Audio-visual Deep Learning

Triantafyllos Afouras

Worcester College

University of Oxford

A thesis submitted for the degree of
Doctor of Philosophy

Hilary 2021

Acknowledgements

This thesis would not have been possible without the help and support of so many.

First and foremost I thank my advisor Andrew Zisserman, for all his trust, guidance and support, for which I will forever be indebted; I could not have asked for a better captain in this journey. And my co-advisor, Joon Son Chung, for being there from the first to the last step in this endeavor, generously dedicating his time and energy from the other side of the world.

To my amazing collaborators: Andrew Owens, Yuki Asano, Prajwal Renukananda, Arsha Nagrani, Honglie Chen, Weidi Xie, Themos Stafylakis, Jaesung Huh, Andrew Senior, and Oriol Vinyals; without them I would have probably lost my sanity pushing for one of those deadlines.

To the phenomenal BSL-crew: Liliane Momeni, Hannah Bull, Gül Varol, and Samuel Albanie, the best team I've ever been a part of and an absolute pleasure to work with.

To my mentors at Facebook: Francois Fagan, Andrea Vedaldi, and Florian Metze, for their invaluable help and support during my internship and our subsequent collaboration.

To all the members of the Visual Geometry Group, of which it has been an incredible privilege to be a member. In particular to Ankush Gupta and David Novotny for the great company in the lab and useful advice during my first year in the VGG, and also Tom Jakab, Tengda Han, and Chuhan Zhang for being awesome deskmates. Also to Ashish Thandavan, Abhishek Dutta and Ernesto Coto for all the crucial engineering support.

To the Zwara Crew: Christos Zalidis, Apostolos Avranas, Despoina Paschalidou, Christos Tsirigotis, Vasilis Choutas, Giorgos Papoudakis, and especially Angelos Katharopoulos, who has offered me invaluable advice on many important decisions throughout those years.

To my AIMS cohort: Oliver Bent, Fabian Fuchs, Adam Golinski, Bradley Gram-Hansen, Xu Ji, Shuyu Lin, Andrea Patane, Sasha Salter, and Edward Wagstaff, with whom I shared an amazing first year in Oxford, and of course to Wendy Poole for being the most fantastic program admin that has ever existed. Also to Jakob Foerster, Greg Farquhar, Nantas Nardelli and Shimon Whiteson, for the tons of fun I had during our collaborations and from which I also learned a great deal about academic team work.

But most importantly to my brother, Giannis, and my parents, Betty and Giorgos, for their endless love and support, the only constant in this world that I can always count on.

Abstract

Human perception and learning are inherently multimodal: we interface with the world through multiple sensory streams, including vision, audition, touch, olfaction and taste. By contrast, automatic approaches for machine perception and learning have traditionally depended on single modalities, by processing, for instance, video, audio or speech separately. The goal of this thesis is instead utilizing the natural co-occurrence of audio and visual information in videos to learn useful tasks.

The thesis is structured around four main themes: (i) lip reading and Audio-Visual Speech Recognition (AVSR); (ii) audio-visual speech enhancement and separation; (iii) audio-visual sound source localization and detection; (iv) sign language recognition;

Lip reading is the ability to recognise speech by observing the speaker's lip movements; it is a challenging task and has many important applications including enabling speech impaired individuals to better communicate. We build and improve on recent breakthroughs by exploring the use of Transformer-based architectures, proposing attention based pooling mechanisms for representation aggregation, as well as using sub-word units instead of character tokenisation. These enhancements, combined with improvements to the training protocol, yield substantial performance boosts, resulting in state-of-the art results on the challenging LRS2 and LRS3 datasets. Moreover, we develop a method for exploiting unlabelled speech video by distilling an Automatic Speech Recognition Model into a lip-reading one. Finally we show that it is possible to identify spoken language just by observing a speaker's lip movements.

Speech enhancement and separation increases the signal-to-noise ratio of noisy speech audio, by filtering out interfering voices or background noise. Until recently, works in this area focused on solving the problem by using the audio modality alone. We first propose tackling this problem audio-visually by conditioning on each speaker's lip movements. We then further improve this approach by making it robust to visual occlusions.

Recent works have shown that it is possible to determine the spatial location of sound-making objects in video frames by exploiting correlations between the audio and video signals. We present a method to improve and extend these techniques, by grouping heat maps into distinct object representations that can be used for various downstream tasks, without the need for face detectors. The resulting method is entirely self-supervised and can be used for extending tasks such as active speaker detection and speech separation in new domains, e.g. videos of cartoons or puppets. We then propose a method that uses similar principles in order to train object detection models without relying on human annotation, by deriving all the necessary supervision from audio-visual correspondence cues.

Finally we consider the problem of automatic sign-language recognition, which to-date remains unsolved, despite all the progress in related vision and natural language processing

tasks. The main blocker is the scarcity of large-scale annotated sign-language datasets. We attempt to solve this problem by using sign-interpreted TV broadcasts footage, combined with subtitles obtained from the corresponding audio speech. Towards achieving this goal we first train Transformer models to identify and temporally localize instances of sings in continuous signed videos, thus automatically generating thousands of annotations for a large sign vocabulary. We then directly tackle the problem of temporally aligning the asynchronous subtitles to the sign language footage.

Contents

1 | Introduction and Background

Human perception and learning are inherently multimodal: we interface with the world not through one but through multiple sensory streams, including vision, audition, touch, olfaction and taste. Imagine, for instance, some common scenes from everyday life: humans speaking and interacting in crowded restaurants, vehicles moving, birds chirping, rain drops hitting the ground: all these examples involve visual experiences that are usually accompanied by associated sounds. To form informed perceptions of the world, we combine visual and auditory information, which might be ambiguous and noisy individually, but when combined help us form accurate perceptions of the world. Similarly, our oral communication is predominantly based on audio, however visual messages contained in lip motion, body language and hand gestures can be crucial for disambiguation as well as for conveying subtle information not encoded in language, such as different intents and moods. The simultaneous processing of these visual and audio signals is crucial for us in order to understand and learn.

By contrast, automatic approaches for machine perception and learning have traditionally depended on single modalities, for example processing vision, audio or speech separately. Audio-visual information on the other hand is plentiful in internet videos today: resources such as YouTube provide a very rich source of diverse audio-visual scenes.

The goal of this thesis is to utilize the natural co-occurrence of audio and visual information in videos to learn useful tasks. The key ways that we aim to achieve this by are: (i) by using the one modality for supervising the other; (ii) by using the one modality to disambiguate the other; (iii) by using both modalities together as complementary inputs.

We will begin this thesis by providing motivation for audio-visual machine learning from psychology studies on human learning and perception, as well as presenting the key observations that make the idea appealing (Section 1.1); we will move on to Section 1.2 where we introduce the four main themes that we explore; finally in Section 1.3 we will provide a full outline of the thesis.

1.1 Motivation and key ideas

1.1.1 Motivation from psychology

Learning and perception in humans is heavily dependent on the redundancy that results from collecting information from multiple sensory inputs. This redundancy enables functionality even when one of the sensory components is lost [355]. It also means that the different sensory systems can teach each other, in order to bootstrap learning without the need for external supervision or defined tasks. In the psychology literature this idea is termed *reentry* [116, 117] and is discussed in terms of bidirectional exchanges of signals between brain regions which are continuously interrelated in space and time. Through this mechanism, humans can form multiple simultaneous representations of the same information across modalities. The simultaneous perception of these multi-sensory inputs (e.g. visual and auditory) creates a powerful learning mechanism that relies on the separate representations supervising each other.

It has been suggested that humans fuse visual and aural modalities early in the processing stage and that this joint perception is important for both lower level sensorimotor coordination, such as controlling visual orientation as well as more abstract understanding like object categorisation [268]. Indeed human infants are sensitive to spatial properties of sound, being able to direct their gaze toward auditory stimuli even 10 minutes after birth [404], while later developing the ability to focus visual attention on objects that match what they hear [268]. Four-month old infants respond to temporal synchrony between aural and visual stimuli [357]. In fact studies comparing visual attention in deaf and hearing children show that the use of environmental sounds is important for organizing visual attention [355].

Audio-visual information plays an important role during language acquisition as well. For example, studies have shown that this process is sometimes delayed in blind children [262, 401], suggesting that visual information is complementary to speech and important during the language acquisition phase. In fact some studies suggest that human infants can associate phonetic information with lip movements, before learning to understand or speak a language [226, 227, 302].

Although vision and hearing were originally viewed as distinct systems that operate independently, it is now believed that cross-modal interactions are very common [348]. In fact visual inputs can affect the way sound is perceived by the human brain. A prominent example of this is the McGurk Effect [261], an illusion where visual lip motion alters the way that speech is perceived. For example, an audio utterance of the syllable "ba" may be perceived as "fa" or "da" if it is coupled with mouth movements that are associated with those sounds.

Visual sensory input can also heavily influence the perceived spatial location of a sound source, an phenomenon called the "ventriloquist effect". For example, this is experienced in movie theaters, where a video is projected on a screen and a sound track is played from speakers located on the sides, however the viewer perceives the sound to be originating from the screen [402]. The explanation for this effect is that the modality with the higher spatial localization potential (vision) dominates the other one (audition) [10].

The inverse effect is also possible – *i.e.* sound can influence visual perception. Sekuler *et al.* [337] conducted an experiment, where subjects were shown an animation of two identical objects moving towards each other, coinciding, and then moving apart. When the animation was shown without any audio effects, the perception of most participants was that of the two discs continuing in their original directions without collision; however introducing a sound effect at or near the point of coincidence induced the sense of collision followed by bouncing. Other studies moreover suggest that accompanying sound signals may alter temporal aspects of visual perception [348], such as the perceived rate, duration or temporal resolution [146, 333, 388] of visual stimuli.

In this thesis we take inspiration from psychology and study, among other things, ways to improve machine perception and learning, by studying the visual and aural modalities as complementary to each other, or as part of self-supervised frameworks where the one modality is used to provide supervision for the other.

1.1.2 Audio-visual machine learning

The central theme of this thesis is training deep models on videos by exploiting some combination of audio and visual information, with varying levels of supervision. Setting aside the biologically inspired motivation and the relation to human cognition discussed above, this is a promising direction due to two observations: first, audio-visual information is abundant in videos found online. Second, self-supervised methods are becoming a predominant theme in machine learning and audio-visual learning is naturally suited for developing them. We will now briefly examine those two ideas.

A cornucopia of audio-visual information: Videos are abundant on the internet today. Over 500 hours of video are uploaded on YouTube every minute[1] and this number is growing by the day. Platforms like Instagram and TicToc have contributed to an explosion of the multi-modal content that is created by professional and amateur users every day. And what is great from the data-scientist's perspective, is that the visual and audio modalities naturally co-occur and are most often correlated, sharing some common information. Indeed many large-scale datasets already exist with varying levels of curation, e.g. Kinetics [61], Audioset [149], VGGSound [69], HowTo100M [266]. It is common for videos to contain speech, in which case, Automatic Speech Recognition (ASR), a mature and reliable technology, can be used to obtain structured, semantic text representations.

The advent of Self-Supervised Learning (SSL): Deep learning largely owns its success to the availability of large-scale annotated datasets for training [328], and to the development of deep neural networks [224, 351, 367]. These methods originally relied on supervised learning from uni-modal inputs and manual annotations. For basic vision tasks, such as video classification and object detection, supervised models today work very well, however obtaining human annotations is expensive. Instead, self-supervised learning methods have emerged as an alternative that has achieved remarkable results, matching or even surpassing their supervised counterparts [72, 174, 380].

[1] https://www.statista.com/statistics/259477/hours-of-video-uploaded-to-youtube-every-minute/

These approaches usually rely on solving *pretext* tasks for which a part of the input signal is held-out and used as a prediction target, replacing human annotations. The main idea is that in order to solve the pretext task the models need to learn the semantics of the objects, thus developing general representations to be transferred to *downstream* objectives. Examples of this include predicting the natural orientation of images among different rotations [152], learning the relative locations of image patches by solving Jigsaw puzzles [110, 287], or exploiting the temporal coherence in videos [269, 398]. Alternatively strong data augmentations have been broadly used to simulate new views of the same information [174], coupled with contrastive learning methods [167] to learn desired invariances while preserving semantics.

Ideally, self-supervision offers very important advantages compared to supervised methods in terms of both scalability and generalization potential: scaling up is easy as it involves only collecting more data which is fast and cheap; generalization comes from the flexibility to directly apply the same methods to new domains, as no extra annotation effort is needed.

Self-supervision emerging from multisensory input: An alternative to the data augmentations required for uni-modal SSL is provided naturally through the shared information encoded between the two modalities in audio-visual video streams. We could say that one particular incarnation of the concept of reentry in multi- modal machine learning is self-supervision relying on audio-visual co-occurrence: manual annotations can be substituted for the labels by exploiting the correlation between the pattern distributions in audio and video. One way to do this is by minimizing the disparity between the activations produced from separate networks that process the two modalities. De Sa [100], exemplifies this learning process by referring to a person that sees a cow and hears a "mooing" sound at the same time. Even though the appearance of the cow is not accompanied by an explicit "cow" label, its co-occurrence with the "moo" sound helps humans associate the two and learn them as attributes of the particular animal. In order to emulate a similar learning procedure in machines, what is needed is to process the "moo" sound in order to obtain some kind of label (or more generally a supervision signal) to be associated with the cow's appearance and vice versa.

To sum up, in audio-visual self-supervised learning, rather than explicitly telling the machine that it should be associating particular appearances, motions, actions or scenes with sounds, e.g.

speech with lips, a bow scratching a violin with music, etc, the objective is to discover these associations by watching videos and solving appropriate tasks with the help of tailored objectives.

This idea is of course not new; similar lines of work have been studied since decades ago, *e.g.*for visually localizing a sound source [130, 180, 207], separating sound sources [182] or measuring the synchronization of the two modalities [354]. However, the limited availability of data and reliance on statistical modelling and heuristics for obtaining visual features made it difficult to fully exploit the richness of the multi-modal information. Recently, the creation of large video datasets and the advances in deep network architectures has enabled fast progress in this field. Some notable examples include the works of Owens *et al.* [295] who use ambient sounds as supervision to train visual representations or predict the sounds that different objects make when struck with a drumstick [294], of Aytar *et al.* [27] that use pre-trained visual recognition models as teachers to train representations for audio scene classification, and of Harwath *et al.* [172] who learn to match audio captions to images.

These works resurged interest in audio-visual learning and lay the groundwork for the development of more elaborate methods. However, we argue that they have merely scratched the surface on the potential that audio-visual learning offers. We build on this work, by enriching the models, reducing the required amount of supervision, and learning new exciting tasks. Some of the questions that we will attempt to answer on the way are:

- Can the kind of multi-modal processing found in human learning and perception be emulated in machine learning?
- What are the key multi-modal cues in human communication and how do they complement each other?
- Can they be exploited to improve performance in existing machine-learnable tasks and introduce new ones?
- Can we use audio as supervision in videos?
- What ways are there to exploit audio-visual co-occurrence?

1.2 Thesis Topics

The material presented in this thesis is conceptually divided into four units: (a) lip reading and audio-visual speech recognition, (b) audio-visual speech enhancement and separation, (c) audio-visual object localization and detection, and (d) sign language recognition. Some of those concepts are related, and there can be substantial overlap in terms of their applications.

1.2.1 Lip reading and Audio-Visual Speech recognition (AVSR)

Lip reading, the ability to recognise speech by observing the speaker's lip movements, is a challenging task for humans and machines alike. It can deployed in many interesting applications such as enabling silent dictation on mobile devices or dubbing silent films. More importantly it has great potential for medical applications, such as enabling speech impaired individuals to communicate, either by enhancing speech, e.g. for patients suffering from Lou Gehrig's disease, or by helping people with aphonia (loss of voice) to communicate through lip movements [347].

Audio-visual speech recognition extends traditional audio-based Automatic Speech Recognition (ASR), by combining both audio and visual speech signals to transcribe speech into text. The goal is to use lip-movements to disambiguate similarly sounding words, which can be especially useful in noisy environments.

Recent breakthroughs [26, 86, 87, 360] were made possible due to the development of deep learning models and the availability of large scale datasets. We extend and improve these methods by proposing better architectures and more data-efficient learning methods in Part I of this thesis.

1.2.2 Audio-visual speech enhancement and separation

What is commonly referred to as the "cocktail-party problem" can be framed as isolating individual voices in multi-speaker scenarios (separation) or increasing the signal-to-ambient-noise-ratio in noisy audio (enhancement). Until recently, works in this area mostly focused on using only audio to solve the task [255, 313, 392]. However using video cues to solve this

task can be very advantageous. A few prior works that have used video [133, 134, 188] are limited to constrained conditions (e.g. fixed set of phrases or small number of speakers). We are among the first to solve this problem audio-visually and to demonstrate strong performance under general in-the-wild conditions (Part II).

Solving this problem well can enable a diverse range of practical applications, such as facilitating teleconferencing in cars, improving subtitle generation in videos with noisy audio, or developing smart audio-visual hearing devices that can enhance speech based on visual input.

1.2.3 Audio-visual object localization and detection

Sound source localization is the task of determining the spatial location of sound-making objects in video frames. It has been solved with self-supervision using either *correspondence* cues [22, 171, 189, 341, 376], e.g. by training a model to predict whether audio and a single video frame come from the same or different videos, or *synchronization* [293] as the proxy task.

Although these methods obtain compelling saliency maps and are easy to interpret, they are unsatisfactory in the following aspects: (i) they only roughly highlight the location of sound sources, without grouping the objects in a scene, and (ii) have limited practical application. We tackle those problems in Part III.

1.2.4 Sign language recognition

Sign languages are visual languages used as the natural means of communication of deaf communities. Achieving automatic sign localisation would enable various useful applications, such as automatic creation of dictionaries to help learning sign languages, indexing of signing content to enable efficient search and "wake-word" recognition in mobile devices for signers.

However, while there has been substantial progress in machine translation of spoken languages in recent years, automatic sign language recognition remains an unsolved problem. The availability of large-scale annotated datasets, e.g. paired text corpora and transcribed audio speech sequences, has been crucial for the training of strong performing models in these

tasks [17, 77]; But the development of respective sign-language datasets is more challenging. The reason is that compared to other domains, the amount of sign language videos available online is limited and its manual annotation is difficult. Since most of the available footage is found as part of interpreted TV programs, one idea to circumvent the data scarcity issue is to annotate the sign sequences in these videos using the audio speech that they are being interpreted from. However there are some particular challenges to this approach: *e.g.* not all the words in the transcription of a sign-language clip are actually signed; also, the signing video and speech audio are in most cases not naturally aligned (as would be the case with speech audio and lip movements), and the temporal synchronization of the audio to signs is not straightforward. We will discuss ways in which we attempt to solve those problems towards enabling full automatic sign-language translation in Part IV.

1.3 Thesis outline

In this section we provide an outline of the rest of the thesis chapters.

Part I: Lip reading and Audio-Visual Speech recognition (AVSR)

In Chapter 2 we extend our work in [4] and the work of Chung et al [86]. We combine the audio and video modalities, and use modern architectures to create an audio-visual speech recognition pipeline, demonstrating improved recognition performance compared to using only audio, especially under the presence of noise. Unlike previous works, that have focussed on recognising a limited number of words or phrases, we tackle lip reading as an open-world problem, which means training and evaluating on unconstrained natural language sentences, and in the wild videos.

In Chapter 3 we upgrade the lip reading pipeline introduced in Chapter 2 with several architecture and methodology enhancements, including the introduction of a Visual Transformer Pooling (VTP) for aggregating the spatial visual features, and the use of sub-word unit tokenisation instead of characters. The best trained models in the resulting framework significantly improve the state-of-the-art performance on public benchmarks and achieve results comparable to industrial models trained on orders of magnitude more data.

In Chapter 4 we train a network to read lips with cross-modal distillation from a teacher ASR model. We show how arbitrary amounts of unlabeled video data can be exploited to train lip reading models. We achieve state-of-the art results for training only on publicly available datasets.

In Chapter 5 we train models that can identify a spoken language just by interpreting the speaker's lip movements. We show that models can learn to discriminate among 14 different languages using only visual speech information. We evaluate the trained models on challenging examples from bilingual speakers.

Part II: Audio-visual speech enhancement and separation

In Chapter 6 we propose a deep audio-visual enhancement network that can separate a speaker's voice from other voices and background noise in a cocktail party scenario. This is accomplished by predicting an enhanced audio spectrogram conditioned on the lip movements of the target speaker. The performance of the model is evaluated for up to five simultaneous speakers in unconstrained environments, and for speakers and languages unseen at training time, demonstrating strong qualitative and quantitative performance. This work was the first to solve the task under such general conditions (concurrently Ephrat et al [120] and Owens and Efros [293] developed related methods with similar results).

In Chapter 7 this work is extended to deal with visual occlusions when performing video-driven speech enhancement.

Part III: Audio-visual object localization and detection

Chapter 8 presents a method that extracts a set of discrete audio-visual objects from a video clip using self-supervision. Those objects localize and track sounds sources through space and time and can be used as input representations for useful downstream tasks that have required some form of supervision before, such as Active Speaker Detection (ASD) and multi-speaker sound source separation. We demonstrate its generalization power in a new domain, namely videos of cartoons and puppets.

In Chapter 9 we exploit audio-visual correspondence to train an object detector without any manual annotations. This is accomplished by combining noise-contrastive and clustering-based self-supervised learning to generate self-detections (boxes and labels) and using those as targets to train a detector.

Part IV: Sign language recognition In Chapter 10 we train a Transformer model to identify and temporally localise instances of signs among sequences of continuous sign language [383]. In Chapter 11 we design a model to temporally align asynchronous subtitles in sign language videos.

We conclude the thesis with Chapter 12 which contains discussion on the impact that the works presented here have had since their publication and outline ideas for extensions and future work.

1.3.1 Publications

The body of this thesis consists of a number of papers: each of the chapters 2 to 11 contains a paper that has been published, or is about to appear, in a peer-reviewed conference or journal. The papers have been left unmodified, with the exception of formatting changes. Additional details for some of the papers (e.g. architecture and implementation details, further ablations and qualitative results) can be found in the supplementary materials of their online versions. Appendix A contains authorship statements for each of these papers, describing the candidate's contributions to each paper as well as the paper's publication venue. The included papers are:

- Chapter 2: Deep Audio-Visual Speech Recognition:
  Triantafyllos Afouras*, Joon Son Chung*, Andrew Senior, Oriol Vinyals, Andrew Zisserman.
  Published in IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 2019.

- Chapter 3: Sub-word Level Lip-reading with Visual Attention:
  Prajwal Kondajji Renukananda, **Triantafyllos Afouras**, Andrew Zisserman.

To be published in the proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR) 2022.

- Chapter 4: ASR Is All You Need: Cross-modal Distillation For Lip Reading:
  Triantafyllos Afouras, Joon Son Chung, Andrew Zisserman.
  Published in IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2020, pp. 2143-2147.

- Chapter 5: Now You're Speaking My Language: Visual Language IDentification:
  Triantafyllos Afouras, Joon Son Chung, Andrew Zisserman.
  Published in the proceedings of INTERSPEECH, 2020, pp. 2402-2406.

- Chapter 6: The Conversation: Deep Audio-Visual Speech Enhancement:
  Triantafyllos Afouras, Joon Son Chung, Andrew Zisserman.
  Published in the proceedings of INTERSPEECH, 2018, pp. 3244-3248.

- Chapter 7: My Lips Are Concealed: Audio-visual Speech Enhancement Through Obstructions:
  Triantafyllos Afouras, Joon Son Chung, Andrew Zisserman.
  Published in the proceedings of INTERSPEECH, 2019, pp. 4295-4299.

- Chapter 8: Self-Supervised Learning of Audio-Visual Objects from Video:
  Triantafyllos Afouras, Andrew Owens, Joon Son Chung, Andrew Zisserman.
  Published in the proceedings of the European Conference on Computer Vision (ECCV), 2020, pp. 208-224.

- Chapter 9: Self-supervised Object Detection From Audio-visual Correspondence:
  Triantafyllos Afouras*, Yuki M. Asano*, Francois Fagan, Andrea Vedalid, Florian Metze.
  To be published in the proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR) 2022.

- Chapter 10: Read and Attend: Temporal Localisation in Sign Language Videos:
  Gul Varol*, Liliane Momeni*, Samuel Albanie*, **Triantafyllos Afouras***, Andrew

Zisserman.

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

- Chapter 11: Aligning Subtitles in Sign Language Videos:

  Hannah Bull*, **Triantafyllos Afouras***, Gul Varol, Samuel Albanie, Liliane Momeni, Andrew Zisserman.

  Published in the proceedings of the International Conference on Computer Vision (ICCV) 2021.

Several publications have been left out because they either loosely relate to the topics discussed in this thesis, or because of lesser contributions from the candidate:

- Deep Lip Reading: a comparison of models and an online application

  Triantafyllos Afouras, Joon Son Chung, Andrew Zisserman.

  Published in the proceedings of INTERSPEECH, 2018, pp. 3514–3518.

- LRS3-TED: a large-scale dataset for visual speech recognition

  Triantafyllos Afouras, Joon Son Chung, Andrew Zisserman.

  Technical report, available on arXiv: 1809.00496, 2018.

- Seeing Wake Words: Audio-Visual Keyword Spotting

  Liliane Momeni, **Triantafyllos Afouras**, Themos Stafylakis, Samuel Albanie, Andrew Zisserman.

  Published in the proceedings of the British Machine Vision Conference (BMVC), 2020.

- Spot the conversation: speaker diarisation in the wild

  Joon Son Chung*, Jaesung Huh*, Arsha Nagrani*, **Triantafyllos Afouras**, Andrew Zisserman.

  Published in the proceedings of INTERSPEECH, 2020.

- BSL-1K: Scaling up co-articulated sign language recognition using mouthing cues
  Samuel Albanie*, Gul Varol*, Liliane Momeni, **Triantafyllos Afouras**, Andrew Zisserman.
  Published in the proceedings of the European Conference on Computer Vision (ECCV), 2020, pp. 35-53.

- Watch, read and lookup: learning to spot signs from multiple supervisors
  Liliane Momeni*, Gul Varol*, Samuel Albanie*, **Triantafyllos Afouras**, Andrew Zisserman.
  Published in the proceedings of the Asian Conference on Computer Vision (ACCV), 2020. *Best Application Paper Award.*

- SeeHear: Signer Diarisation and a New Dataset
  Samuel Albanie*, Gul Varol*, Liliane Momeni*, **Triantafyllos Afouras**, Andrew Brown, Chuhan Zhang, Ernesto Coto, Necati Cihan Camgöz, Ben Saunders, Abhishek Dutta, Neil Fox, Richard Bowden, Bencie Wol, Andrew Zisserman.
  Technical report, 2021.

- Localizing Visual Sounds the Hard Way
  Honglie Chen, Weidi Xie, **Triantafyllos Afouras**, Arsha Nagrani, Andrea Vedalid, Andrew Zisserman.
  Published in the proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR), 2021.

Part I

Lip reading and Audio-Visual Speech recognition (AVSR)

2 | Deep Audio-Visual Speech Recognition

Triantafyllos Afouras[1]* Joon Son Chung[1]* Andrew Senior[2]
Oriol Vinyals[2] Andrew Zisserman[1,2]

[1]Visual Geometry Group, Oxford [2]Google DeepMind
(* Equal Contribution)

Abstract

The goal of this work is to recognise phrases and sentences being spoken by a talking face, with or without the audio. Unlike previous works that have focussed on recognising a limited number of words or phrases, we tackle lip reading as an *open-world* problem – unconstrained natural language sentences, and in the wild videos.

Our key contributions are: (1) we compare two models for lip reading, one using a CTC loss, and the other using a sequence-to-sequence loss. Both models are built on top of the transformer self-attention architecture; (2) we investigate to what extent lip reading is complementary to audio speech recognition, especially when the audio signal is noisy; (3) we introduce and publicly release a new dataset for audio-visual speech recognition, LRS2-BBC, consisting of thousands of natural sentences from British television.

The models that we train surpass the performance of all previous work on a lip reading benchmark dataset by a significant margin.

Published in IEEE Transactions on Pattern Analysis and Machine Intelligence, 2019.

Lip reading, the ability to recognize what is being said from visual information alone, is an impressive skill, and very challenging for a novice. It is inherently ambiguous at the word level due to homophones – different characters that produce exactly the same lip sequence (*e.g.* 'p' and 'b'). However, such ambiguities can be resolved to an extent using the context of neighboring words in a sentence, and/or a language model.

A machine that can lip read opens up a host of applications: 'dictating' instructions or messages to a phone in a noisy environment; transcribing and re-dubbing archival silent films; resolving multi-talker simultaneous speech; and, improving the performance of automated speech recognition in general.

That such automation is now possible is due to two developments that are well known across computer vision tasks: the use of deep neural network models [224, 351, 367]; and, the availability of a large scale dataset for training [328]. In this case, the lip reading models are based on recent encoder-decoder architectures that have been developed for speech recognition and machine translation [31, 63, 159, 160, 365].

The objective of this paper is to develop neural transcription architectures for lip reading sentences. We compare two models: one using a *Connectionist Temporal Classification* (CTC) loss [159], and the other using a *sequence-to-sequence* (seq2seq) loss [78, 365]. Both models are based on the transformer self-attention architecture [384], so that the advantages and disadvantages of the two losses can be compared head-to-head, with as much of the rest of the architecture in common as possible. The dataset developed in this paper to train and evaluate the models, are based on thousands of hours of videos that have talking faces together with subtitles of what is being said.

