the total duration of the dataset; "# Classes", number of classes in the dataset; "Video", whether videos are available; "AV-C", whether audios and videos correspond, in the sense that the sound source is always visually evident within the video clip.

Audio Recognition. Audio Recognition, namely the problem of classifying sounds, has traditionally been addressed by means of models such as Gaussian Mixture Models (GMM) [Zhuang et al., 2010] and Support Vector Machines (SVM) [Temko and Nadeu, 2006] trained by using hand-crafted low-dimension features such as the Mel Frequency Cepstrum Coefficients (MFCCs) or i-vectors [Huang et al., 2013]. However, the performance of MFCCs in audio recognition degrades rapidly in "unconstrained" environments that include real-world noise [Yapanel et al., 2002; Hansen et al., 2001]. More recently, the success of deep learning has motivated approaches based on CNNs [Takahashi et al., 2016; Hershey et al., 2017] or RNNs [Parascandolo et al., 2016; Choi et al., 2017; Xu et al., 2018]. In this paper, rather than developing complex DNN architectures specific to audio recognition, we choose to illustrate the benefits of our new benchmark dataset by training baselines to serve as comparison for future research. To this end, we train powerful ResNet architectures with

the NetVLAD aggregation method for audio recognition tasks [Arandjelovic et al., 2016; Xie et al., 2019].

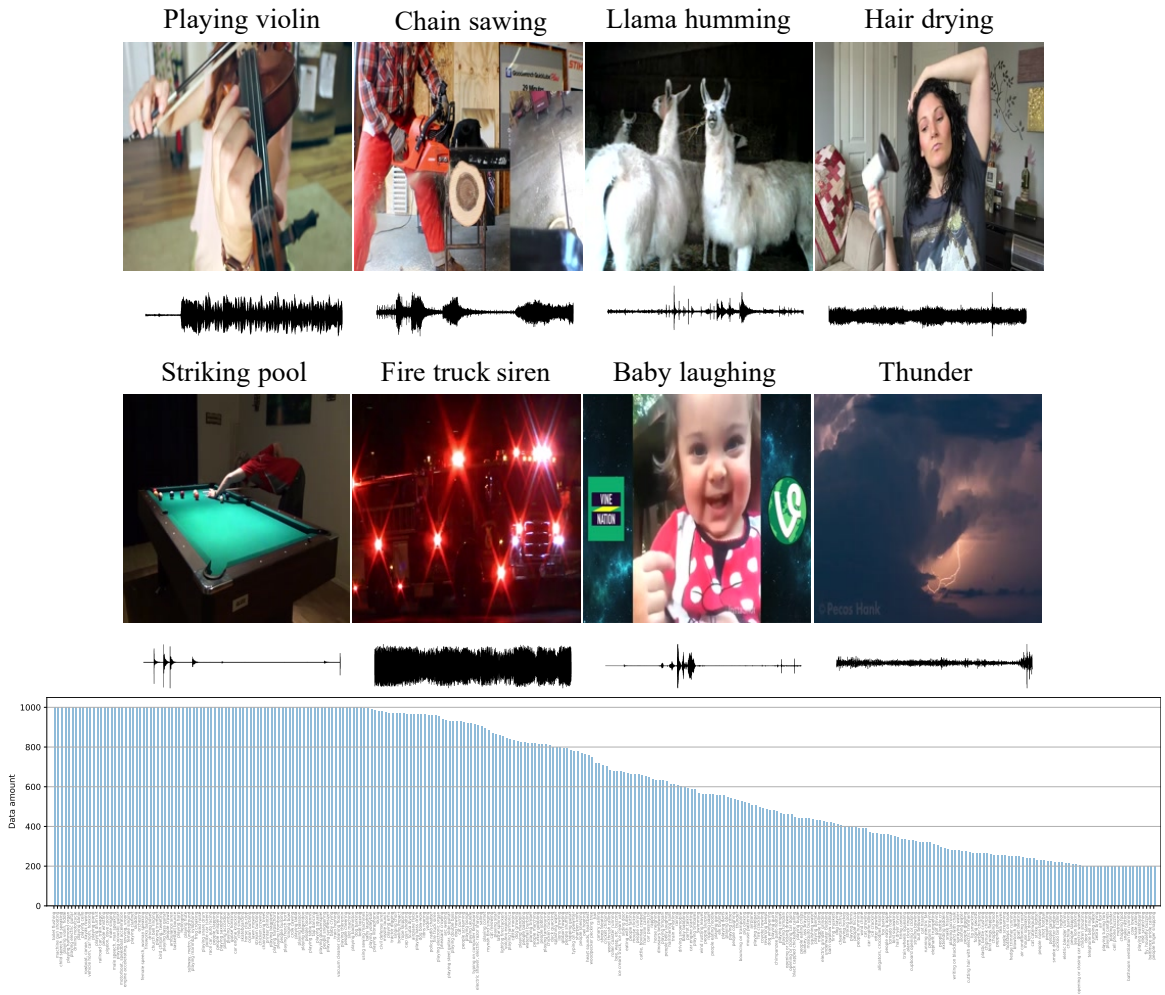


Figure 4.1: The top two rows of this figure shows example video frame and audio pairs of VGG-Sound classes. the bottom bar chart demostrates VGG-Sound classes with sizes of each audio class sorted by descending order.

4.3 The VGG-Sound dataset

VGG-Sound contains over 200k clips for 309 different sound classes. The dataset is audio-visual in the sense that the object that emits each sound is also visible in the corresponding video clip. Figure. 4.1 shows example cropped image frames, corresponding audio waveforms, and a histogram details the statistics for each class. Each sound class contains

200–1000 10s clips, with no more than 2 clips per video. The set of sound labels is flat (*i.e.* there is no hierarchy as in *AudioSet*). Sound classes can be loosely grouped as: people, animals, music, sports, nature, vehicle, home, tools, and others. All clips in the dataset are extracted from videos downloaded from YouTube, spanning a large number of challenging acoustic environments and noise characteristics of real applications.

In the following sections, we describe the multi-stage approach that we have developed to collect the dataset. The process can be described as a cascade that starts from a large number of potential audio-visual classes and corresponding videos, and then progressively filters out classes and videos to leave a smaller number of clips that are annotated reliably. The number of classes and videos after each stage of this process is shown in Table 4.3. The process is extremely scalable and only requires manual input at a few points for well defined tasks.

Stages	Goal	# Classes	# Videos
1	Candidate videos	600	1m
2	Visual verification	470	550k
3	Audio verification	390	260k
4	Iterative noise filtering	309	200k

Table 4.3: The number of classes and videos after each stage of the generation pipeline. Note, classes with less than 100 videos are removed from the dataset.

4.3.1 Stage 1: Obtaining the class list and candidate videos.

The first step is to determine a tentative list of sound classes to include in the dataset. We follow *three* guiding principles in order to generate this list. First, the sounds should be *in the wild*, in the sense that they should occur in real life scenarios, as opposed to artificial sound effects. Second, it must be possible to *ground and verify the sounds visually*. In other words, our sound classes should have a clear visual connotation too, in the sense of being predictable with reasonable accuracy from images alone. For instance, the sound 'electric guitar' is visually recognizable as it is generally possible to visually recognize someone playing a guitar, but 'happy song' and 'pop music' are not: these classes are too abstract

for visual recognition and so they are not included in the dataset. Third, the classes should be *mutually exclusive*. Although we initialize the list using the label hierarchies in existing audio datasets [Gemmeke et al., 2017; Foggia et al., 2015; Fonseca et al., 2017] and other hierarchical on-line sources, our classes are only leaf nodes in these hierarchies. In this manner, the label set in VGG-Sound is flat and contains only one label for each clip. For instance, if the clip contains the sound of a car engine, then the label will be only "car engine"; the more general class "engine" is not included in the list.

The initial list of classes, constructed in this manner, had 600 items. Each class name is used as a search query for YouTube to automatically download corresponding candidate videos. In order to increase the chance of obtaining relevant videos, the class names are further transformed to generate variants of each textual query as follows: (1) forming 'verb+(ing) object' sentences, *e.g.* 'playing electric guitar','ringing church bells', *etc.* (2) submitting the query after translation to different languages, as is done in Carreira et al. [2018], such as English, Spanish and Chinese, etc; (3) adding possible synonym phrase which specify the same sounds, *e.g.* 'steam hissing: water boiling, liquid boiling, *etc*.' In total, over 1m videos were downloaded from YouTube in this manner.

4.3.2 Stage 2: Visual verification.

The purpose of this stage is to verify and localize the visual signature in the downloaded videos. In detail, for each VGG-Sound class, the corresponding visual signature is given by image classifiers. For example, 'playing violin' and 'cat meowing' in VGG-Sound can be matched directly to the OpenImage classifiers [Krasin et al., 2016] 'violin' and 'cat'. These associations are proposed automatically by matching keywords and then verified manually.

However, half of VGG-Sound classes (*e.g.* 'hail', 'playing ukulele') could not be matched directly to OpenImage classifiers in this manner. To tackle this issue, we relaxe the way sound labels are matched to visual labels via semantic word embeddings. Specifically, we convert our 600 sound classes and the 5000 OpenImage classes to word2vec embeddings [Mikolov et al., 2013]. These embedding have 512 dimensions, so this step results in matrices $S \in \mathcal{R}^{600\times512}$ and $O \in \mathcal{R}^{5000\times512}$, respectively for sound and image labels. We

then compute the cosine similarity between the two matrices, resulting in an affinity matrix $A \in \mathcal{R}^{600\times5000} = SO^{\top}$ that represents the strength of the similarity between sound and image classes. The top 20 OpenImage classes for each of the 600 sound classes are then selected as the visual signature of the corresponding sound. For example, 'hail' was matched to 'nature, nature reserve, rain and snow mixed, lightning, thunderstorm, *etc*.' and 'playing electric guitar' to 'electric guitar, guitar, acoustic-electric guitar, musical instrument, *etc*.'.

After determining these associations, the OpenImage pre-trained classifier are run on the downloaded videos, and the 10 frames in the video that receive the highest prediction score are selected provided the score is above an absolute confidence threshold of 0.2. The frames that pass this test are assumed to contain the visual content selected by the classifier. Clips are then created by taking 5 seconds at either side of these representative frames. After this stage, the number of sound classes is reduced from the original 600 to 470, due either an initial scarcity of potential video matches or by failed visual verification.

4.3.3 Stage 3. Audio verification to remove negative clips.

Despite visual verification, our clips are still not guaranteed to contain the desired sound, as an object being visible does not imply that it emits a sound at the same time; in fact, we found that many clips where the correct object was in focus, contained instead generic sounds from humans, such as a narrator describing an image or video, or background music. Since these issues are fairly specific, we address them by finetuning the VGGish model with only three sound classes: speech, music and others. The finetuned classifier is typically reliable as most of the existing datasets offers higly clean data of these classes. We use it to reject clips. For example, using a threshold 0.5, in 'playing bass guitar' videos, we reject any clip for which "speech" is greater than the threshold, but allow music; while for 'dog barking' videos, both speech and music are rejected. After this stage, there are 390 classes left with at least 200 validated video clips. Note that our selection process aims to reject "false positive" *i.e.* inappropriate sounds in each class, we do not attempt to use an audio classifier to select positive clips as that risks losing hard positive audio samples.

4.3.4 Stage 4: Iterative noise filtering.

For this final clean up stage of the process, 20 video clips are randomly sampled from each class and manually checked (both visually and on audio) that they belong to the class. Classes with at least 50% correct are kept and the other classes are discarded. The total set of video clips remaining forms our candidate dataset. Note that, at this stage, the candidate videos can be categorized by audio as one of three types: (i) audios that are clearly of the correct category, *i.e.* easy positives; (ii) audios of the correct category, but with a mixture of sounds, *i.e.* hard positives; or (iii) incorrect audios, *i.e.* false positive.

To further curate the candidate dataset, we make three assumptions: First, there is no systematic bias in the noisy samples, by that we mean, the false positives are not from the same category. Second, Deep CNNs tend to end up with different local minimas and prediction errors, ensembling different models can therefore result in a prediction that is both more stable and often better than the predictions of any individual member model. Third, when objects emit sound, there exists particular visual patterns, *e.g.* a "chimpanzee pant-hooting" will mostly happen with moving bodies.

Exploiting the first two assumptions, the videos of each class are randomly divided into two sets, and an audio classifier is trained on half the candidate videos and used to predict the class of the other half. This process is done twice so that each clip has 2 predictions. To obtain relatively easy and precise postives, we keep the clips whose actual class-label falls into the top-3 of the predictions from the ensembled models. In order to mine the harder positives, we exploit the third assumption by computing visual features for the positive clips, and perform visual retrieval from the rest of data that has been rejected by the audio classifiers. Using a visual classifier can result in similar looking visual clips but disparate hard-positive audio clips. Lastly, we train a new audio classifier (ResNet18) with all easy and hard clips, and retrieve more data from that rejected so far. This increases the number of video clips and forms our final dataset: `VGG-Sound` with 309 classes of over 200k videos, and each class contains 200–1000 audio-visual corresponding clips. Note, we did a deduplication process to remove repeated uploaded clips with different YouTube IDs. This is done by removing the ones with same visual representations.

4.4 Experiments

4.4.1 Experimental setup and Evaluation

Experimental setup. We investigate the audio recognition task on both *AudioSet* and our new VGG-Sound dataset. As the two datasets contain different sound vocabularies, at training time, we use subsets of common classes in both datasets (roughly 400k clips from AudioSet and 120k clips from VGG-Sound). Similarly, at testing time, we select the intersection of *AudioSet* and VGG-Sound to form a single testset called AStest (and remove any videos in AStest that are in the training sets of *AudioSet* or VGG-Sound). This leads 164 classes and 7k clips in AStest. In addition, we investigate how audio recognition performs on VGG and ResNet backbone networks with/without NetVLAD aggregation using our new VGG-Sound datasets.

Finally, we benchmark the audio classification task on the full VGG-Sound dataset, which contains over 200k clips of 309 classes. Details of the train/val/test split are given in Table 4.1.

Evaluation metrics. We adopt the evaluation metrics of Hershey et al. [2017], *i.e.* mean average precision (mAP), AUC, and equivalent d-prime class separation.

4.4.2 Implementation details

During training, we follow Hershey et al. [2017] for data preprocessing for models trained with VGGish models. For models trained on ResNet18, we randomly sample 5s from the 10s audio clip and apply a short-time Fourier transform on the sample, resulting a 257×500 spectrogram. During testing, we directly feed the entire 10s audio (257×1000 spectrogram) into the network.

All experiments were trained using the Adam optimizer with cross entropy loss. The learning rate starts with 10^{-3} and is reduced by a factor of 10 after training plateaus. We use a sigmoid layer when training on *AudioSet* data since each video clip has multiple labels. For models trained on VGG-Sound data, we use a softmax layer in the last layer.

Note, as AudioSet audios contain multiple labels (3 on average), TopK accuracies are not applicable.

4.4.3 Results

	Model	Agg	Pre	Train	Test	mAP	AUC	d-prime	Top1	Top5
A	VGGish	/	✓	AudioSet (c)	ASTest	0.286	0.899	1.803	/	/
B	VGGish	/	✓	VGG-Sound (c)	ASTest	0.326	0.916	1.950	0.331	0.570
C	VGGish	/	×	VGG-Sound (c)	ASTest	0.301	0.910	1.900	0.318	0.560
D	ResNet18	AP	×	VGG-Sound (c)	ASTest	0.328	0.923	2.024	0.354	0.637
E	ResNet18	NV	×	VGG-Sound (c)	ASTest	0.369	0.927	2.058	0.375	0.647
F	ResNet18	AP	×	VGG-Sound	ASTest	0.404	0.944	2.253	0.404	0.679
G	ResNet18	NV	×	VGG-Sound	ASTest	0.434	0.950	2.327	0.421	0.706
H	ResNet18	AP	×	VGG-Sound	VGG-Sound	0.516	0.968	2.627	0.488	0.746
I	ResNet18	NV	×	VGG-Sound	VGG-Sound	0.512	0.970	2.660	0.484	0.741

Table 4.4: We compare the results using various combination of models, training sets and test sets. "Agg" represents Aggregation, "AP" and "NV" refer to Average-Pool and NetVLAD respectively. "Pre" refers to whether the model was pretrained on YouTube-8M dataset, "VGG-Sound (c)" is the subset of the VGG-Sound training dataset that only contains classes in common with AudioSet, "ASTest" is the intersection of the AudioSet and VGG-Sound testsets, "Top1" and "Top5" refers to the top 1 and top 5 accuracy, respectively.

From the experimental results in Table 4.4, we can draw the following conclusions: First, when we adopt a pretrained model from Hershey et al. [2017] and finetune on *AudioSet* (Model-A) and `VGG-Sound` (Model-B), despite *AudioSet* containing more data than `VGG-Sound`, model-B still outperforms model-A on all metrics, we conjecture that this is because the noise ratio of `VGG-Sound` is lower than that of *AudioSet*, as we wished in our initial design objective. Second, training model-C (a VGGish model from scratch) on `VGG-Sound`, gets slightly worse performance on all metrics than model-B, which shows that pretraining on a large dataset helps boost the model's performance. Third, when comparing model-D (trained with average pooling) and model-E (trained with NetVLAD), we demonstrate the effectiveness of NetVLAD aggregation over the naïve global average pooling also beats the pretrained VGGish model (model-B). Finally, we train on the the full training set of `VGG-Sound` and test on ASTest or the full `VGG-Sound` testing set. In addition, both average pooling and NetVLAD aggregation are evluated (model-F∼I), the best result achieves an mAP of 0.516 (model-H). Note, testing on full `VGG-Sound` test

set (model-I) shows a better result comparing to the one testing only on ASTest (model-G). This is because ASTest contains audioset testing clips which tend to have multiple sounds. These are hard examples for model to predict, this can also be seen from the Top5 accuracy as the gap between (model-I) and (model-G) is largely reduced.

4.5 Conclusion

In this paper, we propose an automated pipeline for collecting a large-scale audio-visual dataset – VGG-Sound, which contains more than 200k videos and 309 classes for "unconstrained" conditions. We also compare CNN architectures and aggregation methods to provide baseline results for audio recognition on VGG-Sound.

Acknowledgement.

Financial support was provided by the EPSRC Programme Grant Seebibyte EP/M013774/1.

Chapter 5

Localizing Visual Sounds the Hard Way

Honglie Chen Weidi Xie Triantafyllos Afouras Arsha Nagrani
Andrea Vedaldi Andrew Zisserman

Visual Geometry Group, University of Oxford

Abstract

The objective of this work is to localize sound sources that are visible in a video without using manual annotations. Our key technical contribution is to show that, by training the network to explicitly discriminate challenging image fragments, even for images that do contain the object emitting the sound, we can significantly boost the localization performance. We do so elegantly by introducing a mechanism to mine hard samples and add them to a contrastive learning formulation automatically. We show that our algorithm achieves state-of-the-art performance on the popular Flickr SoundNet dataset. Furthermore, we introduce the VGG-Sound Source (VGG-SS) benchmark, a new set of annotations for the recently-introduced VGG-Sound dataset, where the sound sources visible in each video clip are explicitly marked with bounding box annotations. This dataset is 20 times larger than analogous existing ones, contains 5K videos spanning over 200 categories, and, differently from Flickr SoundNet, is video-based. On VGG-SS, we also show that our algorithm achieves state-of-the-art performance against several baselines.

Published in the proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR), 2021.

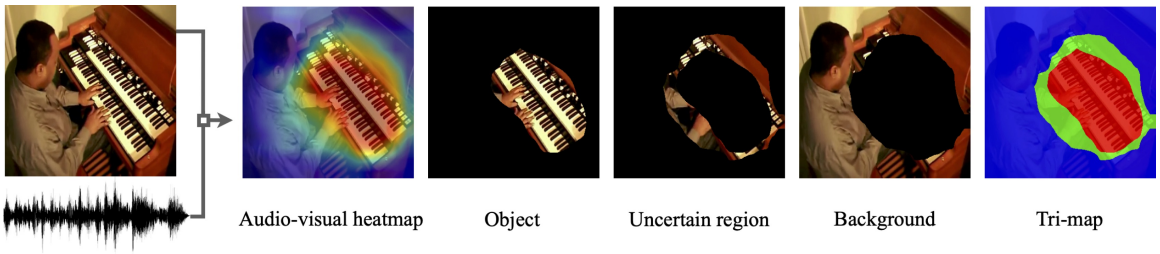


Figure 5.1: **Visual Sound Source localization:** We localize sound sources in videos without manual annotation. Our key contribution is an automatic negative mining technique through differentiable thresholding of a cross-modal correspondence score map, the background regions with low correlation to the given sound as 'hard negatives', and the regions in the Tri-map is'ignored' in a contrastive learning framework.

5.1 Introduction

While research in computer vision largely focuses on the visual aspects of perception, natural objects are characterized by much more than just appearance. Most objects, in particular, emit sounds, either in their own right, or in their interaction with the environment — think of the bark of a dog, or the characteristic sound of a hammer striking a nail. A full understanding of natural objects should not ignore their acoustic characteristics. Instead, modelling appearance and acoustics jointly can often help us understand them better and more efficiently. For example, several authors have shown that it is possible to use sound to discover and localize objects automatically in videos, without the use of any manual supervision [Arandjelovic and Zisserman, 2018; Senocak et al., 2018; Harwath et al., 2018; Owens and Efros, 2018; Hu et al., 2019; Afouras et al., 2020b].

In this paper, we consider the problem of localizing 'visual sounds', *i.e.* visual objects that emit characteristics sounds in videos. Inspired by prior works [Arandjelovic and Zisserman, 2018; Senocak et al., 2018; Harwath et al., 2018], we formulate this as finding the correlation between the visual and audio streams in videos. These papers have shown that not only can this correlation be learned successfully, but that, once this is done, the resulting convolutional neural networks can be 'dissected' to localize the sound source spatially, thus imputing it to a specific object. However, other than in the design of the architecture itself, there is little in this prior work meant to improve the localization capabilities of the resulting models. In particular, while several models [Arandjelovic and Zisserman,

2018; Afouras et al., 2020b; Senocak et al., 2018] do incorporate a form of spatial attention which should also help to localize the sounding object as a byproduct, these may still fail to provide a good *coverage* of the object, often detecting too little or too much of it.

In order to address this issue, we propose a new training scheme that explicitly seeks to spatially localize sounds in video frames. Similar to object detection [Viola and Jones, 2001], in most cases only a small region in the image contains an object of interest, in our case a 'sounding' object, with the majority of the image often being 'background' which is not linked to the sound. Learning accurate object detectors involves explicitly seeking for these background regions, prioritizing those that could be easily confused for the object of interest, also called *hard negatives* [Viola and Jones, 2001; Dalal and Triggs, 2005; Girshick et al., 2014; Shrivastava et al., 2016; Ren et al., 2016; Lin et al., 2017]. Given that we lack supervision for the location of the object making the sound, however, we are unable to tell which boxes are positive or negative. Furthermore, since we seek to solve the localization rather than the detection problem, we do not even have bounding boxes to work with, as we seek instead a segmentation of the relevant image area.

In order to incorporate hard evidence in our unsupervised (or self-supervised) setting, we propose an automatic background mining technique through differentiable thresholding, *i.e.*regions with low correlation to the given sound are incorporated into a negatives set for contrastive learning. Instead of using hard boundaries, we note that some regions may be uncertain, and hence we introduce the concept of a Tri-map into the training procedure, leaving an 'ignore' zone for our model. To our knowledge, this is the first time that background regions have been explicitly considered when solving the sound source localization problem. We show that this simple change significantly boosts sound localization performance on standard benchmarks, such as Flickr SoundNet [Senocak et al., 2018].

To further assess sound localization algorithms, we also introduce a new benchmark, based on the recently-introduced VGG-Sound dataset [Chen et al., 2020a], where we provide high-quality bounding box annotations for 'sounding' objects, *i.e.* objects that produce a sound, for more than 5K videos spanning 200 different categories. This dataset is $20\times$

larger and more diverse than existing sound localization benchmarks, such as Flickr SoundNet (the latter is also based on still images rather than videos). We believe this new benchmark, which we call VGG-Sound Source, or VGG-SS for short, will be useful for further research in this area. In the experiments, we establish several baselines on this dataset, and further demonstrate the benefits of our new algorithm.

5.2 Related Work

5.2.1 Audio-Visual Sound Source Localization

Learning to localize sound sources by exploiting the natural co-occurrence of visual and audio cues in videos has a long history. Early attempts to solve the task used shallow probabilistic models [Hershey and Movellan, 1999; Fisher III et al., 2000; Kidron et al., 2005], or proposed segmenting videos into spatio-temporal tubes and associating those to the audio signal through canonical correlation analysis (CCA) [Izadinia et al., 2012].

Modern approaches solve the problem using deep neural networks — typically employing a dual stream, trained with a contrastive loss by exploiting the audio-visual correspondence, *i.e.*matching audio and visual representations extracted from the same video. For example, Arandjelovic and Zisserman [2018]; Senocak et al. [2018]; Harwath et al. [2018]; Ramaswamy and Das [2020] associate the appearance of objects with their characteristic sounds or audio narrations; Hu *et al.* [Hu et al., 2019] first cluster audio and visual representations within each modality, followed by associating the resulting centroids with contrastive learning; Qian *et al.* [Qian et al., 2020] proposed a weakly supervised approach, where the approximate locations of the objects are obtained from CAMs to bootstrap the model training. Apart from using correspondence, Owens and Efros [Owens and Efros, 2018] also localize sound sources through synchronization, a related objective also investigated in earlier works [Marcheret et al., 2015; Chung and Zisserman, 2016a], while [Khosravan et al., 2019] incorporate explicit attention in this model. Afouras *et al.* [Afouras et al., 2020b] also exploit audio-visual concurrency to train a video model that can distinguish and group instances of the same category.