We also investigate how lip reading can contribute to *audio* based speech recognition. There is a large literature on this contribution, particularly in noisy environments, as well as the converse where some derived measure of audio can contribute to lip reading for the deaf or hard of hearing. To investigate this aspect we train a model to recognize characters from both audio and visual input, and then systematically disturb the audio channel.

Our models output at the character level. In the case of the CTC, these outputs are independent of each other. In the case of the sequence-to-sequence loss a language model is learnt implicitly, and the architecture incorporates a novel dual attention mechanism that can operate over visual input only, audio input only, or both. The architectures are described in Section 2.2. Both models are decoded with a beam search, in which we can optionally incorporate an external language model.

Section 2.3, we describe the generation and statistics of a new large scale dataset, *LRS2-BBC*, that is used to train and evaluate the models. The dataset contains talking faces together with subtitles of what is said. The videos contain faces 'in the wild' with a significant variety of pose, expressions, lighting, backgrounds and ethnic origin. Section 2.4 describes the network training, where we report a form of curriculum learning that is used to accelerate training. Finally, Section 2.5 evaluates the performance of the models, including for visual (lips) input only, for audio and visual inputs, and for synchronization errors between the audio and visual streams.

On the content: This submission is based on the conference paper [86]. We replace the WLAS model in the original paper with two variants of a Transformer-based model [384]. One variant was published in [4], and the second variant (using the CTC loss) is an original contribution in this paper. We also update the visual front-end with a ResNet-based one proposed by [360]. The new front-end and back-end architectures contribute to over 22% absolute improvement in Word Error Rate (WER) over the model proposed in [86]. Finally, we publicly release a new dataset, LRS2-BBC, that supersedes the original LRS dataset in [86] which could not be made public due to license restrictions.

2.1 Background

2.1.1 CTC vs sequence-to-sequence architectures

For the most part, end-to-end deep learning approaches for sequence prediction can be divided into two types.

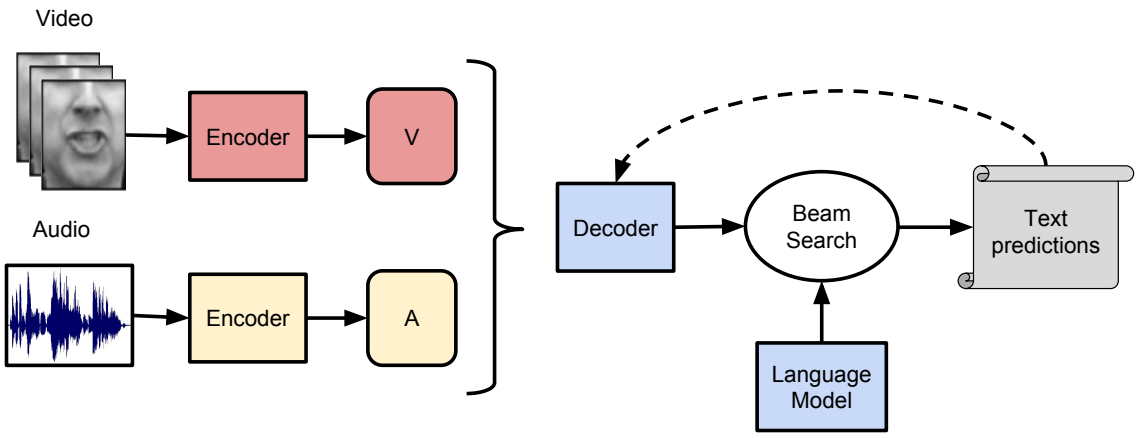


Figure 2.1: Outline of the audio-visual speech recognition pipeline.

The first type uses a neural network as an emission model which outputs the likelihood of each output symbol (*e.g.* phonemes) given the input sequence (*e.g.* audio). These methods generally employ a second phase of decoding using a Hidden Markov Model [184]. One such version of this variant is the Connectionist Temporal Classification (CTC) [159], where the model predicts frame-wise labels and then looks for the optimal alignment between the frame-wise predictions and the output sequence. The main weakness of CTC is that the output labels are not conditioned on each other (it assumes each unit is independent), and hence a language model is employed as a post-processing step. Note that some alternatives to jointly train the two step process has been proposed [158]. Another limitation of this approach is that it assumes a monotonic ordering between input and output sequences. This assumption is suitable for ASR and transcription for example, but not for machine translation.

The second type is sequence-to-sequence models [78, 365] (seq2seq) that first read all of the input sequence before predicting the output sentence. A number of papers have adopted this approach for speech recognition [80, 81]: for example, Chan *et al.* [63] proposes an elegant sequence-to-sequence method to transcribe audio signal to characters. Sequence-to-sequence decodes an output symbol at time t (*e.g.* character or word) conditioned on previous outputs $1,...,t-1$. Thus, unlike CTC-based models, the model implicitly learns a language model over output symbols, and no further processing is required. However, it has been shown [63, 204] that it is beneficial to incorporate an external language model in the decoding of sequence-to-sequence models as well. This way it is possible to leverage larger text-only

corpora that contain much richer natural language information than the limited aligned data used for training the acoustic model.

Regarding architectures, while CTC-based or seq2seq approaches traditionally relied on recurrent networks, recently there has been a shift towards purely convolutional models [32]. For example, fully convolutional networks have been used for ASR with CTC [400, 437] or a simplified variant [93, 246, 429].

2.1.2 Related works

Lip reading. There is a large body of work on lip reading using non deep learning methods. These methods are thoroughly reviewed in [447], and we will not repeat this here. A number of papers have used Convolutional Neural Networks (CNNs) to predict phonemes [285] or visemes [217] from still images, as opposed to recognising to full words or sentences. A *phoneme* is the smallest distinguishable unit of sound that collectively make up a spoken word; a *viseme* is its visual equivalent.

For recognising full words, Petridis *et al.* [303] train an LSTM classifier on a discrete cosine transform (DCT) and deep bottleneck features (DBF). Similarly, Wand *et al.* [391] use an LSTM with HOG input features to recognise short phrases. The shortage of training data in lip reading presumably contributes to the continued use of hand crafted features. Existing datasets consist of videos with only a small number of subjects, and also a limited vocabulary (<60 words), which is also an obstacle to progress. Chung and Zisserman [87] tackles the small-lexicon problem by using faces in television broadcasts to assemble the LRW dataset with a vocabulary size of 500 words. However, as with any word-level classification task, the setting is still distant from the real-world, given that the word boundaries must be known beforehand. Assael *et al.* [26] uses a CNN and LSTM-based network and (CTC) [159] to compute the labelling. This reports strong speaker-independent performance on the constrained grammar and 51 word vocabulary of the GRID dataset [94].

A deeper architecture than LipNet [26] is used by [360], who propose a residual network with 3D convolutions to extract more powerful representations. The network is trained

with a cross-entropy loss to recognise words from the LRW dataset. Here, the standard ResNet architecture [175] is modified to process 3D image sequences by changing the first convolutional and pooling blocks from 2D to 3D.

In our earlier work [86], we proposed a WLAS sequence-to-sequence model based on the LAS ASR model of [63] (the acronym WLAS are for Watch, Listen, Attend and Spell, and LAS for Listen, Attend and Spell). The WLAS model had a dual attention mechanism – one for the visual (lip) stream, and the other for the audio (speech) stream. It transcribed spoken sentences to characters, and could handle an input of vision only, audio only, or both.

In independent and concurrent work, Shillingford *et al.* [347], design a lip reading pipeline that uses a network which outputs phoneme probabilities and is trained with CTC loss. At inference time, they use a decoder based on finite state transducers to convert the phoneme distributions into word sequences. The network is trained on a very large scale lip reading dataset constructed from YouTube videos and achieves a remarkable 40.9% word error rate.

Audio-visual speech recognition. The problems of audio-visual speech recognition (AVSR) and lip reading are closely linked. Mroueh *et al.* [277] employs feed-forward Deep Neural Networks (DNNs) to perform phoneme classification using a large non-public audio-visual dataset. The use of HMMs together with hand-crafted or pre-trained visual features have proved popular – [370] encodes input images using DBF; [137] used DCT; and [286] uses a CNN pre-trained to classify phonemes; all three combine these features with HMMs to classify spoken digits or isolated words. As with lip reading, there has been little attempt to develop AVSR systems that generalise to real-world settings.

Petridis *et al.* [304] use an extended version of the architecture of [360] to learn representations from raw pixels and waveforms which they then concatenate and feed to a bidirectional recurrent network that jointly models the audio and video sequences and outputs word labels.

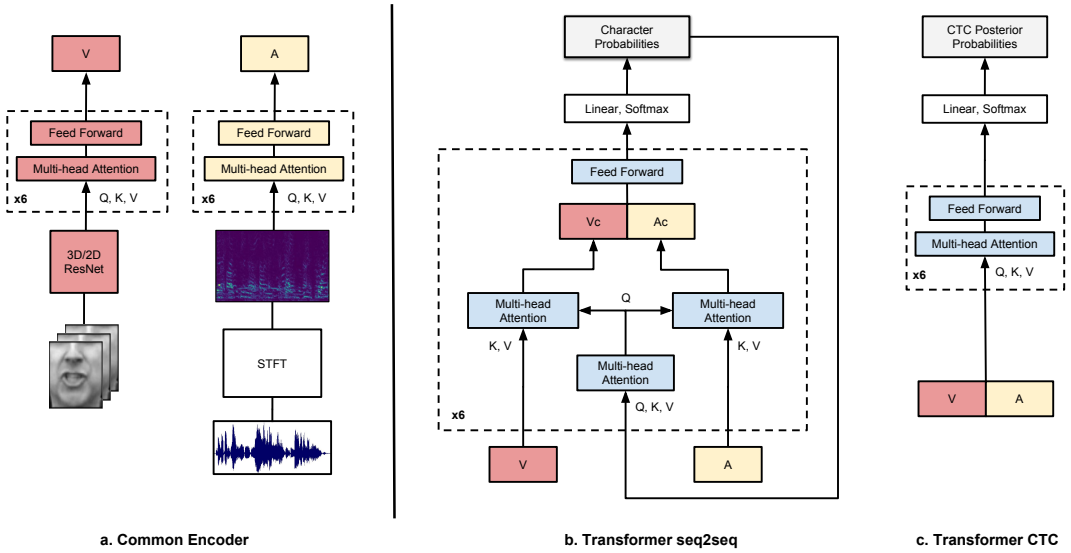


Figure 2.2: Audio-visual speech recognition models. **(a) Common encoder:** The visual image sequence is processed by a spatio-temporal ResNet, while the audio features are the spectrograms obtained by applying Short Time Fourier Transform (STFT) to the audio signal. Each modality is then encoded by a separate Transformer encoder. **(b) TM-seq2seq**: a Transformer model. On every decoder layer, the video (V) and audio (A) encodings are attended to separately by independent multi-head attention modules. The context vectors produced for the two modalities, V_c and A_c respectively, are concatenated channel-wise and fed to the feed forward layers. K, V and Q denote the Key, Value and Query tensors for the multi-head attention blocks. For the self-attention layers it is always $Q = K = V$, while for the encoder-decoder attentions, $K = V$ are the encodings (V or A), while Q is the previous layer's output (or, for the first layer, the prediction of the network at the previous decoding step). **(c) TM-CTC**: Transformer CTC, a model composed of stacks of self-attention and feed forward layers, producing CTC posterior probabilities for every input frame. For full details on the multi-head attention and feed forward blocks refer to the Appendix.

2.2 Architectures

In this section, we describe model architectures for audio-visual speech recognition, for which we explore two variants, based on the recently proposed Transformer model [384]: i) an encoder-decoder attention structure for training in a seq2seq manner and ii) a stack of self-attention blocks for training with CTC loss. The architecture is outlined in Figure 2.2. The general model receives two input streams, one for video (V) and one for audio (A).

2.2.1 Audio Features

For the acoustic representation we use 321-dimensional spectral magnitudes, computed with a 40ms window and 10ms hop-length, at a 16 kHz sample rate. Since the video is sampled at 25 fps (40 ms per frame), every video input frame corresponds to 4 acoustic feature frames. We concatenate the audio features in groups of 4, in order to reduce the input sequence length as is common for stable CTC training [76, 330], while at the same time achieving a common temporal-scale for both modalities.

2.2.2 Vision Module (VM)

The input images are 224×224 pixels, sampled at 25 fps and contain the speaker's face. We crop a 112×112 patch covering the region around the mouth, as shown in Figure 2.3. To extract visual features representing the lip movement, we use a spatio-temporal visual front-end that is based on [360]. The network applies 3D convolutions on the input image sequence, with a filter width of 5 frames, followed by a 2D ResNet that gradually decreases the spatial dimensions with depth. The layers are listed in full detail in the Appendix. For an input sequence of $T \times H \times W$ frames, the output is a $T \times \frac{H}{32} \times \frac{W}{32} \times 512$ tensor (*i.e.* the temporal resolution is preserved) that is then average-pooled over the spatial dimensions, yielding a 512-dimensional feature vector for every input video frame.

2.2.3 Common self-attention Encoder

Both variants that we consider use the same self-attention-based encoder architecture. The encoder is a stack of multi-head self-attention layers, where the input tensor serves as the query, key and value for the attention at the same time. A separate encoder is used for each modality as shown in Figure 2.2 (a). The information about the sequence order of the inputs is fed to the model via fixed positional embeddings in the form of sinusoid functions.

2.2.4 Sequence-to-sequence Transformer (TM-seq2seq)

In this variant, separate attention heads are used for attending on the video and audio embeddings. In every decoder layer, the resulting video and audio contexts are concatenated over the channel dimension and propagated to the feedforward block. The attention mechanisms for both modalities receive as queries the output of the previous decoding layer (or the decoder input in the case of the first layer). The decoder produces character probabilities which are directly matched to the ground truth labels and trained with a cross-entropy loss. More details about the multi-head attention and feed-forward building blocks are given in the Appendix.

2.2.5 CTC Transformer (TM-CTC)

The TM-CTC model concatenates the video and audio encodings and propagates the result through a stack of self-attention / feedforward blocks, same as the one used in the encoders. The outputs of the network are the CTC posterior probabilities for every input frame and the whole stack is trained with CTC loss.

2.2.6 External Language Model (LM)

For decoding both variants, during inference, we use a character-level language model. It is a recurrent network with 4 unidirectional layers of 1024 LSTM cells each. The language model is trained to predict one character at a time, receiving only the previous character as input. Decoding for both models is performed with a left-to-right beam search where the LM log-probabilities are combined with the model's outputs via shallow fusion [204]. More details on decoding are given in the Appendix.

2.2.7 Single modality models

The audio-visual models described in this section can be used when only one of the two modalities is present. Instead of concatenating the attention vectors for TM-seq2seq or the encodings for TM-CTC, only the vector from the available modality is used.

2.3 Dataset

Figure 2.3: Top: Original still images from videos used in the making of the LRS2-BBC dataset. **Bottom:** The mouth motions from two different speakers. The network sees the areas inside the red squares.

In this section, we describe the multi-stage pipeline for automatically generating a large-scale dataset, *LRS2-BBC*, for audio-visual speech recognition. Using this pipeline, we have been able to collect thousands of hours of spoken sentences and phrases along with the corresponding facetrack. We use a variety of BBC programs from Dragon's Den to Top Gear and Countryfile.

The processing pipeline is summarised in Figure 2.4. Most of the steps are based on the methods described in [87] and [88], but we give a brief sketch of the method here.

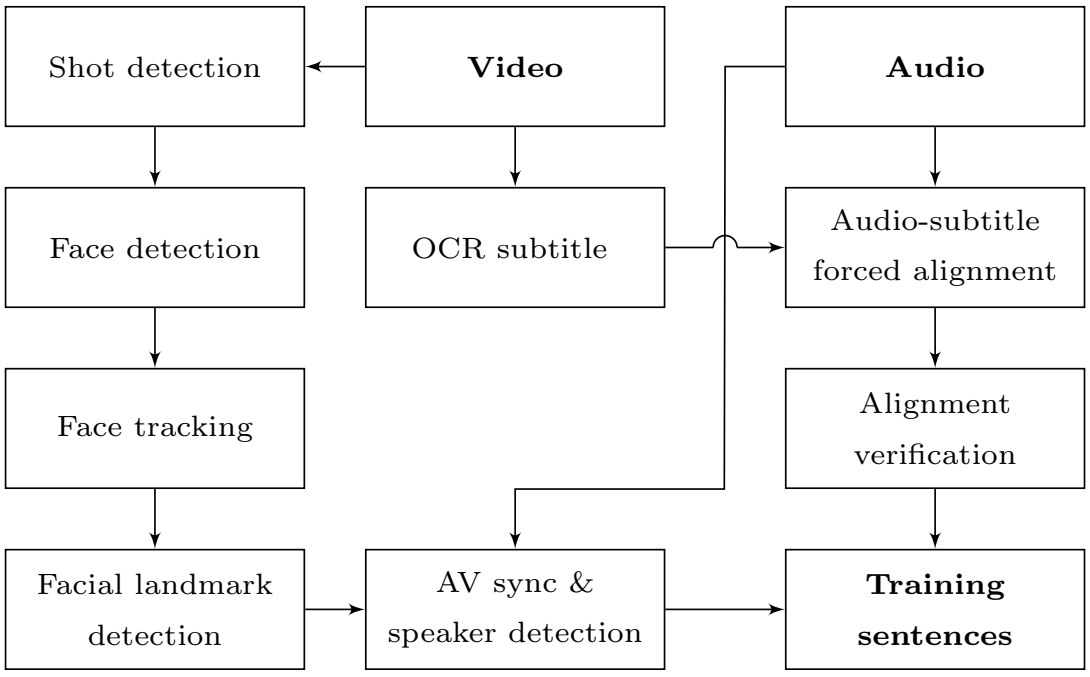


Figure 2.4: Pipeline to generate the dataset.

Video preparation. A CNN face detector based on the Single Shot MultiBox Detector (SSD) [250] is used to detect face appearances in the individual frames. Unlike the HOG-based detector [211] used by previous works, the SSD detects faces from all angles, and shows a more robust performance whilst being faster to run.

The shot boundaries are determined by comparing color histograms across consecutive frames [244]. Within each shot, face tracks are generated from face detections based on their positions, as feature-based trackers such as KLT [252] often fail when there are extreme changes in viewpoints.

Audio and text preparation. The subtitles in television are not broadcast in sync with the audio. The Penn Phonetics Lab Forced Aligner [426] is used to force-align the subtitle to the audio signal. Errors exist in the alignment as the transcript is not verbatim – therefore the aligned labels are filtered by checking against the commercial IBM Watson Speech to Text service.

AV sync and speaker detection. In broadcast videos, the audio and the video streams can be out of sync by up to around one second, which can cause problems when the facetrack corresponding to a sentence is being extracted. A multi-view adaptation [90] of the two-stream network described in [88] is used to synchronise the two streams. The same network is also used to determine which face's lip movements match the audio, and if none matches, the clip is rejected as being a voice-over.

Sentence extraction. The videos are divided into individual sentences/ phrases using the punctuations in the transcript. The sentences are separated by full stops, commas and question marks; and are clipped to 100 characters or 10 seconds, due to GPU memory constraints. We do not impose any restrictions on the vocabulary size.

The LRS2-BBC dataset is divided into development (train/val) and test sets according to broadcast date. The dataset also has a "*pre-train*" set that contains sentence excerpts which may be shorter or longer than the full sentences included in the development set, and are annotated with the alignment boundaries of every word. The statistics of these sets are given in Table 2.1. The table also compares the *'Lip Reading Sentences'* (LRS) series of

datasets to the largest existing public datasets. In addition to LRS2-BBC, we use MV-LRS and LRS3-TED for training and evaluation.

Dataset	Source	Split	Dates	# Spk.	# Utt.	Word inst.	Vocab	# hours
GRID [94]	-	-	-	51	33,000	165k	51	27.5
MODALITY [98]	-	-	-	35	5,880	8,085	182	31
LRW [87]	BBC	Train-val	01/2010 - 12/2015	-	514k	514k	500	165
		Test	01/2016 - 09/2016	-	25k	25k	500	8
LRS [86] †	BBC	Train-val	01/2010 - 02/2016	-	106k	705k	17k	68
		Test	03/2016 - 09/2016	-	12k	77k	6,882	7.5
MV-LRS [90] †	BBC	Pre-train	01/2010 - 12/2015	-	430k	5M	30k	730
		Train-val	01/2010 - 12/2015	-	70k	470k	15k	44.4
		Test	01/2016 - 09/2016	-	4,305	30k	4,311	2.8
LRS2-BBC	BBC	Pre-train	01/2010 - 02/2016	-	96k	2M	41k	195
		Train-val	01/2010 - 02/2016	-	47k	337k	18k	29
		Test	03/2016 - 09/2016	-	1,243	6,663	1,693	0.5
		Text-only	01/2016 - 02/2016	-	8M	26M	60k	-
LRS3-TED [5]	TED & TEDx (YouTube)	Pre-train	-	5,075	132k	4.2M	52k	444
		Train-val	-	3,752	32k	358k	17k	30
		Test	-	452	1,452	11k	2,136	1
		Text-only	-	5,075	1.2M	7.2M	57k	-

Table 2.1: Statistics on the **Lip Reading Sentences (LRS) audio-visual datasets,** and other existing large-scale lip reading datasets. Division of training, validation and test data; and the number of utterances, number of word instances and vocabulary size of each partition. **Utt:** Utterances. †: Not available to the public due to license restrictions.

Datasets for training external language models. To train the language models used for evaluation on each audio-visual dataset, we use a text corpus containing the full subtitles of the videos from which the dataset's training set was generated. The text-only corpus contains $26M$ words.

2.4 Training strategy

In this section, we describe the strategy used to effectively train the models, making best use of the limited amount of data available. The training proceeds in four stages: i) the visual front-end module is trained; ii) visual features are generated for all the training data using the vision module; iii) the sequence processing module is trained on the frozen visual features; iv) the whole network is trained end-to-end.

2.4.1 Pre-training visual features

We pre-train the visual front-end on word excerpts from the MV-LRS [90] dataset, using a 2-layer temporal convolution back-end to classify every clip with a word label similarly to [360]. We perform data augmentation in the form of horizontal flipping, removal of random frames [26, 360], and random shifts of up to ± 5 pixels in the spatial dimensions and of ± 2 frames in the temporal dimension.

2.4.2 Curriculum learning

Sequence to sequence learning has been reported to converge very slowly when the number of timesteps is large, because the decoder initially has a hard time extracting the relevant information from all the input steps [63]. Even though our models do not contain any recurrent modules, we found it beneficial to follow a curriculum instead of immediately training on full sentences.

We introduce a new strategy where we start training only on single word examples, and then let the sequence length grow as the network trains. These short sequences are parts of the longer sentences in the dataset. We observe that the rate of convergence on the training set is several times faster, while the curriculum also significantly reduces overfitting, presumably because it works as a natural way of augmenting the data.

The networks are first trained on the frozen features of the *pre-train* sets from MV-LRS, LRS2-BBC and LRS3-TED. We deal with the difference in utterance lengths by zero-padding the sequences to a maximum length, which we gradually increase. We then separately fine-tune end-to-end on the *train-val* set of LRS2-BBC or LRS3-TED, according to which set we are evaluating on.

2.4.3 Training with noisy audio & multi-modal training

The audio-only models are initially trained with clean input audio. Networks with multi-modal inputs can often be dominated by one of the modes [125]. In our case we observe that for the audio-visual models the audio signal dominates, because speech recognition is a significantly

easier problem than lip reading. To help prevent this from happening, we add babble noise with 0dB SNR to the audio stream with probability $p_n = 0.25$ during training.

To assess and improve tolerance to audio noise, we then fine-tune the audio-only and audio-visual models in a setting where babble noise with 0dB SNR is always added to the original audio. We synthesize the babble noise samples by mixing the signals of 20 different audio samples from the LRS2-BBC dataset.

2.4.4 Implementation details

The output size of the network is 40, accounting for the 26 characters in the alphabet, the 10 digits, and tokens for `[space]` and `[pad]`. For TM-seq2seq we use an extra `[sos]` token and for TM-CTC the `[blank]` token. We do not model punctuation, as the transcriptions of the datasets do not contain any.

The TM-seq2seq is trained using teacher forcing – we supply the ground truth of the previous decoding step as the input to the decoder, while during inference we feed back the decoder prediction.

Our implementation is based on the TensorFlow library [1] and trained on a single GeForce GTX 1080 Ti GPU with 11GB memory. The network is trained using the ADAM optimiser [212] with the default parameters and an initial learning rate of 10^{-4}, which is reduced by a factor of 2 every time the validation error plateaus, down to a final learning rate of 10^{-6}. For all the models we use dropout with $p = 0.1$ and label smoothing.

2.5 Experiments

In this section we evaluate and compare the proposed architectures and training strategies. We also compare our methods to the previous state of the art.

We train as described in section 2.4.2 and evaluate the fine-tuned models for LRS2-BBC and LRS3-TED on the independent test set of the respective dataset. The inference and

Method \ Dataset	M	LRS2-BBC	LRS2-BBC + extLM	LRS3-TED	LRS3-TED + extLM
Google S2T†	A	20.9%		10.4%	
WAS [86]	V	70.4%	-	-	-
TM-CTC	V	65.0%	54.7%	74.7%	66.3%
TM-CTC	A	15.3%	10.1%	13.8%	8.9%
TM-CTC	AV	13.7%	8.2%	12.3%	7.5%
TM-seq2seq	V	49.8%	48.3%	59.9%	58.9%
TM-seq2seq	A	10.5%	9.7%	9.0%	8.3%
TM-seq2seq	AV	9.4%	8.5%	8.0%	7.2%
Noisy					
Google S2T†	A	86.3%		70.3%	
TM-CTC	A	64.7%	53.4%	65.6%	56.3%
TM-CTC	AV	33.5%	23.6%	37.2%	27.7%
TM-seq2seq	A	58.0%	57.4%	60.5%	57.9%
TM-seq2seq	AV	35.9%	34.2%	44.3%	42.5%

Table 2.2: Word error rates (WER) on the LRS2-BBC and LRS3-TED datasets. The second column (M) specifies the input modalities: V, A, and AV denote video-only, audio-only, and audio-visual models respectively, while + extLM denotes decoding with the external language model. † `https://cloud.google.com/speech-to-text`, accessed 3 July 2018.

evaluation procedures are described below.

Test time augmentation. During inference we perform 9 random transforms (horizontal flipping of the video frames and spatial shifts up to ± 5 pixels) on every video sample, and pass the perturbed sequences through the network, in addition to the original. For TM-seq2seq we average the resulting logits whereas for TM-CTC we average the visual features.

Beam search. Decoding is performed with beam search of width 35 for TM-Seq2seq and 100 for TM-CTC (the values were determined on a held-out validation set from the *train-val* split of LRS2-BBC).

Evaluation protocol. For all experiments, we report the Word Error Rate (WER) which is defined as $\texttt{WER} = (S + D + I)/N$, where S, D and I are the number of substitutions, deletions, and insertions respectively to get from the reference to the hypothesis, and N is the number of words in the reference.

Experimental setup. The rest of this section is structured as follows: First we present results on lip reading, where only the video is used as input. We then use the full models for audio-visual speech recognition, where the video and audio are assumed to be properly synchronised. To assess the robustness of our models in noisy environments we also train and test in a setting where babble noise is artificially added to the utterances. Finally we present some experiments on non-synchronised video and audio. The results for all experiments are summarized in Table 2.2, where we report word error rates depending on whether a language model is used during decoding or not.

2.5.1 Lips only

Results. The best performing network is TM-seq2seq, which achieves a WER of 48.3% on LRS2-BBC when decoded with a language model, an absolute improvement of over 22% compared to the previous 70.4% state-of-the-art [86]. This model also sets a baseline for LRS3-TED at 58.9%.

In Figure 2.5 we show how the WER changes as a function of the number of words in a test sentence. Figure 2.6 shows the performance of the models on the 30 most common words. Figure 2.7 shows the effect of increasing the beam width for the video-only TM-seq2seq model when evaluating on LRS2-BBC. It is noteworthy that increasing the beam width is more beneficial when decoding with the external language model (+ extLM).

Decoding examples. The model learns to correctly predict complex unseen sentences from a wide range of content – examples are shown in Table 2.3.

2.5.2 Audio-visual speech recognition

The visual information can be used to improve the performance of ASR, particularly in environments with background noise [277, 286, 304]. Here, we analyse the performance of the audio-visual models described in Section 2.2.

but this particular reality was not inevitable
it would have been completely alien to the rest of london
comes from one of the most beautiful parts of the world
everyone has gone home happy and that's what it's all about
especially when it comes to climate change
but it's a different type of animal I want to show you right now
but these are one of the most wary birds in the world
there's always historical treasures to look at
and so how does your brain give you that detail
but this is the source of innovation
the choices don't make sense because it's the wrong question
but it's a global phenomenon
mortality is not going down it's going up

Table 2.3: Examples of unseen sentences that TM-seq2seq correctly predicts (video only).

Results. The results in Table 2.2 demonstrate that the mouth movements provide important cues in speech recognition when the audio signal is noisy; and give an improvement in performance even when the audio signal is clean – for example the word error rate is reduced from 10.1% for audio only to 8.2%, when using the audio-visual TM-CTC model. The gains when using the audio-visual TM-seq2seq compared to the audio-only model are similar.

Decoding examples. Table 2.4 shows some of the many examples where the model fails to predict the correct sentence from the lips or the audio alone, but successfully deciphers the words when both streams are present.

Alignment and attention visualisation. The encoder-decoder attention mechanism of the TM-seq2seq model generates explicit alignment between the input video frames and the hypothesised character output. Figure 2.9 visualises the alignment of the characters "comes from one of the most beautiful parts of the world" and the corresponding video frames. Since the architecture contains multiple attention heads, we obtain the alignment by averaging the attention masks over all the decoder layers in the log domain.