Alternative approaches solve the task using an audio-visual source separation objective. For example, Zhao *et al.* [Zhao et al., 2018a] employ a mix-and-separate approach to learn to associate pixels in video frames with separated audio sources, while Zhao *et al.* [Zhao et al., 2019] extends this method by providing the model with motion information through optical flow. Rouditchenko *et al.* [Rouditchenko et al., 2019] train a two-stream model to co-segment video and audio, producing heatmaps that roughly highlight the object according to the audio semantics. These methods rely on the availability of videos containing single-sound sources, usually found in well curated datasets. In other related work, Gan *et al.* [Gan et al., 2019] learn to detect cars from stereo sound, by distilling video object detectors, while Gao *et al.* [Gao and Grauman, 2019] lift mono sound to stereo by leveraging spatial information.

5.2.2 Audio-Visual Localization Benchmarks

Existing audio-visual localization benchmarks are summarised in Table 5.1 (focusing on the test sets). The Flickr SoundNet sound source localization benchmark [Senocak et al., 2018] is an annotated collection of single frames randomly sampled from videos of the Flickr SoundNet dataset [Aytar et al., 2016; Thomee et al., 2016]. It is currently the standard benchmark for the sound source localization task; we discuss its limitations in Section 5.4, where we introduce our new benchmark. The Audio-Visual Event (AVE) dataset [Tian et al., 2018], contains 4,143 10 second video clips spanning 28 audio-visual event categories with temporal boundary annotations. LLP [Tian et al., 2020] contains of 11,849 YouTube video clips spanning 25 categories for a total of 32.9 hours collected from AudioSet [Gemmeke et al., 2017]. The development set is sparsely annotated with object labels, while the test set contains dense video and audio sound event labels on the frame level. Note that the AVE and LLP test sets contain only temporal localization of sounds (at the frame level), with no spatial bounding box annotation.

Benchmark Datasets	# Data	# Classes	Video	BBox
Flickr SoundNet [Senocak et al., 2018]	250	∼50‡	×	✓
AVE [Tian et al., 2018]†	402	28	✓	×
LLP [Tian et al., 2020]†	1,200	25	✓	×
VGG-SS	5,158	220	✓	✓

Table 5.1: Comparison with the existing sound-source localization benchmrks. Note that VGG-SS has more images and classes. †These datasets contain only temporal localization of sounds, not spatial localization. ‡ We determined this via manual inspection.

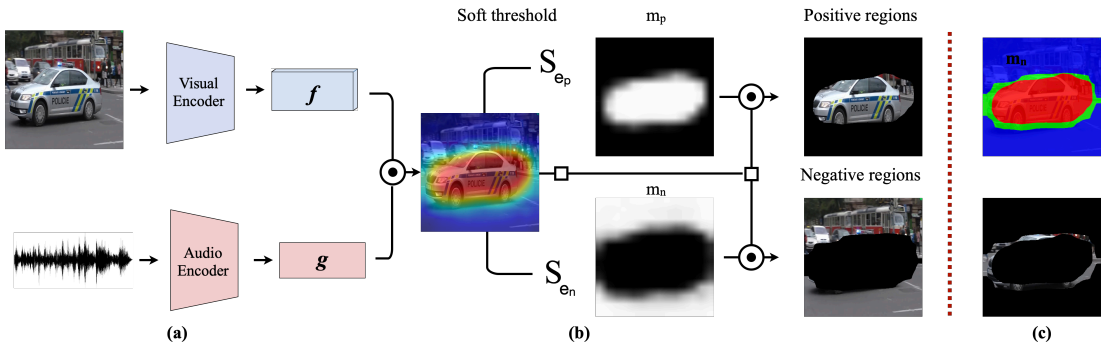


Figure 5.2: **Architecture Overview**. We use an audio-visual pair as input to a dual-stream network shown in **(a)**, $f(\cdot;\theta_1)$ and $g(\cdot;\theta_2)$, denoting the visual and audio feature extractor respectively. Cosine similarity between the audio vector and visual feature map is then computed, giving us a heatmap of size 14×14. **(b)** demonstrates the soft threshold being applied twice with different parameters, generating positive, negative regions. The final Tri-map and the uncertain region are highlighted in **(c)**.

5.3 Method

Our goal is to localize objects that make characteristic sounds in videos, without using any manual annotation. Similar to prior work [Arandjelovic and Zisserman, 2018], we use a two-stream network to extract visual and audio representations from unlabeled video. For localization, we compute the cosine similarity between the audio representation and the visual representations extracted convolutionally at different spatial locations in the images. In this manner, we obtain a positive signal that pulls together sounds and relevant spatial locations. For learning, we also need an opposite negative signal. A weak one is obtained by correlating the sound to locations in other, likely irrelevant videos. Compared to prior work [Arandjelovic and Zisserman, 2018; Afouras et al., 2020b], our key contribution is to

also explicitly seek for hard negative locations that contain background or non-sounding objects in the *same* images that contain the sounding ones, leading to more selective and thus precise localization. An overview of our architecture can be found in Figure 5.2.

While the idea of using hard negatives is intuitive, an effective implementation is less trivial. In fact, while we seek for hard negatives, there is no hard evidence for whether any region is in fact positive (sounding) or negative (non-sounding) as videos are unlabeled. An incorrect classification of a region as positive or negative can throw off the localization algorithm entirely. We solve this problem by using a robust contrastive framework that combines soft thresholding and Tri-maps, which enables us to handle uncertain regions effectively.

In sections 5.3.1 to 5.3.3 we first describe the task of audio-visual localization using contrastive learning in its *oracle* setting, assuming, for each visual-audio pair, we do have the ground-truth annotation for which region in the image is emitting the sound. In section 5.3.4, we introduce our proposed idea, which replaces the *oracle*, and discuss the difference between our method and existing approaches.

5.3.1 Audio-Visual Feature Representation

Given a short video clip with N visual frames and audio, and considering the center frame as visual input, *i.e.* $X = \{I, a\}, I \in \mathbb{R}^{3\times H_v \times W_v}, a \in \mathbb{R}^{1\times H_a \times W_a}$. Here, I refers to the visual frame, and a to the spectrogram of the raw audio waveform. In this manner, representations for both modalities can be computed by means of CNNs, which we denote respectively $f(\cdot;\theta_1)$ and $g(\cdot;\theta_2)$. For each video X_i, we obtain visual and audio representations:

$$V_i = f(I_i;\theta_1), \quad V_i \in \mathbb{R}^{c\times h\times w}, \tag{5.1}$$

$$A_i = g(a_i;\theta_2), \quad A_i \in \mathbb{R}^{c}. \tag{5.2}$$

Note that both visual and audio representation have the same number of channels c, which allows to compare them by using dot product or cosine similarity. However, the video representation also has a spatial extent $h \times w$, which is essential for spatial localization.

5.3.2 Audio-Visual Correspondence

Given the video and audio representations of eqs. (5.1) and (5.2), we put in correspondence the audio of clip i with the image of clip j by computing the cosine similarity of the representations, using the audio as a probe vector:

$$[S_{i\to j}]_{uv} = \frac{\langle A_i, [V_j]_{:uv}\rangle}{\|A_i\|\,\|[V_i]_{:uv}\|}, \quad uv \in [h]\times[w].$$

This results in a map $S_{i\to j} \in \mathbb{R}^{h\times w}$ indicating how strongly each image location in clip j responds to the audio in clip i. To compute the cosine similarity, the visual and audio features are L^2 normalized. Note that we are often interested in correlating images and audio from the same clip, which is captured by setting $j = i$.

5.3.3 Audio-Visual Localization with an Oracle

In the literature, training models for audio-visual localization has been treated as learning the correspondence between these two signals, and formulated as contrastive learning [Senocak et al., 2018; Arandjelovic and Zisserman, 2018; Hu et al., 2019; Afouras et al., 2020b; Qian et al., 2020].

Here, before diving into the self-supervised approach, we first consider the *oracle* setting for the contrastive learning where ground-truth annotations are available. This means that we are given a training set $\mathcal{D} = \{d_1, d_2, \ldots, d_k\}$, where each training sample $d_i = (X_i, m_i)$ consists of a audio-visual sample X_i, as given above, plus a segmentation mask $m_i \in \mathbb{B}^{h\times w}$ with ones for those spatial locations that overlap with the object that emits the sounds, and zeros elsewhere. During training, the goal is therefore to jointly optimize $f(\cdot;\theta_1)$ and $g(\cdot;\theta_2)$, such that $S_{i\to i}$ gives high responses only for the region that emits the sound present in the audio. In this paper, we consider a specific type of contrastive learning, namely InfoNCE [Oord et al., 2018; Han et al., 2019].

Optimization. For each clip i in the dataset (or batch), we define the positive and negative responses as:

$$P_i = \frac{1}{|m_i|}\langle m_i,\ S_{i\to i}\rangle,$$
$$N_i = \underbrace{\frac{1}{|\mathbf{1}-m_i|}\langle \mathbf{1}-m_i,\ S_{i\to i}\rangle}_{\text{hard negatives}} + \underbrace{\frac{1}{hw}\sum_{i\neq j}\langle \mathbf{1},\ S_{i\to j}\rangle}_{\text{easy negatives}}.$$

where $\langle\cdot,\cdot\rangle$ denotes Frobenius inner product. To interpret this equation, note that the inner product simply sums over the element-wise product of the specified tensors and that $\mathbf{1}$ denotes a $h \times w$ tensor of all ones. The first term in the expression for N_i refers to the *hard negatives*, calculated from the "background" (regions that do not emit the characteristic sound) within the same image, and the second term denotes the easy negatives, coming from other images in the dataset. The optimization objective can therefore be defined as:

$$\mathcal{L} = -\frac{1}{k}\sum_{i=1}^{k}\left[\log\frac{\exp(P_i)}{\exp(P_i)+\exp(N_i)}\right]$$

Discussion. Several existing approaches [Senocak et al., 2018; Harwath et al., 2018; Arandjelovic and Zisserman, 2018; Afouras et al., 2020b] to self-supervised audio-visual localization are similar. The key difference lies in the way of constructing the positive and negative sets. For example, in [Senocak et al., 2018] a heatmap generated by using the soft-max operator is used to pool the positives and images from other video clips are treated as negatives; instead, in [Arandjelovic and Zisserman, 2018], positives come from max pooling the correspondence map, $S_{i\to i}$ and the negatives from max pooling $S_{i\to j}$ for $j \neq i$. Crucially, all such approaches have missed the *hard negatives* term defined above, computed from the background regions within the same images that do contain the sound. Intuitively this term is important to obtain a shaper visual localization of the sound source; however, while this is easy to implement in the oracle setting, obtaining hard negatives in self-supervised training requires some care, as discussed next.

5.3.4 Self-supervised Audio-Visual Localization

In this section, we describe a simple approach for replacing the oracle, and continuously bootstrapping the model to achieve better localization results. At a high level, the proposed idea inherits the spirit of self-training, where predictions are treated as pseudo-ground-truth for re-training.

Specifically, given a dataset $\mathcal{D} = \{X_1, X_2, \ldots, X_k\}$ where only audio-visual pairs are available (but not the masks m_i), the correspondence map $S_{i \to i}$ between audio and visual input can be computed in the same manner as section 5.3.2. To get the pseudo-ground-truth mask $\hat{m}_i$, we could simply threshold the map $S_{i \to i}$:

$$\hat{m}_i = \begin{cases} 1, & \text{if } S_{i \to i} \geq \epsilon \\ 0, & \text{otherwise} \end{cases}$$

Clearly, however, this thresholding, which uses the Heaviside function, is not differentiable. Next, we address this issue by relaxing the thresholding operator.

Smoothing the Heaviside function. Here, we adopt a smoothed thresholding operator in order to maintain the end-to-end differentiability of the architecture:

$$\hat{m}_i = \text{sigmoid}((S_{i \to i} - \epsilon)/\tau)$$

where ϵ refers to the thresholding parameter, and τ denotes the temperature controlling the sharpness.

Handling uncertain regions. Unlike the oracle setting, the pseudo-ground-truth obtained from the model prediction may potentially be noisy, we therefore propose to set up an "ignore" zone between the positive and negative regions, allowing the model to self-tune. In the image segmentation literature, this is often called a Tri-map and is also used for matting [Chuang et al., 2002; Tao et al., 2018]. Conveniently, this can be implemented by applying two different ϵ's, one controlling the threshold for the positive part and the other for the negative part of the Tri-map.

Training objective. We are now able to replace the oracle while computing the positives and negatives automatically. This leads to our final formulation:

$$\hat{m}_{ip} = \text{sigmoid}((S_{i\to i} - \epsilon_p)/\tau)$$
$$\hat{m}_{in} = \text{sigmoid}((S_{i\to i} - \epsilon_n)/\tau)$$
$$P_i = \frac{1}{|\hat{m}_{ip}|}\langle \hat{m}_{ip},\ S_{i\to i}\rangle$$
$$N_i = \frac{1}{|\mathbf{1} - \hat{m}_{in}|}\langle \mathbf{1} - \hat{m}_{in},\ S_{i\to i}\rangle + \frac{1}{hw}\sum_{j\neq i}\langle \mathbf{1},\ S_{i\to j}\rangle$$
$$\mathcal{L} = -\frac{1}{k}\sum_{i=1}^{k}\left[\log\frac{\exp(P_i)}{\exp(P_i)+\exp(N_i)}\right]$$

where ϵ_p and ϵ_n are two thresholding parameters (validated in experiment section), with $\epsilon_p > \epsilon_n$. For example if we set $\epsilon_p = 0.6$ and $\epsilon_n = 0.4$, regions with correspondence scores above 0.6 are considered positive and bellow 0.4 negative, while the areas falling within the $[0.4, 0.6]$ range are treated as "uncertain" regions and ignored during training (Figure 5.2).

5.4 The VGG-Sound Source Benchmark

As mentioned in Section 5.2, the SoundNet-Flickr sound source localization benchmark [Senocak et al., 2018] is commonly used for evaluation in this task. However, we found it to be unsatisfactory in the following aspects: i) both the number of total instances (250) and sounding object categories (approximately 50) that it contains are limited, ii) only certain reference frames are provided, instead of the whole video clip, which renders it unsuitable for the evaluation of video models, and iii) it provides no object category annotations.

In order to address these shortcomings, we build on the recent VGG-Sound dataset [Chen et al., 2020a] and introduce VGG-SS, an audio-visual localization benchmark based on videos collected from YouTube.

5.4.1 Test Set Annotation Pipeline

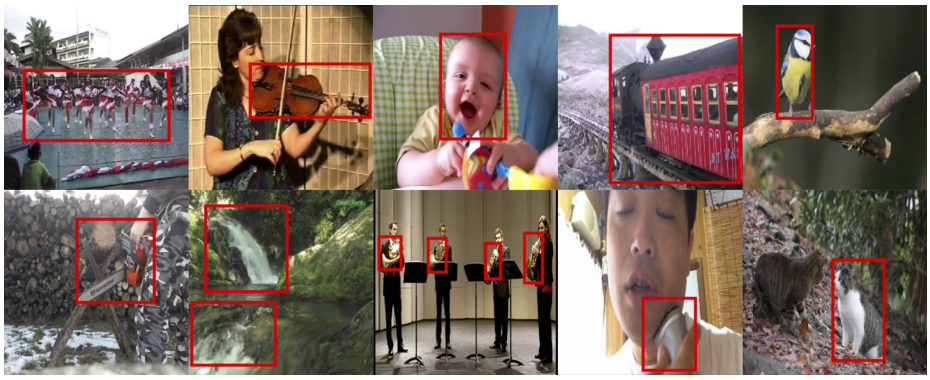
In the following sections, we describe a semi-automatic procedure to annotate the objects that emit sounds with bounding boxes, which we apply to obtain VGG-SS with over 5k video clips, spanning 220 classes.

(1) Automatic bbox generation. We use the entire VGG-Sound test set, containing 15k 10-second video clips, and extract the center frame from each clip. We use a Faster R-CNN object detector [Ren et al., 2016] pretrained on OpenImages to predict the bounding boxes of all relevant objects. Following Chen et al. [2020a], we use a word2vec model to match visual and audio categories that are semantically similar. At this stage, there are roughly 8k frames annotated automatically.

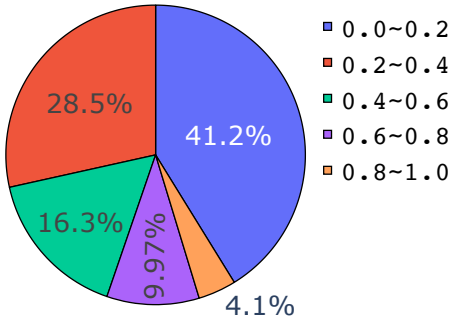
(2) Manual image annotation. We then annotate the remaining frames manually. There are three main challenges at this point: (i) there are cases where localization is extremely difficult or impossible, either because the object is not visible (e.g. in extreme lighting conditions), too small ('mosquito buzzing'), or is diffused throughout the frame ('hail', 'sea waves', 'wind'); (ii) the sound may originate either from a single object, or from the interactions between multiple objects and a consistent annotation scheme must be decided upon; and finally (iii), there could be multiple instances of the same class in the same frame, and it is challenging to know which of the instances are making the sound from a single image.

We address these issues in three ways: First, we remove categories (*e.g.* mainly environmental sounds such as wind, hail etc) that are challenging to localize, roughly 50 classes; Second, as illustrated in Figure 5.3(a), when the sound comes from the interaction of multiple objects, we annotate a tight region surrounding the interaction point; Third, if there are multiple instances of the same sounding object category in the frame, we annotate each separately when there are less than 5 instances and they are separable, otherwise a single bounding box is drawn over the entire region, as shown in the top left image ('human crowd') in Figure 5.3(a).

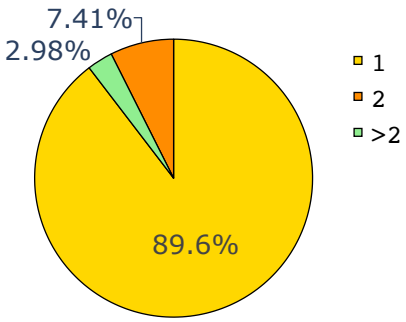
(3) Manual video verification. Finally, we conduct manual verification on videos using the VIA software [Dutta and Zisserman, 2019]. We do this by watching the 5-second video around every annotated frame, to ensure that the sound corresponds with the object in the



(a) VGG-SS benchmark examples



(b) Bounding box areas



(c) Number of bounding boxes

Figure 5.3: **VGG-SS Statistics.** Figure 5.3(a): Example VGG-SS images and annotations showing class diversity (humans, animals, vehicles, tools etc.) Figure 5.3(b): Distribution of bounding box areas in VGG-SS, the majority of boxes cover less than 40% of the image area. Figure 5.3(c) shows the distribution of number of bounding boxes - roughly 10% of the test data is challenging with more than one bounding box per image.

bounding box. This is particularly important for the cases where there are multiple candidate instances present in the frame, however, only one is making the sound, *e.g.* human singing.

The statistics after every stage of the process and the final dataset are summarised in Table 5.2. The first stage generates bounding box candidates for the entire VGG-Sound test set (309 classes, 15k frames); the manual annotation process then removes unclear classes and frames, resulting in roughly 260 classes and 8k frames. Our final video verification further cleans up the the test set, yielding a high-quality large-scale audio-visual benchmark — VGG-Sound Source (VGG-SS), which is 20 times larger than the existing one [Senocak et al., 2018].

Stage	Goal	# Classes	# Videos
1	Automatic BBox Generation	309	15k
2	Manual Annotation	260	8k
3	Video Verification	220	5k

Table 5.2: The number of classes and videos in VGG-SS after each annotation stage.

5.5 Experiments

In the following sections, we describe the datasets, evaluation protocol and experimental details used to thoroughly assess our method.

5.5.1 Training Data

For training our models, we consider two large-scale audio-visual datasets, the widely used Flickr SoundNet dataset and the recent VGG-Sound dataset, as detailed next. Only the center frames of the *raw* videos are used for training. Note, other frames *e.g.*(3/4 of the video) are tried for training, no considerable performance change is observed.

Flickr SoundNet: This dataset was initially proposed in Aytar et al. [2016] and contains over 2 million unconstrained videos from Flickr. For a fair comparison with recent work [Senocak et al., 2018; Hu et al., 2019; Qian et al., 2020], we follow the same data splits, conducting self-supervised training with subsets of 10k or 144k image and audio pairs.

VGG-Sound: VGG-Sound was recently released with over 200k clips for 309 different sound categories. The dataset is conveniently audio-visual, in the sense that the object that emits sound is often visible in the corresponding video clip, which naturally suits the task considered in this paper. Again, to draw fair comparisons, we conduct experiments with

training sets consisting of image and audio pairs of varying sizes, *i.e.* 10k, 144k and the full set.

5.5.2 Evaluation protocol

In order to quantitatively evaluate the proposed approach, we adopt the evaluation metrics used in Senocak et al. [2018]; Qian et al. [2020]: Consensus Intersection over Union (cIoU) and Area Under Curve (AUC) are reported for each model on two test sets, as detailed next.

Flickr SoundNet Testset: Following [Senocak et al., 2018; Hu et al., 2019; Qian et al., 2020], we report performance on the 250 annotated image-audio pairs of the Flickr SoundNet benchmark. Every frame in this test set is accompanied by 20 seconds of audio, centered around it, and is annotated with 3 separate bounding boxes indicating the location of the sound source, each performed by a different annotator.

VGG-Sound Source (VGG-SS): We also re-implement and train several baselines on VGG-Sound and evaluate them on our proposed VGG-SS benchmark, described in section 5.4.

5.5.3 Implementation details

As Flickr SoundNet consists of image-audio pairs, while VGG-Sound contains short video clips, when training on the latter we select the middle frame of the video clip and extract a 3s audio segment around it to create an equivalent image-audio pair. Audio inputs are 257×300 magnitude spectrograms. The dimensions for the audio output from the audio encoder CNN is a 512D vector, which is max-pooled from a feature map of $17 \times 13 \times 512$, where 17 and 13 refer to the frequency and time dimension respectively. For the visual input, we resize the image to a $224 \times 224 \times 3$ tensor without cropping. For both the visual and audio stream, we use a lightweight ResNet18 [He et al., 2016] as a backbone.

Following the baselines [Hu et al., 2019; Qian et al., 2020], we also pretrain the visual encoder on ImageNet. We use $\epsilon_p = 0.65$ and $\epsilon_n = 0.4$, $\tau = 0.03$, that are picked by ablation study. All models are trained with the Adam optimizer using a learning rate of 10^{-4} and a batch size of 256. During testing, we directly feed the full length audio spectrogram into the network.

5.6 Results

Method	Training set	CIoU	AUC
Attention10k [Senocak et al., 2018]	Flickr10k	0.436	0.449
CoarsetoFine [Qian et al., 2020]	Flickr10k	0.522	0.496
AVObject [Afouras et al., 2020b]	Flickr10k	0.546	0.504
Ours	Flickr10k	**0.582**	**0.525**
Ours	VGG-Sound10k	**0.618**	**0.536**
Attention10k [Senocak et al., 2018]	Flickr144k	0.660	0.558
DMC [Hu et al., 2019]	Flickr144k	0.671	0.568
Ours	Flickr144k	**0.699**	**0.573**
Ours	VGG-Sound144k	**0.719**	**0.582**
Ours	VGG-Sound Full	**0.735**	**0.590**

Table 5.3: Quantitative results on Flickr SoundNet testset. We outperform all recent works using different training sets and number of training data.

Model	Pos ϵ	Neg ϵ	Tri-map	CIoU	AUC
a	✓ (0.6)	×	×	0.675	0.568
b	✓ (0.6)	✓ (0.6)	×	0.667	0.544
c	✓ (0.6)	✓ (0.45)	✓	0.700	0.568
d	✓ (0.65)	✓ (0.45)	✓	0.703	0.569
e	✓ (0.65)	✓ (0.4)	✓	**0.719**	**0.582**
f	✓ (0.7)	✓ (0.3)	✓	0.687	0.563

Table 5.4: Ablation study. We investigate the effects of the hyper-parameters for defining positive and negative regions, where the picked value is specified in the bracket.

In the following sections, we first compare our results with recent work on both Flickr SoundNet and VGG-SS dataset in detail. Then we conduct an ablation analysis showing

the importance of the *hard negatives* and the Tri-map in self-supervised audio-visual localization.