Noisy audio. We perform the audio-only and audio-visual experiments with noisy audio,

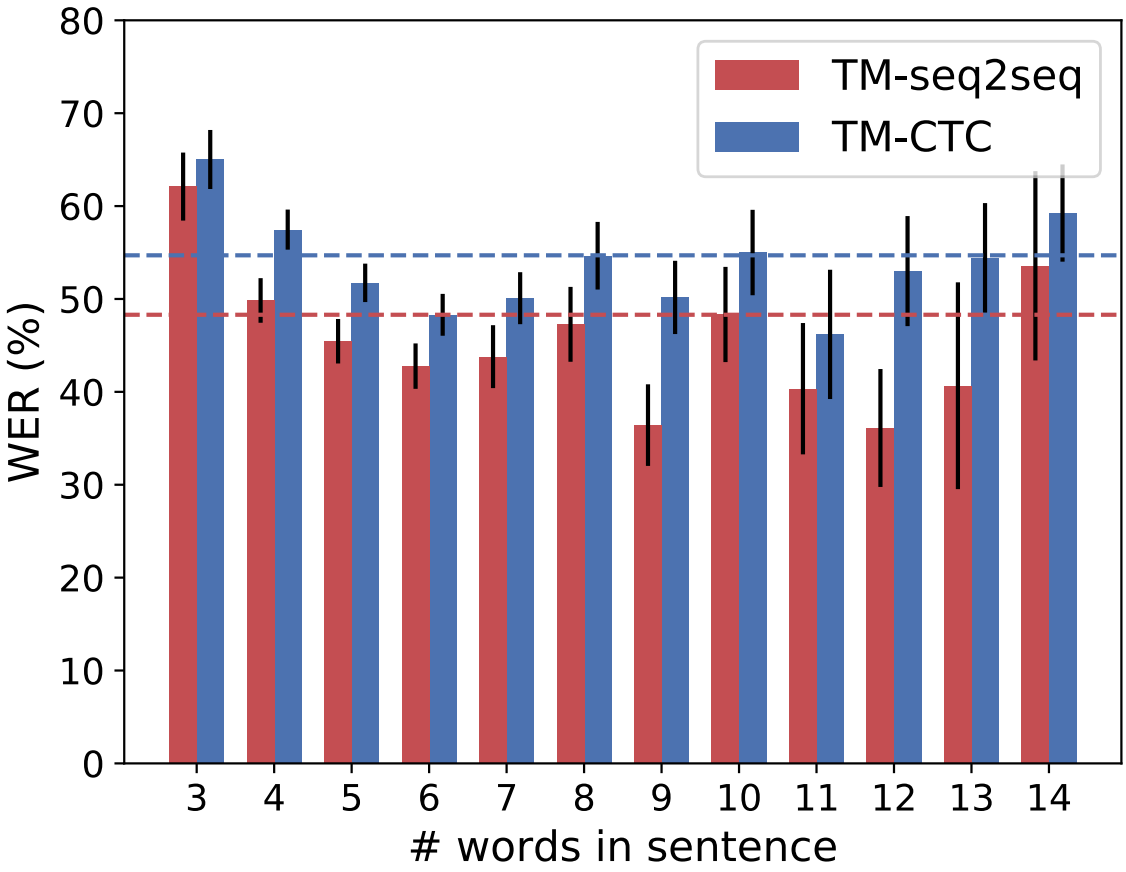


Figure 2.5: Word error rate per number of words in the sentence for the video-only models, evaluated on the test set of LRS2-BBC. We exclude sentence sizes represented by less than 5 samples in the set (i.e. 15, 16 and 19 words). The dashed lines show the average WER over all the sentences. For both models, the WER is relatively uniform for different sentence sizes. However samples with very few words (3) appear to be more difficult, presumably because they provide less context.

synthesized by adding babble noise to the original utterances. Speech recognition in a noisy environment is extremely challenging, as can be seen from the significantly lower performance of the off-the-shelf Google S2T ASR baseline (over 60% performance degradation compared to clean). This difficulty is also reflected on the performance of our audio-only models, that the word error rates similar to the ones obtained when only using the lips. However combining the two modalities provides a significant improvement, with the word error rate dropping significantly, by up to 30%. Notably, the audio-visual models perform much better than either the video-only, or audio-only ones under the presence of loud background noise.

AV attention visualization. In Figure 2.10 we compare the attention masks of different TM-seq2seq models in the presence and absence of additive babble noise in the audio stream.

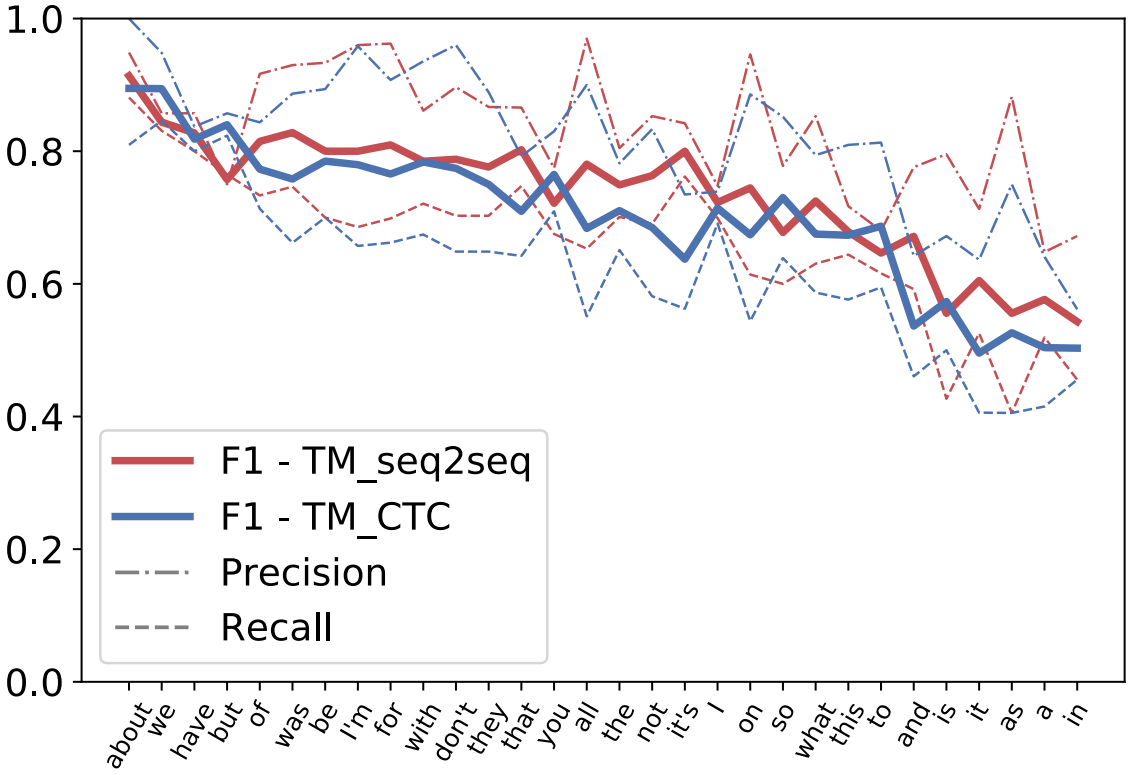


Figure 2.6: Per word F1, Precision and Recall rates, on the 30 most common words in the LRS2-BBC test set, for the video-only models. The measures are calculated via the minimum edit-distance operations (details in the Appendix). For all words and both models, precision is higher than recall.

2.5.3 Out-of-sync audio and video

Here, we assess the performance of the audio-visual models when the audio and video inputs are not temporally aligned. Since the audio and video have been synchronised in our dataset, we synthetically shift the video frames to achieve an out-of-sync effect. We evaluate the performance on de-synchronised samples of the LRS2-BBC dataset. We consider the TM-CTC and TM-seq2seq architectures, with and without fine-tuning on randomly shifted samples. The results are shown in Figure 2.8. It is clear that the TM-seq2seq architecture is more resistant to these shifts. We only need to calibrate the model for one epoch for the out-of-sync effect to practically vanish. This showcases the advantage of employing independent encoder-decoder attention mechanisms for the two modalities. In contrast, TM-CTC, that concatenates the two encodings, struggles to deal with the shifts, even after several epochs of fine-tuning.

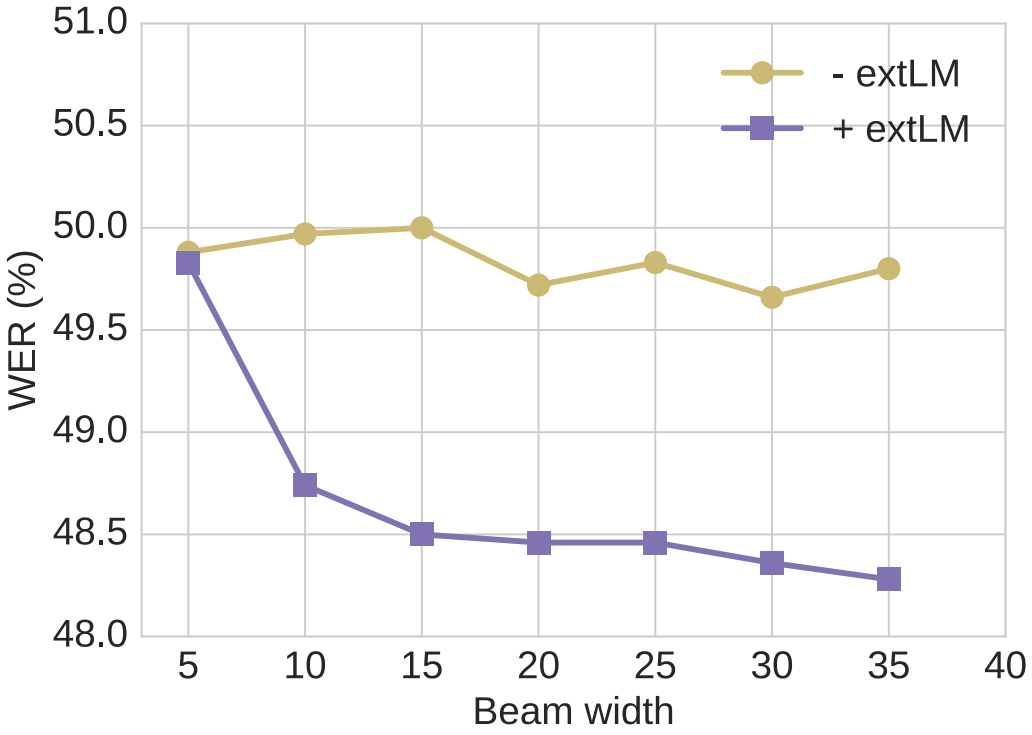


Figure 2.7: The effect of beam width on Word Error Rate for the video-only TM-seq2seq model, when evaluating on LRS2-BBC.

2.5.4 Discussion on seq2seq vs CTC

The TM-seq2seq model performs significantly better for lip-reading in terms of WER, when no audio is supplied. For audio-only or audio-visual tasks, the two methods perform similarly. However the CTC models appear to handle background noise better; in the presence of loud babble noise, both the audio-only and audio-visual TM-seq2seq models perform significantly worse that their TM-CTC counterparts.

Training time. The TM-seq2seq models have a more complex architecture and are harder to train, with the full audio-visual model taking approximately 8 days to complete the full curriculum for both datasets, on a single GeForce Titan X GPU with 12GB memory. In contrast, the audiovisual TM-CTC model trains faster i.e. in approximately 5 days on the same hardware. It should be noted however that since both architectures contain no recurrent modules and no batch normalization, their implementation can be heavily parallelized into multiple GPUs.

	Transcription	WER %
GT	your job needs to be challenging	
V	job is to be challenging	33
A	your child needs to be challenging	16
AV	your job needs to be challenging	0
GT	I mean I thought poetry was just self expression	
V	I mean I thought poetry would just suffer as pressure	44
A	I mean not thought poetry was just self expression	11
AV	I mean I thought poetry was just self expression	0
GT	cluster bombs left behind	
V	unless you perhaps have blind	125
A	close to bombs left behind	25
AV	cluster bombs left behind	0
GT	I was the first non family investor in amazon	
V	I was the first not family of us are absurd	55
A	I was the first non family in bester and amazon	33
AV	I was the first non family investor in amazon	0

Table 2.4: Examples of AVSR results. **GT:** Ground Truth; **A:** Audio only; **V:** Video only; **AV:** Audio-visual.

Inference time. Decoding of the TM-CTC model does not require auto-regression and therefore the CTC probabilities need only be evaluated once, regardless of the beam width W. This is not the case for TM-seq2seq, where for every step of the beam search, the decoder subnetwork needs to be evaluated W times. This makes the decoding of the CTC model faster, which can be an important factor for deployment.

Language modelling. Both models perform better when an external language model is incorporated in the beam search, however the gains are much higher for TM-CTC, since no explicit language consistency is enforced by the visual model alone.

Generalization to longer sequences. We observed that the TM-CTC model generalizes better and adapts faster as the sequence lengths are increased during the curriculum learning. We believe this also affects the training time as the latter takes more epochs to converge.

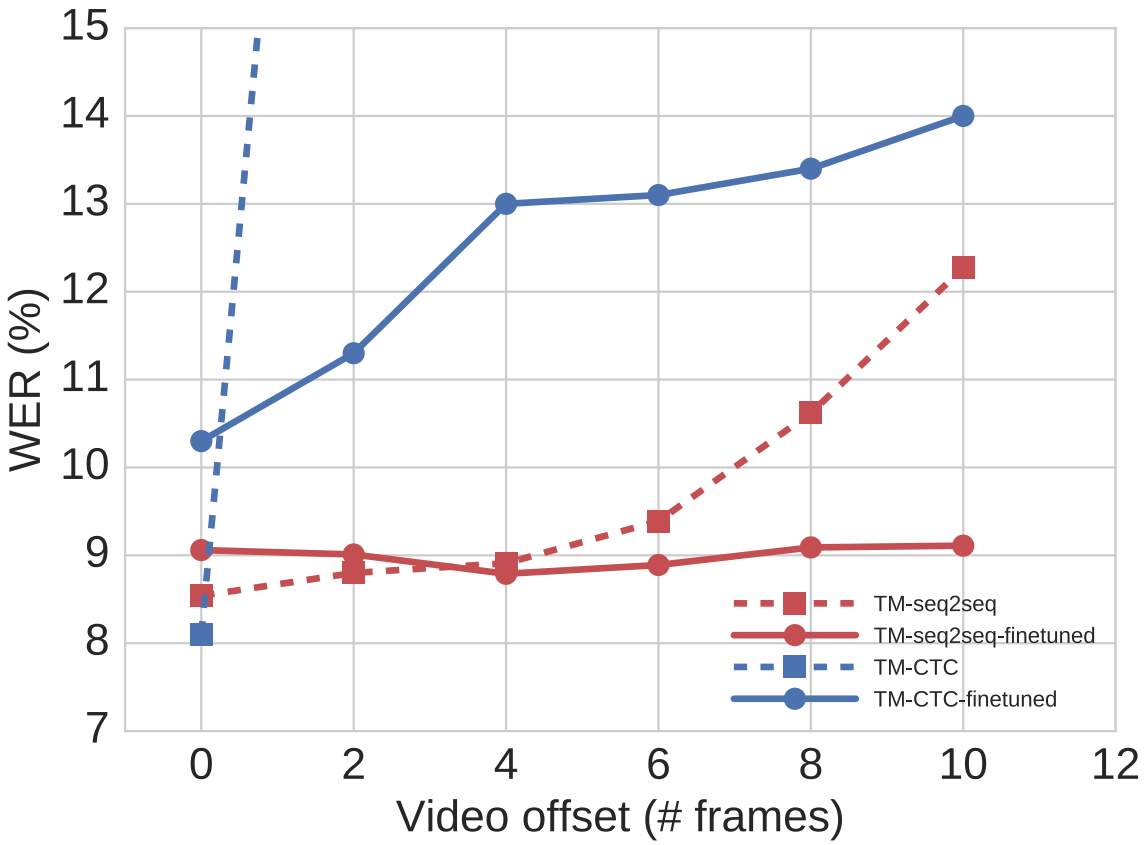


Figure 2.8: WER scored by the audio-visual models on LRS2-BBC when the video frames are artificially shifted by a number of frames compared to audio. The TM-seq2seq model is only fine-tuned for one epoch, while CTC for 4 epochs on the train-val set.

2.6 Conclusion

In this paper, we introduced a large-scale, unconstrained audio-visual dataset, LRS2-BBC, formed by collecting and preprocessing thousands of videos from the British television.

We considered two models that can transcribe audio and video sequences of speech into characters and showed that the same architectures can also be used when only one of the modalities is present. Our best visual-only model surpasses the performance of the previous state-of-the-art on the LRS2-BBC lip reading dataset by a large margin, and sets a strong baseline for the recently released LRS3-TED. We finally demonstrate that visual information helps improve speech recognition performance even when the clean audio signal is available. Especially in the presence of noise in the audio, combining the two modalities leads to a significant improvement.

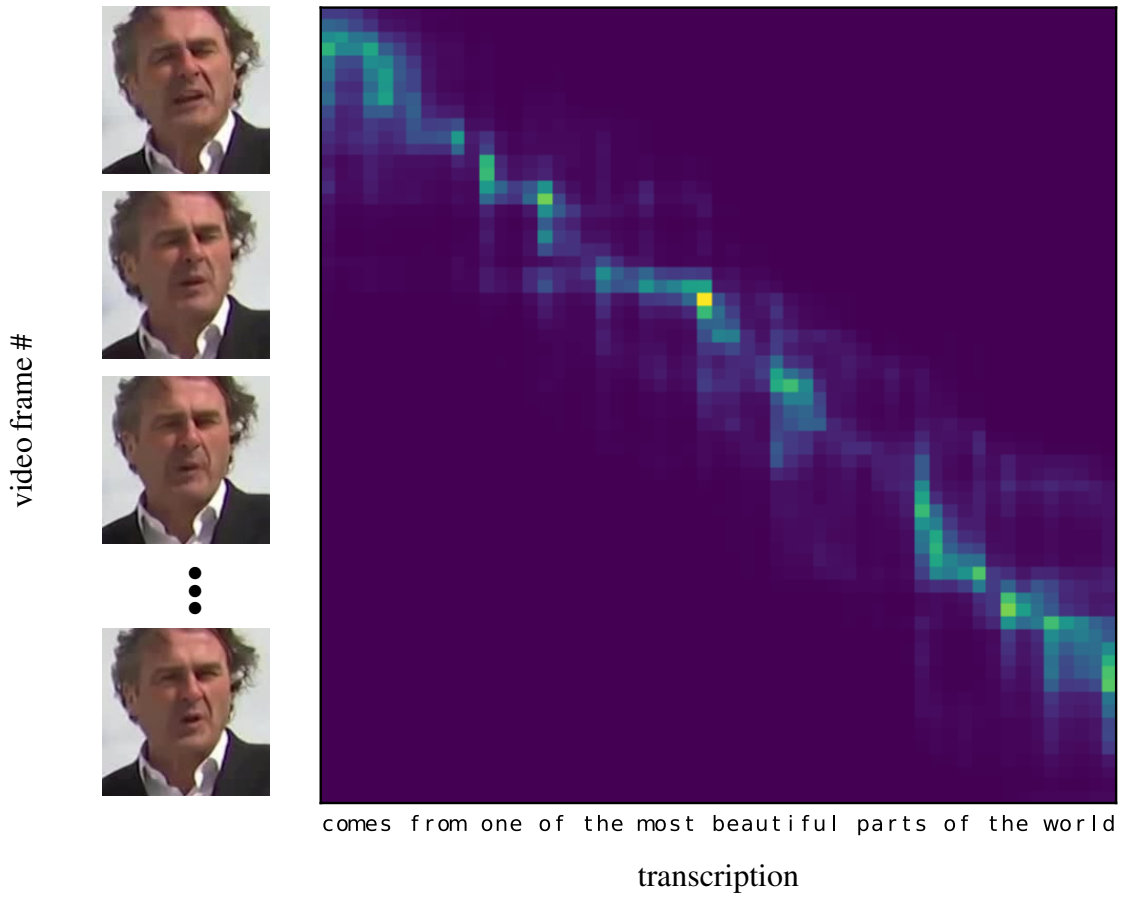


Figure 2.9: Alignment between the video frames and the character output with TM-seq2seq. The alignment is produced by averaging all the encoder-decoder attention heads over all the decoder layers in the log domain.

Appendices

Appendices for this chapter can be found in the online version of the paper.[1]

Statement of authorship

A statement of authorship for this paper is provided in Appendix A.

[1] https://www.robots.ox.ac.uk/~vgg/publications/2019/Afouras19/afouras18c.pdf

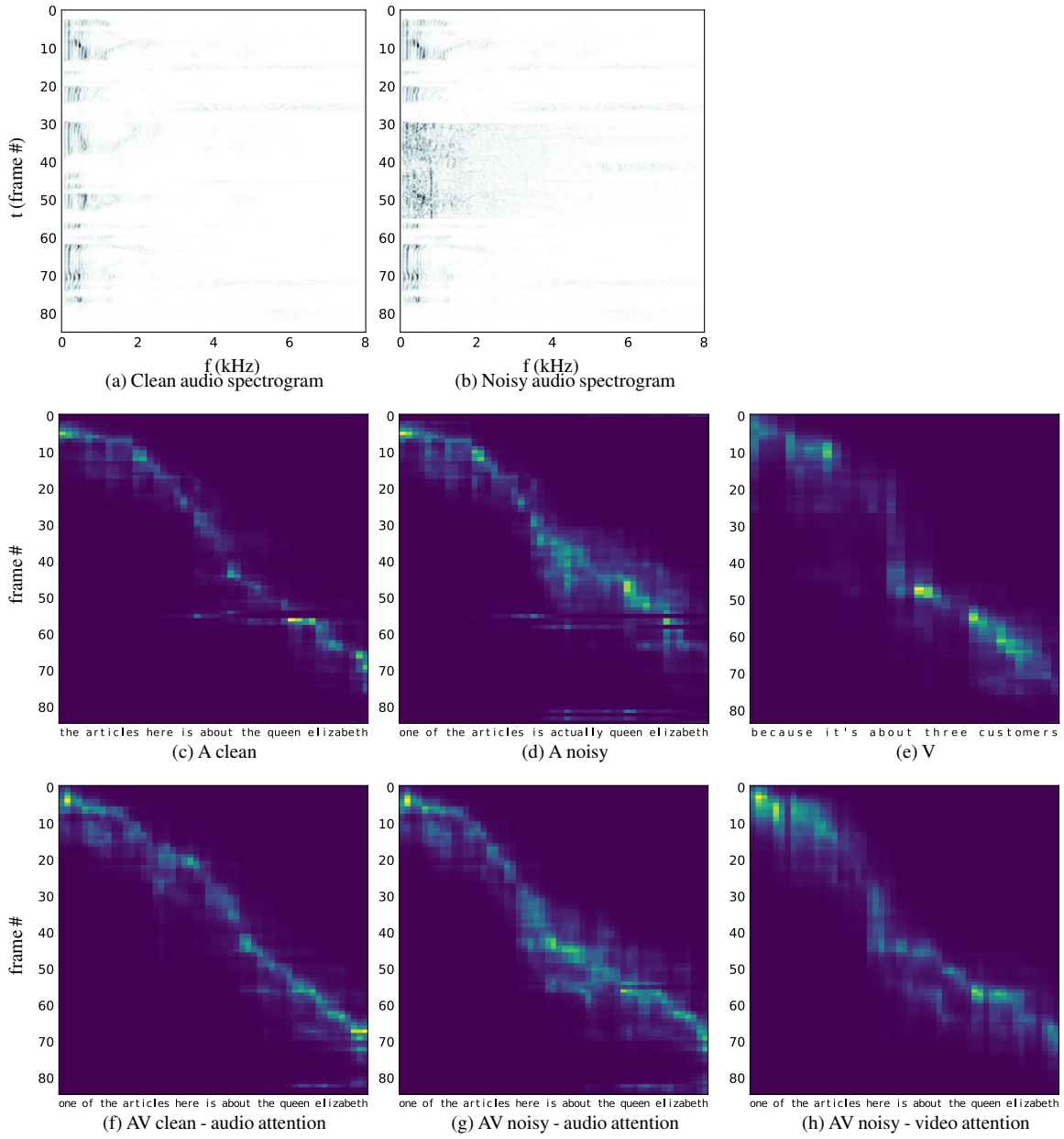


Figure 2.10: Visualization of the effect of additive noise on the attention masks of the different TM-seq2seq models. We show the attentions on (a) the clean audio utterance, and (b) on the noisy utterance which we obtain by adding babble noise to the 25 central audio frames. Comparing (c) with (d), the attention of the audio-only models appears to be more spread around the area where the noise is applied, while the last frames are not attended upon. Similarly for the audio-visual model, the audio attention is more focused when the audio is clean (f) compared to when it is noisy (g). The ground truth transcription of the sentence is "one of the articles there is about the queen elizabeth". Observing the transcriptions, we see that the audio-only model (d) does not predict the central words correctly when noise is added, however the audio-visual model (g & h) successfully transcribes the sentence, by leveraging the visual cues. Interestingly, in this particular example, the transcription that the video-only model outputs (e) is completely wrong; the combination of both modalities however yields a correct prediction. Finally, the attention mask of the AV model on the video input (f) has a clear monotonic trend and is similar to the one of the video-only model (e); this also verifies that the model indeed learns to use the video modality even when audio is present.

3 | Sub-word Level Lip-reading with Visual Attention

Prajwal Kondajji Renukananda[1] Triantafyllos Afouras[1] Andrew Zisserman[1]

[1]Visual Geometry Group, Oxford

Abstract

The goal of this paper is to learn strong lip reading models that can recognise speech in silent videos. Most prior works deal with the open-set visual speech recognition problem by adapting existing automatic speech recognition techniques on top of trivially pooled visual features. Instead, in this paper we focus on the unique challenges encountered in lip reading and propose tailored solutions. To that end we make the following contributions: (1) we propose an attention-based pooling mechanism to aggregate visual speech representations; (2) we introduce a novel data augmentation method based on dropping parts of the input and output sequences; (3) we use sub-word units for lip-reading for the first time and show that this allows us to better model the ambiguities of the task; (4) we propose a training pipeline that balances the lip-reading performance with other key factors such as data and compute efficiency. Following the above, we obtain state-of-the-art results on the challenging LRS2 and LRS3 benchmarks when training on public datasets, and even achieve results comparable with works trained on large-scale industrial datasets by using an order of magnitude less data. Our best model achieves 31.3% word error rate on the LRS2 dataset, a performance unprecedented for lip-reading models, significantly reducing the performance gap between lip-reading and automatic speech recognition.

To be published in the proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR) 2022.

3.1 Introduction

Lip reading, or visual speech recognition, is the task of recognising speech from silent video. It has many practical applications which include improving speech recognition in noisy environments, enabling silent dictation, or dubbing and transcribing archival silent films [196]. It also has important medical applications, such as helping speech impaired individuals, e.g. people suffering from Lou Gehrig's disease speak [347], or enabling people with aphonia (loss of voice) to communicate just by using lip movements.

Lip reading and audio-based automatic speech recognition (ASR) both have the common goal of transcribing speech, however they differ regarding the input: while in ASR the input signal is an audio waveform, in essence a one-dimensional time-series, lip reading has to deal with high-dimensional video inputs that have both temporal and spatial complexity. This added complexity makes training large end-to-end models harder due to GPU memory and computation constraints. Furthermore, understanding speech from visual information alone is challenging due to inherent ambiguities present in the visual stream, i.e. the existence of homophemes, different characters that are visually indistinguishable (e.g. 'pa', 'ba' and 'ma'). That lip reading is a much harder task is also supported by the fact that although humans can understand speech reasonably well even in the presence of noise and through a variety of accents, they perform relatively poorly on lip reading [26, 87].

Designing a lip reading model requires both a visual component – mouth movements need to be identified – as well as a temporal sequence modelling component, which typically involves learning a language model that can resolve ambiguities in individual lip shapes. Recent developments in deep learning models and the availability of large scale annotated datasets has led to breakthroughs, surpassing human performance [87]. However, most of these works have taken the approach of adapting techniques used for ASR and machine translation, without catering to the particularities of the vision problem.

The conjecture in this paper is that the performance of lip reading, in terms of both accuracy and data efficiency, can be improved if the model is designed from the start taking account of the peculiarities of the visual, rather than the audio domain. To this end, we make four contributions to this design.

Visual encoding. Our first contribution is the design of a novel visual backbone for lip reading. The spatio-temporal complexity in lip reading requires dealing with problems such as tracking the mouth in moving talking heads. This is usually achieved with complicated pre-processing pipelines based on facial landmarks. However, those are sub-optimal in many cases. For example, landmarks don't work well in profile views [200]. Moreover, it is unclear what is the optimal region-of-interest for lip reading: it has been shown that besides the lips, other parts of the face, e.g. the cheeks, may also contain useful discriminative information [439]. Also,

this region-of-interest can vary drastically in terms of scale, aspect ratio across identities and utterances. Thus, in this work, we propose an end-to-end trainable attention-based pooling mechanism that learns to track and aggregate the lip movement representations, resulting in a significant performance boost.