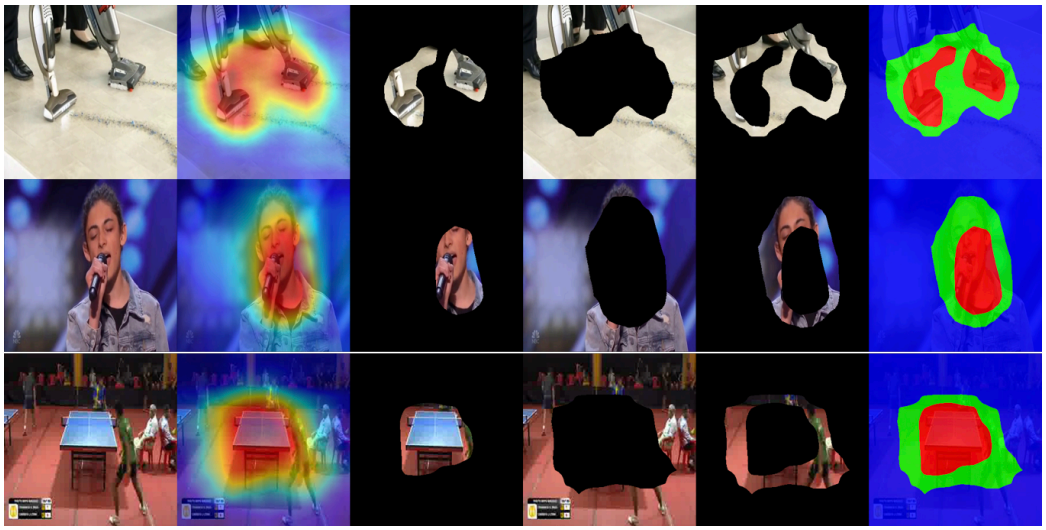
5.6.1 Comparison on the Flickr SoundNet Test Set

In this section, we compare to recent approaches by training on the same amount of data (using various different datasets). As shown in Table 5.3, we first fix the training set to be Flickr SoundNet with 10k training samples and compare our method with Arandjelovic and Zisserman [2018]; Qian et al. [2020]; Harwath et al. [2018]. Our approach clearly outperforms the best previous methods by a substantial gap (0.546% vs. 0.582%). Second, we also train on VGG-Sound using 10k random samples, which shows the benefit of using VGG-Sound for training. Third, we switch to a larger training set consisting of 144k samples, which gives us a further 5% improvement compared to the previous state-of-the-art method [Hu et al., 2019]. In order to tease apart the effect of various factors in our proposed approach, *i.e.* introducing *hard negative* and using a Tri-map vs different training sets, *i.e.* Flickr144k vs. VGG-Sound144k, we conduct an ablation study, as described next.

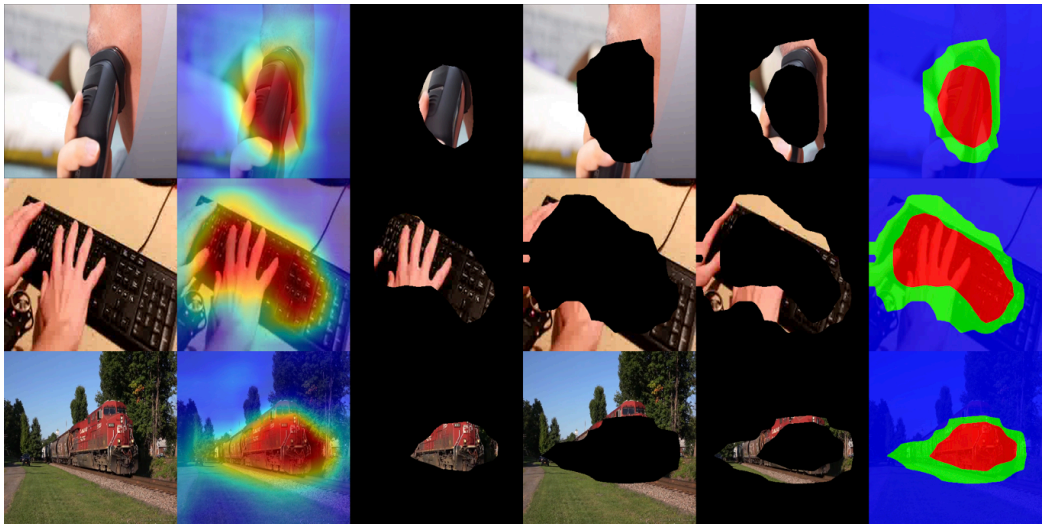
5.6.2 Ablation Analysis

In this section, we train our method using the 144k-samples training data from VGG-Sound and evaluate it on the Flickr SoundNet test set, as shown in Table 5.4.

On introducing hard negative and Tri-map. While comparing **model a** trained using only positives and **model b** adding negatives from the complementary region decreases performance slightly. This is because all the non-positive areas have been counted as negatives, whereas regions around the object are often hard to define. Therefore deciding for all pixels whether they are positive or negative is problematic. Second, comparing **model b** and **model c-f** where some areas between positives and negatives are ignored during training by using the Tri-map, we obtain a large gain (around 2-4%), demonstrating the importance of defining an "uncertain" region and allowing the model to self-tune.



(a)



(b)

Figure 5.4: **Example Tri-map visualizations.** We show images, heatmaps and Tri-maps here. The Tri-map effectively identify the objects and the uncertain region let the model only learn controlled hard negatives.

On hyperparameters. we observe the model is generally robust to different set of hyper-parameters on defining the positive and negative regions, **model-e** ($\epsilon_p = 0.65$ and $\epsilon_n = 0.4$) strives the best balance.

Method	CIoU	AUC
Attention10k [Senocak et al., 2018]	0.185	0.302
AVobject [Afouras et al., 2020b]	0.297	0.357
Ours	**0.344**	**0.382**

Table 5.5: Quantitative results on the VGG-SS testset. All models are trained on VGG-Sound 144k and tested on VGG-SS.

5.6.3 Comparison on VGG-Sound Source

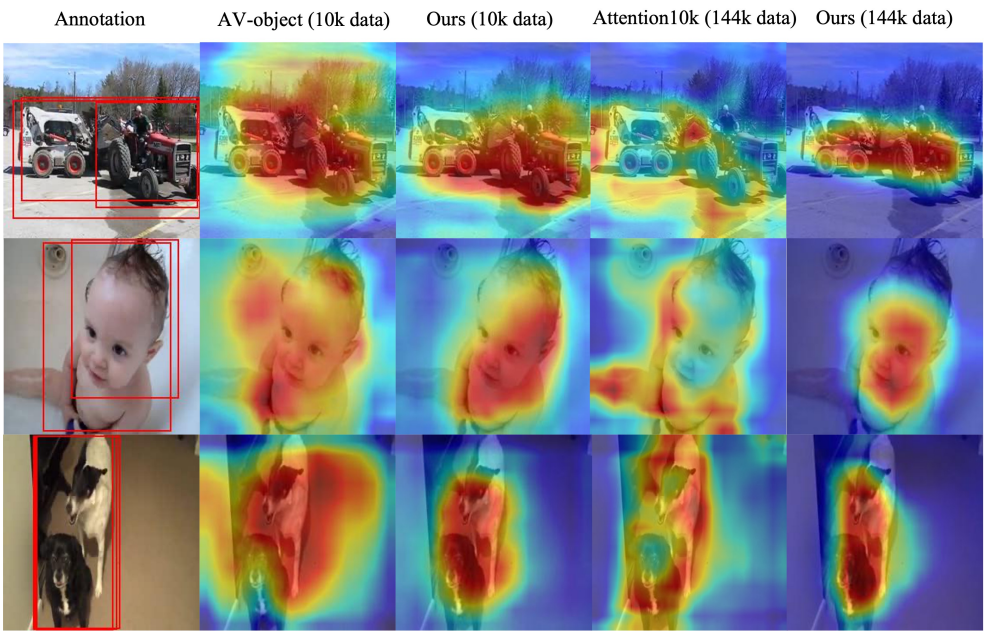
In this section, we evaluate the models on the newly proposed VGG-SS benchmark. As shown in Table 5.5, the CIoU is reduced significantly for all models compared to the results in Table 5.3, showing that VGG-SS is a more diverse and challenging benchmark than Flickr SoundNet. However, our proposed method still outperforms all other baseline methods by a large margin of around 5%.

5.6.4 Qualitative results

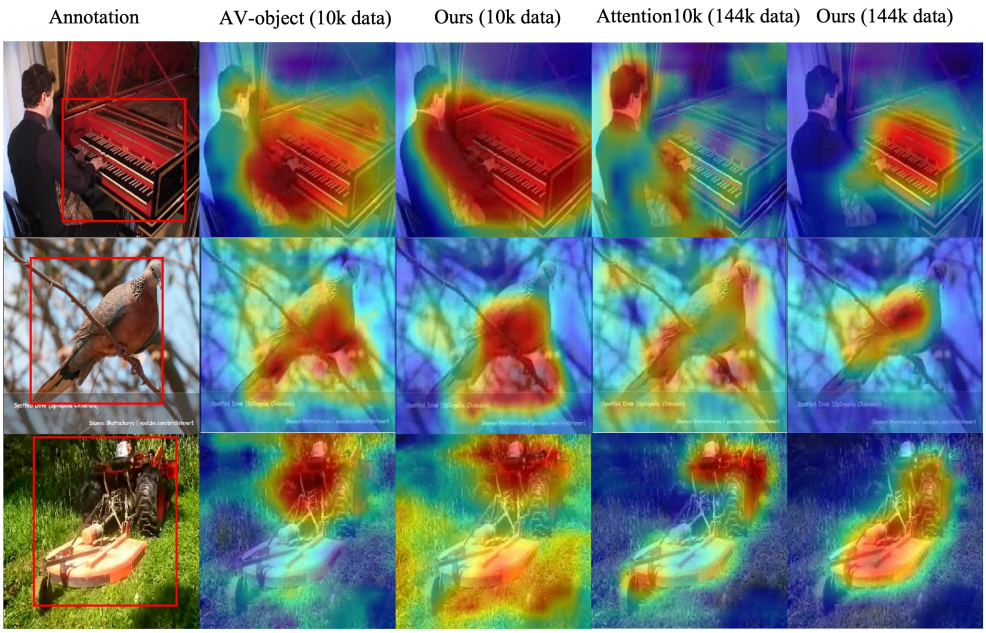
In Figure 5.4, we threshold the heatmaps with different thresholds, *e.g.* $\epsilon_p = 0.65$ and $\epsilon_n = 0.4$ (same as the ones used during training). The objects and background are accurately highlighted in the positive region and negative region respectively, so that the model can learn proper amount of hard negatives. We visualize the prediction results in Figure 5.5, and note that the proposed method presents much cleaner heatmap outputs. This once again indicates the benefits of considering hard negatives during training.

5.6.5 Open Set Audio-visual Localization

We have so far trained and tested our models on data containing the same sound categories (closed set classification). In this section we determine if our model trained on heard/seen categories can generalize to classes that have never been heard/seen before, *i.e.*to an open



(a) Visualization on Flickr SoundNet testset



(b) Visualization on VGG-SS testset

Figure 5.5: **Qualitative results** for models trained on various methods and data amount. The first column shows annotation overlaid on images, the following two column shows predictions trained on 10k data and the last tow column show predictions trained on 144k data. Our method has no false positives in the predictions as the hard negatives are penalised in the training.

set scenario. To test this, we randomly sample 110 categories (seen/heard) from VGG-Sound for training, and evaluate our network on another *disjoint* set of 110 unseen/unheard categories (for a full list please refer to appendix). We use roughly 70k samples for both heard and unheard classes.

# training Data	Test class	CIoU	AUC
70k	Heard 110	0.289	0.362
70k	Unheard 110	0.263	0.347

Table 5.6: Quantitative results on VGG-SS for unheard classes. We vary the training set (classes) and keep the testing set fixed (subset of the VGG-SS).

Heard and unheard evaluations are shown in Table 5.6, where for the heard split we also train the model on 70k samples containing both old and new classes. The difference in performance is only 2%, which demonstrates the ability of our network to generalize to unheard or unseen categories. This is not surprising due to the similarity between several categories. For example, if the training corpus contains human speech, one would expect the model to be capable of localizing human singing, as both classes share semantic similarities in audio and visual features.

5.7 Conclusion

We revisit the problem of unsupervised visual sound source localization. For this task, we introduce a new large-scale benchmark called VGG-Sound Source, which is more challenging than existing ones such as Flickr SoundNet. We also suggest a simple, general and effective technique that significantly boosts the performance of existing sound source locators, by explicitly mining for hard negative image locations in the same image that contains the sounding objecs. A careful implementation of this idea using Tri-maps and differentiable thresholding allows us to significantly outperform the state of the art.

Acknowledgements

This work is supported by the UK EPSRC CDT in Autonomous Intelligent Machines and Systems, the Oxford-Google DeepMind Graduate Scholarship, the Google PhD Fellowship, and EPSRC Programme Grants Seebibyte EP/M013774/1 and VisualAI EP/T028572/1.

5.A Evaluation metric

We follow the same evaluation metrics as in Senocak et al. [2018], and report consensus intersection over union (cIoU) and area under curve (AUC). The Flickr Soundnet dataset contains 3 bounding box annotations from different human annotators. The bounding box annotations are first converted into binary masks $\{\mathbf{b}_j\}_{j=1}^{n}$ where n is the number of bounding box annotations per image. The final weighted ground truth mask is defined as:

$$\mathbf{g} = min(\sum_{j=1}^{n} \frac{\mathbf{b}_j}{C}, 1)$$

where C is a parameter meaning the minimum number of opinions to reach agreement. We choose $C = 2$, the same as Senocak et al. [2018]. Given the ground truth $\mathbf{g}$ and our prediction $\mathbf{p}$, the cIoU is defined as

$$cIoU(\tau) = \frac{\sum_{i \in A(\tau)} g_i}{\sum_i g_i + \sum_{i \in A(\tau) - G} 1}$$

where i indicates the pixel index of the map, τ denotes the threshold to judge positiveness, $A(\tau) = \{i | p_i > \tau\}$, and $G = \{i | g_i > 0\}$. We follow Senocak et al. [2018], and use $\tau = 0.5$. Example predictions and their cIoUs are shown in Figure 5.6.

cIoU = 0.673 cIoU = 0.535

Figure 5.6: Example predictions with calculated cIoU.

Since the $cIoU$ is calculated for each testing image-audio pair, the success ratio is defined as number of successful samples ($cIoU$ greater than a threshold τ_2) / total number of samples. The curve showing success ratio is plotted against the threshold τ_2 varied from 0 to 1 and the area under the curve is reported. The Pseudocode is shown in Algorithm 1.

Algorithm 1 Pseudocode of AUC calculation

```
# cIoUs : [cIoU_1,cIoU_2,...,cIoU_n]

x = [0.05 * i for i in range(21)]
for t in x: # Divide into 20 different thresholds
      score.append(sum(cIoUs > t) / len(cIoUs))
AUC = calulcate_auc(x,score) # sklearn.metrics.auc
```

5.A.1 Tri-map visualization

In addition to video examples, we show more image results of our Tri-maps in Figure 5.7.

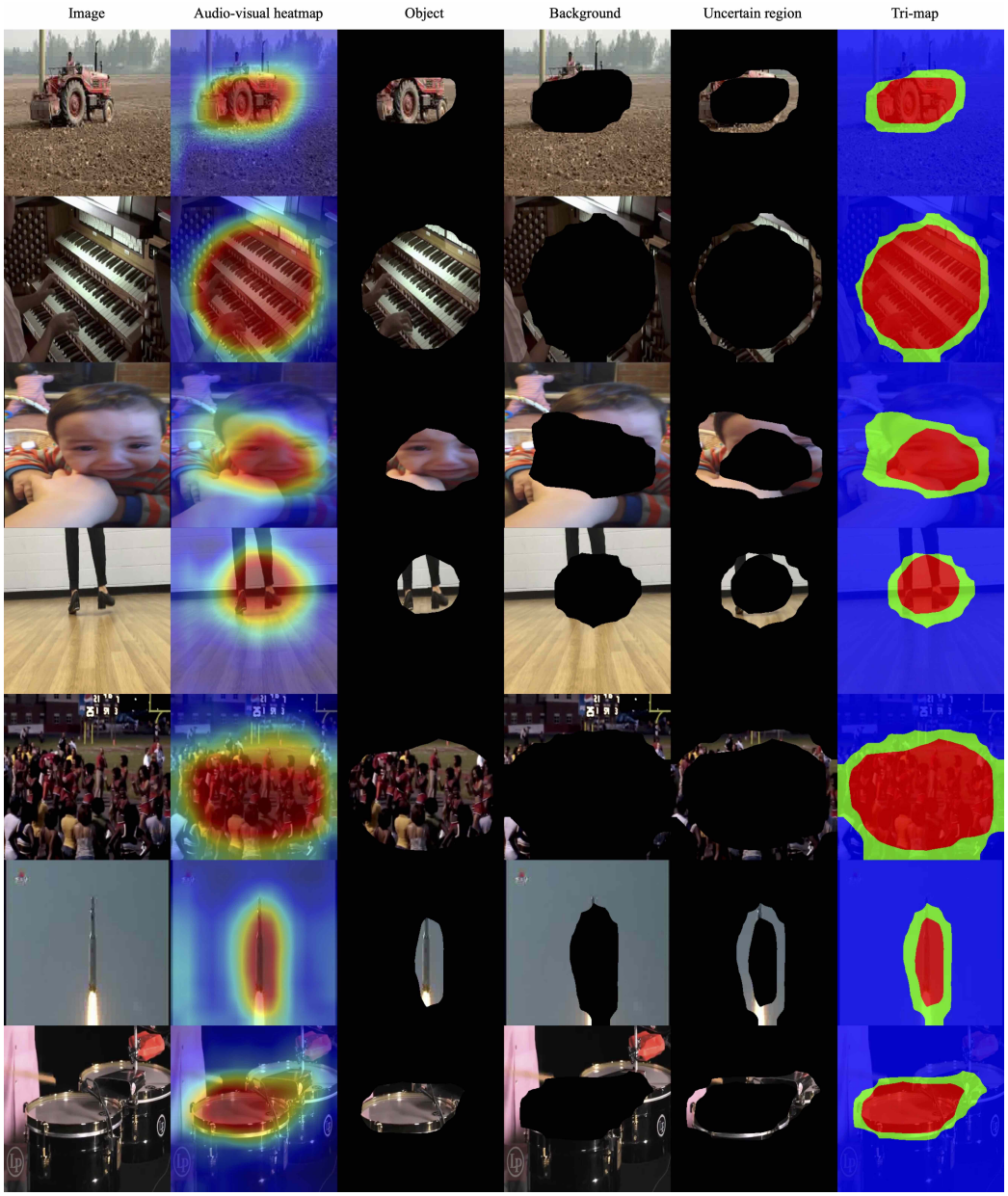


Figure 5.7: Tri-map visualization examples.

5.B VGG-Sound Source (VGG-SS)

We show more dataset examples, the full 220 class list of VGG-SS and the classes we removed from the original VGG-Sound dataset [Chen et al., 2020a] in this section.

5.B.1 VGG-SS annotation interface

We show our manual annotation interface, LISA [Dutta and Zisserman, 2019], in Figure 5.8. The example videos are from the class 'Rapping'. The 'Play' button shows the 5s clip, and 'Show region' recenters to the key frame we want to annotate. We choose 'Yes' only if we hear the correct sound, 'No' for the clips that do not contain the sound of class, and 'Not Sure' if the sound is not within the 5s we choose (original video clip is 10s)

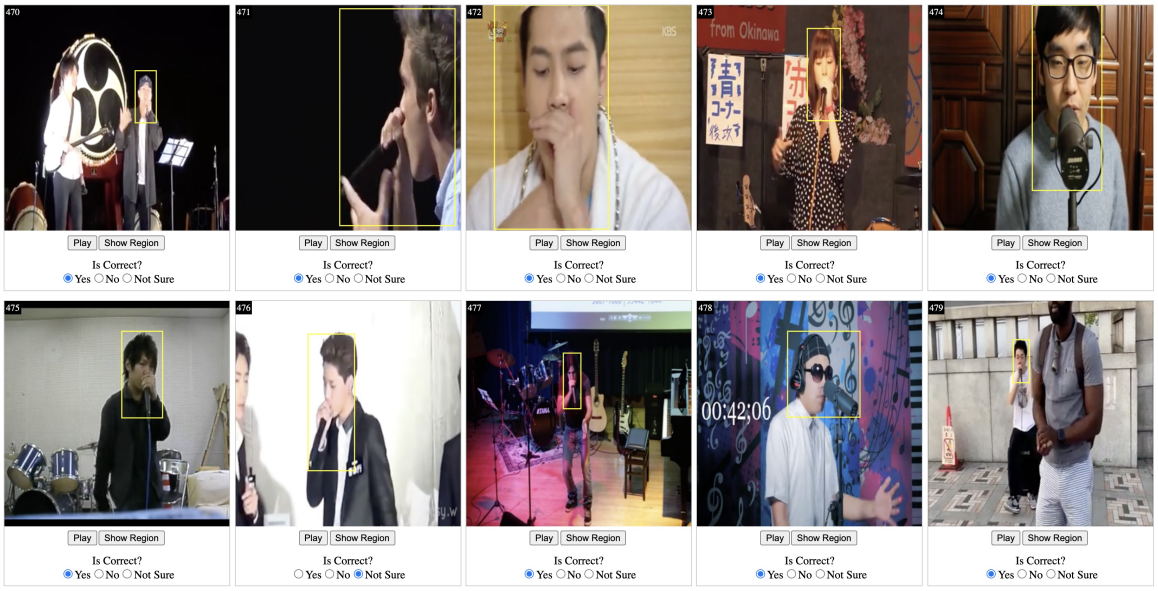


Figure 5.8: LISA Annotation Interface.

5.B.2 VGG-SS examples

We randomly sample from images with 1 bounding box, 2 bounding boxes, and with more than 2 bounding boxes. We show examples with 1 bounding box on the top 4 rows, examples with 2 bounding boxes on the following two rows, and examples with more than 2 bounding boxes on the last row in Figure 5.9.

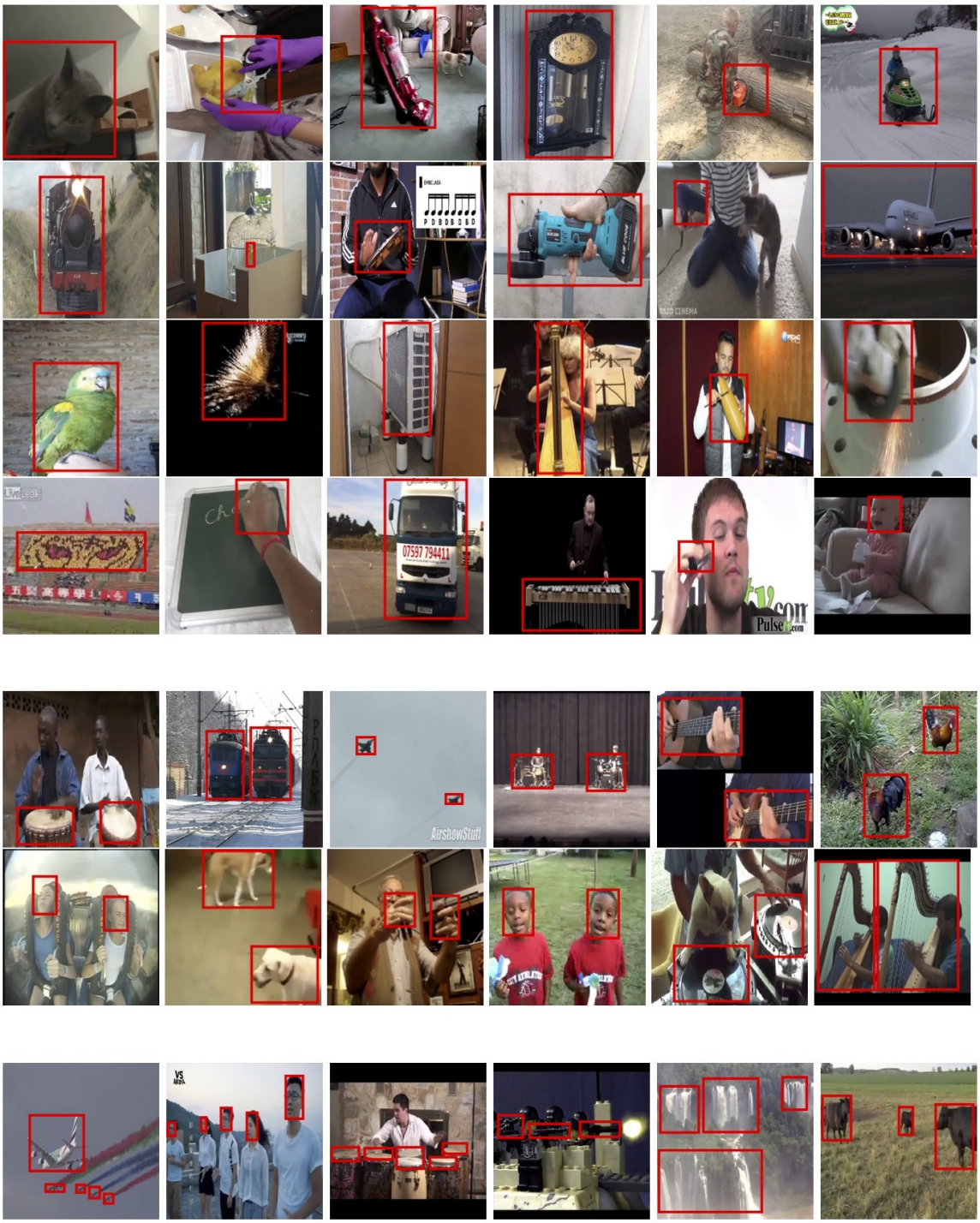


Figure 5.9: We show examples with 1 bounding box on the top 4 rows, examples with 2 bounding boxes on the following two rows, and examples with more than 2 bounding boxes on the last row.