Text tokenisation. Lip reading methods most commonly output character-level tokens. This output representation however is suboptimal as characters are more fine-grained than the input, with multiple characters corresponding to a video frame. Furthermore, characters do not encode any prior knowledge about the language, leading to higher dependence on external language models which must also 'learn to read'. In this work we instead use sub-word tokens (word-pieces) which not only match with multiple adjacent frames but are also semantically meaningful for learning a language easily. Word-pieces result in much shorter (than character) output sequences which greatly reduces the training time of the sequence Transformer. They also provide a language prior, reducing the language modelling burden of the model. We experimentally compare character and word-piece tokenization to justify this choice.

Training curriculum. Curriculum learning has been used in both ASR and lip reading as a way to accelerate model convergence and improve final performance. Prior lip reading works are often trained in two stages [2, 87]: First the visual backbone is pre-trained on a limited context lip reading task (e.g. word-level recognition); the backbone is then frozen and a new sequence model is trained on top of it on a sentence-level task. The second stage follows a curriculum [2, 87] that gradually increases the length of the sentences, a practice that has its roots in audio sequence modelling [63]. Among other findings in this work, we show that (i) it is possible to skip word-level lipreading and pre-train the backbone directly on the sentence transcription task, albeit keeping the sentences to a manageable level, e.g. 2 words, and (ii) that the complicated curriculum that gradually increases the sentence length only gives a slight benefit over simply sub-sampling word sequences as a form of augmentation.

Temporal Augmentation. Although spatial augmentation methods for improving the robustness of the visual backbone are commonly used for lip reading, not many temporal augmentation techniques have been proposed for this task – even though such augmentations

have been shown to improve performance for sequence-to-sequence tasks like Translation, and ASR [300]. A final contribution is to introduce a simple, yet very effective way to reduce overfitting by randomly dropping output words and corresponding input frames.

Besides improving performance on the sentence-level lip reading task itself, obtaining improved lip movement representations can have broader impact, as those are often used for other related downstream tasks – e.g. sound source separation [120], visual keyword spotting [270], and visual language identification [8]. Moreover the temporal augmentation we propose can be extended to other visual speech translation tasks, such as sign language recognition [383].

In summary, we make the following four contributions and show their benefits to improving lip reading performance: (i) a visual backbone architecture using attention based pooling on the spatial feature map; (ii) a temporal augmentation technique based on dropping words and the corresponding visual sub-sequence; (iii) the use of sub-word units, rather than characters for the language tokens; and (iv) a two stage curriculum training schedule that simplifies training, without significant performance cost.

As will be seen, with these design choices and training methodology, the performance of our best models exceeds prior work on standard evaluation benchmarks, and even approaches proprietary models that use an order of magnitude more data for training.

3.2 Related Work

We present an overview of prior work on lip reading, including a discussion of how these methods select and track the visual regions of interest, as well as the output tokenisations they use, followed by a brief overview of temporal augmentations and the use of attention for visual feature aggregation in other domains.

Lip reading. Early works on lip reading relied on hand-crafted pipelines and statistical models for visual feature extraction and temporal modelling [157, 251, 290, 299, 310]; an extensive review of those methods is presented in [447]. The advent of deep learning and the availability of large scale lip reading datasets such as LRS2 [86] and LRS3 [5], rejuvinated this area.

Progress was initially on word level recognition [87, 360], and then moved onto sentence level recognition by adapting models developed for ASR using LSTM sequence-to-sequence [86] or CTC [26, 347] approaches. [305] take a hybrid approach, training an LSTM-based sequence-to-sequence model with an auxiliary CTC loss. One trend in recent work is moving to Transformer based architectures [2], or variants using convolution blocks [435], and hybrid architectures like a Conformer [164]. Another trend is to investigate the benefits of training with larger datasets, either directly by training on proprietary data that is orders of magnitude larger than any public dataset [256], or indirectly by distilling ASR models into lip reading ones [7, 241, 423]. For visual feature extraction and short-term dynamic modelling, most modern pipelines rely on spatio-temporal CNNs consisting of multiple 3D convolutional layers [26, 347], or more lightweight alternatives that comprise a single 3D convolutional layer followed by 2D ones [2, 87, 360] applied frame-wise.

Mouth ROI selection, registration and tracking. A thorow investigation on facial region of interest (ROI) selection for lip reading is provided by [439]. The videos included in datasets like LRS2 and LRS3 are commonly preprocessed with a face detection and tracking pipeline which outputs clips roughly centered around the speaker's face. Many previous works use a central crop on the provided videos as input to the feature extractors [2, 253, 360]. More elaborate pipelines use facial landmarks to register the face to a cannonnical view and/or only extract the crops of the mouth area [26, 223, 256, 304, 347, 435]. [439] propose inputting a large part of the face, combined with Cutout[108] to encourage the model to also use the extra-oral face regions. After selecting which input region to extract the low-level CNN features from, all above works apply Global Avarage Pooling (GAP) on the extracted visual features map; this obtains a compact representation, but discards spatial information. Recent works [435] have shown that replacing GAP with a Spatio-temporal fusion module improves performance.

Text tokenisation. Most prior works on lip reading output character-level predictions [2, 85, 86, 253, 256, 305, 435]. Those approaches usually use an external language model during inference to boost performance[204, 254]. Instead [347] chose to output phoneme sequences, using phonetic dictionaries. This approach has the advantage of a more accurate mapping of lip-movements to sounds, but requires a complicated decoding pipeline involving a proprietary

finite-state-transducer. [126, 218] use a hard-crafted heuristic to map words onto viseme sequences and vice versa, and use viseme tokens for representing the output and target text. In this work we instead propose using sub-word level tokenisation, which greatly reduces the output sequence length, thus accelerating both training and inference, and neatly encodes prior language information improving overall performance.

Temporal augmentation in sequence learning. For automatic speech recognition, simple temporal augmentation methods, including time/frequency spectrogram masking [300] and input speed perturbation [238], have been shown to considerably boost performance. Augmentation techniques based on randomly deleting, adding or replacing tokens in sentences have been proposed to reduce overfitting for machine translation [37, 358, 432], while adversarial methods have been shown to increase the efficacy of these perturbations [74, 115, 264]. For lip reading, temporal augmentations have been limited to random deletion and duplication of input frames [26].

Visual feature aggregation with attention. Our work is also related to methods that use attention for improving visual representations of images or videos. [195, 397] use attention-weighted-averages of visual features as building blocks for various classification and detection tasks, while OCNet [428] uses self-attention to model context between pixels for semantic segmentation. A number of recent papers has replaced convolutions with Transformer [384] blocks in visual representation pipelines. DETR [59] and efficient DETR [420] learn object detectors by applying spatial transformers on top of CNN feature extractors. Similarly, the Visual Transformer [406] tokenises low-level CNN features and then processes them using a Transformer to model relationships between tokens. ViT [111] completely removes CNNs from the visual pipeline, replacing them with Transformer layers applied on image patch sequences, while the Timesformer [38] has been suggested as a purely Transformer-based solution for video representation learning.

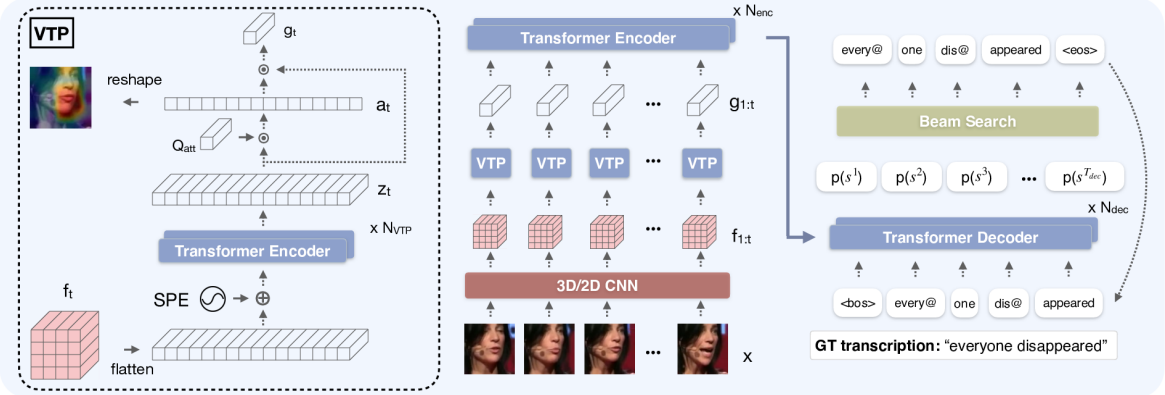


Figure 3.1: Proposed lip reading architecture. *Left:* The input video frames are passed through a spatio-temporal CNN to extract low-level visual features $\boldsymbol{f}$. The feature map corresponding to every input frame is then separately processed by a Visual Transformer Pooling module (VTP). The VTP block adds spatial positional encodings (SPE) to the input features and passes the result through a Transformer encoder to produce a self-attended feature map $\boldsymbol{z}_t$. A query vector $\boldsymbol{Q}_{att}$ is used to compute an attention mask which is in turn used to obtain a spatially weighted average of $\boldsymbol{z}_t$. This produces a compact visual representation of the appearance and lip movement around each input video frame. Concatenating the frame-wise features forms a temporal feature sequence $\boldsymbol{g}$. This is passed as input to an encoder-decoder Transformer (*right*) that auto-regressively predicts sub-word probabilities for one token at a time. An output sentence is eventually inferred from these distributions using a beam search.

3.3 Method

In this section we describe our proposed method. The architecture of the model is outlined in Figure 3.1. Next, we explain each stage of the pipeline and refer the reader to the arXiv version of the paperfor further architecture and training details.

3.3.1 Visual backbone

CNN. The input to the pipeline is a silent video clip of T frames, $\boldsymbol{x} \in \mathcal{R}^{T\times H\times W\times 3}$. A spatio-temporal residual CNN is applied on subclips of 5 frames (i.e. 0.2s) with a unit frame stride, to extract visual spatial feature maps $\boldsymbol{f} \in \mathcal{R}^{T\times h\times w\times C}$. Typically $H = W = 224$, $h = w = 14$, and $C = 512$.

Visual Transformer Pooling.

The CNN feature map $\boldsymbol{f}_t \in \mathcal{R}^{h\times w\times C}$ corresponding to every input frame $t \in \{1,...,T\}$ is processed individually by a shared Visual Transformer Pooling (VTP) block: The feature

map is first flattened, then spatial positional encodings (SPE) are added to it; the result is passed through an encoder consisting of N_{VTP} Transformer layers, to produce an enhanced self-attended feature map

$$\boldsymbol{z}_t = encoder_v(\boldsymbol{f}_t + SPE_{1:hw}) \in \mathcal{R}^{\times hw \times C}.$$

A learnable query vector $\boldsymbol{Q_{att}} \in \mathcal{R}^{C\times 1}$ is then used to extract a visual attention mask

$$\boldsymbol{a}_t = softmax(\boldsymbol{Q}_{\boldsymbol{att}}^{\top}\, \boldsymbol{z}_t) \in \mathcal{R}^{hw\times 1}.$$

The attention mask is used to compute a weighted average over the self-attended feature map

$$\boldsymbol{g}_t = \frac{1}{hw}\sum_{u=1}^{hw} a_t^u z_t^u \in \mathcal{R}^C$$

where a_t^u and z_t^u denote the feature and attention weight respectively, associated with frame t and location $u \in \{1,\ldots,hw\}$. By stacking the resulting vectors $\boldsymbol{g}_t$ in time, we obtain an embedding sequence $\boldsymbol{g} = (\boldsymbol{g}_1, \boldsymbol{g}_2 \cdots, \boldsymbol{g}_T) \in \mathcal{R}^{T\times C}$ which contains a compact spatio-temporal representation for every input frame.

3.3.2 Transformer encoder-decoder

An encoder-decoder Transformer model is used to predict a text token sequence $s = (s_1, s_2 \cdots, s_{T_{dec}})$ from the source video embedding sequence $\boldsymbol{g}$, one token at a time: temporal positional encodings (PE) are added to $\boldsymbol{g}$, and the result is input to an encoder, which consists of N_{enc} multi-head Transformer layers, to produce a self-attended embedding sequence

$$\boldsymbol{g_{enc}} = encoder(\boldsymbol{g} + PE_{1:T}) \in \mathcal{R}^{T\times C}.$$

The decoder, which consists of N_{dec} Transformer layers, then attends on this sequence and predicts the output text token sequence s in an auto-regressive manner, by factorising its joint probability:

$$\log p(s|\boldsymbol{x}) = \sum_{t=1}^{T_{dec}} \log p(s_t|\, \boldsymbol{g_{enc}}(\boldsymbol{x}), s_{1:t-1}) \tag{3.1}$$

where positional encodings have also been added to the auto-regressive decoder inputs as in [384].

The text sentences are encoded into token sequences (and vice versa tokens are decoded into text) using a sub-word level tokeniser, in particular WordPiece [407].

Beam search decoding and rescoring. Decoding is performed with a left-to-right beam search of width B. Additional language knowledge can be incorporated by using an external language model (LM) to rescore [64] the B-best hypotheses $\{s_1 \cdots s_B\}$ that the beam search results in, and obtain the highest scoring one as the final sentence prediction:

$$s_{best} = \underset{s \in \{s_1 \cdots s_B\}}{\operatorname{argmax}} \left[\alpha \log p(s|\boldsymbol{x}) + (1-\alpha) \log p_{LM}(s) \right]$$

3.3.3 Training

Optimisation objective. Given a training dataset $\mathcal{D}$ consisting of pairs (x, s^*) of video clips and their ground truth transcriptions, the model is trained to maximise the log likelihoods of the transcriptions by optimising the following objective

$$\mathcal{L} = -\mathbb{E}_{(x,s^*) \in \mathcal{D}} \log p(s^*|\boldsymbol{x}) \tag{3.2}$$

Teacher forcing. To accelerate training we follow common practice for sequence-to-sequence training with Transformers, and feed in the previous ground truth token as the decoder input at every step, instead of using auto-regression. The tokens are fed into the decoder via a learnable embedding layer.

Temporal augmentation. To increase robustness and reduce overfitting we propose a novel augmentation techinque during training: For a video-text pair $(\boldsymbol{x}, s^*)$ we randomly drop a word with probability p_d from s^* and remove the corresponding video frames from $\boldsymbol{x}$. This requires knowledge of the word boundaries in the training set which can be automatically obtained using forced alignment of the text transcriptions to the audio speech, and is commonly

included in lip reading datasets. Thus, it requires no extra manual annotations and is also simple and computationally inexpensive to implement.

Training protocol. Training is performed in two stages. First the whole network is trained end-to-end on short sentences of 2 words. Following [2, 87] we use frame word-boundaries to crop out training samples from all the possible combinations of 2 consecutive words in the datset, which provides natural augmentation. Once training converges, we freeze the visual backbone, then pre-extract and dump the visual features of all the samples in the training dataset. In the second training stage that follows, we train the encoder-decoder subnetwork on all possible sub-sequences of length 2 or larger that can be generated by combining consecutive word utterances in the dataset.

Discussion. We note that our proposed curriculum is much simpler than the ones commonly used in prior works [2, 87, 305], since (i) the same network and loss are used during the backbone pre-training stage, which provides a good initialization of the entire network and enables a smooth transfer; this is in contrast to other works that pre-train with a different proxy loss and require a separate word classification head which is subsequently discarded; and (ii), the second stage is significantly simpler to implement and requires a single run, unlike curriculums that gradually increase the length of the training sentences and usually require a complicated tuning process with multiple manual restarts. In the following section we include experiments that compare these different curriculum choices in terms of both performance and training time.

3.4 Experiments

3.4.1 Data

LRS2 & LRS3. For training and evaluation we use two publicly available sentence-level lip reading datasets: LRS2 [87] and LRS3 [5] . LRS2 contains video clips from a variety of shows from British television, such as Countryfile and Top Gear; the transcribed content sums up to approximately 224 hours in total. LRS3 has been collected from over 5 thousand TED and

	Training		Evaluation	
Method	Datasets used	Total # hours	LRS2	LRS3
LIBS [445]	LRS2, LRS3	698	65.3	-
Hyb. CTC/Att. [305]	LRS2, LRW	389	63.5	-
TDNN [423]	LRS2	224	48.9	-
Conv-seq2seq [435]	LRS2, LRS3	698	51.7	60.1
CTC + KD [7]	LRS2, LRS3, VoxCeleb2‡	1,032	51.3	59.8
Hyb. + Conformer [253]	LRS2, LRW	389	37.9	-
Hyb. + Conformer [253]	LRS3, LRW	639	-	43.3
Ours	LRS2, LRS3	698	**32.4**	**41.9**
TM-seq2seq [2]	LRS2, LRS3, LRW, MV-LRS†	1,637	48.3	58.9
CTC-V2P [347]	LSVSR†	3,886	-	55.1
RNN-T [256]	YT31k†	31,000	-	**33.6**
Ours	LRS2, LRS3, MV-LRS†	1,472	28.9	39.0
Ours	LRS2, LRS3, MV-LRS†, TEDx_{ext}	2,676	**26.7**	34.5

Table 3.1: Comparison of different lip reading models on the test sets of the LRS2 and LRS3 datasets, including the datasets and the aggregate number of hours used for training each model, in terms of Word Error Rate % (WER, lower is better). Our model achieves state-of-the-art results, outperforming all previous baselines when trained on publicly available data (i.e. LRS2 and LRS3). If we additionally use MV-LRS and TEDx_{ext} for training, then our best model obtains results comparable with those of [256] who train on a very large scale industrial datastet, even though we are only using an order of magnitude less data. This is indicative of the data efficiency of our proposed pipeline. †Large non-public labelled datasets: MV-LRS [2] contains 730 hours, LSVSR [347] 3.9k hours, and YT31k [256] 31k hours of transcribed video. ‡unlabelled dataset. Results shown in blue have been obtained by training (partly or entirely) on data that are non-publicly available.

TEDx talks in English, available on YouTube, totalling 475 hours. Both datasets have been created using a detection and tracking pipeline that produces face-cropped clips roughly centered around the speaker's talking head. All videos are available at a 224×224 pixel resolution and 25 fps. The datasets contain a "pretrain" partition that includes extensive head tracks including word boundaries that have been produced by force-aligning subtitles to the audio. Those word alignments enable training at any granularity. The test sets contain only full sentences.

Additional dataset: TEDx_{ext}. In order to obtain more training data, we create a new dataset from TEDx talks downloaded from YouTube, by using a pipeline similar to [5]. We collect 13,211 TEDx talks in English that are not included in LRS3. Unlike the videos used for the

creation of LRS3, the new videos do not have any manual transcriptions, therefore to obtain text supervision we use the closed-captions automatically produced by the YouTube ASR system. As these captions are only approximately aligned to the audio, we use the Montreal Force Aligner [260] to obtain more accurate alignments for the word boundaries needed by our training pipeline (see Section 3.3 of the main paper). For the rest of the processing (face detection, tracking and cropping) we used the same pipeline as in [5]. The resulting training dataset contains $1{,}204$ hours in total over $318{,}459$ visual speech tracks, including text transcriptions with word boundary alignment. We call this new training set TEDx_{ext}. We note that since this pipeline does not require any manual transcriptions, the supervision comes for free, therefore it is easily scalable. However the supervision is not as strong due to the noise in the training data introduced by the imperfect ASR transcriptions.

3.4.2 Implementation details

During the first training stage we apply random visual augmentations on the input frames to reduce ovefitting: the input videos are first resized to a square 160 pixels resolution, from which a random square 112-pixel crop is extracted. Random horizontal flipping and brightness jittering are also applied before inputting to the lip reading pipeline. During inference we use the central 112-pixel crop, without any augmentations. Our proposed temporal augmentation method described in Section 3.3.3 is applied directly on the dumped visual representations, during the second training stage of training. A word is dropped from a sentence with $p_d = 70\%$ probability. We use the WordPiece tokenizer of the pre-trained BERT model in HuggingFace [1], with a vocabulary of 30522 tokens. We also use an off-the-shelf GPT2 language model for beam rescoring. We set N_{VTP}=3 layers with 8 heads each for the encoder of the VTP module. The encoder-decoder Transformer contains $N_{enc} = 6$ and $N_{dec} = 6$ layers with 8 attention heads per layer everywhere. We use sinusoidal positional encodings [384] for both SPE and PE. For the beam rescoring we set hyperparameter α to 0.7. We train all models with the Adam optimiser[213] with $\beta_1 = 0.9$, $\beta 2 = 0.98$ and $\epsilon = 10^{-9}$. In the first stage of the training we follow a Noam learning rate schedule[384] for the first 30 epochs and then reduce the learning

[1] https://huggingface.co/transformers/pretrained_models.html

Method	WER
TM-seq2seq† baseline	39.6
+ WordPiece	36.9
+ VTP	33.9
+ Temporal augmentations	33.0
+ Beam LM rescoring	32.4

Table 3.2: Ablation on the design improvements proposed in this work. The results reported are for the test set of the LRS2 dataset. It is clear that all the proposed components contribute independently to the performance boost. † The baseline is an improved version of TM-seq2seq[2] (see the arXiv version of the paperfor full details) .

Method	WER	Max. len.	Time
Char.	39.4	100	40h
BPE	36.9	35	20h
WP	35.3	25	15h

Table 3.3: Ablation on the choice of text tokenization. Using sub-word tokenisation over characters yields shorter output sequences and results in both better performance and faster training. From the two sub-word unit tokenization methods we considered, WordPiece outperforms BPE. Char: Character tokenisation; BPE: byte pair encoding; WP: WordPiece. Time: Total training time (second stage);

Method	WER	Time
Curr. 2 - 9 words [2]	35.3	15h
3 - 9 words	38.1	5h
2 - 9 words	35.8	8h
2 - 14 words	35.6	16h

Table 3.4: Training protoccol ablation. Our proposed training schedule is much simpler to implement and obtains very similar performance to the more complicated curriculum of [2] in half the training time. Curr.: Curriculum that gradually increases the length of the training utterances. Time: Total training time in hours (second stage).

rate by a factor of 5 every time the validation loss plateaus, until reaching 10^{-6}. For the second stage, the learning rate is initially set to $5e^{-5}$ and reduced by a factor of 5 on plateau down to 10^{-6}. For our best reported models, the first stage of training takes approximately 10 days on 4 Tesla v100s GPUs. The second stage takes 1.5 days on 1 Tesla v100 GPU.

3.4.3 Results

State-of-the-art lipreading. We compare the results of our method to existing works in Table 3.1. Additional results, including character error rate CER evaluation and larger models are included in the arXiv version of the paper. It is clear that our best model outperforms all prior work trained on public data, on both the LRS2 and LRS3 benchmarks. In particular compared to the strongest baseline of Ma *et al.* [253] our best model performs 5.5% better on LRS2 and 1.4% better on LRS3. When also using MV-LRS for training, we obtain a significant boost, achieving 28.9% and 30.0% WER for LRS2 and LRS3 respectively. Finally, in order to slightly reduce the gap in terms of training data with [256], we train on the extra $1{,}204$ hours of TEDx_{ext}. This gives us a further boost, enabling our best model to reach an unprecedented 26.7 % WER on LRS2, and 34.5% WER on LRS3, only 0.9% higher than the performance of [256]. We

Figure 3.2: Visualization of the visual attention masks $\boldsymbol{a}$ from the VTP module superimposed on the input frames that produce them. The video clips used here are random samples from the LRS3 dataset. It is evident that the model follows the more discriminative mouth region.

note that the later uses 31k hours of training data, while we only train on 2.7k hours, to achieve comparable results, which suggests that our proposed pipeline is much more data efficient.

Ablations. To better understand the influence of our proposed design choices, we perform a number of ablations, starting from a variation of the TM-seq2seq model[2][2], and building up to our full model. We summarize the results of this study in Table 3.2. It is clear that all the proposed improvements give significant performance boosts and are largely orthogonal. In particular, the use of WordPiece tokens contributes a 3.3% absolute improvement on LRS2, while introducing the VTP module decreases the WER by an extra 3%. Adding the temporal augmentation during training and a beam rescoring with an external language model contribute another 0.9% and 0.6% improvement respectively.

Ablation on text tokenisation. In Table 3.3 we investigate different text tokenisation choices. We compare the character-level baseline with two sub-word unit tokenisation methods, namely byte pair encoding (BPE) [136, 340] and WordPiece [407]. We observe that using sub-word tokens results in much shorter output sequence lengths. For example for the utterances of the LRS2 test set, the average sentence length is 75 characters compared to 35 BPE tokens and 25 WordPiece tokens. This difference results in shorter training times for the sub-word

[2]Using the same CNN extractor as our model for fair comparison, see the arXiv version of the paperfor details.

tokenised models, because of the lower computation cost, as well as faster convergence, presumably because of the inherent encoding of language priors that they provide. In terms of WER performance too, it is apparent that the sub-word level models outperform the character baseline. Of the two sub-word tokenization choices, WordPiece provides a small boost against BPE (35.3% vs 36.9% WER).

Training protocol ablation. In Table 3.4 we explore different training protocol settings for the second training stage. We note that for the results reported here we do not use Temporal augmentation or LM rescoring. As the baseline we consider the complex curriculum proposed by [2], where training starts with short sequences of 2 words and gradually moves to longer sequences up to 9 words. This setting indeed gives us the best result, 35.3% WER on the LRS2 test set, but comes at a price of longer training time (15 hours). We then proceed to training on all possible sub-sequences of 2-9 words as described in Section 3.3.3. We observe that compared to the full curriculum, this method obtains only slightly worse results (35.8% WER), but trains almost $2\times$ faster. We therefore conjecture that it is the natural augmentation by considering different sub-sequences that provides most of the performance boost, while the gradual warmup of the model contributes very little, contrary to the assumptions of [2]. Moreover we note that this curriculum is much simpler to implement and run, as it requires minimum manual configuration. We can also note that although skipping the training on word-pairs (training from 3 to 9 word sequences) converges faster, it is detrimental to final performance (2.3 % WER worse). Finally, training on longer sequences (up to 14 words) provides a slight boost 0.2%, however it takes considerably more time.

Visual attention visualization.

In Figure 3.2 we visualize the visual attention maps that the VTP module produces. Note that the lips region is tracked very accurately while the speakers turn their heads around, even for extreme profile views.

3.5 Discussion

Narrowing the gap between lip reading and ASR performance opens up opportunities for useful applications, as noted in the introduction, but also raises privacy issues and the risk of potential malign uses. One issue that is often raised is the potential for malign surveillance, e.g. using CCTV footage from public spaces to eavesdrop on private civilian conversations. However, this is in fact very low risk due to a number of factors: we achieve a low WER on benchmarks containing video material that is professionally produced and at high resolutions and frame-rates, with good lighting conditions. Moreover the speakers are aware of being filmed and collaborate, most of the time speaking while frontally facing the camera. In contrast, CCTV usually operate at much lower resolution and frame rates and from unusual angles. As shown in prior work [90, 256, 347], lip reading performance greatly deteriorates with lower frame rate or input resolutions, or when non-frontal (e.g. profile or overhead viewpoints) rather than frontal speaker views are considered.

We will be making the code and pre-trained models of this work public. This technology is already available to a small handful of corporations that have access to enough data and compute resources for training. We believe that open access is important in order to accelerate progress in the field, as well as enable research on defences against potential adversarial attacks.

Overall, we believe that the benefits of the positive applications of lip reading that we have discussed (e.g. medical) greatly outweight the risk of malevolent uses, the latter ones being inflated, therefore transparent research into this field should be continued and encouraged by the community.

3.6 Conclusion

We have presented an improved architecture for lip reading based on attention-based aggregation of visual representations as well as several enhancements of the training protocol, including temporal augmentations, the use of sub-word output tokenisation, and a more data-efficient learning curriculum. Our best models train and converge faster than other baselines

and achieve state-of-the-art results outperforming prior work trained on public data by a significant margin, while also obtaining performance comparable to that of industrial models trained on orders of magnitude more data.

Statement of authorship. A statement of authorship for this paper is provided in Appendix A.

4 | ASR Is All You Need: Cross-modal Distillation For Lip Reading

Triantafyllos Afouras[1] Joon Son Chung[1,2] Andrew Zisserman[1]

[1]Visual Geometry Group, Oxford [2]Naver Corporation

Abstract

The goal of this work is to train strong models for visual speech recognition without requiring human annotated ground truth data. We achieve this by distilling from an Automatic Speech Recognition (ASR) model that has been trained on a large-scale audio-only corpus. We use a cross-modal distillation method that combines Connectionist Temporal Classification (CTC) with a frame-wise cross-entropy loss. Our contributions are fourfold: (i) we show that ground truth transcriptions are not necessary to train a lip reading system; (ii) we show how arbitrary amounts of unlabelled video data can be leveraged to improve performance; (iii) we demonstrate that distillation significantly speeds up training; and, (iv) we obtain state-of-the-art results on the challenging LRS2 and LRS3 datasets for training only on publicly available data.

Published in IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2020, pp. 2143-2147.

4.1 Introduction

Visual speech recognition (VSR) has received increasing amounts of attention in recent years due to the success of deep learning models trained on corpora of aligned text and face videos [26, 86, 87]. In many machine learning applications, training on very large datasets has proven to have huge benefits, and indeed [256, 347] recently demonstrated significant performance improvements by training on very large-scale proprietary datasets. However, the largest publicly available datasets for training and evaluating visual speech recognition, LRS2 and LRS3 [5, 86], are orders of magnitude smaller than their audio-only counterparts used for training Automatic

Speech Recognition (ASR) models [34, 298]. This indicates that there are potential gains to be made from a scalable method that could exploit vast amounts of unlabelled video data.