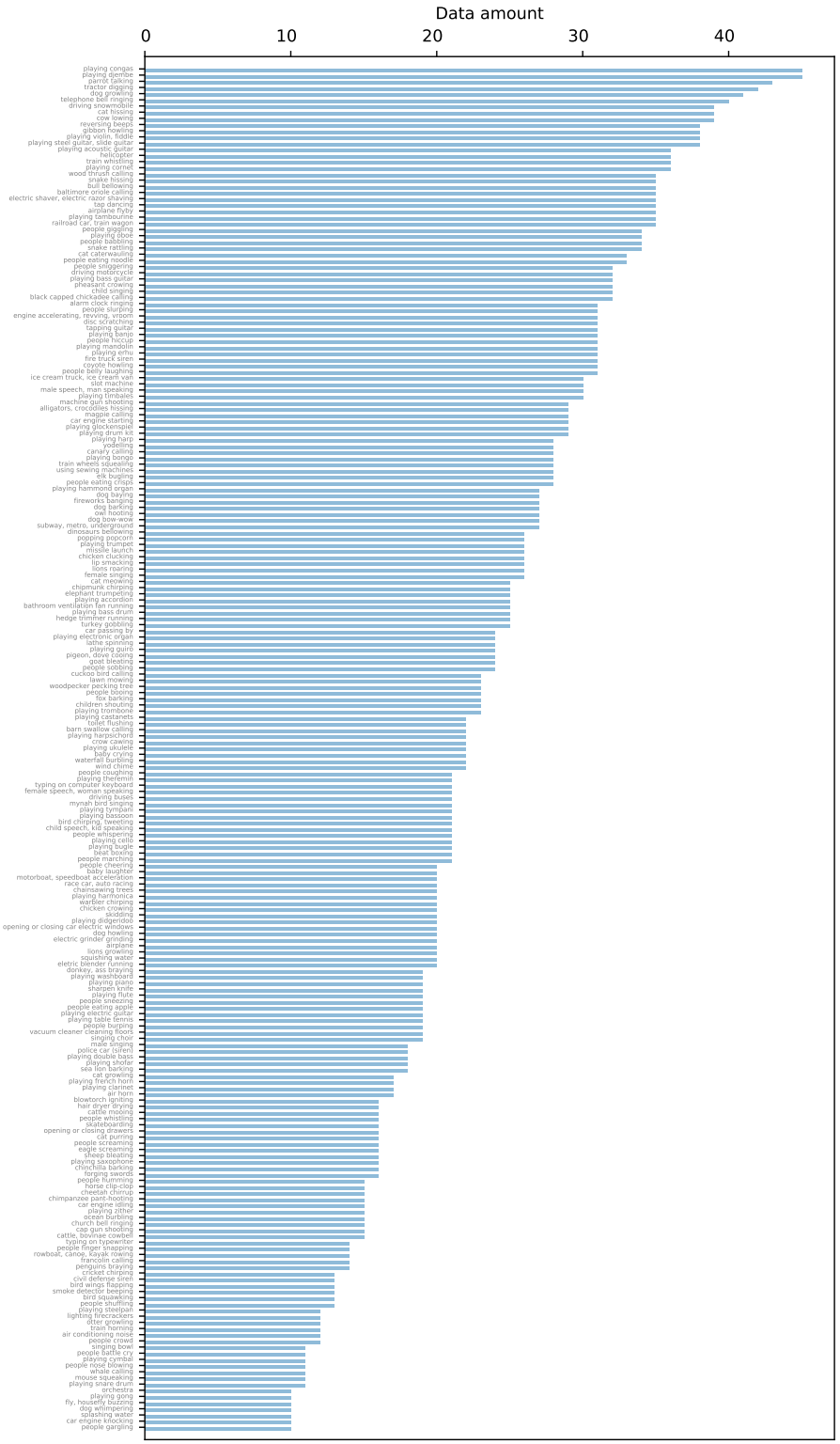


Figure 5.10: VGG-SS benchmark per class statistics.

Chapter 6

Audio-Visual synchronization in the wild

Honglie Chen **Weidi Xie** **Triantafyllos Afouras** **Arsha Nagrani**
Andrea Vedaldi **Andrew Zisserman**

Visual Geometry Group, University of Oxford

Abstract

In this paper, we consider the problem of audio-visual synchronization applied to videos 'in-the-wild' (*i.e.*of general classes beyond speech). As a new task, we identify and curate a test set with high audio-visual correlation, namely VGG-Sound Sync. We compare a number of transformer-based architectural variants specifically designed to model audio and visual signals of arbitrary length, while significantly reducing memory requirements during training. We further conduct an in-depth analysis on the curated dataset and define an evaluation metric for open domain audio-visual synchronization. We apply our method on standard lip reading speech benchmarks, LRS2 and LRS3, with ablations on various aspects. Finally, we set the first benchmark for general audio-visual synchronization with over 160 diverse classes in the new VGG-Sound Sync video dataset. In all cases, our proposed model outperforms the previous state-of-the-art by a significant margin. Project page: https://www.robots.ox.ac.uk/~vgg/research/avs

Published in the Proceedings of the British Machine Vision Conference (BMVC), 2021.

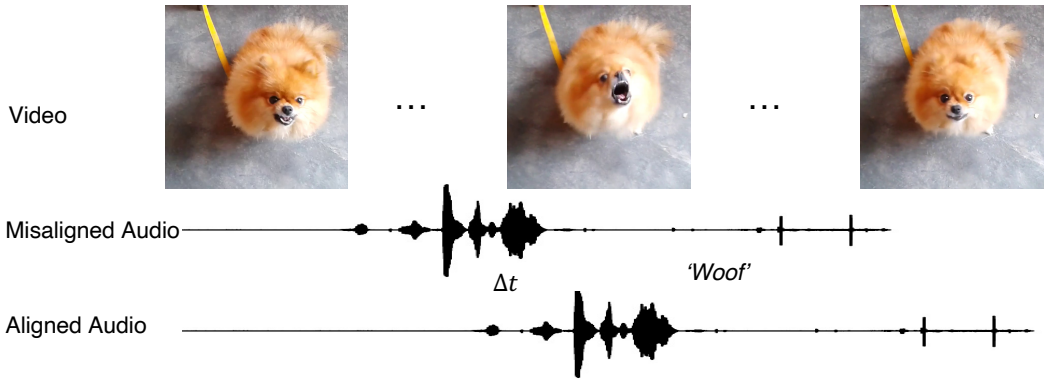


Figure 6.1: **Audio-visual synchronization in the wild**. The goal of this work is to develop an audio-visual synchronization method that performs well on general videos in-the-wild. Unlike speech videos, highly correlated audio and visual events for general classes may occur briefly in the video (e.g. the bark of the dog in the centre of this clip). A short clip sampled randomly from the video might miss this fleeting moment; with longer input videos this becomes less probable. With this in mind, we propose a Transformer based architecture that can operate on long sequences, and is able to perform audio-visual synchronization on videos of 160 general sound classes.

6.1 Introduction

In videos, the audio and visual streams are often strongly correlated, presenting effective signals for self-supervised representation learning [Arandjelovic and Zisserman, 2018; Owens and Efros, 2018]. A useful task in this area is audio-visual synchronization, and several studies have shown promising results even without requiring any manual supervision [Chung and Zisserman, 2016b; Chung et al., 2019; Afouras et al., 2020b]. However, these works study this problem extensively on only one class – human speech – where even a slight offset is easily discernable.

In this paper, rather than focusing on a specialised domain, *e.g.*human speech [Chung and Zisserman, 2016a,b; Chung et al., 2019; Afouras et al., 2020b], or videos with periodic sounds such as the tennis shots in a match [Ebeneze et al., 2021], we aim to explore audio-visual synchronization on general videos in the wild (characterized by more than 160 sound classes). Solving this task would be extremely useful for a number of applications including video conferencing, television broadcasts and video editing, which are largely done by 'off-line' measurements or heavy manual processing [Staelens et al., 2012; Shrestha et al., 2010; Dassani et al., 2019].

There are several challenges in automatic audio-visual synchronization for general classes. First, unlike the task of synchronizing speech [Nagrani et al., 2017b; Afouras et al., 2019a; Chung et al., 2017b; Chung and Zisserman, 2017], which contains audio-visual evidence from the lips most of the time, videos from general classes may contain uniform sounds (*e.g.* airplane engine sound, electric trimmer), ambient sounds (*e.g.* wind, water, crowds, traffic), or small object sound sources (*e.g.* players in an orchestra, birds), which make synchronization extremely challenging or even impossible; Second, for categories with strong audio-visual evidence, localising such signals can also be difficult, for example: temporally, 'dog barking' may happen instantaneously, as shown in Figure 6.1, and spatially, unlike in speech synchronization where visual cues are largely localised to lip motions, in general videos the entire frame must be processed to accommodate different object classes; Third, due to the aforementioned challenges, it is unclear how to evaluate the synchronization in general classes.

In order to address these issues, first, we curate a new benchmark for general audio-visual synchronization called VGG-Sound Sync using a subset of VGG-Sound [Chen et al., 2020a]. Specifically, this is built by selecting classes and video clips that potentially have audio-visual correlation, and removing those classes and video clips that don't, *e.g.* uniform, ambient sound; Second, compared with previous works, we use substantially longer input video sequences, so that the chance of having a synchronized audio and video event in the input increases. We explore several variants of Transformer-based architectures that can elegantly deal with these long sequences of variable lengths, and that use self-attention to implicitly pick out the relevant parts in both space and time. Finally, we conduct a thorough study on the VGG-Sound Sync test set, estimating the chance of audio-visual synchronization for different clip lengths, and also define a set of metrics for evaluation.

Concretely, in this paper, we consider the problem of audio-visual synchronization applied to 'in-the-wild' videos, *i.e.* general classes beyond speech. We make the following contributions: (i) we identify and curate a subset of general classes from VGG-Sound, namely VGG-Sound Sync, with potentially high audio-visual correlation; (ii) we introduce a set of transformer-based architectures for audio-visual synchronization, which can exploit the spatial-temporal correlations between audio and visual streams, such models can train

and predict on variable length video sequences; (iii) we conduct an analysis on the VGG-Sound Sync test set, and define an evaluation metric for audio-visual synchronization on these videos; (iv) we achieve state-of-the art synchronization performance on standard lip reading speech benchmarks, LRS2, LRS3; and more importantly, set the first benchmark for audio-visual synchronization in general (non-speech) classes.

6.2 Related Work

Audio-visual synchronization. Early works studied audio-visual synchronization in talking faces [Hershey and Movellan, 1999; Slaney et al., 2000] using handcrafted features and statistical models. Chung and Zisserman [2016a] developed a model for synchronizing lip movements to audio speech, based on a dual-encoder architecture trained with contrasting learning. Follow-up works improved this pipeline by moving to noise-contrastive objectives [Chung et al., 2019], or directly inferring the audio-visual offset conditional on cross-similarity patterns [Kim et al., 2021]. Lip synchronization is an important component for pipelines used for various visual speech related tasks, such as lipreading [Chung and Zisserman, 2016c; Afouras et al., 2019a], active speaker detection [Chung and Zisserman, 2016a] and sign language recognition [Albanie et al., 2020]. Although these works demonstrate strong synchronization performance, they are limited in terms of deployment as they are applicable only on videos that include speech. Our method generalizes to broader sound source classes and conditions, while also outperforming these works in the speech domain. Other closely related works have investigated lip-syncing [Halperin et al., 2019], i.e. the temporal alignment of video and speech clips from different sources, speech-conditioned face animation [Chung et al., 2017a; Vougioukas et al., 2019], and audio-visual dubbing [Prajwal et al., 2019; Yang et al., 2020]. Audio-visual synchronization has been also used as a pre-text task for learning general visual and audio representations [Owens and Efros, 2018; Korbar et al., 2018; Patrick et al., 2020; Afouras et al., 2020b; Cheng et al., 2020]. Khosravan et al. [2019] investigate the use of attention for audio-visual synchronization on speech data. Ebeneze et al. [2021] train models to detect synchronization errors based on mismatch of event detection between the audio and visual stream. Casanovas and

Cavallaro [2014] propose a method for synchronizing audio-visual recordings of the same events from different cameras. Unlike the works above which use simple concatenation between audio and visual features, we employ encoder-based and decoder-based Transformers to implicitly match the relevant parts.

Audio-visual learning. Our work is more broadly related to various works on audio-visual learning, including audio-visual event detection [Tian et al., 2018; Lin et al., 2019], sound-source localization [Arandjelovic et al., 2016; Qian et al., 2020; Xu et al., 2020; Afouras et al., 2020b; Gan et al., 2019], representation learning [Nagrani et al., 2018c; Alwassel et al., 2019; Asano et al., 2020a], audiovisual fusion [Xiao et al., 2020; Jaegle et al., 2021; Nagrani et al., 2021] and sound source separation [Zhao et al., 2018a; Gao et al., 2018; Tzinis et al., 2021]. More recently, Zhao et al. [2019] proposed to leverage temporal motion information to separate musical instrument sound. Gan et al. [2020b] further improved the sound separation models with explicit keypoint-based representations. Another line of work explored audio synthesis using visual input: Gan et al. [2020a] utilized body keypoints to synthesize music from a silent video, and Koepke et al. [2020] synthesized piano music from overhead views of the hands. Gao and Grauman [2019] converted monaural audio into binaural audio by injecting visual spatial information.

Transformers. Transformers [Vaswani et al., 2017] were originally introduced for NLP tasks, in particular machine translation where they showed improvement over recurrent-based encoder-decoder architectures. Since then they have been widely applied to a great range of problems, including speech recognition [Gulati et al., 2020b], language modelling [Devlin et al., 2018; Dai et al., 2019], object detection [Carion et al., 2020; Yao et al., 2021]. Recent works have even extended their use to visual feature extraction, replacing CNNs, for classification [Dosovitskiy et al., 2021], semantic segmentation [Wu et al., 2020; Dosovitskiy et al., 2021] and video representation learning [Bertasius et al., 2021]. In the multi-modal domain, Lin and Wang [2020]; Tian et al. [2020] explored unimodal and cross-modal temporal contexts simultaneously to detect audio-visual events, and Lee et al. [2021b] alleviated the high memory requirement of a vanilla Transformer by

sharing the weights across layers and modalities. Audio-visual fusion using transformers has also been explored by new architectures such as Perceiver [Jaegle et al., 2021] and MBT [Nagrani et al., 2021].

6.3 Method

In this section, we describe our proposed method, which we call Audio-Visual synchronization with Transformers (AVST). Our goal is to detect audio-visual synchronization without the use of any manual annotation. Similar to prior work [Chung and Zisserman, 2016a; Owens and Efros, 2018; Afouras et al., 2020b], we first use CNN encoders to extract visual and audio representations from unlabelled video (described in section 6.3.1.1). In section 6.3.1.2, we introduce three variants of our Transformer-based module that can jointly process visual and audio features, and discuss the pros and cons for each architecture. Finally in section 6.3.2, we describe the contrastive learning objective used to train the model. An overview of our architecture can be found in Figure 6.2.

6.3.1 Architecture

6.3.1.1 Audio and visual representations

The proposed model has two input streams, one ingesting a short video clip $v_i \in \mathbb{R}^{3\times T\times H_v\times W_v}$ consisting of T visual frames and the other taking in an audio spectrogram $a_j \in \mathbb{R}^{1\times H_a\times W_a}$, where i, j index the source of each modality (e.g. when $i = j$ the visual and audio signals come from the same video and are temporally aligned). We compute representations for each modality using functions $f(\cdot;\theta_1)$ and $g(\cdot;\theta_2)$, which in this case are instantiated using CNN encoders:

$$V_i = f(v_i;\theta_1), \quad V_i \in \mathbb{R}^{c\times t_v\times h\times w} \tag{6.1}$$

$$A_j = g(a_j;\theta_2), \quad A_i \in \mathbb{R}^{c\times t_a} \tag{6.2}$$

Both representations V_i and A_j have the same number of channels c, which allows us to jointly model the input video and audio with cross-modal attention.

6.3.1.2 synchronization module

The visual and audio representations are formulated into a sequence of tokens, and passed through a Transformer [Vaswani et al., 2017] consisting of N layers. We introduce three variants of AVST, each one with a slightly different design choice for modelling audio-visual information.

Encoder variant (AVST_{enc}). The most straightforward step is to simply treat the dense visual features as a sequence of 'visual tokens'. To that end, the visual features are flattened over the spatial dimensions and concatenated to the audio features after also prepending a learnable `class` token ([CLS]), inspired by the BERT model [Devlin et al., 2018]. In order for the model to distinguish the signals from the two modalities and maintain spatio-temporal positional information (as all subsequent Transformer layers are permutation invariant), three types of encodings are also added to the audio and visual features: modality encodings $E_m \in \mathbb{R}^{c\times 2}$, that indicate the type of feature (i.e. audio or visual); temporal encodings $E_{t_{\{v,a\}}} \in \mathbb{R}^{c\times t_{\{v,a\}}}$ and spatial encodings $E_s \in \mathbb{R}^{c\times h\times w}$, that keep track of absolute positions for the tokens:

$$\overline{V_i} = \text{FLATTEN}(V_i) + E_m + E_{t_v} + E_s, \tag{6.3}$$
$$\overline{A_j} = A_j + E_m + E_{t_a}, \tag{6.4}$$
$$Z_{ij} = [[\text{CLS}]; \overline{V_i}; \overline{A_j}] \tag{6.5}$$

where $[;]$ denotes a concatenation operation. The output result, $Z_{ij} \in \mathbb{R}^{c\times(1+hwt_v+t_a)}$, is then fed into a Transformer Encoder [Vaswani et al., 2017], that is composed of a stack of Multihead Self-Attention (MSA), and feed forward networks (FFNs). This module allows the tokens from both modalities to directly interact with each other through the self-attention operations:

$$Y_{ij} = \text{TRANSFORMER-ENCODER}(Z_{ij}). \tag{6.6}$$

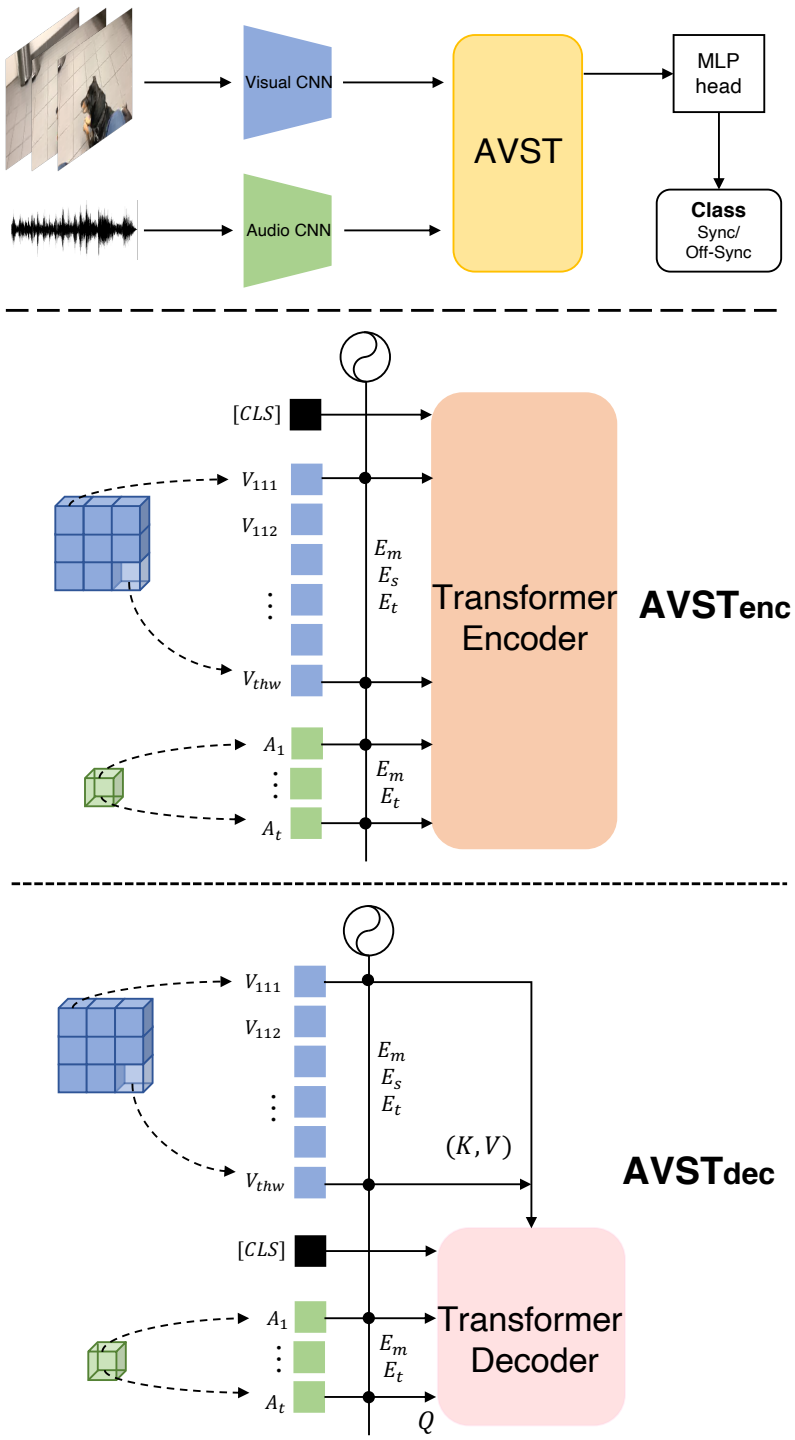


Figure 6.2: **The AVST model architecture and variants.** We use AVST to jointly model visual and audio representations computed from backbone CNNs, with an MLP head to predict audio-visual synchronization (top). On the middle and bottom, we show two variants of the AVST transformer backbone, the Encoder (AVST_{enc}) and Decoder variant (AVST_{dec}). AVST_{enc} uses self-attention for all audio and visual features, whereas in AVST_{dec} the visual information is kept fixed, and the audio latents are used to QUERY the visual information which forms the KEY, VALUE pairs.

Max-pooled encoder variant (AVST$_{\text{enc-mp}}$). Naively feeding all visual features densely into the Transformer is computationally expensive, with a quadratic cost, $\mathcal{O}((hwt_v + t_a)^2)$, which significantly limits the scalability to longer video sequences. Thus, although the architecture is powerful, it heavily limits the audio-visual samples that can be processed in each batch, which in turn limits the number of negatives that can be used for training, resulting in sub-optimal performance, as we will show in the experiments (section 6.4).

Rather than taking dense visual features as input, we propose a cheaper alternative, which consists of a simple Global Max Pooling (GMP) operation spatially on each frame. This reduces the length of the sequence that is input to the Transformer from (hwt_v+t_a+1) to (t_v+t_a+1); and thereby significantly lowers the memory footprint of the MSA module. To obtain AVST$_{\text{enc-mp}}$ we simply replace the flattening operator in Equation 6.4 with GMP:

$$\overline{V_i} = \text{GMP}(V_i) + E_m + E_{t_v} + E_s \tag{6.7}$$

Decoder variant (AVST$_{\text{dec}}$). Using the visual feature from max-pooling is computationally efficient, however, it also removes spatial information in the visual representations, impairing the ability of audio features to probe fine-grained visual information, which may be required for certain general object categories.

To resolve the aforementioned challenge, we consider an alternative architecture that uses a Transformer decoder [Vaswani et al., 2017], as shown in Figure 6.2 (right), where dense visual features are kept fixed without self-attention and passed as the KEY and VALUE inputs to every decoder layer, and audio features (concatenated along a [CLS] token, similarly to AVST$_{\text{enc}}$) are passed as the QUERY inputs:

$$\text{QUERY} = \text{CONCAT}([\text{CLS}], \overline{A}_j), \ \text{KEY} = \text{VALUE} = \overline{V_i} \tag{6.8}$$

$$Y_{ij} = \text{TRANSFORMER-DECODER}(\text{QUERY}, \ \text{KEY}, \ \text{VALUE}) \tag{6.9}$$

6.3.1.3 Output head

For all variants, we only use the first token (Y_{ij}^1), of the output of the final encoder (or decoder) layer, corresponding to the [CLS] position in the input sequence. This functions as an aggregate representation of the whole output sequence and is fed to $h(\cdot;\theta_3)$, which we implement as an MLP head. The output is a synchronization score that indicates to what degree the inputs v_i and a_i are in sync, $s_{ij} = h(Y_{ij}^1;\theta_3)$.

6.3.2 Training objectives

Given mini-batches $\mathcal{B} = \{(v_1, a_1), (v_2, a_2), ..., (v_k, a_k)\}$ of temporally aligned audio-visual pairs, the goal is to jointly optimize the entire pipeline in an end-to-end manner, so that the prediction scores for synchronized pairs (v_i, a_i) are maximised, while the scores of out-of-sync pairs (v_i, a_j) are minimised. Training proceeds by minimising the commonly used InfoNCE loss, defined as:

$$\mathcal{L} = -\frac{1}{k}\sum_{i=1}^{k}\left[\log\frac{\exp(s_{ii})}{\sum_j \exp(s_{ij})}\right]$$

Discussion. Unlike previous works, which simply compute either the Euclidean distance or the cosine similarity between the audio and visual representations obtained from separate CNN streams to predict synchronization, we use a Transformer model that jointly models the relationship between the audio and visual streams using attention over multiple layers. This is useful for attending to longer input sequences, where informative audio and video may only be localised in a short sub-sequence of the video.

6.4 Experiments

In the following sections, we describe the datasets, evaluation protocol and experimental details to thoroughly assess our method.

6.4.1 Datasets

Audio-visual speech datasets: We conduct experiments on two public audio-visual speech datasets, namely, LRS2 [Afouras et al., 2019a; Chung et al., 2017b; Chung and Zisserman, 2017] and LRS3 [Afouras et al., 2018b], which have been created from British television footage and TED talks from YouTube respectively. Both datasets are distributed as short video clips of tightly cropped face tracks around the active speaker's head. Since LRS3 is based on public YouTube videos, we also extract full-frame versions of the same clips for all splits ("pretrain", "trainval" and "test"). To distinguish between these two versions of LRS3, we refer to them as "cropped" and "full-frame" respectively. Note that for LRS2, only the "cropped" version is available.