In this direction, we propose to train a VSR model by *distilling* from an ASR model with a teacher-student approach. This opens up the opportunity to train VSR model on audio-visual datasets that are an order of magnitude larger than LRS2 and LRS3, such as VoxCeleb2 [85] and AVSpeech [120], but lack text annotations. More generally, the VSR model can be trained from *any* available video of talking heads, e.g. from YouTube. Training by distillation eliminates the need for professionally transcribed subtitles, and also removes the costly step of forced-alignment between the subtitles and speech required to create VSR training data [87].

Our aim is to to pretrain on large unlabelled datasets in order to boost lip reading performance. In the process we also discover that human-generated captions are actually not necessary to train a good model. The approach we follow, as shown in Fig. 4.1, combines a distillation loss with conventional Connectionist Temporal Classification (CTC) [159]. An alternative option to exploit the extra data, would have been to train solely with CTC on the ASR transcriptions. However we find that compared to that approach, distillation provides a significant acceleration to training.

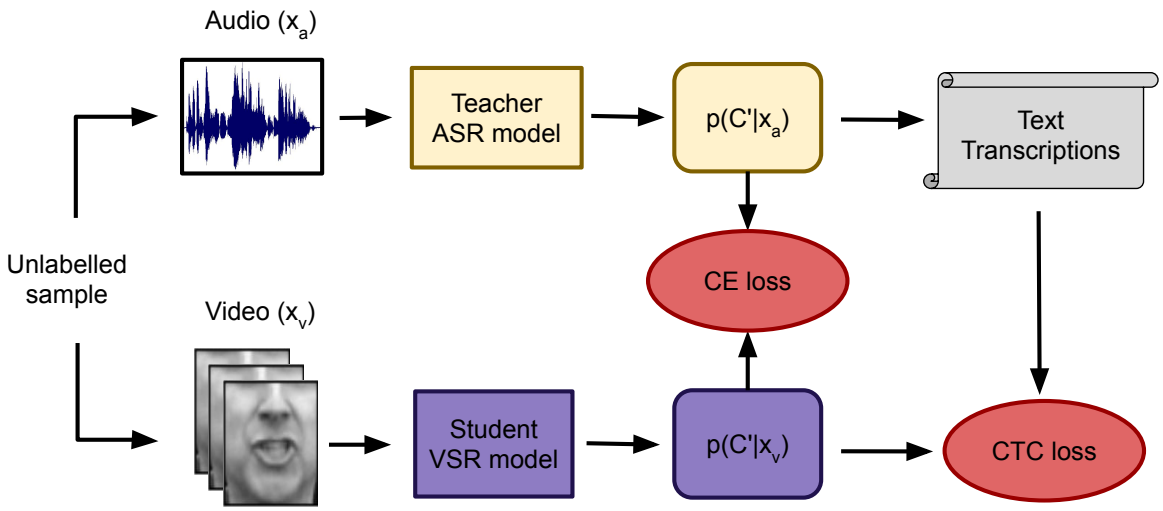


Figure 4.1: Cross-modal distillation of an ASR teacher into a student VSR model. CTC loss on the ASR-generated transcripts is combined with minimizing the KL-divergence between the student and teacher posterior distributions.

4.1.1 Related Work

Supervised lip reading. There have been a number of recent works on lip reading using datasets such as LRS2 [86] and LRS3 [5]. Works on word-level lip reading [87] have proposed CNN models and temporal fusion methods for word-level classification. [360] combines a deeper residual network and an LSTM classifier to achieve the state-of-the-art on the same task. Of more relevance to this work is open set character-level lip reading, for which recent work can be divided into two groups. The first uses CTC where the model predicts frame-wise labels and is trained to minimize the loss resulting from all possible input-output alignments under a monotonicity constraint. LipNet [26] and more recently [256, 347] are based on this approach. [256] in particular demonstrates state-of-the-art performance by training on proprietary data that is orders of magnitude larger than any public dataset. The second group is sequence-to-sequence models that predict the output sequence one token at a time in an autoregressive manner, attending to different parts of the input sequence on every step. Some examples are the sequence-to-sequence LSTM-with attention model used by [86] and the Transformer-based model used by [4] or a convolutional variant by [435]. [2, 305] take a hybrid approach that combines the two ideas, namely using a CTC loss with attention-based models. Both approaches can use external language models during inference to boost performance[204, 254]

Knowledge distillation (KD). Distilling knowledge between two neural networks has been popularised by [185]. Supervision provided by the teacher is used to train the student on potentially unlabelled data, usually from a larger network into a smaller network to reduce model size. There are two popular ways of distilling information: training the student to regress the teacher's pre-softmax logits [28], and minimising the cross-entropy between the probability outputs [185, 239].

Sequence and CTC distillation. KD has also been studied in the context of sequence modeling. For example it has been used to compress sequence-to-sequence models for neural machine translation [210] and ASR [208]. Distillation of acoustic models trained with CTC has also been investigated for distilling a BLSTM model into a uni-directional LSTM so that it can

be used online [209], transferring a deep BLSTM model into a shallower one [109], and the posterior fusion of multiple models to improve performance [229].

Cross-modal distillation. Our approach falls into a group of works that use networks trained on one modality to transfer knowledge to another, in a teacher-student manner. There have been many variations on this idea, such as using a visual recognition network (trained on RGB images) as a teacher for student networks which take depth or optical flow [165], or audio [27] as inputs. More specific examples include using the output of a pre-trained face emotion classifier to train a student network that can recognize emotions in speech [12] or visual recognition of human pose to train a network to recognize pose from radio signals [444]. The closest work to ours is Wei *et al.* [241] who apply cross-modal distillation from ASR for learning audio-visual speech recognition. An interesting finding is that the student surpasses the teacher's performance, by exploiting the extra information available in the video modality. However, their method is focused on improving ASR by incorporating visual information, rather than learning to lip read from the video signal alone, and they train the teacher model with ground truth supervision on the same dataset as the student one. Consequently, their method does not apply naturally to unlabelled audio-visual data.

4.2 Datasets

A summary of audio-visual speech datasets found in the literature is given in Table 4.1. LRS2 and LRS3 are public audio-visual datasets that contain transcriptions but are relatively small. LRS2 is from BBC programs and LRS3 from TED talks, and there is a domain gap between them. Librispeech is large, transcribed, and diverse regarding the number of speakers, but audio-only. On the other hand VoxCeleb2, which is similar in scale, is audio-visual but lacks transcriptions. YT31k, LSVSR and MV-LRS contain aligned ground truth transcripts and have been used to train state-of-the-art lip reading models [2, 256, 347]. However, these datasets are not publicly available which hinders reproduction and comparison. In this paper we focus on using only publicly available datasets. We use our distillation method to pretrain on VoxCeleb2 and then fine-tune and evaluate the resulting model on LRS2 and LRS3.

Table 4.1: Statistics of modern audio-visual datasets. **Tran.:** Indicates if the dataset is labelled, i.e. includes aligned transcriptions; **Mod.:** Modalities included (A=audio-only, AV=audio + video). *VoxCeleb2 (clean)* refers to the subset of VoxCeleb2 we obtain after filtering according to Section 4.2.

Dataset	# Utter.	# Hours	Mod.	Tran.	Public
YT31k [256]	-	31k	AV	✓	✗
LSVSR [347]	2.9M	3.8k	AV	✓	✗
MV-LRS [87]	500k	775	AV	✓	✗
Librispeech [298]	292k	1k	A	✓	✓
VoxCeleb2 [85]	1.1M	2.3k	AV	✗	✓
LRS2 (pre-train) [86]	96k	195	AV	✓	✓
LRS2 (main) [86]	47k	29	AV	✓	✓
LRS2 (test) [86]	1.2k	0.5	AV	✓	✓
LRS3 (pre-train) [5]	132k	444	AV	✓	✓
LRS3 (train-val) [5]	32k	30	AV	✓	✓
LRS3 (test) [5]	1.3k	1	AV	✓	✓
VoxCeleb2 (clean)	140k	334	AV	✗	✓

To enable the use of an unlabelled speech dataset for training lip reading models for English, we first filter out unsuitable videos. For example, in VoxCeleb2, the language spoken is not always English, while the audio in many samples can be noisy and therefore hard for an ASR model to comprehend. We first run the trained teacher ASR model (details in section 4.3) to obtain transcriptions on all the unlabelled videos. We then use a simple proxy to select good samples: for each utterance we calculate the percentage of words with 4 characters or more in the ASR output that are valid english words and keep only the samples for which this is 90% or more.

As a second refinement stage, we obtain transcriptions from a separate ASR model. We use a model similar to wave2letter [246] trained on Librispeech. We then compare the generated transcriptions with the ones from the teacher model and only keep an utterance when the overlap in terms of Word Error Rate is below 28%. For VoxCeleb2, the above process discards a large part of the dataset, resulting in approximately $140k$ clean utterances out of the 1M in total.

4.3 Cross-modal distillation

As a *teacher*, we use the state-of-the-art Jasper 10x5 acoustic model [238] for ASR, a deep 1D-convolutional residual network. The *student* model for lip reading uses an architecture

Table 4.2: Architecture of Jasper-lip 5x3. To modify the Jasper model for lip-reading, we replace the first strided convolutional layer with a transposed convolution (stride=0.5).

# Blocks	Block	Kernel	# Output Channels	Dropout	# Sub Blocks
1	Conv1	11 *stride=0.5*	256	0.2	1
1	B1	11	256	0.2	3
1	B2	13	384	0.2	3
1	B3	17	512	0.2	3
1	B4	21	640	0.3	3
1	B5	25	768	0.3	3
1	Conv2	29 *dilation=2*	896	0.4	1
1	Conv3	1	1024	0.4	1
1	Conv4	1	# graphemes + 1	0	1

similar to the teacher's. More specifically, we adapt the Jasper acoustic model for lip reading as shown in Table 4.2. The input to this network are visual features extracted from a spatio-temporal residual CNN [360].

4.3.1 CTC loss on transcriptions

CTC provides a loss function that enables training networks on sequence to sequence tasks without the need for explicit alignment of training targets to input frames. The CTC output token set C' consists of an output grapheme alphabet C augmented with a blank symbol '$-$': $C' = C \bigcup \{-\}$. The network consumes the input sequence and outputs a probability distribution p_t^{ctc} over C' for each frame t. A CTC path $\pi \in C'^T$ is a sequence of grapheme and blank labels with the same length T as the input. Paths π can be mapped to possible output sequences with a many-to-one function $B : C'^T \rightarrow C^{\leq T}$ that removes the blank labels and collapses repeated non-blank labels. The probability of an output sequence y given input sequence x is obtained by marginalizing over all the paths that are mapped to y through B: $p(y|x) = \sum_{\pi \in B^{-1}(y)} \prod_{t=1}^{T} p_t^{ctc}(\pi(t)|x)$. [159] computes and differentiates this sum w.r.t. the posteriors p_t^{ctc} efficiently, enabling one to train the network by minimizing the CTC loss over input-output sequence pairs x, y^*:

$$\mathcal{L}_{CTC}(x, y^*) = -log(p(y^*|x))$$

4.3.2 Distillation loss

To distill the acoustic model into the target lip-reading model, we minimize the KL-divergence between the teacher and student CTC posterior distributions or, equivalently, the frame level cross-entropy loss:

$$\mathcal{L}_{KD}(x_a,x_v) = -\sum_{t\in T}\sum_{c\in C'} log p_t^a(c|x_a) p_t^v(c|x_v)$$

where p_t^a and p_t^v denote the CTC posteriors for frame t obtained from the teacher and student model respectively. This type of distillation has been used by other authors when distilling acoustic CTC models within the same modality (audio) and is referred to as frame-wise KD [209, 339, 369].

4.3.3 Combined loss

As shown on Fig. 4.1, given the transcription of an utterance and corresponding teacher posteriors, we combine the CTC and KD loss terms into a common objective:

$$\mathcal{L}(x_a,x_v,y^*) = \lambda_{CTC}\mathcal{L}_{CTC}(x_v,y^*) + \lambda_{KD}\mathcal{L}_{KD}(x_a,x_v)$$

where λ_{CTC} and λ_{KD} are balancing hyperparameters.

4.4 Experimental Setup

We train on the VoxCeleb2, LRS2 and LRS3 datasets and evaluate on LRS2 and LRS3 test sets (Table 4.1). In this context, we investigate the following training scenarios:

Full supervision. We use annotated datasets only (LRS2, LRS3), and train with CTC loss on the ground truth transcriptions, similarly to [4, 26]. This is the baseline method.

No supervision. We do not use any ground truth transcriptions and rely solely on the transcriptions and posteriors of the ASR teacher model for the training signal.

Unsupervised pre-training and fine-tuning. We first pre-train the model using distillation on unlabeled data. We then fine-tune the model on the transcribed target dataset (either LRS2 or LRS3) with full supervision. We perform two sets of experiments in this setting: (i) we

use the ground truth annotations of all the samples in the dataset that we are fine-tuning on, or (ii) we only use the ground truth of the "main" and "train-val" subsets of LRS2 and LRS3 respectively (see Table 4.1), which contain a small fraction of the total hours.

4.4.1 Implementation details

Our implementation is based on the Nvidia Seq2Seq framework [225]. As a teacher, we use the 10x5 Jasper model trained on Librispeech. To extract visual features from videos we use the publicly available visual frontend from [4], pre-trained on word-level lip reading. We train the student model with the NovoGrad optimizer and the settings of [238] on 4 GPUs with 11GB memory and a batch size of 64 on each. We set $\lambda_{CTC} = 0.1$ and $\lambda_{KD} = 10$; these values were empirically determined to give similar gradient norms for each term during training. Decoding is performed with a 8192-width beam search that uses a 6-gram language model trained on the Librispeech corpus text.

4.5 Experiments

We summarize our results in Table 4.3. The baseline method (CTC, GT) obtains 58.5% WER on LRS2 and 68.8% on LRS3 when trained and evaluated on each dataset separately. In the same setting, and without any ground truth transcriptions, our method achieves similar performance on LRS2 (58.2%) and even better on LRS3 (65.6%). This result demonstrates that human-annotated videos are not necessary in order to effectively train lip reading models. Fine-tuning with limited ground truth transcriptions, as described in Section 4.4, reduces this to 57.9% for LRS2 and 65.1% for LRS3. For training on LRS2 alone, these results outperform the previous state-of-the art which was 63.5% by [305].

Using our method to train on the extra available data, but without any ground truth transcriptions, we further reduce the WER to 54.2% and 61.7% for LRS2 and LRS3 respectively. If we moreover fine-tune with a small amount of ground truth transcriptions, the WER drops to 52.2% (LRS2) and 61.0% (LRS3). Finally, training on each dataset with full supervision after unsupervised pre-training, yields the best results, 51.3% for LRS2 and 59.8% for LRS3. Comparing these numbers to the results we obtained when training on each dataset individually,

Table 4.3: Word Error Rate % (WER, lower is better) evaluation. **CTC:** Model trained with CTC loss. **CTC + KD:** Combined loss. GT denotes using all the ground truth transcriptions of the dataset, ASR the transcriptions obtained from the teacher ASR model, and ASR/GT first pre-training with the ASR transcriptions and then fine-tuning with a small fraction of the ground truth ones. **Vox.:** VoxCeleb2 (clean). †Trained on large non-public labelled datasets: YT31k [256], LSVSR [347], and MV-LRS [2] (see Table 4.1). ‡Concurrent work.

	Trained on			**Evaluated on**	
Method	Vox.	LRS2	LRS3	LRS2	LRS3
Hyb. CTC/Att. [305]	✗	GT	✗	63.5	-
TM-seq2seq† [2]	✗	GT	GT	48.3	58.9
CTC-V2P† [347]	✗	✗	✗	-	55.1
RNN-T† [256]	✗	✗	✗	-	33.6
Conv-seq2seq‡ [435]	✗	GT	GT	**51.7**	**60.1**
CTC	✗	GT	✗	58.5	-
CTC + KD	✗	ASR	✗	58.2	-
CTC + KD	✗	ASR/GT	✗	57.9	-
CTC	✗	✗	GT	-	68.8
CTC + KD	✗	✗	ASR	-	65.6
CTC + KD	✗	✗	ASR/GT	-	65.1
CTC + KD	ASR	ASR	ASR	54.2	-
CTC + KD	ASR	ASR/GT	ASR	52.2	-
CTC + KD	ASR	GT	ASR	**51.3**	-
CTC + KD	ASR	✗	ASR	-	61.7
CTC + KD	ASR	✗	ASR/GT	-	61.0
CTC + KD	ASR	✗	GT	-	**59.8**

one concludes that using extra unlabelled audio-visual speech data is indeed an effective way to boost performance.

Distillation significantly accelerates training, even compared to using ground truth transcriptions. In Fig. 4.2 we indicatively compare the learning curves of the baseline model, trained with CTC loss on ground truth transcriptions, and our proposed method, trained on transcriptions and posteriors from the teacher model. Our intuition is that the acceleration is due to the distillation providing explicit alignment information, contrary to CTC which only provides an implicit signal.

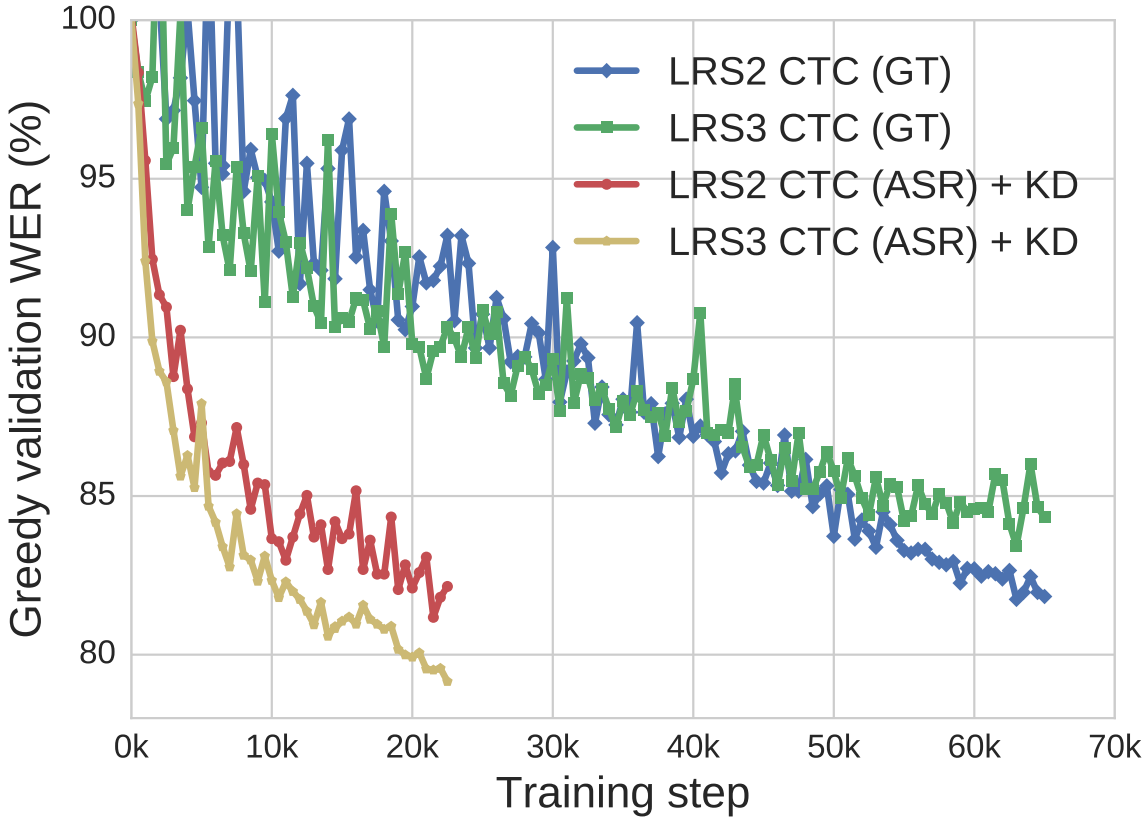


Figure 4.2: Progression of the greedy WER (validation) during training. Our method accelerates training significantly compared to training with CTC alone.

4.6 Discussion and future work

In this paper we demonstrated an effective strategy to train strong models for visual speech recognition by *distilling* knowledge from a pre-trained ASR model. This training method does not require manually annotated data and is therefore suitable for pre-training on unlabeled datasets. It can be optionally fine-tuned on a small amount of annotations and achieves performance that exceeds all existing lip reading systems aside from those trained using proprietary data.

In concurrent work, [435] also obtain state-of-the-art results on LRS2 and LRS3 that are very close to ours. We note that their improvements come from changes in the architecture, which should be orthogonal to our methodology; the two could be combined in future work for even better results.

There are many languages for which annotated data for visual speech recognition is very limited. Since our method is applicable to any video with a talking head, given access

to a pretrained ASR model and unlabelled data for a new language, we could naturally extend to lip reading that language.

Several authors [109, 229, 339, 369] have reported difficulties distilling acoustic models trained with CTC, stemming from the misalignment between the teacher and student spike timings. From the solutions proposed in the literature we only experimented with sequence-level KD [369] but did not observe any improvements. Investigating the extent of this problem in the cross-modal distillation domain is left to future work.

The method we have proposed can be scaled to arbitrarily large amounts of data. Given resource constraints we only utilized VoxCeleb2 and trained a relatively small network. In future work we plan to scale up in terms of both dataset and model size to develop models that can match and surpass the ones trained on very large-scale annotated datasets.

Statement of authorship

A statement of authorship for this paper is provided in Appendix A.

5 | Now You're Speaking My Language: Visual Language IDentification

Triantafyllos Afouras[1] Joon Son Chung[1,2] Andrew Zisserman[1]

[1]Visual Geometry Group, Oxford [2]Naver Corporation

Abstract

The goal of this work is to train models that can identify a spoken language just by interpreting the speaker's lip movements. Our contributions are the following: (i) we show that models can learn to discriminate among 14 different languages using only visual speech information; (ii) we compare different designs in sequence modelling and utterance-level aggregation in order to determine the best architecture for this task; (iii) we investigate the factors that contribute discriminative cues and show that our model indeed solves the problem by finding temporal patterns in mouth movements and not by exploiting spurious correlations. We demonstrate this further by evaluating our models on challenging examples from bilingual speakers.

Published in the proceedings of INTERSPEECH, 2020, pp. 2402-2406.

5.1 Introduction

Language identification from audio is a relatively easy task for humans. Indeed we can distinguish between languages that we do not speak or understand [233]. Moreover, automatic language identification (LID) from audio speech, is a well studied problem [57, 101, 258, 320], and determining the spoken language is often a first step for multilingual speech recognition [156, 279].

But is it possible to infer the language spoken by only *looking* at the speaker's lip movements, without the audio? There is evidence that humans can infer the spoken language by observing the lip movements of the speaker [325, 356, 403]. Moreover, Newman and Cox [282, 283] have shown that, under controlled visual conditions, visual language identification can also be automated.

Our objective in this paper is visual language identification '*in the wild*' – speaker independent, and text (content) independent identification. To this end, we train and evaluate visual language identification (VLID) models on a large multilingual audio-visual speech dataset, composed of public datasets of TEDx talks. We show that VLID can be accomplished under more general conditions, with good accuracy and for a large number of languages. To ensure that the models are indeed distinguishing between languages by finding patterns in the mouth movement, and not instead using other factors (e.g. inferring ethnicity from appearance cues) or spurious correlations, we compare with a face recognition baseline and also evaluate the models on a dataset from a different domain, VoxCeleb2 [85].

VLID opens up a host of interesting applications such as automatically recognising the language in silent films, automatically detecting dubbing in films, or recognising the spoken language from a distance. Most importantly, from a practical perspective, it can be used to pre-condition lip reading models, which are highly dependent on context, and to make audio-based language identification more robust in noisy environments. Please see our website `http://www.robots.ox.ac.uk/~vgg/research/vlid` for video examples.

5.1.1 Related Work

Audio language identification. Research in audio language identification has a long history, and the performance given reasonably long speech segments is very high. The architectures, aggregation methods and loss functions used in the LID task are similar to those in speaker recognition. For example, Geng et al. [150] investigate the use of RNNs for temporal aggregation in language identification. Cai et al. [49] explore the encoder and loss function for LID and propose some efficient temporal aggregation strategies, while Chen et al. [71] use NetVLAD [19] for temporal aggregation. In more recent work [50] use a 2D CNN as feature extractor with a BLSTM backend for temporal modelling and a self-attentive pooling layer for utterance level aggregation. The experiments show that decision-level fusion of different architectures yields the best results. Miao et al. [263] propose the use of a CNN-LSTM-TDNN encoder in combination with attention mechanisms in both time and frequency. Padi et al. [297] use a BLSTM-based attention model, obtaining state-of-the-art results on the NRE17 dataset.

Table 5.1: Statistics of audio-visual datasets used for training and evaluating our VLID models and baselines. **# videos:** Number of original YouTube videos. **# hours:** Total number of hours **# clips:**. Number of clips (each video is separated into multiple clips). For each statistic, we shown the minimum per language in parenthesis.

dataset	# hours	# videos	# clips
LRS3-Lang+ (dev)	1,707 (38)	19,300 (342)	683k
LRS3-Lang+ (test)	166 (0.9)	1,816 (30)	59k
VoxCeleb2-Lang	9 (0.8)	1,595 (98)	8.8k
VoxCeleb2-Biling	20.7 (0.7)	921 (26)	15k

Wan et al. [390] and Mazzawi et al. [259] also investigate LSTM based architectures for this dataset. Titus et al. [378] explore the effect of accent in language identification performance and train models robust to accented speech.

Visual language identification. The ability of humans to recognize languages by observing the lip movements of the speaker has been researched in psycholinguistics. Soto et al. [356] first report that facial speech information alone is sufficient for language identification. Weikum et al. [403] study visual speech identification in infants, while Ronquest et al. [325] investigate if humans are able to distinguish between English and Spanish based on visual speech.

However, there is limited research in using the visual modality to automatically identify the spoken language. Previous works by Newman and Cox [282, 283] are of closest relevance to ours: they introduce visual language identification as a classification problem, and show that languages can be classified by using only lip motion. However, the videos used are constrained to studio conditions, with a small number of subjects reading a set text, and their method does not use deep learning methods. Also related is [65] that identifies language in music videos by using both audio and video cues, while [359] used facial landmarks to classify between two languages, English and French. Brahme et al. [44] use constrained local models to the solve same task.

Lip reading. The methods used in visual language identification are closely related to those used for lip reading. There has been significant progress in the recent years, mainly due to the advances in deep learning and the creation of large scale datasets. While earlier work in the field used neural networks to predict phonemes [285] or words [87, 360], it has been proven more

recently that automatic lip reading can be generalised to continuous speech in unconstrained domains [4, 86, 305, 347, 435]. Recent works have shown that lip reading models trained on very large datasets can achieve word error rates as low as 33% on a real-world dataset, far exceeding the performance of professional lip readers [256].

5.2 Datasets

For training and evaluation, we use the LRS3-Lang [8] and LRS3 [5] datasets, as well as VoxCeleb2 [85] as a second multilingual test set. We show aggregate statistics of all datasets used in Table 5.1.

5.2.1 LRS3-Lang+

LRS3-Lang [8] is a multilingual audio-visual dataset based on videos collected from TEDx talks. The dataset covers 13 different (non-English) languages with a total of over 1,300 hours of video. For English we use the "pretrain" set of LRS3 [5], where the videos come from the same domain (TED(x) talks) and the exact same process has been followed to collect the data. The test set of LRS3 is small and contains short segments of no more than 6 seconds long. Therefore we re-split the "pretrain" set into a development and test set containing disjoint speakers. We incorporate this new split into LRS3-Lang as the English part to create a composite multilingual dataset of 14 languages, which we call LRS3-Lang+. The relative distribution of languages in our composite dataset are shown in Figure 5.1.

5.2.2 VoxCeleb2

VoxCeleb2 is an audio-visual speech dataset which consists of 5,994 speakers with a total of 1,092,009 clips in the development set, and 118 speakers with 36,237 clips in the test set. To assess the cross-domain generalization capabilities of the models and baselines (trained on LRS3-Lang+), we create two subsets from the development set of VoxCeleb2, which we use as test sets.

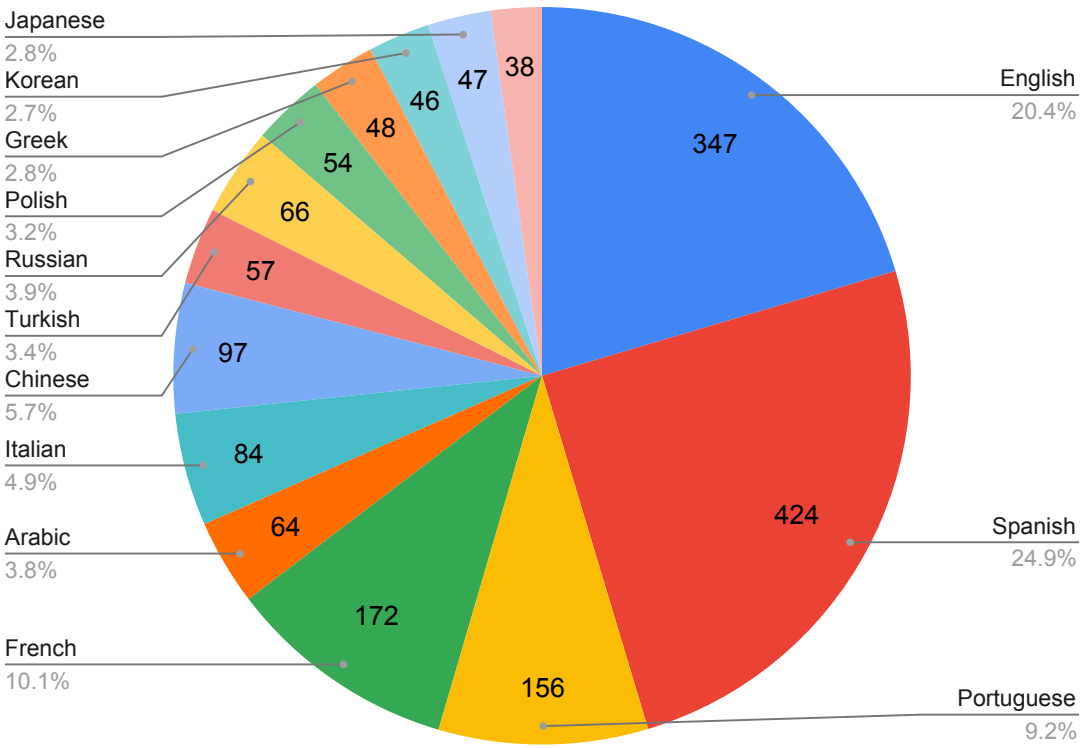


Figure 5.1: Language distribution of the `LRS3-Lang+` dataset in number of hours.