General sound dataset: Here, we construct a new benchmark called VGG-Sound Sync using a subset of VGG-Sound [Chen et al., 2020a], which was recently released with over $200k$ clips, and each clip is labelled as one of the 300 different sound categories. This dataset is conveniently audio-visual, in the sense that the object that emits sound is likely to be visible in the corresponding video clip. In the next section, we will detail the curation process.

6.4.2 Evaluation protocol

Depending on the downstream benchmarks, we consider two different evaluation protocols.

6.4.2.1 Audio-visual synchronization on speech

For LRS2 and LRS3, we follow previous works and use an input of 5 frames, extracted at 25fps. During testing, the synchronization scores were computed densely between each 5-frame video feature and all audio features in ± 15 frame range. Synchronization was then determined to be correct if the lip-sync error was not detectable to a human, *i.e.* the maximum score between two streams is within ± 1 frame ($\pm 0.04s$) from the ground truth Chung et al. [2019].

6.4.2.2 Audio-visual synchronization on general classes

Compared to speech videos with audio-visual cues (the lip motion and speech) spanning almost the entire clip, evaluating synchronization on general videos potentially incurs two challenges: (1) videos with only ambient or uniform sound, *e.g.* wind, wave, engine sound, are unlikely to have any cues that can be used for synchronization; (2) the audio-visual cues for synchronization are sometimes instantaneous, *e.g.* a dog barking may only last for less than 1s. In the following, we describe the evaluation benchmark and how it was constructed.

Seconds	1s	2s	3s	4s	5s	6s
Audio-visual evident	50%	56%	57%	62%	60%	59%
Uniform/ambient sound	30%	34%	35%	31%	35%	38%
No sound/object	20%	10%	8%	7%	5%	3%

Table 6.1: Categorisation of video clips as duration of video varies.

Categorising video clips. Here, we analyse the statistics of videos in the VGG-Sound test set, by categorising each video clip into three classes, namely, audio-visual evident, uniform / ambient sound, missing sound / visual object. Specifically, we randomly sample 1200 video clips, where each clip is of different lengths between 1s and 6s for manual verification. As shown in Table 6.1, the following phenomenon can be observed: *First*, the proportion of clips with uniform or ambient sound remains roughly constant, as this error is caused by all the video clips of particular sound categories; *Second*, as expected, with the increase of temporal lengths, the chance of having audio-visual cues for synchronization increases. Notably, when clips are over 2s, the error rate drops to around 10%.

At this stage, we curate a subset of VGG-Sound by filtering the sound categories to remove ones that are potentially dominated by uniform / ambient sound, resulting in a test set of over 160 classes, 95k training videos and 5k testing videos (each of them lasts 10 seconds).

Verifying synchronization of YouTube videos. In this section, we conduct manual verification to serve two purposes: *first*, as the video clips in VGG-Sound are all sourced from YouTube, their audio-visual alignments are not always guaranteed, we aim to understand the chance of these videos being audio-visual synchronized, at least from the perspective of an ordinary human observer; *second*, we aim to understand the human tolerance, by that we mean, how much temporal misalignment is noticeable for human observers. In a practical evaluation, offsets smaller than such tolerance should be ignored or considered as correct. In detail, we randomly sample 500 example videos from our test set with 25fps, and create 1000 audio-visual pairs, with each lasting 5s. The temporal offsets between both streams vary from $[-0.8, +0.8]$ second, for example, for one visual clip sampled at time t, its paired audio signal can be centered at any time between $t - 0.8, t + 0.8$, we feed these pairs to human observers and ask a binary question: *is the given audio-visual pair synchronized ?* Please check the detailed statistics on proportions of videos considered to be syncd by a manual observer in Appendix 6.C.

Summary. To evaluate the synchronization for videos of general classes, we curate a test set from VGG-Sound, namely VGG-Sound Sync, with ambient, uniform sound categories removed. We only include audio-visual pairs of length between 2 - 6 second, that have a sufficiently high chance of containing informative cues for synchronization. During evaluation, we decode the videos with 25fps, and construct audio or visual input by taking every 5th frame, note that, this has the same effect as using input decoded from 5fps. The synchronization scores are computed for all audio-visual pairs with $[-15, -14, \ldots, +14, +15]$ frame gaps. Considering the challenging nature for audio-visual synchronization in natural videos, synchronization is determined to be correct if the synchronization error is not detectable by a human, *i.e.* the maximum score between two streams is within ± 5 frames ($\pm 0.2s$) from the ground truth.

6.4.3 Implementation details

Training curriculum. Following prior work [Korbar et al., 2018; Afouras et al., 2020b], we train our models in two stages: in the first stage, we construct the mini-batches by sampling audio-visual clips from different videos, this provides easy (correspondence) negatives that helps the training converge. In the second stage, all the clips in a mini-batch are sampled from the same video, which provides harder (synchronization) negatives.

Training hyper-parameters. On a P40 GPU with 24GB memory, we train AVST_{enc} with a batch-size of 4 (due to memory restrictions), for $\text{AVST}_{\text{enc-mp}}$ and AVST_{dec}, we use a batch-size of 16 and 12 respectively, thereby allowing more negatives per batch.

Architectural Details. Unless otherwise specified, our Transformer encoder consists of 3 layers, 4 attention heads and a hidden unit dimension of 512. Typically $H = W = 224$ and $h = w = 14$. We refer the readers to Appendix 6.A for more details.

6.4.4 Results on speech datasets

We first report experimental results on LRS2 and LRS3, and perform a number of ablations on different architectural design choices. We also analyse the model's robustness on cases, where the visual or audio signal is partially unavailable.

Architectures comparison. To compare our proposed architecture variants and assess their trade-offs, we train and evaluate them on the "full-frame" version of LRS3 and show results of all three Transformer variants in Table 6.2. Due to the memory restrictions, we can only train AVST_{enc} with a fixed length of 5 frames, whereas for the other two architectures, training is done with variable sequence length and larger batch size (see section 6.4.3). We observe a large gap (6% – 7%) between the performance of AVST_{enc} and the other two variants, which indicates that AVST_{enc} suffers from the reduced number of negatives. We also note that AVST_{dec} can localise sound sources because it preserves spatial information, but shows slightly worse performance than $\text{AVST}_{\text{enc-mp}}$ on speech datasets. We conclude that $\text{AVST}_{\text{enc-mp}}$ is a light-weight solution that offers the best performance when

the sounding objects (*e.g.*lips) are clear and unique, which need little fine-grained spatial information.

Comparison to the state-of-the-art. We compare our method to previous work on "full-frame" LRS3 in the top half of Table 6.2. We show a significant improvement compared to the AVobjects baseline (16% gain) on short input (5 frames) reaching up to an almost saturated 98.6% accuracy with 15 frames. In the bottom half of Table 6.2, we further summarise our results for experiments on the "cropped" LRS2 dataset. Here too, we observe that our method greatly outperforms both the SyncNet [Chung and Zisserman, 2016b] and PM [Chung et al., 2019] baselines, and achieves almost perfect accuracy with 15 frames of input during test time.

Since $\text{AVST}_{\text{enc-mp}}$ shows superior performance on speech datasets using a light-weight architecture, we conduct the rest of the analysis on speech data using $\text{AVST}_{\text{enc-mp}}$. In addition, in order to compare with SyncNet and PM, we use the same fixed length of 5 frames during training and testing.

Number of Transformer Layers. We ablate the Transformer depth on the LRS2 dataset in Table 6.3. As more layers are added, the performance consistently improves, achieving the best performance with 3 layers. This confirms the effectiveness of self-attention in jointly modelling audio and visual information.

Robustness test. To mimic real-world scenarios, where sound sources and their corresponding sound might not appear together at every frame, we further conduct experiments to assess the robustness of our model on the LRS2 dataset by randomly masking input audio or video frames. We mask 1 frame for each or both modalities. As can be seen in Table 6.4, we find that for short inputs this causes a significant performance drop, however with longer inputs, we achieve comparable results to the non-masked case in Table 6.2.

Model	# Params.	Var.	Dataset	Clip Length in frames (seconds)					
				5(0.2s)	7(0.28s)	9(0.36s)	11(0.44s)	13(0.52s)	15(0.6s)
AVobjects	69.4M	✗	LRS3	61.8	72.0	79.7	85.4	89.5	91.8
$AVST_{enc}$	42.6M	✗	LRS3	70.2	77.1	83.3	88.4	92.0	94.4
$AVST_{dec}$	44.5M	✓	LRS3	75.7	86.4	89.4	94.0	95.1	96.9
$AVST_{enc\text{-}mp}$	42.4M	✓	LRS3	**77.3**	**88.0**	**93.3**	**96.4**	**97.8**	**98.6**
SyncNet	13.6M	✗	LRS2	75.8	82.3	87.6	91.8	94.5	96.1
PM	13.6M	✗	LRS2	88.1	93.8	96.4	97.9	98.7	99.1
$AVST_{enc\text{-}mp}$	42.4M	✓	LRS2	**91.9**	**97.0**	**98.8**	**99.6**	**99.8**	**99.9**

Table 6.2: **Architecture comparison on LRS3 and LRS2.** We use the 'full-frame' dataset. 'Var': whether models are trained and tested using variable length inputs. '5-15' refers to the number of input frames to corresponding models.

# Layers	Clip Length (frames)					
	5	7	9	11	13	15
1	89.1	94.0	96.8	98.4	99.1	99.4
2	91.6	95.4	97.6	98.8	99.1	99.6
3	92.0	95.5	97.7	98.8	99.3	99.6

Table 6.3: **Ablation on Transformer depth (LRS2).** Performance increases with depth.

Mask	Clip Length (frames)					
	5	7	9	11	13	15
Audio	73.1	85.3	92.6	96.1	98.0	99.2
Visual	76.5	87.3	93.4	96.9	98.2	99.3
Both	71.7	84.0	91.2	95.6	97.7	99.1

Table 6.4: **Robustness test on LRS2.** 1 frame is masked during train and test.

6.4.5 Results on general sound classes

In this section, we report audio-visual synchronization results on the VGG-Sound Sync dataset consisting of videos with general sound classes, and compare with several strong baselines. Results are provided in Table 6.5. First, while comparing with the recent AVobjects [Afouras et al., 2020a] method, both of our models show superior results on all input lengths, this is because (1) we trained on variable input lengths, where longer samples contain richer audio-visual evidence; and (2) the use of Transformer based architectures ($AVST_{enc}$ and $AVST_{dec}$) can implicitly discover the important temporal parts in long sequences. Second, in contrast to the results in speech datasets (Table 6.2), we note that $AVST_{dec}$ has higher accuracy than $AVST_{enc}$ on general videos. The reason is that general videos contain complex visual scenes and, compared to other variants, $AVST_{dec}$ can extract fine-grained spatial information in such situation by explicitly computing the attention between image regions and the audio sequence, therefore showing better performance. Finally, we analyse the performance for each class of VGG-Sound Sync dataset in Fig-

Method	Clip Length in frames (seconds)				
	10(2s)	15(3s)	20(4s)	25(5s)	30(6s)
AVobjects	37.2	42.6	45.1	47.3	49.4
$\text{AVST}_{\text{enc-mp}}$	39.0	44.1	46.8	49.7	51.8
AVST_{dec}	40.6	44.6	47.6	50.4	52.7

Table 6.5: **Audio-visual synchronization results on VGG-Sound Sync**. We show results for sequences of $10 - 30$ frames. Our model outperforms the state of the art AVObjects Afouras et al. [2020b] on sequence lengths $\geqslant 10$.

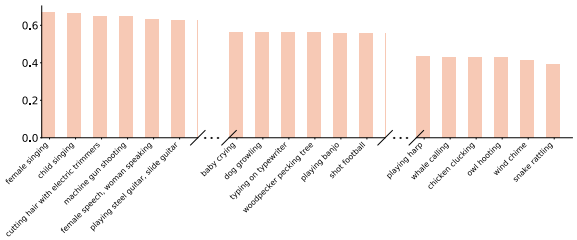


Figure 6.3: **Per-class accuracy on VGG-Sound Sync**.

ure 6.3, and find that the performance is highly class dependant, with the best class ('child singing') achieving 75.7%, and most highly performing classes containing strong audio-visual correlations, *e.g.*'female singing','playing steel guitar', etc.

6.4.5.1 Visualisation of attention heatmaps

We visualise the attention heatmaps of our model on samples from VGG-Sound Sync in Figure 6.4 (refer to Appendix 6.D for more qualitative results). For general object classes in VGG-Sound Sync, the model manages to pick up on interesting sources of motion that produce sound. When comparing the heatmaps produced by AVST_{dec} to the current *state-of-the-art* sound source localization method [Chen et al., 2021a], we notice that our method attends to regions that *create or modify* sound, *e.g.*hands, lips, helicopter rotor, etc, while Chen et al. [2021a] tends to localise the entire object.

6.5 Conclusion

We revisit the problem of audio-visual synchronization in human speech and introduce a new general class audio-visual synchronization benchmark called VGG-Sound Sync. We experiment with different variants of Transformer-based architectures, analyse several critical components, and conduct thorough ablation studies to validate their necessity. Con-

(a) Localise Visual Sound [Chen et al., 2021a]

(b) AVST_{dec}

Figure 6.4: **Attention heatmaps on VGG-Sound Sync.** We compare the heatmaps that we obtain with the AVST_{dec} model to the state-of-the-art method for sound source localization [Chen et al., 2021a]. It is interesting to note that while [Chen et al., 2021a] highlights discriminative parts of the objects that are generally associated with the sound and are therefore *sufficient to identify it – i.e.* the entire musical instrument, firetruck and helicopter – our method focuses on the parts that exhibit some motion – *i.e.* the player's hands, the firetruck siren and the helicopter's rotor – that *modify or create sound* and are necessary to solve the much more challenging synchronization task.

sequently, our proposed architecture sets new *state-of-the-art* results on LRS2 and LRS3, and provides baselines for general sound audio-visual synchronization.

Acknowledgements

This work is supported by the UK EPSRC CDT in Autonomous Intelligent Machines and Systems, the Oxford-Google DeepMind Graduate Scholarship, the Google PhD Fellowship, and EPSRC Programme Grants VisualAI EP/T028572/1.

6.A Implementation details

Here, we describe the architecture details, and hyper-parameters used during training.

Input Features. We follow previous works and use an input of $5 \sim 15$ frames in LRS2 and LRS3 extracted at 25 FPS, and use $5 \sim 30$ frames in VGG-Sound Sync dataset with 5 FPS. Visual frames are resized to a $224 \times 224 \times 3 \times T$ tensor without cropping, indicating height, width, channel, frames. The visual feature map before max-pool/reshape has a dimension of 14×14. For the visual stream, we use a ResNet18 2D+3D as backbone. For instance, an input of $224 \times 224 \times 3 \times 5$ would result the output visual feature a dimension of $14 \times 14 \times 512 \times 5$. Audio spectrograms are extracted using an FFT with window size of 320 and hop length 40. A lightweight VGG-M is then used to process the audio spectrogram giving tensors of dimension $512 \times T$. Encoders are initialised from scratch.

Training hyperparameters. We use a learning rate of $1e^{-4}$, Adam optimiser, trained for 100 epochs on 2 P40 GPUs.

6.B Robustness test

The aim of this experiment is to mimic real-world scenarios, as sound and its corresponding sound source might not happen at the same time. We mask random n frame length of input for one or both modalities, *i.e.*replace video frames or audio segments with zeros, we show results here when more than 1 frame is masked. With short input sequence, *e.g.*5 frames, the performance drops significantly as the mask length increases, *e.g.*3 frames, however, as the input length increases, our model shows robust performance at 15 frames.

6.C synchronization on general sound classes

Human tolerance on general classes. Here, we analyse the synchronization tolerance on general classes by manually creating misaligned audio-visual clips and asking human observers to verify whether the given clip is synchronized. Specifically, we randomly sample

Mask modality	# Mask Length	Clip Length in frames (seconds)					
		5(0.2s)	7(0.28s)	9(0.36s)	11(0.44s)	13(0.52s)	15(0.6s)
Audio	1 Frame	73.1	85.3	92.6	96.1	98.0	99.2
Visual	1 Frame	76.5	87.3	93.4	96.9	98.2	99.3
Both	1 Frame	71.7	84.0	91.2	95.6	97.7	99.1
Both	2 Frames	66.8	83.2	91.0	95.5	97.5	99.1
Both	3 Frames	56.1	79.6	90.3	95.2	97.2	99.0

Table 6.6: **Robustness test on LRS2.** As the input sequence length increases, even when we mask more frames, the performance remains robust.

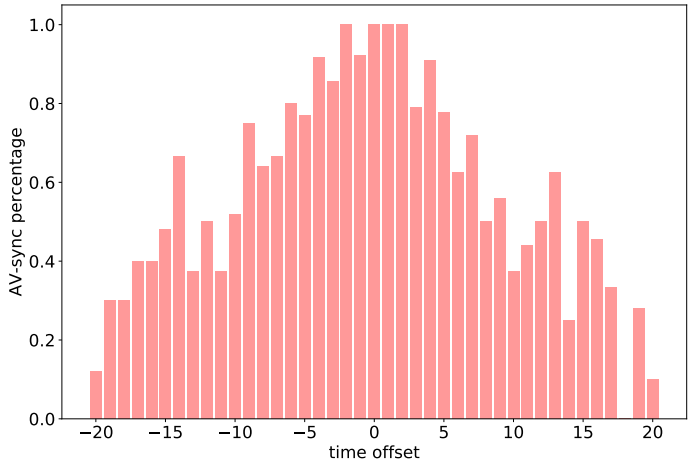


Figure 6.5: Proportions of videos considered to be syncd by a manual observer, Histogram demonstrating if the object-sync error is detectable for different offset values.

500 videos from all classes (160 classes) in VGG-Sound Sync. The misaligned audio-visual pairs are then generated by randomly choosing a offset within ± 20 frames (0.8s). During manual verification, we create 25 audio-visual pairs for each offset, with 1000 audio-visual pairs in total. As shown in Figure 6.5, all sample videos are manually verified as synchronized at time offset 0, indicating the downloaded videos contain high-quality audio-visual synchronization naturally. In addition, an error below ± 5 frames (0.2s) is indistinguishable for human (Approximately, 90% of the videos are verified as synchronized within ± 5 frames). Furthermore, the proportions of videos considered to be synchronized decrease significantly beyond such offsets. Therefore, we allow a prediction offset of up to ± 5 frames during evaluation for general sound classes.

Per-class synchronization accuracy. Please refer to Figure 6.6 for per class accuracy in VGG-Sound Sync dataset. We list the 160 classes we select from VGG-Sound in the end of this document.

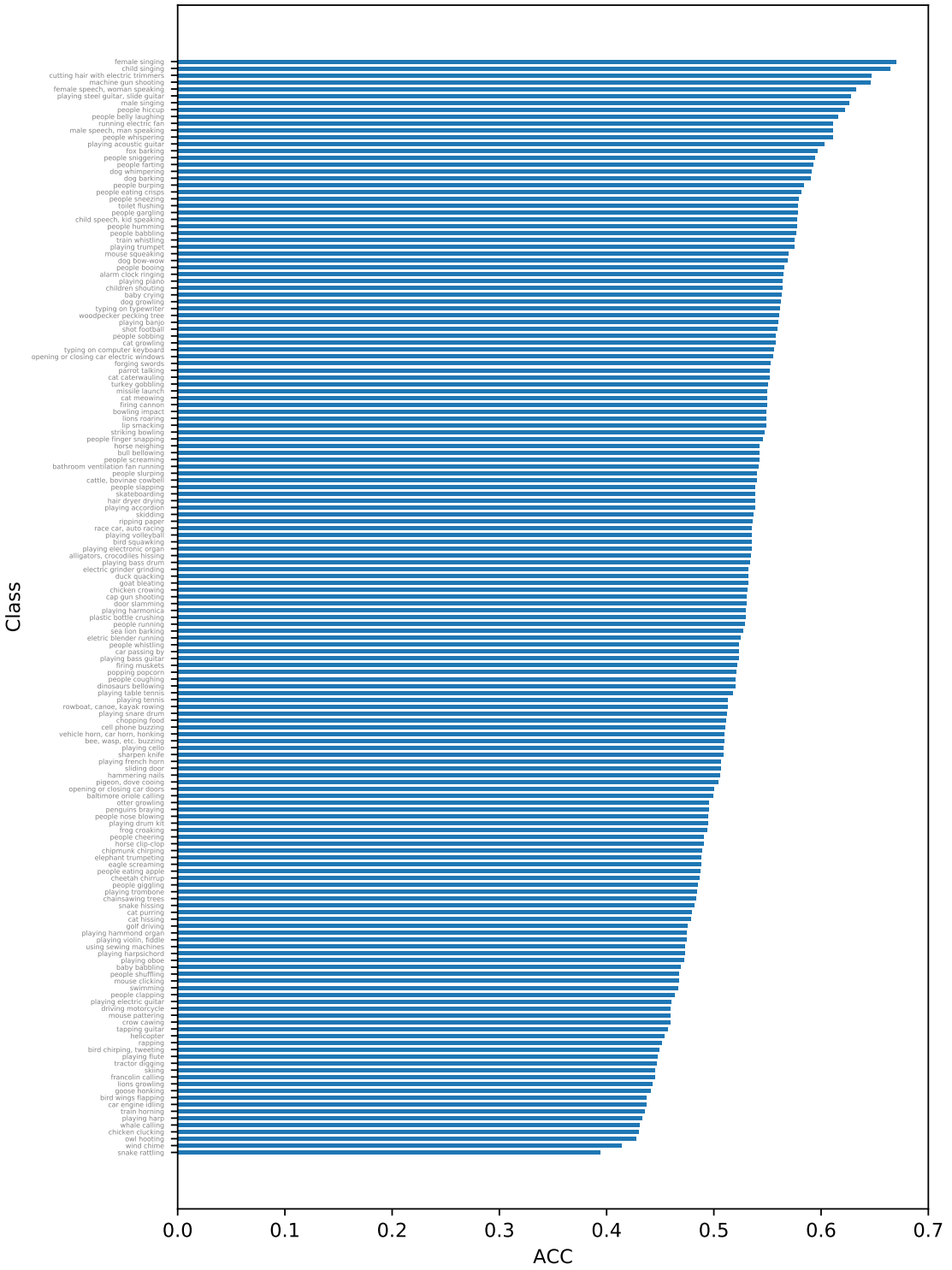
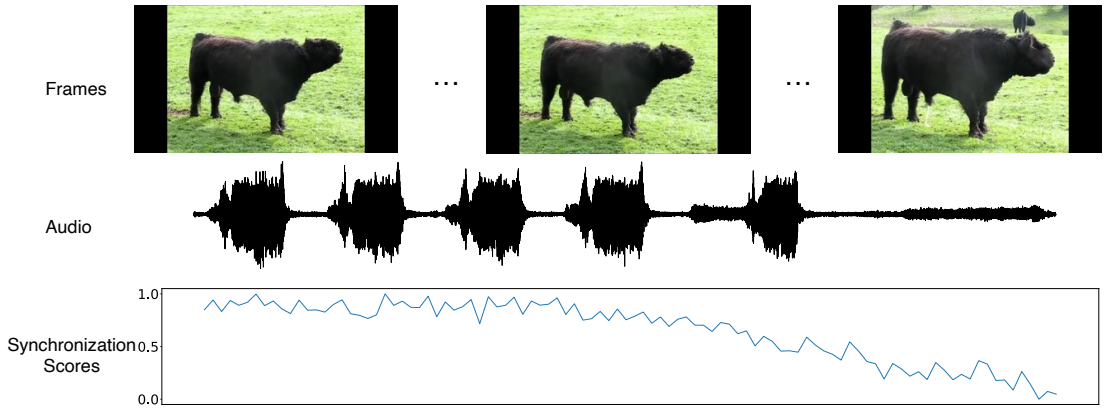


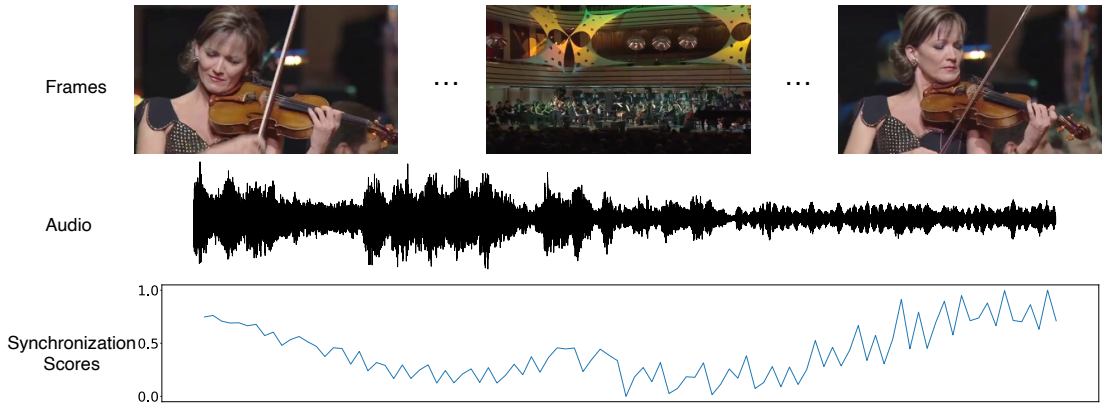
Figure 6.6: Per-class accuracy on VGG-Sound Sync.

Visualising the temporal synchronization. In order to understand the challenges in synchronizing general sound, we show video frames, audio and the model predicted synchronization score from top to bottom, in Figure 6.7. Specifically, we input 6s of audio and the corresponding frames (5 fps) with the same timestamp, to generate a synchronization score using our AVST_{dec} model. The bottom synchronization score figure is then generated using a moving window with stride of 1 frame. Each video is 10s long in Figure 6.7.

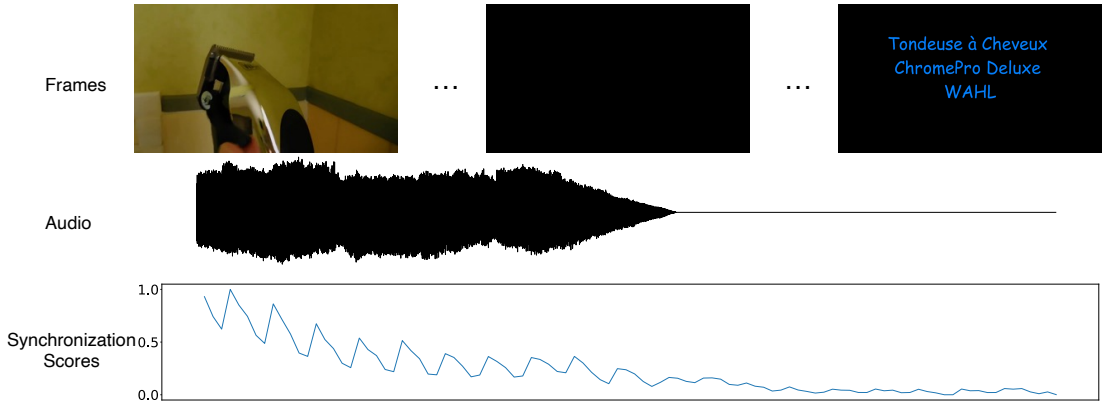
Ideally, we would like to see the scores being always around 1.0, however, in practise, synchronization of general sound can be challenging due to the following reasons: 1), missing audio information, as shown in Figure 6.7(a), the bull makes sound only for the first few seconds, resulting in weak synchronization in the later period. 2), missing visual information, in Figure 6.7(b), the violin player is not present for the entire clip, causing low synchronization scores. 3), missing information of both modalities. The last example in Figure 6.7(c) contains random frames with no sound; this severely decreases the synchronization scores. These hard cases suggest that experimenting on even longer input sequences may be beneficial, we therefore plan to investigate this in future work.



(a) **Bull bellowing.** In this case, the bull makes sound only in the first few seconds, although visual information exist over the entire clip, the missing audio causes the low score.



(b) **Playing violin, fiddle.** On the other hand, this example contains violin sound throughout the video clip, however, the violin player is not present all the time, causing the low score.



(c) **Electrical trimmer.** In this example, the last few seconds contain no information for both modalities, therefore the synchronization score is extremely low.

Figure 6.7: Synchronization scores along temporal axis. We analyse the hard cases which contain missing information of audio/video/both.

6.D Attention heatmaps visualisation

We show more image heatmap results from LRS3 in Figure 6.8 and results from VGG-Sound Sync in Figure 6.9. Both models are trained using AVST_{dec} model. The heatmaps are created by normalizing the attention matrix between the audio feature and visual feature in AVST_{dec} model. In Figure 6.8, we can accurately localise the human mouth. In Figure 6.9, we localise the sound source in various classes, *e.g.* running electric fan, sharpen knife, cat meowing, etc. Note, rather than localise the entire sounding object, *e.g.*piano, sewing machine, our model focus on the interaction points between the objects, *e.g.*hands, sewing machine needle.

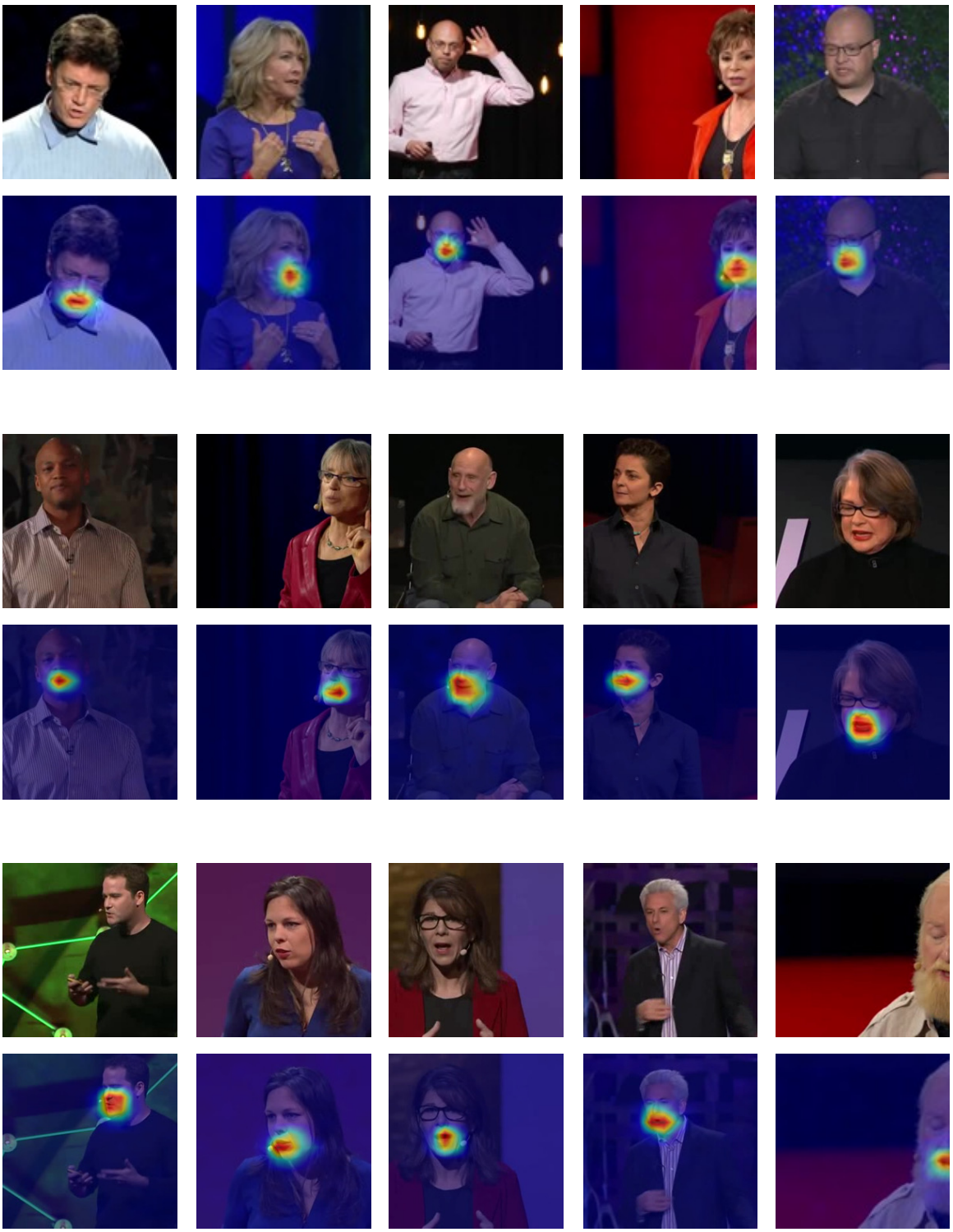


Figure 6.8: Attention heatmap visualisations on LRS3 dataset.

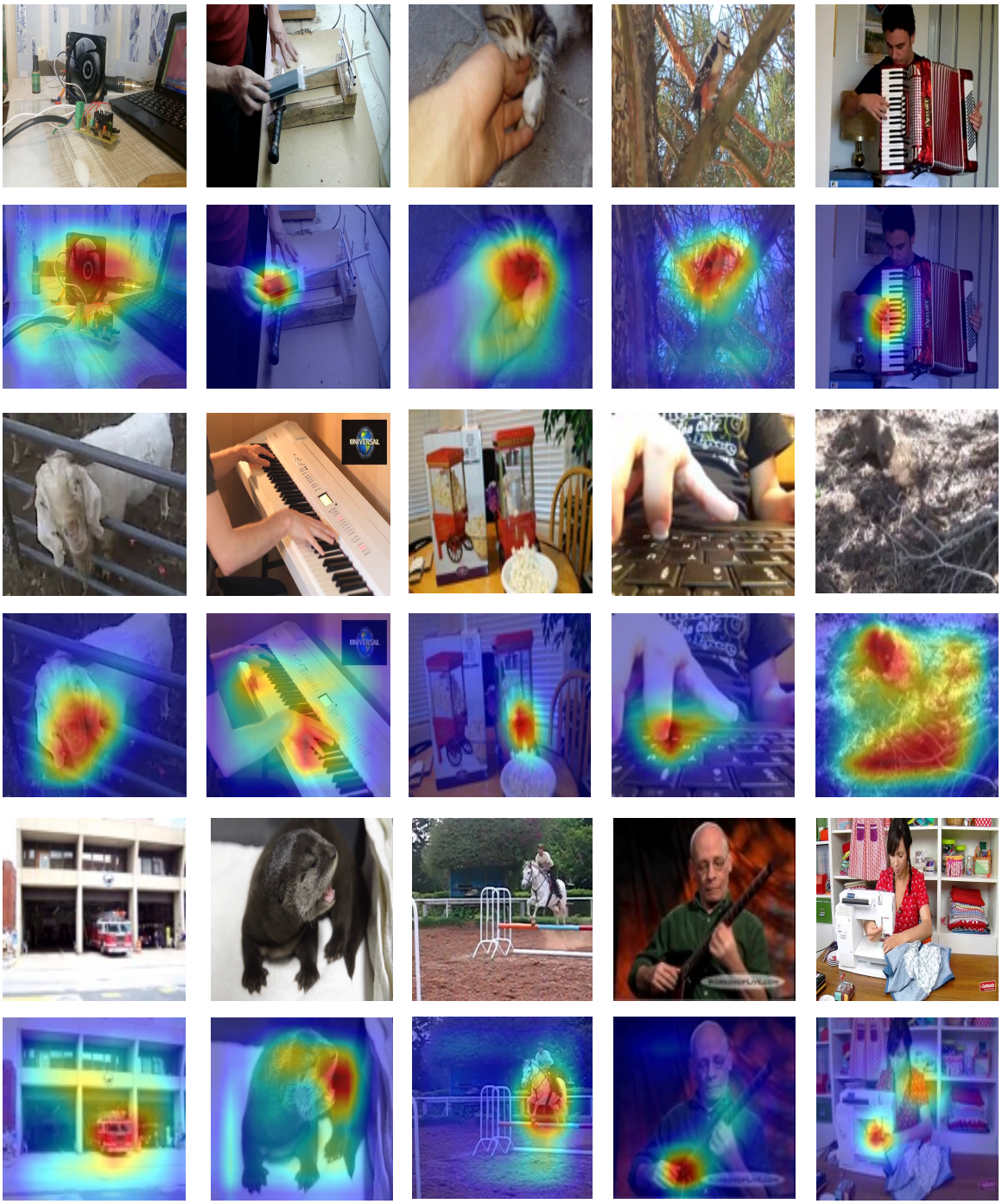


Figure 6.9: Attention heatmap visualisations on VGG-Sound Sync dataset.

Class list

1. female singing
2. child singing
3. cutting hair with electric trimmers
4. machine gun shooting
5. female speech, woman speaking
6. playing steel guitar, slide guitar
7. male singing
8. people hiccup
9. people belly laughing
10. running electric fan
11. male speech, man speaking
12. people whispering
13. playing acoustic guitar
14. fox barking
15. people sniggering
16. people farting
17. dog whimpering
18. dog barking
19. people burping
20. people eating crisps
21. people sneezing
22. toilet flushing
23. people gargling
24. child speech, kid speaking
25. people humming
26. people babbling
27. train whistling
28. playing trumpet
29. mouse squeaking
30. dog bow-wow
31. people booing
32. alarm clock ringing
33. playing piano
34. children shouting
35. baby crying
36. dog growling
37. typing on typewriter
38. woodpecker pecking tree
39. playing banjo
40. shot football
41. people sobbing

42. cat growling
43. typing on computer keyboard
44. opening or closing car electric windows
45. forging swords
46. parrot talking
47. cat caterwauling
48. turkey gobbling
49. missile launch
50. cat meowing
51. firing cannon
52. bowling impact
53. lions roaring
54. lip smacking
55. striking bowling
56. people finger snapping
57. horse neighing
58. bull bellowing
59. people screaming
60. bathroom ventilation fan running
61. people slurping
62. cattle, bovinae cowbell
63. people slapping
64. skateboarding
65. hair dryer drying
66. playing accordion
67. skidding
68. ripping paper
69. race car, auto racing
70. playing volleyball
71. bird squawking
72. playing electronic organ
73. alligators, crocodiles hissing
74. playing bass drum
75. electric grinder grinding
76. duck quacking
77. goat bleating
78. chicken crowing
79. cap gun shooting
80. door slamming
81. playing harmonica
82. plastic bottle crushing
83. people running

84. sea lion barking
85. eletric blender running
86. people whistling
87. car passing by
88. playing bass guitar
89. firing muskets
90. popping popcorn
91. people coughing
92. dinosaurs bellowing
93. playing table tennis
94. playing tennis
95. rowboat, canoe, kayak rowing
96. playing snare drum
97. chopping food
98. cell phone buzzing
99. vehicle horn, car horn, honking
100. bee, wasp, etc. buzzing
101. playing cello
102. sharpen knife
103. playing french horn
104. sliding door
105. hammering nails
106. pigeon, dove cooing
107. opening or closing car doors
108. baltimore oriole calling
109. otter growling
110. penguins braying
111. people nose blowing
112. playing drum kit
113. frog croaking
114. people cheering
115. horse clip-clop
116. chipmunk chirping
117. elephant trumpeting
118. eagle screaming
119. people eating apple
120. cheetah chirrup
121. people giggling
122. playing trombone
123. chainsawing trees
124. snake hissing
125. cat purring
126. cat hissing

127. golf driving
128. playing hammond organ
129. playing violin, fiddle
130. using sewing machines
131. playing harpsichord
132. playing oboe
133. baby babbling
134. people shuffling
135. mouse clicking
136. swimming
137. people clapping
138. playing electric guitar
139. driving motorcycle
140. mouse pattering
141. crow cawing
142. tapping guitar
143. helicopter
144. rapping
145. bird chirping, tweeting
146. playing flute
147. tractor digging
148. skiing
149. francolin calling
150. lions growling
151. goose honking
152. bird wings flapping
153. car engine idling
154. train horning
155. playing harp
156. whale calling
157. chicken clucking
158. owl hooting
159. wind chime
160. snake rattling

Chapter 7

Conclusion

In this thesis, we have discussed a number of methods for training deep neural networks without using manual annotations including correcting freely obtained noisy labels or self-supervision from multimodal cues. Here we first summarize the main contributions and highlight the achievements in this thesis (Section 7.1). Next, we discuss potential directions and notable recent publications (Section 7.2).

7.1 Achievements and Impact

Automatic label correction. In Chapter 3, we proposed to correct geometric noisy labels automatically using three key ideas: a consistency loss which guides the learning even without knowing the ground truth, a self-supervised loss that automatically generates the training data, and finally an inductive method which corrects multiple objects in a image by building a spatial memory mask indicating the past predictions, forming an auto-regressive model. We demonstrated the model performance by showing superior alignment results to previous methods on standard benchmarks. Finally, we released a new satellite image dataset - railway tracks dataset including 35k images with binary mask annotations indicating the position of the railway tracks.

Since the publication of our original conference paper, MapRepair [Zorzi et al., 2020] was proposed to extend our method to transform misaligned footprints and, at the same time, detect obsolete footprints and segment constructions that lack annotations. Jiang et al. [2021] on the other hand, further improved our method by correcting the registration noise

in the vector representation to guarantee label continuity. Similarly, class labels are represented by geometric shapes (*e.g.*, spatial points, polylines) before refinement in Jiang et al. [2020]. Furthermore, the idea of using inductive models and self-supervised learning has been adopted by a number of works in several domains such as Human pose [Le et al., 2020] and video segmentation [Zhu et al., 2020].

VGG-Sound. In Chapter 4, we introduced a large-scale audio-visual dataset called VGG-Sound which contains more than 200k videos and 309 classes "in the wild". We achieved the state-of-the-art audio recognition performance using models trained on VGG-Sound. More importantly, a automated and scalable pipeline for collecting the audio-visual dataset was released.

The VGG-Sound dataset has enabled development in a number of audio-visual domains, such as representation learning [Asano et al., 2020a; Feng et al., 2020], counting [Zhang et al., 2021], cross-modal retrieval [Oncescu et al., 2021], separation [Zhu and Rahtu, 2021b], localization [Chen et al., 2021a; Sanguineti et al., 2021], detection [Afouras et al., 2021] and audio recognition [Kazakos et al., 2021]. In addition, the automatic pipeline has prompted others to develop similar ways for collecting a dataset using audio-visual correspondence, Lee et al. [2021a] propose to curate an audio-visual dataset by maximizing mutual information between audio and visual channels in videos.

Localize sound source in visual scene. We investigated the problem of unsupervised visual sound source localization in Chapter 5. First, a new large-scale benchmark, VGG-Sound Source was released with 5k video clips spanning 220 classes. This is 20x larger than the previous audio-visual localization benchmark. Furthermore, we also developed a general method for explicitly mining the hard negative regions in the image during training using Tri-maps, significantly surpassing the previous works on standard benchmarks.

To assist the community and enable further research, we have open-sourced the code implementation and pretrained models for "localizing visual sound". Our work has spurred various followup works in audio-visual domain. For instance, Zhu and Rahtu [2021b] perform both visual sound separation and localization via a efficient three-stream framework

including Visual frame, Slow spectrogram, and Fast spectrogram. Afouras et al. [2021] on the other hand, consider to localize and classify the sound source in a visual scene by using self-labels and self-boxes generated during audio-visual correspondence learning.

Synchronize audible and visual modality. In Chapter 6, we extended the traditional audio-visual synchronization problem in human speech and introduced a new general class audio-visual synchronization benchmark. To the best of our knowledge, we are the first to formalize the self-supervised task in this manner. To learn this representation, we made use of vast amounts of video data and cross-modal self-supervision. Moreover, we proposed a novel transformer-based architecture to deal with long sequence sensory inputs. Our method outperforms the previous state-of-the-art by a significant margin on LRS2 and LRS3, and provide baselines for general sound audio visual synchronization. We expect this work to inspire future research in how to fuse the cross-modal information without using simple late-stage fusion of representations across modalities.

7.2 Future work

We conclude the thesis by highlighting the scope for possible extensions which includes thoughts on broader objectives and more abstract ideas.

Dataset. While our VGG-Sound dataset is of sufficient diversity to enable generalization to learn "in the wild", it can be further improved. A limitation of VGG-Sound is that it does not contain temporally-precise annotations, *i.e.*, fine-grained second-level annotations in the video clips. Recently, Hershey et al. [2021] extended and adapted AudioSet [Gemmeke et al., 2017] to create a strong-labeled training subset of 67k clips. Fine-tuning with a mix of weak (clip-level) and strongly (second-level) labeled data can substantially improve audio classifier performance. A bootstrap strategy could be applied to automatically generate strong labels for VGG-Sound based on state-of-the-art audio classifier. Another line of extension could consider constructing a new version of VGG-Sound with more classes and larger size. [Lee et al., 2021a] use an interesting pipeline to curate 100M video clips with competitive audio-visual correspondence (69% after human evaluation).

Audio-visual fusion. While many machine perception models bind cross-modal representations using middle fusion [Wang et al., 2020b; Seichter et al., 2020], or late fusion methods [Wang et al., 2020a], a transformed-based architecture is proposed in Chapter 6 to handle configurations of different modalities, the self attention operation of transformers can provide a natural mechanism to connect multimodal signals. However, our model is constrained by the long sequence and deep models, which suffers from the quadratic scaling problem in the all-to-all attention mechanism in the vanilla transformer. One promising direction is proposed by Jaegle et al. [2021], who introduce a small set of latent units that forms an attention bottleneck, allowing it to scale to handle very large inputs. Nagrani et al. [2021] further improve this idea with modality fusion at multiple layers.

General audio-visual learning. In Chapter 4 - 6, we proposed new benchmarks and methods for general audio-visual tasks such as localization and synchronization. However, many other audio-visual applications including audio-visual separation and generation, have not yet been solved in a wider domain with general classes. In fact, audio-visual learning in the open world remains challenging. Association between visual/audio features and disambiguation between instances of the same class are pivotal to achieve this goal.

Bibliography

Triantafyllos Afouras, Joon Son Chung, and Andrew Zisserman. The conversation: Deep audio-visual speech enhancement. In *INTERSPEECH*, 2018a.

Triantafyllos Afouras, Joon Son Chung, and Andrew Zisserman. Lrs3-ted: a large-scale dataset for visual speech recognition. *arXiv preprint arXiv:1809.00496*, 2018b.

Triantafyllos Afouras, Joon Son Chung, Andrew Senior, Oriol Vinyals, and Andrew Zisserman. Deep audio-visual speech recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019a.

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

Triantafyllos Afouras, Joon Son Chung, and Andrew Zisserman. Asr is all you need: Cross-modal distillation for lip reading. In *International Conference on Acoustics, Speech, and Signal Processing*, 2020a.

Triantafyllos Afouras, Andrew Owens, Joon Son Chung, and Andrew Zisserman. Self-supervised learning of audio-visual objects from video. In *Proceedings of the European Conference on Computer Vision*, 2020b.

Triantafyllos Afouras, Yuki M. Asano, Francois Fagan, Andrea Vedaldi, and Florian Metze. Self-supervised object detection from audio-visual correspondence. *arXiv preprint arXiv:2104.06401*, 2021.

Samuel Albanie, Gül Varol, Liliane Momeni, Triantafyllos Afouras, Joon Son Chung, Neil Fox, and Andrew Zisserman. BSL-1K: Scaling up co-articulated sign language recognition using mouthing cues. In *Proceedings of the European Conference on Computer Vision*, 2020.

Rasha Alshehhi, Prashanth Reddy Marpu, Wei Lee Woon, and Mauro Dalla Mura. Simultaneous extraction of roads and buildings in remote sensing imagery with convolutional neural networks. *ISPRS Journal of Photogrammetry and Remote Sensing*, 2017.

Humam Alwassel, Dhruv Mahajan, Bruno Korbar, Lorenzo Torresani, Bernard Ghanem, and Du Tran. Self-supervised learning by cross-modal audio-video clustering. *Advances in Neural Information Processing Systems*, 2019.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *Proceedings of the International Conference on Computer Vision*, December 2015.

Relja Arandjelovic and Andrew Zisserman. Look, listen and learn. In *Proceedings of the International Conference on Computer Vision*, 2017.

Relja Arandjelovic and Andrew Zisserman. Objects that sound. In *Proceedings of the European Conference on Computer Vision*, 2018.

Relja Arandjelovic, Petr Gronat, Akihiko Torii, Tomas Pajdla, and Josef Sivic. NetVLAD: CNN architecture for weakly supervised place recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lui, and Cordelia Schmid. Vivit: A video vision transformer. *ArXiv*, abs/2103.15691, 2021.

Yuki M. Asano, Mandela Patrick, Christian Rupprecht, and Andrea Vedaldi. Labelling unlabelled videos from scratch with multi-modal self-supervision. In *Advances in Neural Information Processing Systems*, 2020a.

Yuki M Asano, Christian Rupprecht, and Andrea Vedaldi. Self-labelling via simultaneous clustering and representation learning. In *International Conference on Learning Representations*, 2020b.

Yusuf Aytar, Carl Vondrick, and Antonio Torralba. Soundnet: Learning sound representations from unlabeled video. In *Advances in Neural Information Processing Systems*, 2016.

Yoshua Bengio and James Bergstra. Slow, decorrelated features for pretraining complex cell-like networks. In *Advances in Neural Information Processing Systems*, 2009.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the International Conference on Machine Learning*, 2009.

Leonard Berrada, Andrew Zisserman, and M. Pawan Kumar. Smooth loss functions for deep top-k classification. In *Proceedings of the International Conference on Learning Representations*, 2018.

Marcelo Bertalmio, Guillermo Sapiro, Vincent Caselles, and Coloma Ballester. Image inpainting. In *Proceedings of the Annual Conference on Computer Graphics and Interactive Techniques*, 2000.

Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? In *Proceedings of the International Conference on Machine Learning*, 2021.

Richard W. Brislin. Back-translation for cross-cultural research. *Journal of Cross-Cultural Psychology*, 1970.

Gabriel J. Brostow, Jamie Shotton, Julien Fauqueur, and Roberto Cipolla. Segmentation and recognition using structure from motion point clouds. In *Proceedings of the European Conference on Computer Vision*, 2008.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, 2020.

Susanne Burger, Qin Jin, Peter F. Schulam, and Florian Metze. Noisemes: Manual Annotation of Environmental Noise in Audio Streams. 2012.

Jonathon Cai, Richard Shin, and Dawn Xiaodong Song. Making neural programming architectures generalize via recursion. In *Proceedings of the International Conference on Learning Representations*, 2017.

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *Proceedings of the European Conference on Computer Vision*, 2020.

Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *Proceedings of the European Conference on Computer Vision*, 2018.