VoxCeleb2-Lang. `VoxCeleb2` contains no language labels, however the identity of the speakers and their nationality are known. We therefore obtain language labels from two sources. The first is training an audio-only model on `LRS3-Lang+` (details in Section 5.3) and using it to classify the audio of the speakers in `VoxCeleb2`. The second source is using the nationality of the speakers: each language is assigned a list of nationalities – i.e. countries where the language is predominantly spoken. For example, English is associated with American, British, Australian, and Scottish nationalities; Spanish is associated with Spanish, Mexican, and Argentinean nationalities etc. For every speaker, we then use their nationality to list a set of possible languages. This narrows down the search space for each language considerably. The final language pseudo-labels are obtained by exploiting the redundancy between these two sources: For a given video, we only assign a language label when the audio-only model predicts one of the languages associated with the nationality of the speaker with a probability higher than a strict threshold (90%). This process gives us very accurate pseudo-labels, however leaves very

few samples (less than 0.5 hour in total) for Japanese, Arabic and Greek. We therefore exclude these languages during evaluation on this dataset. The above procedure results in 11 languages, each containing material from at least 98 original YouTube videos each (see Table 5.1).

VoxCeleb2-Biling. To assess our models on bilingual speakers, we isolate individual speakers in `VoxCeleb2-Lang` who, across multiple videos, appear to be speaking both in English and in a non-English language with a high confidence, as determined by the audio model prediction. This is common due to the Celebrity content of the `VoxCeleb2` dataset (international actors, football players, politicians etc). We then create pairs of mother-tongue and English clips for those speakers. We refer to the resulting split as `VoxCeleb2-Biling`.

5.3 Architecture

We implement two types of models: an audio baseline, using audio features for LID, and our lip models using video features for VLID.

5.3.1 Input representation

Audio features. The input to the audio LID network is 80-dimensional log-mel spectrograms, extracted at every 10ms with 25ms frame length.

Video features. We extract embeddings modelling the lip movement with a spatio-temporal (3D/2D) ResNet18 network [175, 360] pretrained on word-level lip reading in English [4]. The model ingests a sequence of video frames (converted to grayscale) and outputs 512-dimensional visual features densely, one for every input frame.

5.3.2 Sequence modeling

We consider variations of Time-Delay Neural Networks (TDNN) and BLSTM [187, 336] encoders for the back-end. Those models ingest the visual features and convert them to repre-

sentations more discriminative for the language recognition task, whilst potentially modelling longer term temporal dependencies. We experiment with 3 different encoder architectures.

TDNN model. This is a 10-layer residual temporal (1D) convolutional network. We use depth-wise separable convolutions [79] which we find to train faster and overfit less. The kernel width is set to 5, the number of channels to 512, and the temporal stride to 1 for all the layers.

TDNN + BLSTM. This model uses a TDNN as described above, followed by a bi-directional LSTM (BLSTM) with a cell dimension of 512.

3×BLSTM. This model, inspired by [259], uses a stack of 3 BLSTMs with cell size 512.

Utterance level aggregation. In line with the common practices in the audio LID literature, we also experiment with 3 different utterance-level aggregation techniques.

Temporal average pooling (TAP). The TAP layer simply takes the mean of the features along the time domain.

Self-attentive pooling (SAP). Unlike the TAP layer that equally pools the features over time, [49] introduces a self-attentive pooling layer that pays attention to the frames that are more informative for utterance-level speaker recognition.

NetVLAD. We also consider NetVLAD[19], which has been successfully used for temporally aggregating features in speech models for LID [71] and speaker verification [411]. NetVLAD mimics the BoW-derived VLAD[102] descriptor by learning a feature vocabulary from the input representations, then soft-quantising them over this dictionary and finally aggregating the results (in our case temporally).

5.3.3 Face recognition ablation

In order to assess to what extent our models learn to distinguish between spoken languages and are not using other appearance cues that are strongly correlated (e.g. ethnicity), we also consider the following baseline: We take a ResNet50 convolutional network [175] pretrained for face

Table 5.2: Language Identification performance on the test set of `LRS3-Lang+` and the `VoxCeleb2-Lang` split. The average class accuracy is reported everywhere (higher is better). For all lip-reading models, a 3D/2D ResNet18 frontend is implied, and only the sequence-processing backend is varied and listed for comparison. **Mod.:** Input modality; A: Audio; L: Lips; F: Face. **Agg.:** Temporal aggregation strategy; NV.: NetVLAD. For `LRS3-Lang+` we report the average over all 14 languages (chance = 7%), while for `VoxCeleb2-Lang`, the average over 11 languages (excluding Japanese, Korean and Greek, chance = 9%). As the audio model is used to generate the pseudo-labels for the `VoxCeleb2-Lang` dataset, we don't report its accuracy on this test set.

			LRS3-Lang+			**VoxCeleb2**	
Model	**Mod.**	**Agg.**	**5s**	**10s**	**30s**	**5s**	**10s**
TDNN + BLSTM	A	TAP	95.6	96.6	97.3	-	-
ResNet50	F	AP	66.0	67.0	67.5	16.3	20.6
ResNet50 frozen	F	AP	39.9	40.8	41.2	24.5	27.4
TDNN	L	TAP	**67.2**	**76.3**	81.8	56.0	64.8
TDNN	L	SAP	66.4	74.2	76.8	52.9	62.0
TDNN	L	NV	66.3	74.0	75.8	46.4	59.8
TDNN + BLSTM	L	TAP	64.0	75.5	79.1	52.4	61.5
TDNN + BLSTM	L	SAP	65.4	75.2	79.2	52.1	61.1
3×BLSTM	L	TAP	64.8	75.5	82.0	**59.5**	**67.4**
3×BLSTM	L	SAP	64.5	76.0	**84.0**	58.5	66.7

recognition on the VGGFace2 dataset [58] and fine-tune it on the on the VLID task. We consider 2 versions: (i) the model is trained end-to-end; (ii) the model is frozen at the penultimate residual block, i.e. only the last residual block and classification layers are fine-tuned.

5.4 Experimental Setting

Training. All models are trained only on `LRS3-Lang+`. We train the LID and VLID models by randomly sampling a segment of T contiguous frames from a given training clip. To accelerate training for all models we use a curriculum, first setting $T = 64$ and then increasing it to 128 and 256 frames (2.5s, 5s and 10s). During training the batches are balanced for languages. For languages with more samples available, the same frames are seen less often. To run inference with the RNN-based models on sequences longer than 256 frames (max seen during training), we split the sequence into 128 frame segments with 50% overlap and

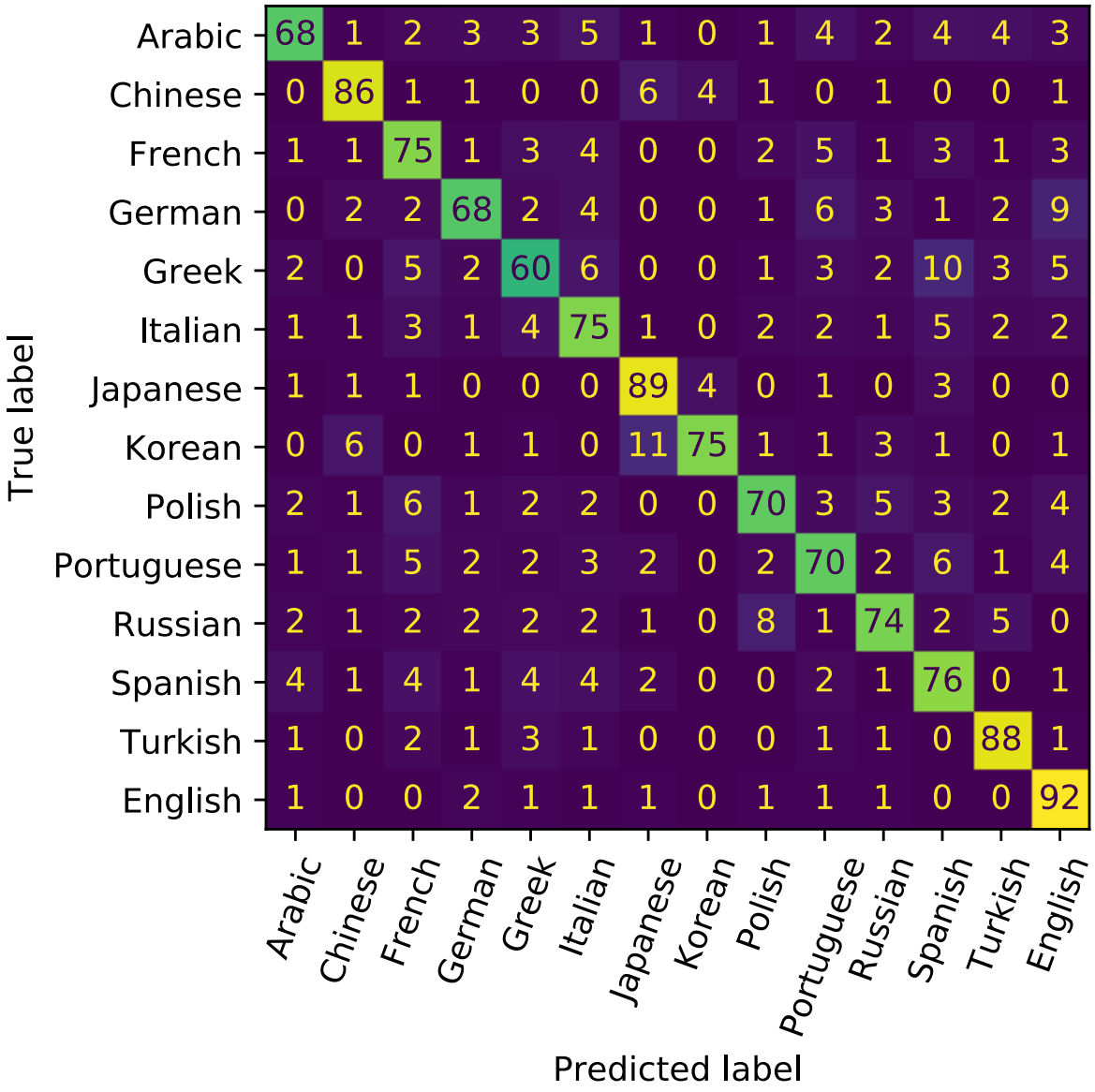


Figure 5.2: Confusion matrix for predictions of *3×BLSTM-SAP* model on the test set of `LRS3-Lang+` (10 seconds experiment).

then average the predictions [390].

The face recognition baselines are trained by feeding one random frame from a clip at a time, with a batch of 32. For a fair comparison with our models, during inference we feed the face recognition models with all the frames of each test clip (e.g. 125 frames for the 5 seconds ones). The prediction is then obtained by averaging the model logits for all the frames.

Evaluation protocol. We evaluate on sequences of 5, 10 and 30 seconds long. As continuous

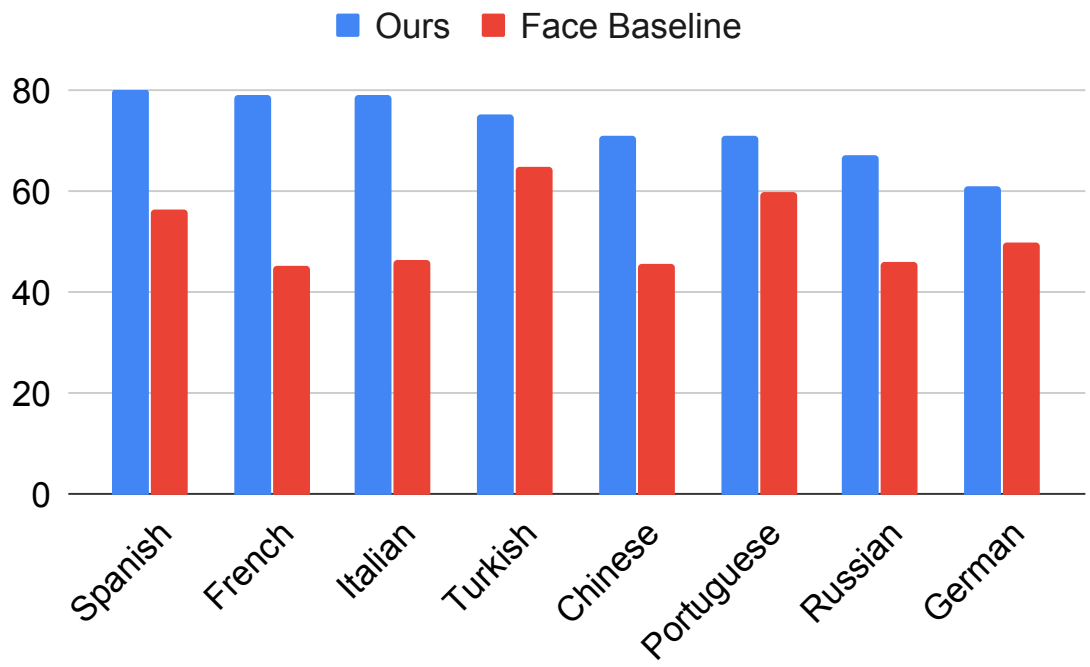


Figure 5.3: Visual language identification accuracy on bilingual test set (`VoxCeleb2-Biling`). The model is tasked with discriminating between each language and English. Utterances of length 5 seconds are used. Chance accuracy is 50%.

Figure 5.4: Challenging examples from `VoxCeleb2-Biling` for which our VLID model correctly predicts the spoken language (indicated by the flag). Modelling of the lip movements is essential to solve this task.

clips of 30 seconds are very scarce in the datasets, we synthesize those by merging smaller clips from the same video together. For all experiments, the metric that we report is the average class language identification accuracy. We evaluate our models and baselines on the test set of `LRS3-Lang+`, on `VoxCeleb2-Lang`, and on `VoxCeleb2-Biling`.

5.5 Results

We summarize the results of our experiments on LRS3-Lang+ and VoxCeleb2-Lang in Table 5.2.

As expected, the audio LID model achieves a very high accuracy. The visual VLID models also perform well. In both cases the model's performance improves as more temporal input is available. Indeed, when the visual models are supplied with 30 seconds of input the accuracy rises as high as 84%.

In terms of architectures, all options that we examine perform reasonably. The simplest of the models, *TDNN* performs best on LRS3-Lang+, except for the 30s case where the *3×BLSTM* model achieves marginally better results. When evaluating the models on the different domain of VoxCeleb2-Lang, the advantage of using the *3×BLSTM* is more apparent. Adding a BLSTM layer on top of the *TDNN* model impairs performance. In terms of utterance-level aggregation, neither SAP or NetVLAD clearly outperform simple temporal pooling. We conjecture that these results are due to overfitting in the more complicated models.

We show the confusion matrix for the predictions of the *3×BLSTM-SAP* model in Figure 5.2. We note that the languages that are most commonly confused have phonetic similarities (e.g. German-English, Greek-Spanish, Korean-Japanese, Russian-Polish).

We next turn to the question of whether the visual model is indeed modelling the temporal mouth patterns to recognize the language or is just relying on appearance cues, such as face shape or skin tone. It is worth noting that (i) the visual features only use monochrome (not RGB) inputs, and (ii) they are trained on a word-level lip reading task on videos from British television and then frozen. This limits the extent of the information that they can access from the raw frames. In contrast, the baselines have a varying degree of access to the raw frames – and it can be seen that they can exploit this in solving the task. Examining the performance of the ResNet50-based face models, we notice that the model trained end-to-end obtains good results on LRS3-Lang+. However, when evaluated on VoxCeleb2-Lang the same model performs very poorly. On the other hand, the evaluations of the model based on the

frozen ResNet50, pretrained on face recognition, shows relatively worse performance on LRS3-Lang+, but its generalization on VoxCeleb2-Lang is better. The above suggest that the end-to-end model finds some shortcut which leads it to greatly overfitting the dataset. We conjecture that this might be due to background landmarks or camera artefacts correlated with the location of shooting of the TEDx events.

VoxCeleb2-Lang. We note that there is a significant domain shift between LRS3-Lang+, where the models have been trained, and VoxCeleb2 as well as that the speaker identities between the two datasets are disjoint. As can be seen, the VLID models exhibit strong performance despite this domain shift. The face baselines, in contrast and as discussed above, drop in performance to near chance level. This demonstrates again that the VLID models are indeed using the mouth shape (visemes) and temporal changes for LID, and not employing shortcuts from the face and raw frames.

Bilingual speakers. On figure 5.3 we show results on bilingual speakers from VoxCeleb2. As expected, the accuracy of the face baseline fluctuates around the random performance (50%), as inferring the spoken language given the same face is very hard without any lip movement modelling. Our model significantly outperforms the baseline, reaching 80% accuracy for Spanish.

We show some qualitative examples of clips of bilingual speakers that our model predicts correctly in Figure 5.4. Please refer to our website for video examples.

5.6 Conclusion

We can give a qualified answer to the question posed in the introduction: "Yes, it is possible to infer the spoken language only by observing the speaker's lips, and to a remarkably good accuracy". Our experiments have shown that using lip movements for this task exceeds using appearance cues captured by face embeddings. Finally, by performing analysis on bilingual speakers we demonstrated that our trained models can even distinguish between different languages spoken by the same person.

In future work we plan to investigate which lip movements provide the most discriminative cues, as well as explore the visual similarities and differences between languages – e.g. determine if certain viseme combinations are more prominent for some groups of languages than in others.

Statement of authorship

A statement of authorship for this paper is provided in Appendix A.

Part II

Audio-visual speech enhancement

6 | The Conversation: Deep Audio-Visual Speech Enhancement

Triantafyllos Afouras[1] Joon Son Chung[1] Andrew Zisserman[1]

[1]Visual Geometry Group, Oxford

Abstract

Our goal is to isolate individual speakers from multi-talker simultaneous speech in videos. Existing works in this area have focussed on trying to separate utterances from known speakers in controlled environments. In this paper, we propose a deep audio-visual speech enhancement network that is able to separate a speaker's voice given lip regions in the corresponding video, by predicting both the magnitude and the phase of the target signal. The method is applicable to speakers unheard and unseen during training, and for unconstrained environments. We demonstrate strong quantitative and qualitative results, isolating extremely challenging real-world examples.

Published in the proceedings of INTERSPEECH, 2020, pp. 2402-2406.

6.1 Introduction

In the film *The Conversation* (dir. Francis Ford Coppola, 1974), the protagonist, played by Gene Hackman, goes to inordinate lengths to record a couple's converservation in a crowded city square. Despite many ingenious placements of microphones, he did not use the lip motion of the speakers to suppress speech from others nearby. In this paper we propose a new model for this task of audio-visual speech enhancement, that he could have used.

More generally, we propose an audio-visual neural network that can isolate a speaker's voice from others, using visual information from the target speaker's lips: Given a noisy audio signal and the corresponding speaker video, we produce an enhanced audio signal containing only the target speaker's voice with the rest of the speakers and background noise suppressed.

Rather than synthesising the voice from scratch, which would be a challenging task, we instead predict a mask that filters the noisy spectrogram of the input. Many speech enhancement approaches focus on refining only the magnitude of the noisy input signal and use the noisy phase for the signal reconstruction. This works well for high signal-to-noise-ratio scenarios, but as the SNR decreases, the noisy phase becomes a bad approximation of the ground truth one [132]. Instead, we propose correction modules for both the magnitude and phase. The architecture is summarised in Figure 6.1. In training, we initialize the visual stream with a network pre-trained on a word-level lip-reading task, but after this, we train from unlabelled data (Section 6.3.1) where no explicit annotation is required at the word, character or phoneme-level.

There are many possible applications of this model; one of them is automatic speeech recognition (ASR) – while machines can recognise speech relatively well in noiseless environments, there is a significant deterioration in performance for recognition in noisy environments [18]. The enhancement method we propose could address this problem, and improve, for example,

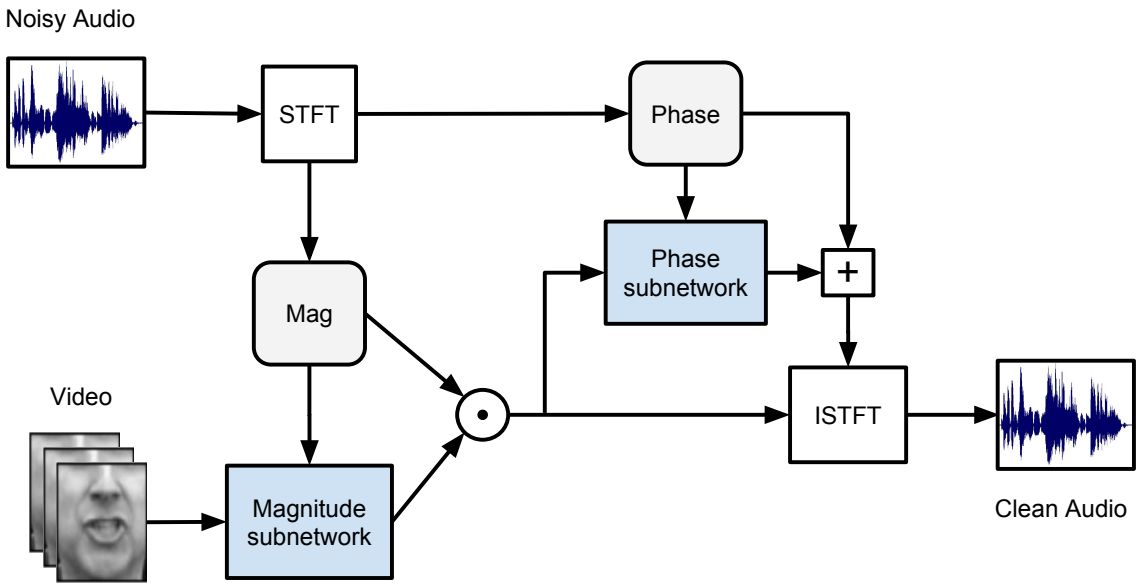


Figure 6.1: Audio-visual enhancement architecture overview. It consists of two modules: a magnitude sub-network and a phase sub-network. The first sub-network receives the magnitude spectrograms of the noisy signal and the speaker video as inputs and outputs a soft mask. We then multiply the input magnitudes element-wise with the mask to produce a filtered magnitude spectrogram. The magnitude prediction, along with the phase spectrogram obtained from the noisy signal are then fed into the second sub-network, which produces a phase residual. The residual is added to the noisy phase, producing the enhanced phase spectrograms. Finally the enhanced magnitude and phase spectra are transformed back to the time domain, yielding the enhanced signal.

ASR for mobile phones in a crowded environment, or automatic captioning for YouTube videos.

The performance of the model is evaluated for up to five simultaneous voices, and we demonstrate both strong qualitative and quantitative performance. The trained model is evaluated on unconstrained 'in the wild' environments, and for speakers and languages unseen at training time. To the best of our knowledge, we are the first to achieve enhancement under such general conditions. We provide supplementary material with interactive demonstrations on `http://www.robots.ox.ac.uk/~vgg/demo/theconversation`.

6.1.1 Related works

Various works have proposed methods to isolate multi-talker simultaneous speech. The majority of these are based on methods that only use the audio, *e.g.* by using voice characteristics of a known speaker [199, 255, 313, 315, 392]. Compared to audio-only methods, we not only separate the voices but also properly assign them to the speakers, by using the visual information.

Speech enhancement methods have traditionally only dealt with filtering the spectral magnitudes, however many approaches have been recently been proposed for jointly enhancing the magnitude and phase spectra [112, 122, 132, 186, 275, 276, 329]. The prevalent method for estimating phase spectra from given magnitudes in speech synthesis is the one proposed by Griffin and Lim [161].

Prior to deep learning, a large number of previous works have been developed for audio-visual speech enhancement by predicting masks [205, 249] or otherwise [15, 103, 153, 154, 181, 182, 396], with an overview of audio-visual source separation is provided in [321]. However, we will concentrate from hereon on methods that have built on these using a deep learning framework.

In [119] a deep neural network is developed to generate speech from silent video frames of a speaking person. This model is used in [133] for speech enhancement, where the predicted spectrogram serves as a mask to filter the noisy speech. However, the noisy audio signal is not used in the pipeline, and the network is not trained for the task of speech enhancement. In contrast, [134] synthesizes the clean signal conditioning on both the mixed speech input and the input video. [188] also use a similar audio-visual fusion method, trained to both generate

the clean signal and to reconstruct the video. Both papers use the phase of the noisy input signal as an approximation for the clean phase. However, these methods are limited in that they are only demonstrated under constrained conditions (*e.g.* the utterances consist of a fixed set of phrases in [188]), or for a small number of speakers that have been seen during training.

Our method differs from these works in several ways: (i) we do not treat the spectrograms as images but as temporal signals with the frequency bins as channels; this allows us to build a deeper network with a large number of parameters that trains fast; (ii) we generate a soft mask for filtering instead of directly predicting the clean magnitudes, which we found to be more effective; (iii) we include a phase enhancing sub-network; and, finally, (iv) we demonstrate on previously unheard (and unseen) speakers and on in-the-wild videos.

In concurrent and independent work, [120] develop a similar system, based on dilated convolutions and a bidirectional LSTM, demonstrating good results in unconstrained environments, while [293] train a network for audio-visual synchronisation and successfully use its features for speech separation.

The enhancement method proposed here is complementary to lip reading [26, 87, 360], which has also been shown to improve ASR performance in noisy environments [86, 304].

6.2 Architecture

This section describes the input representations and architectures for the audio-visual speech enhancement network. The network ingests continuous clips of the audio-visual data. The model architecture is given in detail in Figure 6.2.

6.2.1 Video representation

Visual features are extracted from the input image frame sequence with a spatio-temporal residual network similar to the one proposed by [360], pre-trained on a word-level lip reading task. The network consists of a 3D convolution layer, followed by a 18-layer ResNet [175]. For every video frame the network outputs a compact 512 dimensional feature vector f_0^v (where

the subscript 0 refers to the layer number in the audio-visual network). Since we train and evaluate on datasets with pre-cropped faces, we do not perform any extra pre-processing, besides conversion to grayscale and an appropriate scaling.

6.2.2 Audio representation

The acoustic representation is extracted from the raw audio waveforms using Short Time Fourier Transform (STFT) with a Hann window function, which generates magnitude and phase spectrograms. STFT parameters are computed in a similar manner to [134], so that every video frame of the input sequence corresponds to four temporal slices of the resulting spectrogram. Since the videos are at 25fps (40ms per frame), we select a hop length of 10ms with a window length of 40ms at a sample rate of 16Khz. The resulting spectrograms have frequency resolution $F = 321$, representing frequencies from 0 to 8 kHz, and time resolution $T \approx \frac{T_s}{hop}$, where T_s is the duration of the signal in seconds. The magnitude and phase spectrograms are represented as $T \times 321$ and $T \times 642$ tensors respectively, with the real and imaginary components concatenated along the frequency axis for the latter. We convert the magnitudes to mel-scale spectrograms, with 80 frequency bins before feeding them to the magnitude subnetwork, however we conduct the filtering on the original, linear-scale spectrograms.

6.2.3 Magnitude sub-network

The visual feature sequence f_0^v is processed by a residual network of 10 convolutional blocks. Every block consists of a temporal convolution with kernel width 5 and stride 1, preceded by ReLU activation and batch normalization. A shortcut connection adds the block's input to the result of the convolution. A similar stack of 5 convolutional blocks is employed for processing the audio stream. The convolutions are performed along the temporal dimension, with the frequencies of the noisy input spectrogram M_n viewed as the channels. Two of the intermediate blocks perform convolutions with stride 2, overall down-sampling the temporal dimension by 4, in order to bring it down to the video stream resolution. The skip connections of those layers are down-sampled by average pooling with stride 2. The audio and visual streams are then concatenated over the channel dimension: $f_0^{av} = [f_{10}^v; f_5^a]$. The fused tensor

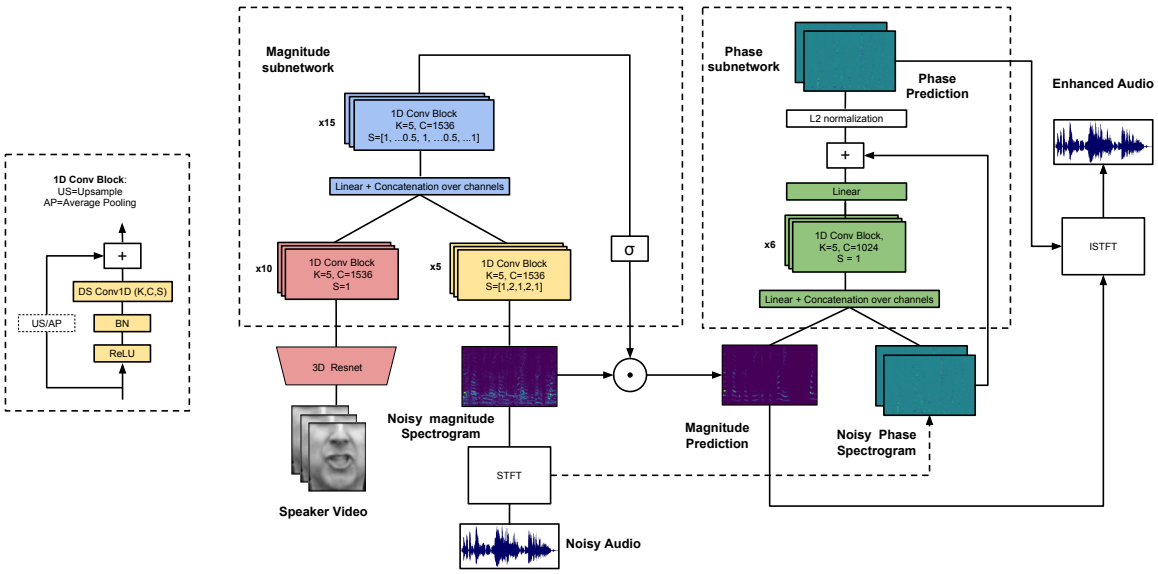


Figure 6.2: Audio-visual enhancement network. **BN**: Batch Normalization, **C**: number of channels; **K**: kernel width; **S**: strides – fractional ones denote transposed convolutions. The network consists of a magnitude and a phase sub-network. The basic building unit is the temporal convolutional block with pre-activation [176] shown on the left. Identity skip connections are added after every convolution layer (and speed up training). All convolutional layers have 1536 channels in the magnitude sub-network and 1024 in the phase subnetwork. Depth-wise separable convolution layers [79] are used, which consist of a separate convolution along the time dimension for every channel, followed by a position-wise projection onto the new channel dimensions (equivalent to a convolution with kernel width 1).

is passed through another stack of 15 temporal convolution blocks. Since we want the output mask to have the same temporal resolution as the input magnitude spectrogram, we include two transposed convolutions, each up-sampling the temporal dimension by a factor of 2, resulting in a factor of 4 in total. The fusion output is projected through position-wise convolutions onto the original magnitude spectrogram dimensions and passed through sigmoid activation in order to output a mask with values between 0 and 1. The resulting tensor is multiplied with the noisy magnitude spectrogram element-wise to produce the enhanced magnitudes:

$$\hat{M} = \sigma(W_m^T f_{15}^{av}) \odot M_n$$

6.2.4 Phase sub-network

Our intuition for the design of the phase enhancement sub-network is that there is structure in speech that induces a correlation between the magnitude and phase spectrograms. As with the magnitudes, instead of trying to predict the clean phase from scratch, we only predict a

residual that refines the noisy phase. The phase sub-network is therefore conditioned on both the noisy phase and the magnitude predictions. These two inputs are fused together through linear projection and concatenation and then processed by a stack of 6 temporal convolution blocks, with 1024 channels each. The phase residual is formed by projecting the result onto the dimensions of the phase spectrogram and is added to the noisy phase. The clean phase prediction is finally obtained by L_2-normalizing the result:

$$\phi_6 = \underbrace{ConvBlock(...ConvBlock([W_{m\phi}^T \hat{M}; W_{n\phi}^T \Phi_n]))}_{\times 6}$$

$$\hat{\Phi} = \frac{(W_\phi^T \phi_6 + \Phi_n)}{||(W_\phi^T \phi_6 + \Phi_n)||_2}$$

In training, the weights of the layers are initialized with small values and zero biases, so that the initial residuals are nearly zero and the noisy phase is propagated to the output.

6.2.5 Loss function

The magnitude subnetwork is trained by minimizing the L_1 loss between the predicted magnitude spectrogram and the ground truth. The phase subnetwork is trained by maximizing the cosine similarity between the phase prediction and ground truth, scaled by the ground truth magnitudes. The overall optimisation objective is:

$$\mathcal{L} = ||\hat{M} - M^*||_1 - \lambda \frac{1}{TF} \sum_{t,f} M_{tf}^* < \hat{\Phi}_{tf}, \Phi_{tf}^* > \tag{6.1}$$

6.3 Experiments

6.3.1 Datasets

The model is trained on two datasets: the first is the BBC-Oxford Lip Reading Sentences 2 (LRS2) dataset [86, 90], which contains thousands of sentences from BBC programs such as Doctors and EastEnders; the second is VoxCeleb2 [82], which contains over a million utterances spoken by over 6,000 different speakers.

Mag	Φ \ # Spk.	SIR (dB) 2	3	4	5	SDR (dB) 2	3	4	5	PESQ 2	3	4	5	WER (%) 2	3	4	5
Mix	**Mix**	–	–	–	–	-0.3	-3.4	-5.4	-6.7	1.73	1.47	1.37	1.21	93.1	99.5	99.9	100
Pr	**GT**	10.8	13.2	13.8	13.7	15.7	13.0	10.8	9.5	3.41	3.05	2.93	2.80	9.4	12.0	16.7	21.5
Pr	**GL**	0.9	2.5	3.6	4.0	-2.9	-2.8	-2.9	-2.7	2.98	2.71	2.52	2.35	10.5	13.7	20.3	27.8
Pr	**Mix**	1.6	2.7	2.5	2.0	10.5	7.8	5.9	4.8	3.02	2.70	2.49	2.33	10.8	14.9	22.0	31.9
Pr	**Pr**	3.9	5.4	5.4	4.8	11.8	9.1	7.1	5.8	3.08	2.79	2.56	2.43	9.7	13.8	20.3	28.9

Table 6.1: Evaluation of speech enhancement performance on the LRS2 dataset, for scenarios with different number of speakers (denoted by # Spk). The magnitude (Mag) and phase (Φ) columns specify if the spectrograms used for the reconstructions are predicted or are obtained directly from the mixed or ground truth signal: **Mix:** Mixed; **Pr:** Predicted; **GT:** Ground Truth; **GL:** Griffin-Lim; **SIR:** Signal to Interference Ratio; **SDR:** Signal to Distortion Ratio; **PESQ:** Perceptual Evaluation of Speech Quality, varies between 0 and 4.5; (higher is better for all three); **WER:** Word Error Rate from off-the-shelf ASR system (lower is better). The WER on the ground truth signal is 8.8%.

The LRS2 dataset is divided into training and test sets by broadcast date, in order to ensure that there is no overlapping video between the sets. The dataset covers a large number of speakers, which encourages the trained model to be speaker agnostic. However, since no identity labels are provided with the dataset, there may be some overlapping speakers between the sets. The ground truth transcriptions are provided with the dataset, which allows us to perform quantitative tests on the intelligibility of the generated audio.

The VoxCeleb2 dataset lacks the text transcriptions, however the dataset is divided into training and test sets by identity, which allows us to test the model explicitly for speaker-independent performance.

The audio and video on these datasets are properly synchronized. Evaluation on videos where this is not the case (*e.g.* TV broadcast), is possible by preprocessing with the pipeline described in [88] to detect and track active speakers and synchronize the video and the audio.

6.3.2 Experimental setup

We examine scenarios where we add 1 to 4 extra interference speakers on the clean signal, therefore we generate signals with 2 to 5 speakers in total. It should be noted that the task of separating the voice of multiple speakers with equal average "loudness" is more challenging than separating the speech signal from background babble noise.

6.3.3 Evaluation protocol

We evaluate the enhancement performance of the model in terms of perceptual speech quality using the blind source separation criteria described in [128] (we use the implementation provided by [118]). The Signal to Interference Ratio (SIR) measures how well the unwanted signals have been suppressed, the Signal to Artefacts Ratio (SAR) accounts for the introduction of artefacts by the enhancement process, and the Signal to Distortion Ratio (SDR) is an overall quality measure, taking both into account. We also report results on PESQ [322], which measures the overall perceptual quality and STOI [368], which is correlated with the intelligibility of the signal. From the metrics presented above, PESQ has been shown to be the one correlating best with listening tests that account for phase distortion[274].

Additionally, we use an ASR system to test for the intelligibility of the enhanced speech. For this, we use the Google Speech Recognition interface, and report the Word Error Rates (WER) on the clean, mixed and generated audio samples.

6.3.4 Training

We pre-train the spatio-temporal visual front-end on a word-level lip reading task, following [360]. This proceeds in two stages: first, training on the LRW dataset [87], which covers near-frontal poses; and then on an internal multi-view dataset of a similar size. To accelerate the subsequent training process, we freeze the front-end, pre-compute and save the visual features for all the videos, and also compute and save the magnitude and phase spectrograms for both the clean and noise audio.

Training takes place in three phases: first, the magnitude prediction sub-network is trained, following a curriculum which starts with high SNR inputs (i.e. only one additional speaker) and then progressively moves to more challenging examples with a greater number of speakers; second, the magnitude sub-network is frozen, and only the phase network is trained ; finally, the whole network is fine-tuned end-to-end. We did not experiment with the hyperparameter balancing the magnitude and phase loss terms, but set it to $\lambda = 1$.

To generate training examples we first select a reference pair of visual and audio features (v_r, a_r) by randomly sampling a 60-frame clean segment, making sure that the audio and visual features correspond and are correctly aligned. We then sample N noise spectrograms $x_n, n \in [1, N]$, and mix them with the reference spectrogram in the frequency domain by summing up the complex spectra, obtaining the mixed spectrogram a_m. This is a natural way to augment our training data since a different combination of noisy audio signals is sampled every time. Before adding in the noise samples, we normalize their energy to have the reference signal's one: $a_m = a_r + \sum_n \frac{rms(x_r)}{rms(a_n)} a_n$.

6.3.5 Results

LRS2. We summarize our results on the test set of the LRS2 dataset in Table 6.1. The performance under the different metrics is listed for the following signal types: The mixed signal which serves as a baseline, and the reconstructions that are obtained using the magnitudes predicted by our network and either the ground truth phase, the phase approximated with the Griffin Lim algorithm, the mixed signal phase or the predicted phase. The signal reconstructed from predicted magnitudes and phases is what we consider the final output of our network.

The evaluation when using the ground truth phase is included as an upper bound to the phase prediction. As can be seen from all measures on the mixed signal, the task becomes increasingly difficult as more speakers are added. In general both the BSS metrics and PESQ correlate well with our observations. It is interesting to note that while more speakers are added, the SIR stays roughly the same, however more overall distortion is introduced. The model is very effective in suppressing cross-talk in the output, however it does so with a trade-off in the quality of the target voice.

The phase predicted by our network performs better than the mixed phase. Even though the improvement is relatively small in numbers, the difference in speech quality is noticeable as the "robotic" effect of having off-sync harmonics is significantly reduced. We encourage the reader to listen to the samples in the supplementary material, where those differences can be

understood better. However, the considerable gap with the performance of the ground truth phase shows that there is much room for improvement in the phase network.

The transcription results using the Google ASR are also in line with these findings. In particular, it is noteworthy that our model is able to generate highly intelligible results from noisy audio that is incomprehensible by a human or an ASR system.

Although the content is mainly carried by the magnitude, we see major improvement in terms of WER when using a better phase approximation. It is interesting to note that, although the phase obtained using the Griffin Lim (GL) algorithm achieves significantly worse performance on the objective measures, it demonstrates relatively strong WER results, even slightly surpassing the predicted phase by a small margin in the case of 5 simultaneous speakers.

VoxCeleb2. In order to explicitly assess whether our model can generalize to speakers unseen during training, we also fine-tune and test on VoxCeleb2, using train and test sets that are disjoint in terms of speaker identities. The results are summarized in Table 6.2, where we showcase an experiment for the 3-speaker scenario. We additionally include evaluation using the SAR and STOI metrics. Overall the performance is comparable to, but slightly worse than, on the LRS2 dataset – which is in line with the qualitative performance. This can be attributed to the visual features not being fine-tuned, and the presence of a lot of other background noise in VoxCeleb2. The results confirm that the method can generalize to unseen (and unheard) speakers.

The last column of the table shows the PESQ evaluation for the original model trained on LRS2, without any fine-tuning on VoxCeleb. The performance is worse than that of the fine-tuned model, however it clearly works. Since LRS2 is constrained to English speakers only, but VoxCeleb2 contains multiple languages, this demonstrates that the model learns to generalise to languages not seen during training.

6.3.6 Discussion

Phase refinement. Training our whole network end-to-end decreases the phase loss and this might suggest that the inclusion of visual features also improves the phase enhance-

Mag	Φ	SIR	SAR	SDR	STOI	PESQ	PESQ-NF
Mix	**Mix**	-	-1.59	-2.99	0.34	1.58	1.58
Pr	**GT**	11.43	16.41	10.30	0.77	3.02	2.79
Pr	**GL**	2.05	3.49	-2.42	0.65	2.59	2.39
Pr	**Mix**	1.72	13.54	6.71	0.65	2.59	2.41
Pr	**Pr**	5.02	13.77	7.91	0.67	2.67	2.45

Table 6.2: Evaluation of speech enhancement performance on the VoxCeleb2 dataset, for 3 simultaneous speakers, Notations are described in the caption of Table 6.1. Additional metrics used here: **SAR:** Signal to Artefacts Ratio; **STOI:** Short-Time Objective Intelligibility, varies between 0 and 1; **PESQ-NF:** PESQ score with a model that has not been fine-tuned on VoxCeleb; Higher is better for all.

ment. However, a thorough investigation to determine if, and to what extent, this is true is left to future work.

AV synchronization. Our method is very sensitive to the temporal alignment between the voice and the video. We use SyncNet for the alignment, but since the method can fail under extreme noise, we need to build some invariance in the model. In future work this will be incorporated in the model.

6.4 Conclusion

In this paper, we have proposed a method to separate the speech signal of a target speaker from background noise and other speakers using visual information from the target speaker's lips. The deep network produces realistic speech segments by predicting both the phase and the magnitude of the target signal; we have also demonstrated that the network is able to generate intelligible speech from very noisy audio segments recorded in unconstrained 'in the wild' environments.

Statement of authorship

A statement of authorship for this paper is provided in Appendix A.

7 | My Lips Are Concealed: Audio-visual Speech Enhancement Through Obstructions

Triantafyllos Afouras[1] Joon Son Chung[1] Andrew Zisserman[1]

[1]Visual Geometry Group, Oxford

Abstract

Our objective is an audio-visual model for separating a single speaker from a mixture of sounds such as other speakers and background noise. Moreover, we wish to hear the speaker even when the visual cues are temporarily absent due to occlusion. To this end we introduce a deep audio-visual speech enhancement network that is able to separate a speaker's voice by conditioning on both the speaker's lip movements and/or a representation of their voice. The voice representation can be obtained by either (i) enrollment, or (ii) by self-enrollment – learning the representation on-the-fly given sufficient unobstructed visual input. The model is trained by blending audios, and by introducing artificial occlusions around the mouth region that prevent the visual modality from dominating. The method is speaker-independent, and we demonstrate it on real examples of speakers unheard (and unseen) during training. The method also improves over previous models in particular for cases of occlusion in the visual modality.

Published in the proceedings of INTERSPEECH, 2018, pp. 3244-3248.

7.1 Introduction

While there has been great progress in the field of automatic speech recognition (ASR) in recent years, some key challenges remain, particularly the understanding of speech in very noisy environments or in cases where multiple people speak simultaneously. In this direction, isolating voices in multi-speaker scenarios, increasing the signal-to-noise ratio in noisy audio, or combinations of both are all important tasks.

Figure 7.1: An audio-visual speech enhancement model may fail when the lip region is occluded by e.g. a microphone. In such cases the input audio is often entirely filtered out and the result is silent output over the occluded frames. The aim of our method is to be robust to this kind of occlusions.

Until very recently, works in this area have only used the audio modality for the task. However, recent works have shown that the use of video can aid tremendously in solving the problem [3, 120, 293].

These audio-visual models have demonstrated impressive results, but given their dependence on the visual input, they may fail when the mouth area is occluded by the speaker's hands, a microphone (e.g. Fig. 7.1), or if the speaker turns their head away. Contemporaneously, it has been shown that an embedding of the speaker's voice can guide the separation of simultaneous speech [393].

In this paper we propose combining the two approaches, i.e. conditioning on both the video input containing the speaker's lip movement and an embedding of their voice, in order to make the audio-visual models robust to occlusions. Our assumption is that the video provides invaluable discriminative information when present, while the speaker embedding can help the model when the video is absent due to occlusions. In the simplest case, the voice embedding can be obtained from pre-enrolled audio.

While it is possible to separate simultaneous speakers using only the audio [183, 421], the

permutation issue in the time-domain remains an unsolved problem. With our approach, even partially occluded video can provide information on the voice characteristics of the speaker and resolve the ambiguity of assigning the separated voice to the speaker.

We make the following contributions: (i) we show how speaker embedding and visual cues can be combined to separate a single speaker from a mixture of voices despite the visual stream (the lips) being occluded; (ii) we propose a neural network model that can operate with video only, enrollment data only, or both; and (iii) we introduce a recurrent model that can bootstrap the computation of the speaker embedding under temporary occlusions, without requiring a prior speaker embedding. We term this *self-enrollment*.

7.1.1 Related Work

Audio-only enhancement and separation. Various methods have been proposed to isolate multi-talker simultaneous speech, the majority of which only use monaural audio, *e.g.* [199, 255, 313, 315, 392]. A number of recent works have addressed the permutation problem to separate unseen speakers. Deep clustering [183] uses embeddings trained to yield a low-rank approximation to an ideal pairwise affinity matrix, whilst Yu *et al.* employ a permutation invariant-loss [421].

Audio-visual speech enhancement. Prior to the advent of deep learning, numerous works have been developed for audio-visual speech enhancement [103, 153, 182, 205, 321, 396]. Several recent methods have used a deep learning framework for the same task – most notably [133, 134, 188]. However, these methods are limited in that they are only demonstrated under constrained conditions (e.g. the utterances consist of a fixed set of phrases), or for a small number of known speakers. Our previous work [3] proposed a deep audio-visual speech enhancement network that is able to separate a speaker's voice given lip regions in the corresponding video, by predicting both the magnitude and the phase of the target signal. Ephrat *et al.* [120] designed a network that conditions on the video input of all the source speakers and outputs complex masks, thus also enhancing both magnitude and phase. Owens

and Efros [293] train a network on audio-visual synchronization and use the learned features for speaker separation. These last works demonstrate general results in-the-wild case.

Enhancement by conditioning on voice only. Wang *et al.* [393] develop a method that separates voices conditioned on pre-learned speaker embeddings, showing that voice characteristics alone can be enough to determine the separation. This however relies on a pretrained model and does not use video.

We propose to combine the two ideas: using both visual input and voice embeddings from the target speaker; our method partially builds on [3, 120, 393].

7.2 Method

This section describes the architecture of the audio-visual speech enhancement network, which is given in Figure 7.2. The network receives three inputs: (i) the noisy audio to be enhanced; (ii) the corresponding video frames; (iii) a reference audio containing speech from the target speaker. We summarize the principal modules below. Details of the architecture are provided in Table 7.1.

Video representation. Input to the network is pre-cropped image frames, such as the face crops found in the LRS datasets [2, 5]. Visual features are extracted from the sequence of image frames using a spatiotemporal residual network described in [360]. The network contains a 3D convolution layer, followed by a common 18-layer 2D ResNet [175]. For every video frame it outputs a compact 512 dimensional feature vector.

Audio representation. As acoustic features, we use the magnitude and phase spectrograms extracted from the audio waveforms using a Short Time Fourier Transform (STFT) with a 25ms window length and a 10ms hop length at a sample rate of 16kHz. This results in spectrograms with a time dimension four times the number of corresponding video frames. We use $T/4$ and T to denote the number of video frames and corresponding time resolution of the spectrograms respectively.

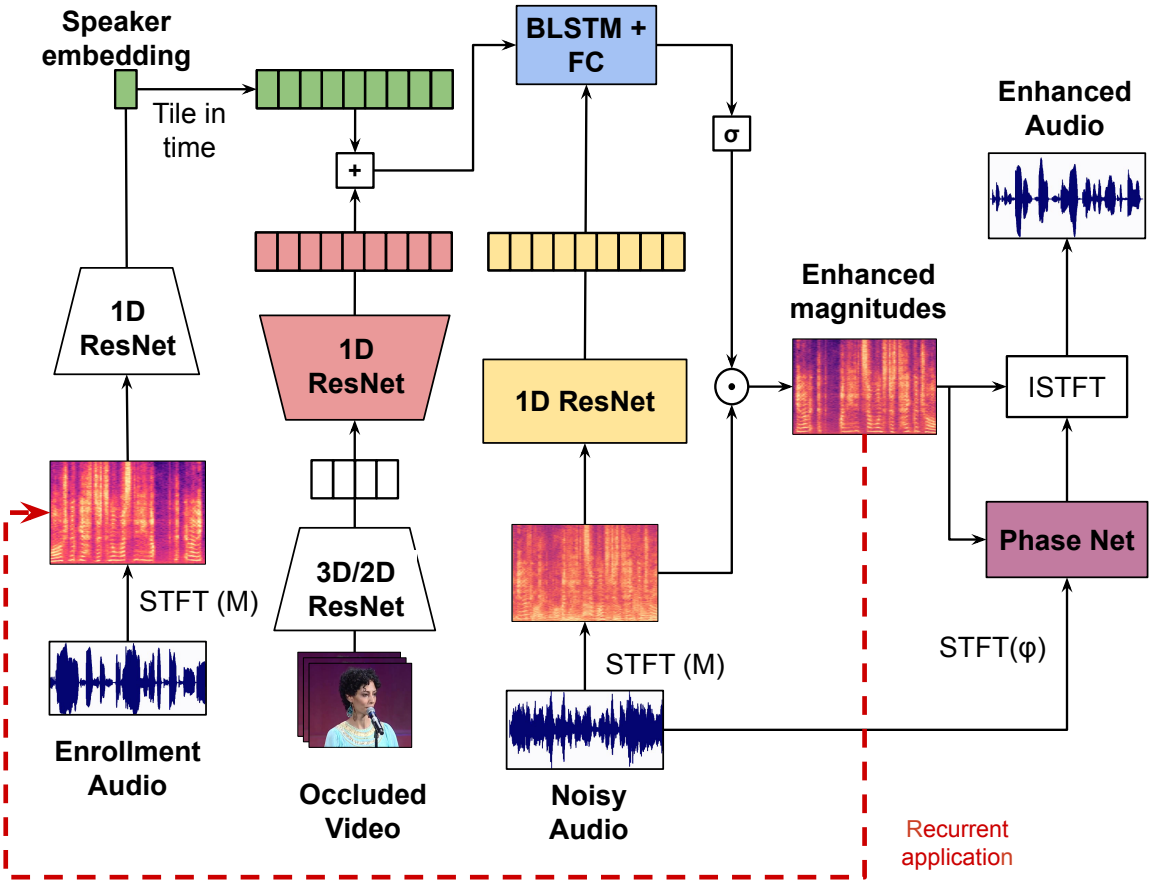


Figure 7.2: The architecture of the audio-visual speech enhancement network: There are 2 audio streams. The one processes the incoming noisy audio, while the other takes as input an enrollment audio sample and creates a speaker embedding that captures the speaker's voice characteristics. A visual stream extracts frame-wise representations from the input video. The visual, speaker and audio embeddings are combined and fed into the BLSTM which outputs a multiplicative mask that filters the noisy spectrograms. When no enrollment audio is provided, the enhanced magnitudes (created by a video-only pass) can be used as the input to the speaker embedding network.

Speaker embedding network. For embedding a reference audio clip into a compact speaker representation, we use the method of Xie *et al.* [411]. To reduce the number of computations, we replace all 2D spatial convolutions with 1D temporal ones which regard the frequency bins as channels and pre-train the modified architecture on the VoxCeleb2 [85] dataset following [411].

Modality combination. As shown in Figure 7.2, the noisy magnitude spectrograms are encoded into audio feature vectors through a shallow temporal ResNet. The video features are upsampled through a network containing two transposed convolution layers to match the temporal dimension of the spectrograms ($4T$). The speaker embedding extracted from the reference audio is tiled temporally and added to the resulting video embeddings to form the conditioning vec-

Table 7.1: Architecture details. a) The 1D ResNet that processes the video features. b) The 1D ResNet that processes the noisy audio spectrogram. c) The BLSTM and FC layers that perform the modality fusion. Notation: **K:** Kernel width; **S:** Stride – fractional strides denote transposed convolutions; **P:** Padding; **Out:** Temporal dimension of the layer's output. The non-transposed convolution layers are all depth-wise separable. Batch Normalization, ReLU activation and a shortcut connection are added after every convolutional layer.

Layer	# filters	K	S	P	Out
fc0	1536	1	1	1	$T/4$
conv1	1536	5	1	2	$T/4$
conv2	1536	5	1	2	$T/4$
conv3	1536	5	½	2	$T/2$
conv4	1536	5	1	2	$T/2$
conv5	1536	5	1	2	$T/2$
conv6	1536	5	1	2	$T/2$
conv7	1536	5	½	2	T
conv8	1536	5	1	2	T
conv9	1536	5	1	2	T
fc10	256	1	1	1	T

(a) Video Stream

Layer	# filters	K	S	P	Out
fc0	1536	1	1	1	T
conv1	1536	5	1	2	T
conv2	1536	5	1	2	T
conv3	1536	5	1	2	T
conv4	1536	5	1	2	T
conv5	1536	5	1	2	T
fc6	256	1	1	1	T

(b) Noisy Audio Stream

Layer	# filters	Out
BLSTM	400	T
fc1	600	T
fc2	600	T
fc_mask	F	T

(c) AV Fusion

tor used for the enhancement. This vector is then fed along with the noisy audio embedding into a one-layer bidirectional LSTM, followed by two fully connected layers. The output has spectrogram dimensions and is passed through a sigmoid activation to produce the enhancement mask.

Phase sub-network. In order to adjust the noisy phases to the enhanced magnitudes, we use the phase network of [3] without any changes.

Self-enrollment. For self-enrollment, the magnitude network is run twice: on the first pass, no speaker embedding is added to the visual one. The magnitudes that are output then serve as input to the speaker embedding network, as indicated by the red feedback arrow, and the network is run for a second time, with speaker embeddings this time.

We minimize the learning objective [3]:

$$\mathcal{L} = ||\hat{M} - M^*||_1 - \frac{1}{TF}\sum_{t,f} M^*_{tf} < \hat{\Phi}_{tf}, \Phi^*_{tf} >$$

where $\hat{M}$, $\hat{\Phi}$ and M^*, Φ^* are the predicted and ground truth magnitude and phase spectrograms

respectively, and T and F their time and frequency resolutions.

7.3 Experimental Setup

Datasets. The network is trained on the MV-LRS [90], LRS2 [2], and LRS3 [5] datasets, and tested on LRS3. MV-LRS and LRS2 contain material from British television broadcasts, while LRS3 was created from videos of TED talks. The speakers appearing in LRS3 are to the best of our knowledge not seen in either of the other two datasets. The datasets share the same format and pipeline including the face detection step, therefore no pre-processing is required in order to utilise them together for training. We remove from the LRS3 training set the few speakers that also appear in the test set, so that there is no overlap of identities between the two. Hence, the test set contains only speakers unseen and unheard during training and is suitable for a speaker-agnostic evaluation of our methods. Moreover, since the test set of LRS3 contains relatively short sentences, for testing we extract some longer sub-sequences from the original material used to make the LRS3 test set. We only use samples from speakers that appear in at least 2 different videos (TED talks), to enable enrollment with audio recorded in a different setting than the target one.

Synthetic data. We generate synthetic examples similarly to other works [3, 120, 393] by first sampling one reference audio-visual utterance from the training dataset and then mixing its audio with interfering audio signals. We consider two scenarios: 2 speakers and 3 speakers, where one and two interfering voices are added to the target signal respectively.

Enrollment. During training we do not know the identities of the speakers. Therefore, we obtain the enrollment signal from the same video but a different, non-overlapping time segment. This effectively reduces the amount of data we can use as we need to discard shorter videos (e.g. if we use 3 seconds, we can only use videos at least 6 seconds long). We use this method for training on datasets where the speaker identities are not known.

During evaluation we experiment with two enrollment methods: (i) *pre-enrollment* – we sample an enrollment segment from a video of the same speaker that is different from the

(a) Training samples **(b)** Test samples

Figure 7.3: Example frames of occluded videos used during training end evaluation.

one used to create the target sample (we do have identity labels for the test set); (ii) *self-enrollment* – we obtain the enrollment audio with a pass through our network that does not use a speaker embedding, as explained in Section 7.2.

Occlusions. For training, we artificially add occlusions to the video frames in the form of random patches as shown in Figure 7.3a. We randomly occlude sub-sequences of 15 to 25 contiguous frames, maintaining the clear-to-occluded frames ratio at 1:3. This is more realistic than simply zeroing out the incoming visual frames, as occluded video frames still produce valid feature vectors. For evaluation however, instead of random patches, we place jittering emojis on the videos as shown in Figure 7.3b. This type of visual noise has not been seen during training. The emojis are used to occlude the video from the start and the end, while the middle of the utterance is kept clear.

Training. The spatio-temporal visual front-end is pre-trained on a word-level lip reading task [360]. We then freeze the front-end and pre-compute the visual features. The features are extracted on a version of the videos where we have added random occlusions.

Training is conducted in four phases. We first pre-train the magnitude subnetwork only with speaker embedding inputs. For this we first use mixtures of two and then three speakers. Second, the visual modality is added and the magnitude network is trained on the saved visual features for the three simultaneous speakers scenario; third the magnitude network is frozen and the phase network is trained; finally the whole network is trained end-to-end.