João Carreira, Pulkit Agrawal, Katerina Fragkiadaki, and Jitendra Malik. Human pose estimation with iterative error feedback. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

João Carreira, Eric Noland, Andras Banki-Horvath, Chloe Hillier, and Andrew Zisserman. A short note about kinetics-600. *arXiv preprint arXiv:1808.01340*, 2018.

Anna Llagostera Casanovas and Andrea Cavallaro. Audio-visual events for multi-camera synchronization. *Multimedia Tools and Applications*, 2014.

Ken Chatfield, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Return of the devil in the details: Delving deep into convolutional nets. In *Proceedings of the British Machine Vision Conference*, 2014.

Honglie Chen, Weidi Xie, Andrea Vedaldi, and Andrew Zisserman. Vgg-sound: A large-scale audio-visual dataset. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2020a.

Honglie Chen, Weidi Xie, Triantafyllos Afouras, Arsha Nagrani, Andrea Vedaldi, and Andrew Zisserman. Localizing visual sounds the hard way. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2021a.

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

Lele Chen, Zhiheng Li, Ross K Maddox, Zhiyao Duan, and Chenliang Xu. Lip movements generation at a glance. In *Proceedings of the European Conference on Computer Vision*, 2018.

Lele Chen, Ross K Maddox, Zhiyao Duan, and Chenliang Xu. Hierarchical cross-modal talking face generation with dynamic pixel-wise loss. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *Proceedings of the International Conference on Machine Learning*, 2020b.

Xie Chen, Yu Wu, Zhenghao Wang, Shujie Liu, and Jinyu Li. Developing real-time streaming transformer transducer for speech recognition on large-scale dataset. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2021b.

Xinlei Chen and Abhinav Gupta. Spatial memory for context reasoning in object detection. In *Proceedings of the International Conference on Computer Vision*, 2017.

Ying Cheng, Ruize Wang, Zhihao Pan, Rui Feng, and Yuejie Zhang. Look, listen, and attend: Co-attention network for self-supervised audio-visual representation learning. *Proceedings of the ACM Multimedia Conference*, 2020.

Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. Generating long sequences with sparse transformers. *ArXiv*, abs/1904.10509, 2019.

Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. In *Conference on Empirical Methods in Natural Language Processing*, 2014.

Keunwoo Choi, Gyrgy Fazekas, Mark Sandler, and Kyunghyun Cho. Convolutional recurrent neural networks for music classification. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2017.

Sumit Chopra, Raia Hadsell, and Yann LeCun. Learning a similarity metric discriminatively, with application to face verification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2005.

Krzysztof Marcin Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamas Sarlos, Peter Hawkins, Jared Quincy Davis, Afroz Mohiuddin, Lukasz Kaiser, David Benjamin Belanger, Lucy J Colwell, and Adrian Weller. Rethinking attention with performers. In *International Conference on Learning Representations*, 2021.

Yung-Yu Chuang, Aseem Agarwala, Brian Curless, David H. Salesin, and Richard Szeliski. Video matting of complex scenes. *ACM Trans. Graph*, 2002.

Joon Son Chung and Andrew Zisserman. Lip reading in the wild. In *Proceedings of the Asian Conference on Computer Vision*, 2016a.

Joon Son Chung and Andrew Zisserman. Out of time: automated lip sync in the wild. In *Workshop on Multi-view Lip-reading, ACCV*, 2016b.

Joon Son Chung and Andrew Zisserman. Signs in time: Encoding human motion as a temporal image. In *Workshop on Brave New Ideas for Motion Representations, ECCV*, 2016c.

Joon Son Chung and Andrew Zisserman. Lip reading in profile. In *Proceedings of the British Machine Vision Conference*, 2017.

Joon Son Chung, Amir Jamaludin, and Andrew Zisserman. You said that? In *Proceedings of the British Machine Vision Conference*, 2017a.

Joon Son Chung, Andrew Senior, Oriol Vinyals, and Andrew Zisserman. Lip reading sentences in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017b.

Joon Son Chung, Arsha Nagrani, and Andrew Zisserman. VoxCeleb2: Deep speaker recognition. In *INTERSPEECH*, 2018.

Soo-Whan Chung, Joon Son Chung, and Hong-Goo Kang. Perfect match: Improved cross-modal embeddings for audio-visual synchronization. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2019.

Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

Antonio Criminisi, Patrick Prez, and Kentaro Toyama. Region filling and object removal by exemplar-based image inpainting. *IEEE Transactions on Image Processing*, 2004.

Kim Dahun, Donghyeon Cho, Donggeun Yoo, and Inso Kweon. Learning image representations by completing damaged jigsaw puzzles. In *Proceedings of the IEEE Conference on Applications of Computer Vision*, 2018.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V Le, and Ruslan Salakhutdinov. Transformer-XL: Attentive language models beyond a fixed-length context. In *Association for Computational Linguistics*, 2019.

Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2005.

Vansh Dassani, Jon Bird, and Dave Cliff. Automated composition of picture-synched music soundtracks for movies. In *European Conference on Visual Media Production*, 2019.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, K. Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 2019.

Carl Doersch, Abhinav Gupta, and Alexei A. Efros. Unsupervised visual representation learning by context prediction. In *Proceedings of the International Conference on Computer Vision*, 2015.

Piotr Dollar, Christian Wojek, Bernt Schiele, and Pietro Perona. Pedestrian detection: A benchmark. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

Jeff Donahue, Lisa Anne Hendricks, Marcus Rohrbach, Subhashini Venugopalan, Sergio Guadarrama, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *arXiv preprint arXiv:1411.4389*, 2014.

Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Learning a deep convolutional network for image super-resolution. In *Proceedings of the European Conference on Computer Vision*, 2014.

Linhao Dong, Shuang Xu, and Bo Xu. Speech-transformer: A no-recurrence sequence-to-sequence model for speech recognition. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2018.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

John Driver. Enhancement of selective listening by illusory mislocation of speech sounds due to lip-reading. *Nature*, 1996.

Abhishek Dutta and Andrew Zisserman. The via annotation software for images, audio and video. In *Proceedings of the ACM Multimedia Conference*, 2019.

Joshua P. Ebeneze, Yongjun Wu, Hai Wei, Sriram Sethuraman, and Zongyi Liu. Detection of audio-video synchronization errors via event detection. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2021.

Gerald Edelman and Joseph Gally. Reentry: a key mechanism for integration of brain function. *Frontiers in Integrative Neuroscience*, 2013.

Gerald M. Edelman. *Neural Darwinism : the theory of neuronal group selection / Gerald M. Edelman*. Basic Books New York, 1987.

Gamaleldin F. Elsayed, Dilip Krishnan, Hossein Mobahi, Kevin Regan, and Samy Bengio. Large margin deep networks for classification. In *Advances in Neural Information Processing Systems*, 2018.

Ariel Ephrat, Tavi Halperin, and Shmuel Peleg. Improved speech reconstruction from silent video. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2017.

Ariel Ephrat, Inbar Mosseri, Oran Lang, Tali Dekel, Kevin Wilson, Avinatan Hassidim, William T Freeman, and Michael Rubinstein. Looking to listen at the cocktail party: a speaker-independent audio-visual model for speech separation. *ACM Transactions on Graphics*, 37(4):112, 2018.

Mark Everingham, Luc Van Gool, Chris K. I. Williams, John Winn, and Andrew Zisserman. The PASCAL Visual Object Classes (VOC) challenge. *International Journal of Computer Vision*, 2010.

Mark Everingham, S. M. Ali Eslami, Luc Van Gool, Chris K. I. Williams, John Winn, and Andrew Zisserman. The pascal visual object classes challenge: A retrospective. *International Journal of Computer Vision*, 2015.

Bo Fan, Lijuan Wang, Frank K. Soong, and Lei Xie. Photo-real talking head with deep bidirectional lstm. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2015.

Ali Farhadi, Mohsen Hejrati, Mohammad Amin Sadeghi, Peter Young, Cyrus Rashtchian, Julia Hockenmaier, and David Forsyth. Every picture tells a story: Generating sentences from images. In *Proceedings of the European Conference on Computer Vision*, 2010.

Zishun Feng, Ming Tu, Rui xia, Yuxuan Wang, and Ashok Krishnamurthy. Self-supervised audio-visual representation learning for in-the-wild videos. In *IEEE International Conference on Big Data*, 2020.

Basura Fernando, Hakan Bilen, Efstratios Gavves, and Stephen Gould. Self-supervised video representation learning with odd-one-out networks. In *Proceedings of the International Conference on Computer Vision*, 2017.

Chelsea Finn, Ian Goodfellow, and Sergey Levine. Unsupervised learning for physical interaction through video prediction. In *Advances in Neural Information Processing Systems*, 05 2016.

John W Fisher III, Trevor Darrell, William T Freeman, and Paul A Viola. Learning joint statistical models for audio-visual fusion and segregation. In *Advances in Neural Information Processing Systems*, 2000.

Pasquale Foggia, Nicolai Petkov, Alessia Saggese, Nicola Strisciuglio, and Mario Vento. Reliable detection of audio events in highly noisy environments. *Pattern Recognition Letters*, 2015.

Eduardo Fonseca, Jordi Pons, Xavier Favory, Frederic Font, Dmitry Bogdanov, Andrés Ferraro, Sergio Oramas, Alastair Porter, and Xavier Serra. Freesound datasets: a platform for the creation of open audio datasets. In *ISMIR*, 2017.

Cdric Fvotte, Nancy Bertin, and Jean-Louis Durrieu. Nonnegative matrix factorization with the itakura-saito divergence: With application to music analysis. *Neural Computation*, 21(3):793–830, 2009. doi: 10.1162/neco.2008.04-08-771.

Peter Fldik. Learning invariance from transformation sequences. *Neural Computation*, 3: 194–200, 06 1991. doi: 10.1162/neco.1991.3.2.194.

Chuang Gan, Hang Zhao, Peihao Chen, David Cox, and Antonio Torralba. Self-supervised moving vehicle tracking with stereo sound. In *Proceedings of the International Conference on Computer Vision*, 2019.

Chuang Gan, Deng Huang, Peihao Chen, Joshua B. Tenenbaum, and Antonio Torralba. Foley music: Learning to generate music from videos. In *Proceedings of the European Conference on Computer Vision*, 2020a.

Chuang Gan, Deng Huang, Hang Zhao, Joshua B. Tenenbaum, and Antonio Torralba. Music gesture for visual sound separation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2020b.

Haoyuan Gao, Junhua Mao, Jie Zhou, Zhiheng Huang, Lei Wang, and Wei Xu. Are you talking to a machine? dataset and methods for multilingual image question answering. In *Advances in Neural Information Processing Systems*, 05 2015.

Ruohan Gao and Kristen Grauman. 2.5d visual sound. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

Ruohan Gao, Rogério Schmidt Feris, and Kristen Grauman. Learning to separate object sounds by watching unlabeled video. In *Proceedings of the European Conference on Computer Vision*, 2018.

Pablo Garrido, Levi Valgaerts, Hamid Sarmadi, Ingmar Steiner, Kiran Varanasi, Patrick Pérez, and Christian Theobalt. Vdub: Modifying face video of actors for plausible visual alignment to a dubbed audio track. 2015.

William W. Gaver. What in the world do we hear?: An ecological approach to auditory event perception. *Ecological Psychology*, 5(1):1–29, 1993.

Jort F. Gemmeke, Daniel P. W. 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 *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2017.

Aritra Ghosh, Himanshu Kumar, and P. S. Sastry. Robust loss functions under label noise for deep neural networks. In *AAAI*, 2017.

Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. In *Proceedings of the International Conference on Learning Representations*, 2018.

Nicolas Girard, Guillaume Charpiat, and Yuliya Tarabalka. Aligning and updating cadaster maps with aerial images by multi-task, multi-resolution deep learning. In *Proceedings of the Asian Conference on Computer Vision*, 2018.

Ross Girshick. Fast r-cnn. In *Proceedings of the International Conference on Computer Vision*, 2015.

Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

Yuan Gong, Yu-An Chung, and James Glass. Ast: Audio spectrogram transformer. *arXiv preprint arXiv:2104.01778*, 2021.

Ross Goroshin, Joan Bruna, Jonathan Tompson, David Eigen, and Yann LeCun. Unsupervised learning of spatiotemporally coherent metrics. In *Proceedings of the International Conference on Computer Vision*, 2015.

Anmol Gulati, Chung-Cheng Chiu, James Qin, Jiahui Yu, Niki Parmar, Ruoming Pang, Shibo Wang, Wei Han, Yonghui Wu, Yu Zhang, and Zhengdong Zhang. Conformer: Convolution-augmented transformer for speech recognition. In *Interspeech*, 2020a.

Anmol Gulati, Chung-Cheng Chiu, James Qin, Jiahui Yu, Niki Parmar, Ruoming Pang, Shibo Wang, Wei Han, Yonghui Wu, Yu Zhang, and Zhengdong Zhang, editors. *Conformer: Convolution-augmented Transformer for Speech Recognition*, 2020b.

Qipeng Guo, Xipeng Qiu, Xiangyang Xue, and Zheng Zhang. Low-rank and locality constrained self-attention for sequence modeling. 2019.

Ankush Gupta, Andrea Vedaldi, and Andrew Zisserman. Inductive visual localization: Factorised training for superior generalisation. In *Proceedings of the British Machine Vision Conference*, 2018.

Tavi Halperin, Ariel Ephrat, and Shmuel Peleg. Dynamic temporal alignment of speech to lips. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2019.

Tengda Han, Weidi Xie, and Andrew Zisserman. Video representation learning by dense predictive coding. In *Workshop on Large Scale Holistic Video Understanding, ICCV*, 2019.

Tengda Han, Weidi Xie, and Andrew Zisserman. Memory-augmented dense predictive coding for video representation learning. In *Proceedings of the European Conference on Computer Vision*, 2020a.

Tengda Han, Weidi Xie, and Andrew Zisserman. Self-supervised co-training for video representation learning. In *Advances in Neural Information Processing Systems*, 2020b.

John Hansen, Ruhi Sarikaya, Umit Yapanel, and Bryan Pellom. Robust speech recognition in noise: an evaluation using the spine corpus. In *INTERSPEECH*, 2001.

David Harwath, Antonio Torralba, and James R. Glass. Unsupervised learning of spoken language with visual context. In *Advances in Neural Information Processing Systems*, 2016.

David Harwath, Adria Recasens, Dídac Surís, Galen Chuang, Antonio Torralba, and James Glass. Jointly discovering visual objects and spoken words from raw sensory input. In *Proceedings of the European Conference on Computer Vision*, 2018.

Simon Haykin and Zhe Chen. The cocktail party problem. *Neural Computation*, 17(9): 1875–1902, 2005. doi: 10.1162/0899766054322964.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020.

John Hershey and Michael Casey. Audio-visual sound separation via hidden markov models. In *Advances in Neural Information Processing Systems*, 2002.

John Hershey and Javier Movellan. Audio-vision: Locating sounds via audio-visual synchrony. In *Advances in Neural Information Processing Systems*, 1999.

John Hershey and Javier Movellan. Audio vision: Using audio-visual synchrony to locate sounds. In *Advances in Neural Information Processing Systems*, 2000.

Shawn Hershey, Sourish Chaudhuri, Daniel P. W. Ellis, Jort F. Gemmeke, Aren Jansen, R. Channing Moore, Manoj Plakal, Devin Platt, Rif A. Saurous, Bryan Seybold, Malcolm Slaney, Ron J. Weiss, and Kevin W. Wilson. CNN architectures for large-scale audio classification. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2017.

Shawn Hershey, Daniel P. W. Ellis, Eduardo Fonseca, Aren Jansen, Caroline Liu, R. Channing Moore, and Manoj Plakal. The benefit of temporally-strong labels in audio event classification. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2021.

Geoffrey Hinton, Li Deng, Dong Yu, George E. Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara N. Sainath, and Brian Kingsbury. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine*, 29(6):82–97, 2012. doi: 10.1109/MSP.2012.2205597.

Jonathan Ho, Nal Kalchbrenner, Dirk Weissenborn, and Tim Salimans. Axial attention in multidimensional transformers, 2020.

Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei A. Efros, and Trevor Darrell. Cycada: Cycle consistent adversarial domain adaptation. In *Proceedings of the International Conference on Machine Learning*, 2018.

Sungeun Hong, Woobin Im, and Hyun S. Yang. Cbvmr: Content-based video-music retrieval using soft intra-modal structure constraint. In *Proceedings of the ACM on International Conference on Multimedia Retrieval*, 2018.

Di Hu, Feiping Nie, and Xuelong Li. Deep multimodal clustering for unsupervised audio-visual learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June 2019.

Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

Jiuxiang Hu, Anshuman Razdan, John C. Femiani, Ming Cui, and Peter Wonka. Road network extraction and intersection detection from aerial images by tracking road footprints. *IEEE Transactions on Geoscience and Remote Sensing*, 2007.

Zhen Huang, Y.-C Cheng, Kehuang Li, V. Hautamki, and Chin-Hui Lee. A blind segmentation approach to acoustic event detection based on i-vector. In *INTERSPEECH*, 2013.

Jarmo Hurri and Aapo Hyvrinen. Simple-Cell-Like Receptive Fields Maximize Temporal Coherence in Natural Video. *Neural Computation*, 15(3):663–691, 03 2003.

Hamid Izadinia, Imran Saleemi, and Mubarak Shah. Multimodal analysis for identification and segmentation of moving-sounding objects. *IEEE Trans. Multimed.*, 2012.

Andrew Jaegle, Felix Gimeno, Andrew Brock, Andrew Zisserman, Oriol Vinyals, and João Carreira. Perceiver: General perception with iterative attention. In *Proceedings of the International Conference on Machine Learning*, 2021.

Dinesh Jayaraman and Kristen Grauman. Slow and steady feature analysis: higher order temporal coherence in video. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

Xu Ji, Joao F. Henriques, and Andrea Vedaldi. Invariant information clustering for unsupervised image classification and segmentation. In *Proceedings of the International Conference on Computer Vision*, pages 9865–9874, 2019.

Zhe Jiang, M. Kirby, Wenchong He, and Arpan Man Sainju. Deep learning for earth image segmentation based on imperfect polyline labels with annotation errors. *ArXiv*, abs/2010.00757, 2020.

Zhe Jiang, Wenchong He, Marcus Kirby, Sultan Asiri, and Da Yan. Weakly supervised spatial deep learning based on imperfect vector labels with registration errors. 2021.

Longlong Jing, Xiaodong Yang, Jingen Liu, and Yingli Tian. Self-supervised spatiotemporal feature learning via video rotation prediction. *arXiv: Computer Vision and Pattern Recognition*, 2018.

Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and Francois Fleuret. Transformers are rnns: Fast autoregressive transformers with linear attention. In *Proceedings of the International Conference on Machine Learning*, 2020.

Will Kay, João Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Apostol Natsev, Mustafa Suleyman, and Andrew Zisserman. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017.

Evangelos Kazakos, Arsha Nagrani, Andrew Zisserman, and Dima Damen. Slow-fast auditory streams for audio recognition. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2021.

Naji Khosravan, Shervin Ardeshir, and Rohit Puri. On attention modules for audio-visual synchronization. 2019.

Einat Kidron, Yoav Schechner, and Michael Elad. Pixels that sound. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2005.

Changil Kim, Hijung Valentina Shin, Tae-Hyun Oh, Alexandre Kaspar, Mohamed Elgharib, and Wojciech Matusik. On learning associations of faces and voices. In *Proceedings of the Asian Conference on Computer Vision*, 2018.

Dahun Kim, Donghyeon Cho, and In-So Kweon. Self-supervised video representation learning with space-time cubic puzzles. In *AAAI*, 2019.

Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Accurate image super-resolution using very deep convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

You Jin Kim, Hee-Soo Heo, Soo-Whan Chung, and Bong-Jin Lee. End-to-end lip synchronization based on pattern classification. *2021 IEEE Spoken Language Technology Workshop (SLT)*, pages 598–605, 2021.

Nikita Kitaev, Lukasz Kaiser, and Anselm Levskaya. Reformer: The efficient transformer. In *Proceedings of the International Conference on Learning Representations*, 2020.

Barbara Knight and Alan Johnston. The role of movement in face recognition. *Visual Cognition*, 1997.

Sophia Koepke, Olivia Wiles, Yael Moses, and Andrew Zisserman. Sight to sound: An end-to-end approach for visual piano transcription. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2020.

Bruno Korbar, Du Tran, and Lorenzo Torresani. Co-training of audio and video representations from self-supervised temporal synchronization. *CoRR*, 2018.

Marek Kowalski, Jacek Naruniec, and Tomasz Trzcinski. Deep alignment network: A convolutional neural network for robust face alignment. *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2017.

Ivan Krasin, Tom Duerig, Neil Alldrin, Andreas Veit, Sami Abu-El-Haija, Serge Belongie, David Cai, Zheyun Feng, Vittorio Ferrari, and Victor Gomes. Openimages: A public dataset for large-scale multi-label and multi-class image classification. 2016.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, 2009.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. ImageNet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, pages 1106–1114, 2012.

Hilde Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso A. Poggio, and Thomas Serre. HMDB: A large video database for human motion recognition. In *Proceedings of the International Conference on Computer Vision*, pages 2556–2563, 2011.

Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Tom Duerig, et al. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *arXiv preprint arXiv:1811.00982*, 2018.

Iro Laina, Christian Rupprecht, and Nassir Navab. Towards unsupervised image captioning with shared multimodal embeddings. In *Proceedings of the International Conference on Computer Vision*, 2019.

Barbara Landau and Lila R. Gleitman. *Language and experience : evidence from the blind child*. Harvard University Press Cambridge, Mass, 1985. ISBN 0674510259.

Ivan Laptev, Tony Lindeberg, Wolfgang Eckstein, Carsten Steger, and Albert Baumgartner. Automatic extraction of roads from aerial images based on scale space and snakes. *Machine Vision and Applications*, 2000.

Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. Learning representations for automatic colorization. In *Proceedings of the European Conference on Computer Vision*, pages 577–593. Springer, 2016.

Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. Colorization as a proxy task for visual understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

Linh Le, Ying Xie, Saisangararamaleengam Alagapan, Sumit Chakravarty, Pablo Ordonez, Michael Hales, and John Johnson. Deep pose alignment. In *2020 IEEE International Conference on Big Data*, 2020.

Thomas Le Cornu and Ben Milner. Generating intelligible audio speech from visual speech. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2017.

Yann LeCun. The MNIST database of handwritten digits. http://yann.lecun.com/exdb/mnist/.

Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 2015.

Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew P. Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, and Wenzhe Shi. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 105–114, 2017.

Hsin-Ying Lee, Jia-Bin Huang, Maneesh Kumar Singh, and Ming-Hsuan Yang. Unsupervised representation learning by sorting sequence. In *Proceedings of the International Conference on Computer Vision*, 2017.

Kuang-Huei Lee, Xi Chen, Gang Hua, Houdong Hu, and Xiaodong He. Stacked cross attention for image-text matching. In *Proceedings of the European Conference on Computer Vision*, 2018.

Sangho Lee, Jiwan Chung, Youngjae Yu, Gunhee Kim, Thomas Breuel, Gal Chechik, and Yale Song. Acav100m: Automatic curation of large-scale datasets for audio-visual video representation learning. In *Proceedings of the International Conference on Computer Vision*, 2021a.

Sangho Lee, Youngjae Yu, Gunhee Kim, Thomas Breuel, Jan Kautz, and Yale Song. Parameter efficient multimodal transformers for video representation learning. In *Proceedings of the International Conference on Learning Representations*, 2021b.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *ArXiv*, abs/1910.13461, 2020.

Dong Li, Wei-Chih Hung, Jia-Bin Huang, Shengjin Wang, Narendra Ahuja, and Ming-Hsuan Yang. Unsupervised visual representation learning by graph-based consistent constraints. In *Proceedings of the European Conference on Computer Vision*, 2016.

Jian Li, Zhaopeng Tu, Baosong Yang, Michael R. Lyu, and T. Zhang. Multi-head attention with disagreement regularization. In *Conference on Empirical Methods in Natural Language Processing*, 2018a.

Zhuwen Li, Qifeng Chen, and Vladlen Koltun. Interactive image segmentation with latent diversity. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018b.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollar, and Larry Zitnick. Microsoft coco: Common objects in context. In *Proceedings of the European Conference on Computer Vision*, September 2014.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollr. Focal loss for dense object detection. In *Proceedings of the International Conference on Computer Vision*, 2017.

Yan-Bo Lin and Yu-Chiang Frank Wang. Audiovisual transformer with instance attention for audio-visual event localization. In *Proceedings of the Asian Conference on Computer Vision*, November 2020.

Yan-Bo Lin, Yu-Jhe Li, and Yu-Chiang Frank Wang. Dual-modality seq2seq network for audio-visual event localization. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 2002–2006, 2019. doi: 10.1109/ICASSP.2019.8683226.

Peter J. Liu*, Mohammad Saleh*, Etienne Pot, Ben Goodrich, Ryan Sepassi, Lukasz Kaiser, and Noam Shazeer. Generating wikipedia by summarizing long sequences. In *Proceedings of the International Conference on Learning Representations*, 2018.

Yang Liu, Samuel Albanie, Arsha Nagrani, and Andrew Zisserman. Use what you have: Video retrieval using representations from collaborative experts. In *Proceedings of the British Machine Vision Conference*, 2019.

Yang Liu, Qingchao Chen, and Samuel Albanie. Adaptive cross-modal prototypes for cross-domain visual-language retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 14954–14964, June 2021a.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Ro{bert}a: A robustly optimized {bert} pretraining approach, 2020.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. *arXiv preprint arXiv:2103.14030*, 2021b.

David Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

Andrew L. Maas, Awni Y. Hannun, and Andrew Y. Ng. Rectifier nonlinearities improve neural network acoustic models. In *ICML Workshop on Deep Learning for Audio, Speech and Language Processing*, 2013.

Etienne Marcheret, Gerasimos Potamianos, Josef Vopicka, and Vaibhava Goel. Detecting audio-visual synchrony using deep neural networks. In *Proc. ICSA*, 2015.

Michaël Mathieu, C. Couprie, and Y. LeCun. Deep multi-scale video prediction beyond mean square error. In *Proceedings of the International Conference on Learning Representations*, volume abs/1511.05440, 2016.

Harry McGurk and John MacDonald. Hearing lips and seeing voices. *Nature*, 264:746–748, 12 1976.

Soroush Mehri, Kundan Kumar, Ishaan Gulrajani, Rithesh Kumar, Shubham Jain, Jose Sotelo, Aaron Courville, and Yoshua Bengio. Samplernn: An unconditional end-to-end neural audio generation model. In *Proceedings of the International Conference on Learning Representations*, 2016.

Annamaria Mesaros, Toni Heittola, and Tuomas Virtanen. TUT database for acoustic scene classification and sound event detection. In *European Signal Processing Conference*, 2016.

Annamaria Mesaros, Aleksandr Diment, Benjamin Elizalde, Toni Heittola, Emmanuel Vincent, Bhiksha Raj, and Tuomas Virtanen. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2019.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, 2013.

Volodymyr Mnih. *Machine Learning for Aerial Image Labeling*. PhD thesis, University of Toronto, 2013.

Volodymyr Mnih and Geoffrey E. Hinton. Learning to label aerial images from noisy data. In *Proceedings of the International Conference on Machine Learning*, 2012.

Hossein Mobahi, Ronan Collobert, and Jason Weston. Deep learning from temporal coherence in video. In *Proceedings of the International Conference on Machine Learning*, 2009.

Mathew Monfort, Alex Andonian, Bolei Zhou, Kandan Ramakrishnan, Sarah Adel Bargal, Tom Yan, Lisa Brown, Quanfu Fan, Dan Gutfruend, Carl Vondrick, et al. Moments in time dataset: one million videos for event understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

MTerrell N. Mundhenk, Daniel Ho, and Barry Y. Chen. Improvements to context based self-supervised learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

Arsha Nagrani, Joon Son Chung, and Andrew Zisserman. VoxCeleb: a large-scale speaker identification dataset. In *INTERSPEECH*, 2017a.

Arsha Nagrani, Joon Son Chung, and Andrew Zisserman. VoxCeleb: A large-scale speaker identification dataset. In *Proc. INTERSPEECH*, 2017b.

Arsha Nagrani, Samuel Albanie, and Andrew Zisserman. Seeing voices and hearing faces: Cross-modal biometric matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018a.

Arsha Nagrani, Samuel Albanie, and Andrew Zisserman. Learnable pins: Cross-modal embeddings for person identity. In *Proceedings of the European Conference on Computer Vision*, 2018b.

Arsha Nagrani, Samuel Albanie, and Andrew Zisserman. Learnable pins: Cross-modal embeddings for person identity. *arXiv preprint arXiv:1805.00833*, 2018c.

Arsha Nagrani, Chen Sun, David Ross, Rahul Sukthankar, Cordelia Schmid, and Andrew Zisserman. Speech2action: Cross-modal supervision for action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020.

Arsha Nagrani, Shan Yang, Anurag Arnab, Aren Jansen, Cordelia Schmid, and Chen Sun. Attention bottlenecks for multimodal fusion. *NeurIPS*, 2021.

Samer A. Nene, Shree K. Nayar, and Hiroshi Murase. Columbia object image library (coil-20). Technical Report CUCS-005-96, Department of Computer Science, Columbia University, February 1996.

Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *Proceedings of the European Conference on Computer Vision*, pages 69–84. Springer, 2016.

Mehdi Noroozi, Ananth Vinjimoor, Paolo Favaro, and Hamed Pirsiavash. Boosting self-supervised learning via knowledge transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

David Novotny, Samuel Albanie, Diane Larlus, and Andrea Vedaldi. Self-supervised learning of geometrically stable features through probabilistic introspection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

Markus Oberweger, Paul Wohlhart, and Vincent Lepetit. Training a feedback loop for hand pose estimation. In *Proceedings of the International Conference on Computer Vision*, 2015.

Andreea-Maria Oncescu, A. S. Koepke, João F. Henriques, Zeynep Akata, and Samuel Albanie. Audio retrieval with natural language queries. In *INTERSPEECH*, 2021.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Alice O'Toole, Dana Roark, and Herv Abdi. Recognizing moving faces: A psychological and neural synthesis. *Trends in cognitive sciences*, 6:261–266, 07 2002. doi: 10.1016/S1364-6613(02)01908-3.

Andrew Owens and Alexei A. Efros. Audio-visual scene analysis with self-supervised multisensory features. In *Proceedings of the European Conference on Computer Vision*, 2018.

Andrew Owens, Phillip Isola, Josh H. McDermott, Antonio Torralba, Edward H. Adelson, and William T. Freeman. Visually indicated sounds. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

Giambattista Parascandolo, Heikki Huttunen, and Tuomas Virtanen. Recurrent neural networks for polyphonic sound event detection in real life recordings. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2016.

Niki Parmar, Ashish Vaswani, Jakob Uszkoreit, Lukasz Kaiser, Noam M. Shazeer, Alexander Ku, and Dustin Tran. Image transformer. In *Proceedings of the International Conference on Machine Learning*, volume abs/1802.05751, 2018a.

Niki J. Parmar, Ashish Vaswani, Jakob Uszkoreit, Lukasz Kaiser, Noam Shazeer, Alexander Ku, and Dustin Tran. Image transformer. In *Proceedings of the International Conference on Machine Learning*, 2018b.

Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2536–2544, 2016.

Mandela Patrick, Yuki M. Asano, Polina Kuznetsova, Ruth Fong, João F. Henriques, Geoffrey Zweig, and Andrea Vedaldi. Multi-modal self-supervision from generalized data transformations. 2020.

Giorgio Patrini, Alessandro Rozza, Aditya Krishna Menon, Richard Nock, and Lizhen Qu. Making deep neural networks robust to label noise: A loss correction approach. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

Kedar Anil Patwardhan, Guillermo Sapiro, and Marcelo Bertalmío. Video inpainting under constrained camera motion. *IEEE Transactions on Image Processing*, 2007.

Renukananda Prajwal, Kondajji, Rudrabha Mukhopadhyay, Jerin Philip, Abhishek Jha, Vinay Namboodiri, and C V Jawahar. Towards automatic face-to-face translation. In *Proceedings of the 27th ACM International Conference on Multimedia*, page 14281436. Association for Computing Machinery, 2019.

Rui Qian, Di Hu, Heinrich Dinkel, Mengyue Wu, Ning Xu, and Weiyao Lin. Multiple sound sources localization from coarse to fine. In *Proceedings of the European Conference on Computer Vision*, 2020.

Alec Radford and Karthik Narasimhan. Improving language understanding by generative pre-training. 2018.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2018.

Janani Ramaswamy and Sukhendu Das. See the sound, hear the pixels. In *Proceedings of the IEEE Conference on Applications of Computer Vision*, 2020.

Scott E. Reed and Nando de Freitas. Neural programmer-interpreters. In *Proceedings of the International Conference on Learning Representations*, 2016.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems*, 2016.

Bernardino Romera-Paredes and Philip Torr. Recurrent instance segmentation. In *Proceedings of the European Conference on Computer Vision*, 2015.

Andrew Rouditchenko, Hang Zhao, Chuang Gan, Josh McDermott, and Antonio Torralba. Self-supervised audio-visual co-segmentation. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2019.

Lenny I. Rudin, Frederic Guichard, and Ping Yu. Video super-resolution via contrast-invariant motion segmentation and frame fusion (with applications to forensic video evidence). In *IEEE International Conference on Image Processing*, 1999.

Shunta Saito, Yakayoshi Yamashita, and Yoshimitsu Aoki. Multiple object extraction from aerial imagery with convolutional neural networks. *Journal of Imaging Science and Technology*, 2016.

Justin Salamon, Christopher Jacoby, and Juan Pablo Bello. A dataset and taxonomy for urban sound research. In *Proceedings of the ACM Multimedia Conference*, 2014.

Jorge Sánchez, Florent Perronnin, Thomas Mensink, and Jakob J. Verbeek. Image classification with the fisher vector: Theory and practice. *International Journal of Computer Vision*, 105(3):222–245, 2013.

Valentina Sanguineti, Pietro Morerio, Alessio Del Bue, and Vittorio Murino. Audio-visual localization by synthetic acoustic image generation. pages 2523–2531, May 2021.

Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

Daniel Seichter, Mona Köhler, Benjamin Lewandowski, Tim Wengefeld, and Horst-Michael Gross. Efficient RGB-D semantic segmentation for indoor scene analysis. In *IEEE International Conference on Robotics and Automation*, volume abs/2011.06961, 2020.

Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017.

Arda Senocak, Tae-Hyun Oh, Junsik Kim, Ming-Hsuan Yang, and In So Kweon. Learning to localize sound source in visual scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

Shuai Shao, Zeming Li, Tianyuan Zhang, Chao Peng, Gang Yu, Xiangyu Zhang, Jing Li, and Jian Sun. Objects365: A large-scale, high-quality dataset for object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

Evan Shelhamer, Jonathan Long, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

Prarthana Shrestha, Mauro Barbieri, Hans Weda, and Dragan Sekulovski. Synchronization of multiple camera videos using audio-visual features. *IEEE Transactions on Multimedia*, 12(1):79–92, 2010. doi: 10.1109/TMM.2009.2036285.

Abhinav Shrivastava, Abhinav Gupta, and Ross Girshick. Training region-based object detectors with online hard example mining. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *Proceedings of the European Conference on Computer Vision*, 2012.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015.

Malcolm Slaney, Michele Covell, and Facesync Is. Facesync:a linear operator for measuring synchronization of video facial images and audio tracks. In *Advances in Neural Information Processing Systems*, 2000.

Paris Smaragdis and Judith. Brown. Non-negative matrix factorization for polyphonic music transcription. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 177–180, 2003. doi: 10.1109/ASPAA.2003.1285860.

Linda Smith and Michael Gasser. The development of embodied cognition: Six lessons from babies. *Artificial Life*, 11:13–30, 2005.

David R. So, Chen Liang, and Quoc V. Le. The evolved transformer. *ArXiv*, abs/1901.11117, 2019.

Khurram Soomro, Amir Zamir, and Mubarak Shah. UCF101: A dataset of 101 human actions classes from videos in the wild. *CoRR*, abs/1212.0402, 2012.

Nitish Srivastava, Elman Mansimov, and Ruslan Salakhutdinov. Unsupervised learning of video representations using lstms. In *Proceedings of the International Conference on Machine Learning*, 2015.

Nicolas Staelens, Jonas De Meulenaere, Lizzy Bleumers, Glenn Van Wallendael, Jan De Cock, Koen Geeraert, Nick Vercammen, Wendy Van den Broeck, Brecht Vermeulen, Rik Van de Walle, et al. Assessing the importance of audio/video synchronization for simultaneous translation of video sequences. *Multimedia systems*, 18(6):445–457, 2012.

Chen Sun, Austin Myers, Carl Vondrick, Kevin P. Murphy, and Cordelia Schmid. Videobert: A joint model for video and language representation learning. *Proceedings of the International Conference on Computer Vision*, pages 7463–7472, 2019.

Narayanan Sundaram, Thomas Brox, and Kurt Keutzer. Dense point trajectories by GPU-accelerated large displacement optical flow. In *Proceedings of the European Conference on Computer Vision*, 2010.

Didac Surs, Amanda Duarte, Amaia Salvador, Jordi Torres, and Xavier Gir-i Nieto. Cross-modal embeddings for video and audio retrieval. In *Proc. ECCV workshop*, 01 2018.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112, 2014.

Naoya Takahashi, Michael Gygli, Beat Pfister, and Luc Van Gool. Deep convolutional neural networks and data augmentation for acoustic event detection. In *INTERSPEECH*, 2016.

Xin Tao, Hongyun Gao, Xiaoyong Shen, Jue Wang, and Jiaya Jia. Scale-recurrent network for deep image deblurring. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

Andriy Temko and Climent Nadeu. Classification of acoustic events using svm-based clustering schemes. *Pattern Recognition*, 2006.

Bart Thomee, David A Shamma, Gerald Friedland, Benjamin Elizalde, Karl Ni, Douglas Poland, Damian Borth, and Li-Jia Li. Yfcc100m: the new data in multimedia research. *Commun. ACM*, 2016.

Yapeng Tian, Jing Shi, Bochen Li, Zhiyao Duan, and Chenliang Xu. Audio-visual event localization in unconstrained videos. In *Proceedings of the European Conference on Computer Vision*, 2018.

Yapeng Tian, Dingzeyu Li, and Chenliang Xu. Unified multisensory perception: Weakly-supervised audio-visual video parsing. In *Proceedings of the European Conference on Computer Vision*, 2020.

Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. *arXiv preprint arXiv:1906.05849*, 2019.

Michael E. Tipping and Christopher M. Bishop. Bayesian imge super-resolution. In S. Thrun, S. Becker, and K. Obermayer, editors, *Advances in Neural Information Processing Systems*, volume 15, pages 1279–1286, Cambridge, MA, 2003. MIT Press.

Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Deep end2end voxel2voxel prediction. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2016.

Efthymios Tzinis, Scott Wisdom, Aren Jansen, Shawn Hershey, Tal Remez, Dan Ellis, and John R. Hershey. Into the wild with audioscope: Unsupervised audio-visual separation of on-screen sounds. In *Proceedings of the International Conference on Learning Representations*, 2021.

Aäron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew W. Senior, and Koray Kavukcuoglu. WaveNet: A generative model for raw audio. In *ISCA Speech Synthesis Workshop*, 2016.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, 2017.

Andreas Veit, Neil Gordon Alldrin, Gal Chechik, Ivan Krasin, Abhinav Gupta, and Serge J. Belongie. Learning from noisy large-scale datasets with minimal supervision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

Ruben Villegas, Jimei Yang, Seunghoon Hong, Xunyu Lin, and Honglak Lee. Decomposing motion and content for natural video sequence prediction. *ArXiv*, abs/1706.08033, 2017.

Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. Show and tell: A neural image caption generator. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3156–3164, 2015.

Paul Viola and Michael Jones. Robust real-time object detection. In *Proc. SCTV Workshop*, 2001.

Carl Vondrick, Abhinav Shrivastava, Alireza Fathi, Sergio Guadarrama, and Kevin Murphy. Tracking emerges by colorizing videos. In *Proceedings of the European Conference on Computer Vision*, September 2018.

Konstantinos Vougioukas, Stavros Petridis, and Maja Pantic. Realistic speech-driven facial animation with gans. *International Journal of Computer Vision*, pages 1–16, 2019.

Apoorv Vyas, Angelos Katharopoulos, and Francois Fleuret. Fast transformers with clustered attention. In *Proceedings of the International Conference on Neural Information Processing Systems*, December 2020.

Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge J. Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011.

Fan Wang, Qixing Huang, and maks guibas. Unsupervised multi-class joint image segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

Weiyao Wang, Du Tran, and Matt Feiszli. What makes training multi-modal classification networks hard? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020a.

Yikai Wang, Wenbing Huang, Fuchun Sun, Tingyang Xu, Yu Rong, and Junzhou Huang. Deep multimodal fusion by channel exchanging. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020b.

Yandong Wen, Mahmoud Al Ismail, Weiyang Liu, B. Raj, and Rita Singh. Disjoint mapping network for cross-modal matching of voices and faces. In *Proceedings of the International Conference on Learning Representations*, volume abs/1807.04836, 2019.

Laurenz Wiskott and Terrence J. Sejnowski. Slow Feature Analysis: Unsupervised Learning of Invariances. *Neural Computation*, 14(4):715–770, 04 2002.

Bichen Wu, Chenfeng Xu, Xiaoliang Dai, Alvin Wan, Peizhao Zhang, Masayoshi Tomizuka, Kurt Keutzer, and Peter Vajda. Visual transformers: Token-based image representation and processing for computer vision. *CoRR*, abs/2006.03677, 2020.

Fanyi Xiao, Yong Jae Lee, Kristen Grauman, Jitendra Malik, and Christoph Feichtenhofer. Audiovisual slowfast networks for video recognition. *arXiv preprint arXiv:2001.08740*, 2020.

Tong Xiao, Tian Xia, Yi Yang, Chang Huang, and Xiaogang Wang. Learning from massive noisy labeled data for image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

Weidi Xie, Arsha Nagrani, Joon Son Chung, and Andrew Zisserman. Utterance-level aggregation for speaker recognition in the wild. In *International Conference on Acoustics, Speech, and Signal Processing*, 2019.

Haoming Xu, Runhao Zeng, Qingyao Wu, Mingkui Tan, and Chuang Gan. Cross-modal relation-aware networks for audio-visual event localization. In *Proceedings of the ACM Multimedia Conference*, 2020.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Aaron Courville, Ruslan Salakhutdinov, Richard Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. *arXiv preprint arXiv:1502.03044*, 2015.

Yong Xu, Qiuqiang Kong, Wenwu Wang, and Mark D. Plumbley. Large-scale weakly supervised audio classification using gated convolutional neural network. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2018.

Baosong Yang, Zhaopeng Tu, Derek F. Wong, Fandong Meng, Lidia S. Chao, and T. Zhang. Modeling localness for self-attention networks. In *Conference on Empirical Methods in Natural Language Processing*, 2018.

Yi Yang, Brendan Shillingford, Yannis Assael, Miaosen Wang, Wendi Liu, Yutian Chen, Yu Zhang, Eren Sezener, Luis C Cobo, Misha Denil, et al. Large-scale multilingual audio visual dubbing. *arXiv preprint arXiv:2011.03530*, 2020.

Zhuyu Yao, Jiangbo Ai, Boxun Li, and Chi Zhang. Efficient DETR: improving end-to-end object detector with dense prior. *CoRR*, abs/2104.01318, 2021.

Umit H. Yapanel, Xianxian Zhang, and John H. L. Hansen. High performance digit recognition in real car environments. In *INTERSPEECH*, 2002.

Armand Zampieri, Guillaume Charpiat, Nicolas Girard, and Yuliya Tarabalka. Multimodal image alignment through a multiscale chain of neural networks with application to remote sensing. In *Proceedings of the European Conference on Computer Vision*, 2018.

Wojciech Zaremba, Tomas Mikolov, Armand Joulin, and Rob Fergus. Learning simple algorithms from examples. In *Proceedings of the International Conference on Machine Learning*, 2016.

Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *Proceedings of the European Conference on Computer Vision*, pages 649–666. Springer, 2016.

Yunhua Zhang, Ling Shao, and Cees G. M. Snoek. Repetitive activity counting by sight and sound. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14070–14079, June 2021.

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 *Proceedings of the International Conference on Computer Vision*, 10/2017 2017.

Hang Zhao, Chuang Gan, Andrew Rouditchenko, Carl Vondrick, Josh McDermott, and Antonio Torralba. The sound of pixels. In *Proceedings of the European Conference on Computer Vision*, 2018a.

Hang Zhao, Chuang Gan, Wei-Chiu Ma, and Antonio Torralba. The sound of motions. In *Proceedings of the International Conference on Computer Vision*, 2019.

Mingmin Zhao, Tianhong Li, Mohammad Abu Alsheikh, Yonglong Tian, Hang Zhao, Antonio Torralba, and Dina Katabi. Through-wall human pose estimation using radio signals. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June 2018b.

Bolei Zhou, Aditya Khosla, Àgata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016a.

Tinghui Zhou, Philipp Krähenbühl, Mathieu Aubry, Qixing Huang, and Alexei A. Efros. Learning dense correspondence via 3d-guided cycle consistency. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016b.

Xiaowei Zhou, Menglong Zhu, and Kostas Daniilidis. Multi-image matching via fast alternating minimization. In *Proceedings of the International Conference on Computer Vision*, 2015.

Yipin Zhou, Zhaowen Wang, Chen Fang, Trung Bui, and Tamara L. Berg. Visual to sound: Generating natural sound for videos in the wild. *CoRR*, abs/1712.01393, 2017.

Fangrui Zhu, Li Zhang, Yanwei Fu, Guodong Guo, and Weidi Xie. Self-supervised video object segmentation. *CoRR*, abs/2006.12480, 2020.

Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the International Conference on Computer Vision*, 2017.

Lingyu Zhu and Esa Rahtu. Visually guided sound source separation using cascaded opponent filter network. In *Proceedings of the Asian Conference on Computer Vision*, 2020.

Lingyu Zhu and Esa Rahtu. Leveraging category information for single-frame visual sound source separation. In *European Workshop on Visual Information Processing (EUVIP)*, pages 1–6. IEEE, 2021a.

Lingyu Zhu and Esa Rahtu. V-slowfast network for efficient visual sound separation. *arXiv preprint arXiv:2109.08867*, 2021b.

Xiaodan Zhuang, Xi Zhou, Mark Hasegawa-Johnson, and Thomas S. Huang. Real-world acoustic event detection. *Pattern Recognition Letters*, 2010.

Michael Zibulevsky and Barak A. Pearlmutter. Blind source separation by sparse decomposition in a signal dictionary. *Neural Computation*, 13(4):863–882, 2001. doi: 10.1162/089976601300014385.

Stefano Zorzi, Ksenia Bittner, and Friedrich Fraundorfer. Map-repair: Deep cadastre maps alignment and temporal inconsistencies fix in satellite images. *arXiv preprint arXiv:2007.12470*, 2020.

Appendix A

Statements of Authorship

A statement of authorship is provided for each multi-authored apper included in this thesis. The statements describe the candidates and co-authors independent research contributions in the thesis publications. For each publication there exists a complete statement that is filled out and signed by the candidate and supervisor.

Statment of Authorship for multi-authored paper in Chapter 3: AutoCorrect: Deep Inductive Alignment of Noisy Geometric Annotations.

Paper title	AutoCorrect: Deep Inductive Alignment of Noisy Geometric Annotations
Publication status	Published
Authors	**Honglie Chen**, Weidi Xie, Andrea Vedaldi, Andrew Zisserman
Details	Published in the Proceedings of the British Machine Vision Conference (BMVC), 2019

Student Confirmation

Student Name	Honglie Chen		
Contribution to the paper	1. Joint conception of the idea 2. Research of prior work 3. Design and implementation of models 4. Running of all experiments 5. Writing and presentation of the paper		
Signature	Honglie	Date	25 / 09 / 2021

Supervisor Confirmation

By signing the Statement of Authorship, you are certifying that the candidate made a substantial contribution to the publication, and that the description above is accurate.

Supervisor Name and Title	Professor Andrew Zisserman		
Supervisor Comments			
Signature		Date	

Statment of Authorship for multi-authored paper in Chapter 4: VGG-Sound: A Large-scale Audio-Visual Dataset.

Paper title	VGG-Sound: A Large-scale Audio-Visual Dataset
Publication status	Published
Authors	**Honglie Chen**, Weidi Xie, Andrea Vedaldi, Andrew Zisserman
Details	Published in IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2020.

Student Confirmation

Student Name	Honglie Chen		
Contribution to the paper	1. Joint conception of the idea 2. Research of prior work 3. Data pre-processing 4. Implementation of decoding module 5. Writing and presentation of the paper		
Signature	Honglie	Date	25 / 09 / 2021

Supervisor Confirmation

By signing the Statement of Authorship, you are certifying that the candidate made a substantial contribution to the publication, and that the description above is accurate.

Supervisor Name and Title	Professor Andrew Zisserman		
Supervisor Comments			
Signature		Date	

Statment of Authorship for multi-authored paper in Chapter 5: Localizing Visual Sounds the Hard Way.

Paper title	Localizing Visual Sounds the Hard Way
Publication status	Published
Authors	**Honglie Chen**, Weidi Xie, Triantafyllos Afouras, Arsha Nagrani, Andrea Vedaldi, Andrew Zisserman
Details	Published in the proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR), 2021

Student Confirmation

Student Name	Honglie Chen		
Contribution to the paper	1. Joint conception of the idea 2. Research of prior work 3. Design and implementation of models 4. Running of all experiments 5. Writing and presentation of the paper		
Signature	Honglie	Date	25 / 09 / 2021

Supervisor Confirmation

By signing the Statement of Authorship, you are certifying that the candidate made a substantial contribution to the publication, and that the description above is accurate.

Supervisor Name and Title	Professor Andrew Zisserman		
Supervisor Comments			
Signature		Date	

Statment of Authorship for multi-authored paper in Chapter 6: Audio-Visual synchronization in the wild.

Paper title	Audio-Visual synchronization in the wild
Publication status	Under Review
Authors	**Honglie Chen**, Weidi Xie, Triantafyllos Afouras, Arsha Nagrani, Andrea Vedaldi, Andrew Zisserman
Details	Published in the Proceedings of the British Machine Vision Conference (BMVC), 2021

Student Confirmation

Student Name	Honglie Chen		
Contribution to the paper	1. Joint conception of the idea 2. Research of prior work 3. Design and implementation of models 4. Running of all experiments 5. Writing and presentation of the paper		
Signature	Honglie	Date	25 / 09 / 2021

Supervisor Confirmation

By signing the Statement of Authorship, you are certifying that the candidate made a substantial contribution to the publication, and that the description above is accurate.

Supervisor Name and Title	Professor Andrew Zisserman		
Supervisor Comments			
Signature		Date	