7.4 Experiments

7.4.1 Evaluation protocol

To evaluate the performance of our model we use the Signal to Distortion Ration (SDR) [128], a common metric expressing the ratio between the energy of the target signal and of the errors contained in the enhanced output. Furthermore to assess the intelligibility of the output, we use the Google Cloud ASR system – we compute the Word Error Rate (WER) between the prediction of the ASR system on the enhanced audio and the ground truth transcriptions of the utterances contained in the segments used for evaluation. We evaluate on fixed length video segments of 8 seconds (200 frames).

7.4.2 Baseline models

We compare our proposed approach to the following baselines and ablations, which we train and evaluate both with and without visual occlusions.

PIT. We implement a blind source separation model that uses only the noisy audio input stream of Fig. 7.2 and is trained with a permutation invariant loss following [421]. This model is tailored to a predefined number of speakers.

V-Conv. This is the convolutional, visually conditioned baseline of Afouras *et al.* [3]. The model uses a series of 1D convolutional blocks for fusing the audio and video modalities instead of a BLSTM. Moreover, the video features are not upsampled by the video stream as in our proposed model, but the audio-visual fusion is performed at the temporal resolution of the video frames. The 1D convolutional stack then upsamples the fused input to the dimensions of the spectrograms.

V-BLSTM. This model is similar to our proposed architecture but conditions only on video features.

Table 7.2: Evaluation of speech enhancement performance on samples from the LRS3 dataset, for 2 and 3 simultaneous speakers. All the samples are 8 seconds long, of which 6 seconds are occluded (3 from either side) when occlusions are used. **Tr. Occ:** (Train Occlusions) Denotes that the model has been trained using artificial occlusions; **T. Occ:** (Test Occlusions) Denotes evaluation with occlusions; **pre:** Pre-enrollment: the enrollment audio is obtained from a different video of the target speaker; **self:** Self-enrollment; **SDR:** Signal to Distortion Ratio (higher is better); **WER:** Word Error Rate from off-the-shelf ASR system (lower is better).

	Tr. Occ.	T. Occ.	Enr.	SDR (dB)		WER (%)	
# Spk. / Method				2	3	2	3
GT signal	-	-	-	-	-	20.0	
Mixed signal	-	-	-	0.0	-3.7	82.6	93.2
PIT [421]	-	-	-	10.7	6.4	38.6	60.2
VoiceFilter [393]	-	-	pre	11.7	5.7	31.6	56.0
No occlusion during evaluation							
V-Conv [3]	✗	✗	-	12.7	9.1	24.9	33.7
V-Conv [3]	✓	✗	-	12.9	9.3	25.0	35.1
V-BLSTM	✗	✗	-	12.9	9.7	24.3	33.5
V-BLSTM	✓	✗	-	13.0	9.5	25.3	35.7
VS	✓	✗	pre	12.8	9.2	26.5	38.3
VS	✓	✗	self	12.8	9.3	26.6	40.3
80% occlusion during evaluation							
V-Conv [3]	✗	✓	-	0.8	-3.0	63.0	78.0
V-Conv [3]	✓	✓	-	2.7	-1.6	54.4	74.1
V-BLSTM	✗	✓	-	5.8	2.4	49.3	67.1
V-BLSTM	✓	✓	-	11.6	6.3	31.3	54.3
VS	✓	✓	pre	12.1	**7.3**	30.7	**50.0**
VS	✓	✓	self	**12.2**	7.2	**30.3**	50.3

VoiceFilter. This model conditions only on speaker embeddings and is equivalent to the sub-network used during the first stage of the training process. It is essentially a *VoiceFilter* [393] implementation with a slightly modified architecture, trained on our dataset.

VS. Our proposed architecture, which receives both video and speaker embedding inputs. As discussed in Section 7.2, we investigate two variants, *VS-pre* and *VS-self*, that correspond to the different enrollment methods employed during evaluation.

7.4.3 Results

We summarize the results of our experiments in Table 7.2. When no occlusions are used, the *V-BLSTM* model only slightly outperforms *V-Conv*. When 80% of the visual input frames are occluded, the models that haven't been trained with occlusions fail. Even when we include occlusions during the training of *V-Conv*, it cannot deal with the missing visual information, since its receptive field is limited (about 1 second to either side). On the contrary, *V-BLSTM* uses its memory and learns to deal with local occlusions. Overall however, the proposed *VS* models that explicitly condition on the expected speaker embedding give the best performance.

The results furthermore verify that both the *VoiceFilter* and *VS-pre* model perform well when evaluated using enrollment signals from sources different from the target one, even though they have never been trained in this setting.

The effect of occluding different amounts of the visual input is studied in Fig. 7.4. The *V-BLSTM* model that has not been trained on occlusions does not perform well when even small parts of the video input are occluded. When trained with occlusions, *V-BLSTM* becomes much more resilient, however it still gives bad results for high occlusion percentages and completely fails when the entire video is occluded.

The *VS-pre* model outperforms *V-BLSTM* when half or more of the input is occluded and gives similar results for cleaner inputs.

Self-enrollment. For very high occlusion levels, the initial enhanced estimation of *VS-self* is bad and evidently unable to capture the target voice characteristics. However, if more than 20% of the frames are clean, self-enrollment performs best. Therefore, apart from the higher occlusion levels, *VS* with self-enrollment provides an advantage compared to *V-BLSTM*.

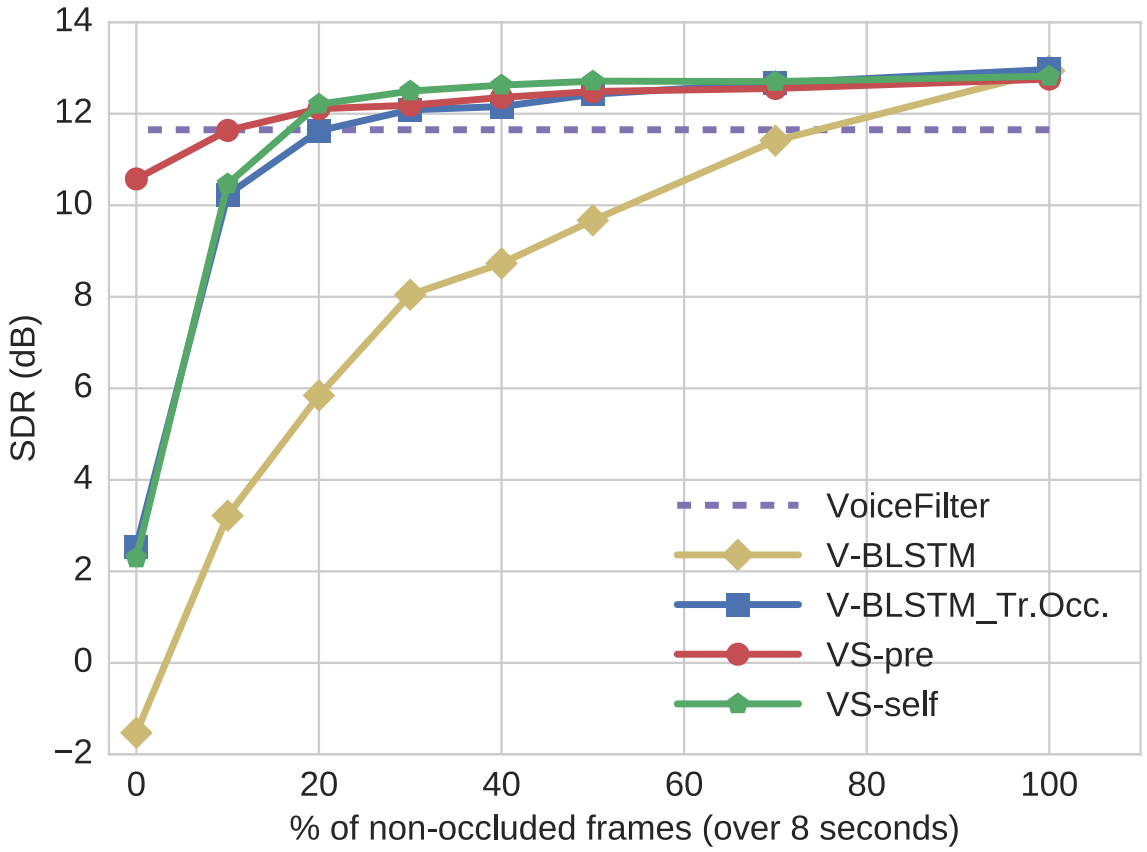
7.5 Conclusion

In this paper, we proposed a deep audio-visual speech enhancement network that is able to separate a speaker's voice by conditioning on both the speaker's lip movements and/or

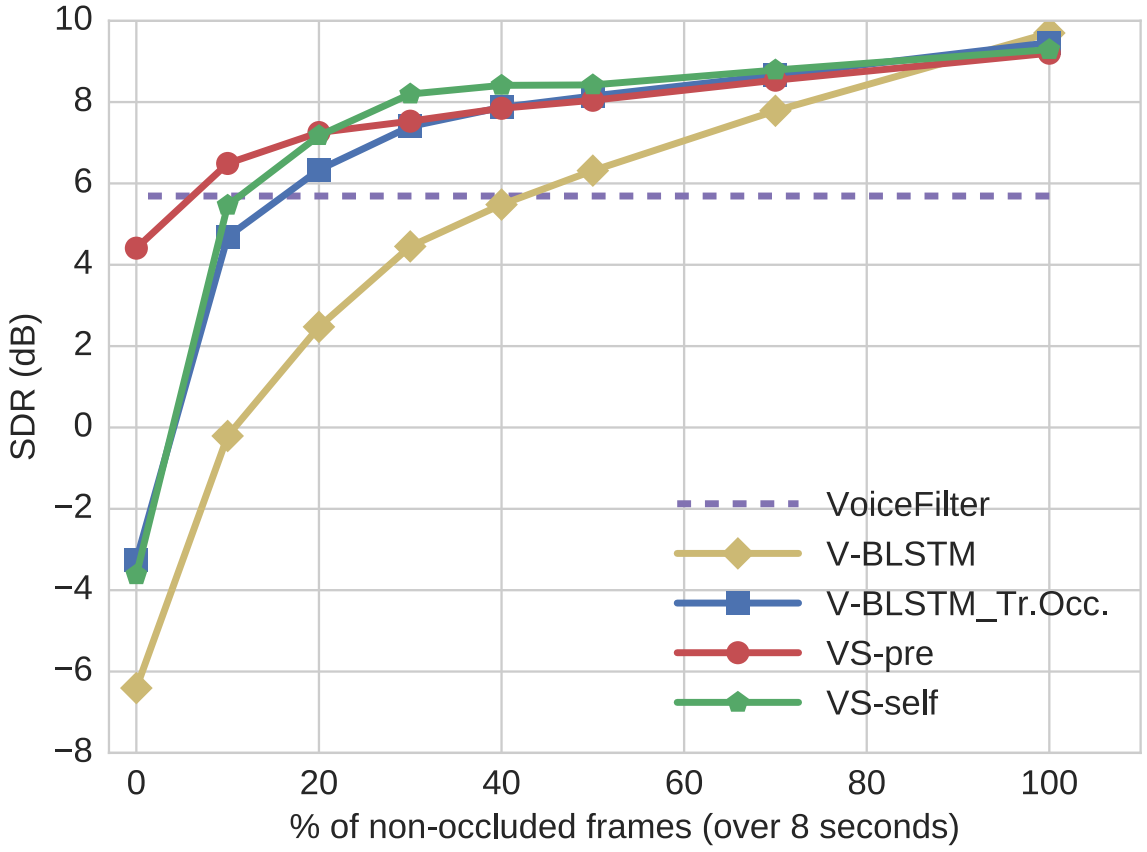
a representation of their voice. The network is robust to partial occlusions, and the voice representation can be self-enrolled from the unoccluded part of the input when it is not possible to obtain segments for pre-enrollment. The methods are evaluated on the challenging LRS3 dataset, and demonstrate performance that exceeds that of previous state-of-the-art [3] when the video input is partially occluded.

Statement of authorship

A statement of authorship for this paper is provided in Appendix A.



(a) 2 Speakers



(b) 3 Speakers

Figure 7.4: Enhancement performance when occluding varying amounts of the visual input for the 2 Speakers and 3 Speakers scenarios. Model notations are explained in the caption of Table 7.2.

Part III

Audio-visual object localization and detection

8 | Self-Supervised Learning of Audio-Visual Objects from Video

Triantafyllos Afouras[1]* Andrew Owens[2] Joon Son Chung[1,3] Andrew Zisserman[1]

[1]Visual Geometry Group, Oxford [2]University of Michigan [3]Naver Corporation

Abstract

Our objective is to transform a video into a set of discrete audio-visual objects using self-supervised learning. To this end, we introduce a model that uses attention to localize and group sound sources, and optical flow to aggregate information over time. We demonstrate the effectiveness of the audio-visual object embeddings that our model learns by using them for four downstream speech-oriented tasks: (a) multi-speaker sound source separation, (b) localizing and tracking speakers, (c) correcting misaligned audio-visual data, and (d) active speaker detection. Using our representation, these tasks can be solved entirely by training on unlabeled video, without the aid of object detectors. We also demonstrate the generality of our method by applying it to non-human speakers, including cartoons and puppets. Our model significantly outperforms other self-supervised approaches, and obtains performance competitive with methods that use supervised face detection.

Published in the proceedings of European Conference on Computer Vision (ECCV), 2020, pp. 208-224.

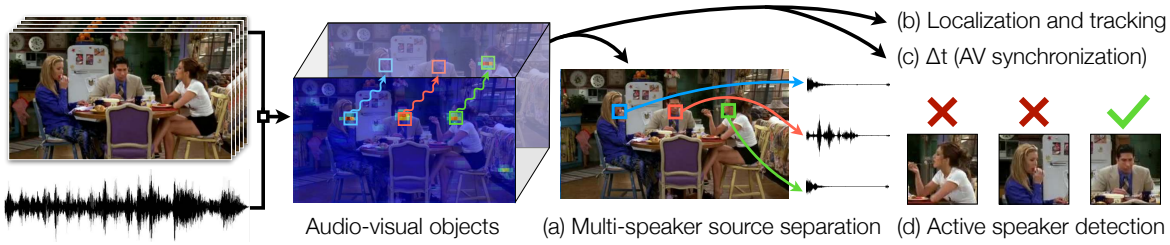


Figure 8.1: We learn through self-supervision to represent a video as a set of discrete *audio-visual objects*. Our model groups a scene into object instances and represents each one with a feature embedding. We use these embeddings for speech-oriented tasks that typically require object detectors: (a) multi-speaker source separation, (b) speaker localization, (c) synchronizing misaligned audio and video, and (d) active speaker detection. Using our representation, these tasks can be solved without any labeled data, and on domains where off-the-shelf detectors are not available, such as cartoons and puppets. Please see our webpage for videos: http://www.robots.ox.ac.uk/∼vgg/research/avobjects.

8.1 Introduction

When humans organize the visual world into objects, hearing provides cues that affect the perceptual grouping process. We group different image regions together not only because they look alike, or move together, but also because grouping them together helps us explain the *causes* of co-occurring audio signals.

In this paper, our objective is to replicate this organizational capability, by designing a model that can ingest raw video and transform it into a set of *discrete audio-visual objects*. The network is trained using only self-supervised learning from audio-visual cues. We demonstrate this capability on videos containing talking heads.

This organizational task must overcome a number of challenges if it is to be applicable to raw videos in the wild: (i) there are potentially many visually similar sound generating objects in the scene (multiple heads in our case), and the model must correctly attribute the sound to the actual sound source; (ii) these objects may move over time; and (iii) there can be multiple other objects in the scene (clutter) as well.

To address these challenges, we build upon recent works on self-supervised audio-visual localization. These include video methods that find motions temporally synchronized with audio onsets [88, 221, 293], and single-frame methods [22, 171, 296, 341] that find regions that are likely to co-occur with the audio. However, their output is a typically a "heat map" that indicates whether a given pixel is likely (or unlikely) to be attributed to the audio; they do not group a scene into *discrete objects*; and, if only using semantic correspondence, then they cannot distinguish which, of several, object instances is making a sound.

Our first contribution is to propose a network that addresses all three of these challenges; it is able to use synchronization cues to detect sound sources, group them into distinct instances, and track them over time as they move. Our second contribution is to demonstrate that object embeddings obtained from this network facilitate a number of audio-visual downstream tasks that have previously required hand-engineered supervised pipelines.

As illustrated in Figure 8.1, we demonstrate that the embeddings enable: (a) multi-speaker sound source separation [3, 120]; (b) detecting and tracking talking heads; (c) aligning misaligned recordings [85, 92]; and (d) detecting active speakers, i.e. identifying which speaker is talking [88, 326]. In each case, we significantly outperform other self-supervised localization methods, and obtain comparable (and in some cases better) performance to prior methods that are trained using stronger supervision, despite the fact that we learn to perform them entirely from a raw audio-visual signal.

The trained model, which we call the Look Who's Talking Network (LWTNet), is essentially "plug and play" in that, once trained on unlabeled data (without preprocessing), it can be applied directly to other video material. It can easily be fine-tuned for other audio-visual domains: we demonstrate this functionality on active speaker detection for non-human speakers, such as animated characters in *The Simpsons* and puppets in *Sesame Street*. This demonstrates the generality of the model and learning framework, since this is a domain where off-the-shelf supervised methods, such as methods that use face detectors, cannot transfer without additional labeling.

8.2 Related work

Sound source localization. Our task is closely related to the *sound source localization* problem, i.e. finding the location in a video that is the source of a sound. Early work performed localization [33, 130, 180, 207] and segmentation [194] by doing inference on simple probabilistic models, such as methods based on canonical correlation analysis.

Recent efforts learn audio and video representations using self-supervised learning [88, 221, 293] with *synchronization* as the proxy task: the network has to predict whether video and audio are temporally aligned (or synthetically shifted). Owens and Efros [293] show via heat-map visualizations that their network often attends to sound sources, but do not quantitatively evaluate their model. Recent work [206] added an attention mechanism to this model. Other work has detected sound-making objects using *correspondence* cues [22, 171, 189, 191, 296, 314, 341, 376], e.g. by training a model to predict whether audio and a single video frame come from the same (or different) videos. Since these models do not use motion and are

trained only to find the correspondence between object appearance and sound, they would not be able to identify which of several objects of the same category is the actual source of a sound. In contrast, our goal is to obtain discrete audio-visual objects from a scene, even when they bellong to the same category (e.g. multiple talking heads). In a related line of work, [138] distill visual object detectors into an audio model using stereo sound, while [141] use spatial information in a scene to convert mono sound to stereo.

Active speaker detection (ASD). Early work on active speaker detection trained simple classifiers on hand-crafted feature sets [97]. Later, Chung and Zisserman [88] used synchronization cues to solve the active speaker detection problem. They used a hand-engineered face detection and tracking pipeline to select candidate speakers, and ran their model only on cropped faces. In contrast, our model learns to do ASD entirely from unlabeled data. Chung *et al.*[84] extended the pipeline by enrolling speaker models from visible speaking segments. Recently, Roth *et al.* [326] proposed an active speaker detection dataset and evaluated a variety of supervised methods for it.

Source separation. In recent years, researchers have proposed a variety of methods for separating the voices of multiple speakers in a scene [3, 120, 133, 293]. These methods either only handle a single on-screen speaker [293] or use hand-engineered, supervised face detection pipelines. Afouras *et al.* [3] and Ephrat *et al.* [120], for example, detect and track faces and extract visual representations using off-the-shelf packages. In contrast, we use our model to separate multiple speakers entirely via self-supervision.

Other recent work has explored separating the sounds of musical instruments and other sound-making objects. Gao *et al.* [140, 142] use semantic object detectors trained on instrument categories, while [327, 441] do not explicitly group a scene into objects and instead either pool the visual features or produce a per-pixel map that associates each pixel with a separated audio source. Recently, [440] added motion information from optical flow. We, too, use flow in our model, but instead of using it as a *cue* for motion, we use it to integrate information from moving objects over time [135, 308] in order to track them. In concurrent work [191] propose a model that groups and separates sound sources.

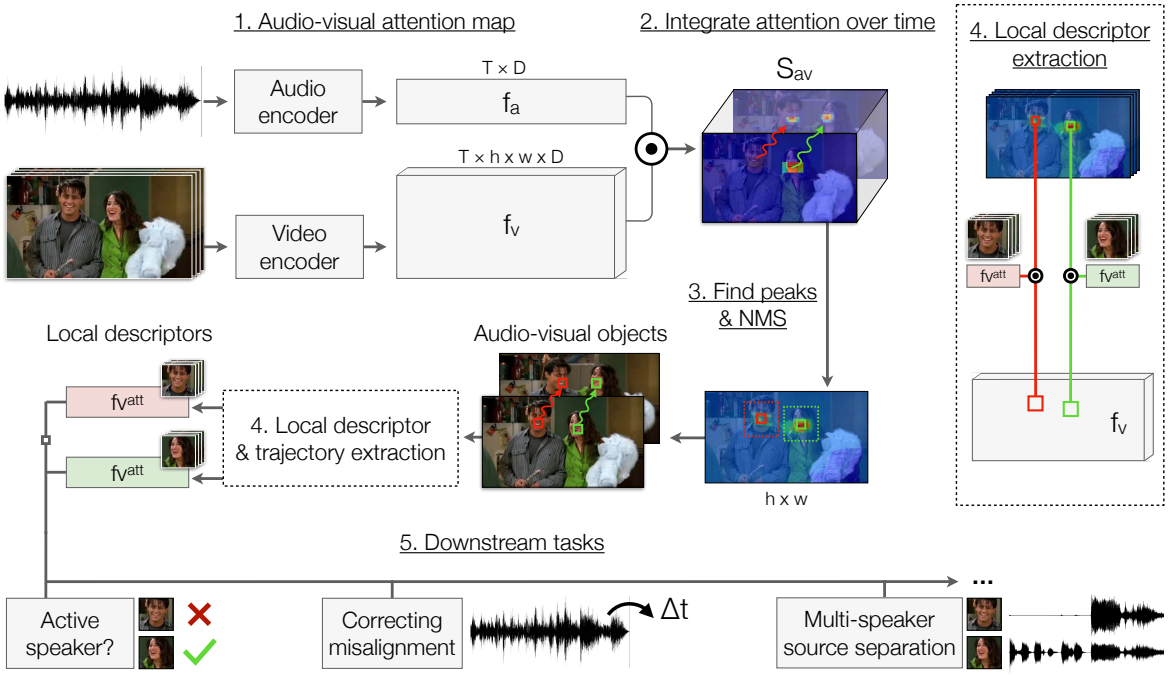


Figure 8.2: The Look Who's Talking Network (LWTNet): (1) Computes an audio-visual attention map S_{av} by solving a synchronization task, (2) accumulates attention over time, (3) selects *audio-visual objects* by computing the N highest peaks with non-maximum suppression (NMS) from the accumulated attention map, each corresponding to a trajectory of the pixel over time; (4) for every audio-visual object, it extracts embedding vectors from a spatial window ρ, using the local attention map S_{av} to select visual features, and (5) provides the audio-visual objects as inputs to downstream tasks.

Representation learning. In recent years, researchers have proposed a variety of self-supervised learning methods for learning representations from images [72, 110, 174, 178, 267, 292, 377, 398], videos [168, 169] and multimodal data [20, 221, 280, 294, 296]. Often the representation learned by these methods is a feature set (e.g., CNN weights) that can be adapted to downstream tasks by fine-tuning. By contrast, we learn an additional *attention mechanism* that can be used to group discrete objects of interest for downstream speech tasks.

8.3 From unlabeled video to audio-visual objects

Given a video, the function of our model is to detect and track (possibly several) audio-visual objects, and extract embeddings for each of them. We represent an audio-visual object as the trajectory of a potential sound source through space and time, which in the domain that we experiment on is often the track of a "talking head". Having obtained these trajectories, we

use them to extract embeddings that can be then used for downstream tasks.

In more detail, our model uses a bottom-up grouping procedure to propose discrete audio-visual objects from raw video. It first estimates local (per-pixel and per-frame) synchronization evidence, using a network design that is more fine-grained in space and time than prior models. It then aggregates this evidence over time via optical flow, thereby allowing the model to obtain robustness to motions, and groups the aggregated attention into sound sources by detecting local maxima. The model represents each object as a separate embedding, temporal track, and attention map that can be adjusted in downstream tasks.

We will now give an overview of the model, which is shown in Figure 8.2, followed by the learning framework which uses self-supervision based on synchronization. For architecture details, please refer to the the arXiv version.

8.3.1 Estimating audio-visual attention

Before we group a scene into sound sources, we estimate a per-pixel attention map that picks out the regions of a video whose motions have a high degree of synchronization with the audio. We propose an attention mechanism that provides highly localized spatio-temporal attention, and which is sensitive to speaker motion. As in [22, 171], we estimate audio-visual attention via a multimodal embedding (Figure 8.2, step 1). We learn vector embeddings for each audio clip and embedding vectors for each pixel, such that if a pixel's vector has a high dot product with that of the audio, then it is likely to belong to that sound source. For this, we use a two-stream architecture similar to those in other sound-source localization work [22, 171, 341], with a network backbone similar to [84].

Video encoder. Our video feature encoder is a spatio-temporal VGG-M [66] with a 3D convolutional layer first, followed by a stack of 2D convolutions. Given a $T \times H \times W \times 3$ input RGB video, it extracts a video embedding map $f_v(x,y,t)$ with dimensions $T \times h \times w \times D$.

Audio encoder. The audio encoder is a VGG-M network operating on log-mel spectrograms, treated as single-channel images. Given an audio segment, it extracts a D-dimensional

embedding $f_a(t)$ for every corresponding video frame t.

Computing fine-grained attention maps. For each space-time pixel, we ask: how correlated is it with the events in the audio? To estimate this, we measure the similarity between the audio and visual features at every spatial location. For every space-time feature vector $f_v(x,y,t)$, we compute the cosine similarity with the audio feature vector $f_a(t)$:

$$S_{av}(x,y,t) = f_v(x,y,t) \cdot f_a(t), \tag{8.1}$$

where we first l_2 normalize both features. We refer to the result, $S_{av}(x,y,t)$, as the *audio-visual attention map*.

8.3.2 Extracting audio-visual objects

Given the audio-visual evidence, we parse a video into object representations.

Integrating evidence over time. Audio-visual objects may only intermittently make sounds. Therefore, we need to integrate sparse attention evidence over time. We also need to group and track sound sources *between* frames, while accounting for camera and object motion. To make our model more robust to these motions, we aggregate information over time using optical flow (Figure 8.2, step 2). We extract dense optical flow for every frame, chain the flow values together to obtain long-range tracks, and average the attention scores over these tracks. Specifically, if $\mathcal{T}(x,y,t)$ is the tracked location of pixel (x,y) from frame 1 to the later frame t, we compute the score:

$$S^{tr}_{av}(x,y) = \frac{1}{T}\sum_{t=1}^{T} S_{av}(\mathcal{T}(x,y,t),t), \tag{8.2}$$

where we perform the sampling using bilinear interpolation. The result is a 2D map containing a score for the future trajectory of every pixel of the initial frame through time. Note that any tracking method can be used in place of optical flow (e.g. with explicit occlusion handling); we use optical flow for simplicity.

Grouping a scene into instances. To obtain discrete audio-visual objects, we detect spatial local maxima (peaks) on the temporally aggregated synchronization maps, and apply

Figure 8.3: Intermediate representations from our model. We show the per-frame attention maps $S_{av}(t)$, the aggregated attention map S^{tr}_{av} and the two highest scoring extracted audio-visual objects. We show the audio-visual objects for a single frame, with a square of constant width.

non-maximum suppression (NMS). More specifically, we find peaks in the time-averaged synchronization map, $S^{tr}_{av}(x,y)$, and sort them in decreasing order; we then choose the peaks greedily, each time suppressing the ones that are within a $\rho \times \rho$ box. The selected peaks can be now viewed as distinct audio-visual objects. Examples of the intermediate representations extracted at the steps described so far are shown in Figure 8.3.

Extracting object embeddings. Now that the sound sources have been grouped into distinct audio-visual objects, we can extract feature embeddings for each one of them that we can use in downstream tasks. Before extracting these features, we locate the position of the sound source in each frame. A simple strategy for this would be to follow the object's optical flow track throughout the video. However, these tracks are imprecise and may not correspond precisely to the location of the sound source. Therefore, we "snap" to the track location to the nearest peak in the attention map. More specifically, in frame t, we search in an area of $\rho \times \rho$ centered on the tracked location $\mathcal{T}(x,y,t)$, and select the pixel location with largest attention value. Then, having tracked the sound source in each frame, we select the corresponding spatial feature vector from the visual feature map f_v (Figure 8.2, step 4). These per-frame embedding features, $f^{att}_v(t)$, can then be used to solve downstream tasks (Section 8.4). One can equivalently view this procedure as an audio-visual attention mechanism that operates on f_v.

8.3.3 Learning the attention map

Training our model amounts to learning the attention map S_{av} on which the audio-visual objects are subsequently extracted. We obtain this map by solving a self-supervised audio-visual synchronization task [88, 221, 293]: we encourage the embedding at each pixel to be correlated with the true audio and uncorrelated with shifted versions of it. We estimate the synchronization evidence for each frame by aggregating the per-pixel synchronization scores. Following common practice in multiple instance learning [22], we measure the per-frame evidence by the maximum spatial response:

$$S_{av}^{att}(t) = \max_{x,y} S_{av}(x,y,t). \tag{8.3}$$

We maximize the similarity between a video frame's true audio track while minimizing that of N shifted (i.e. misaligned) versions of the audio. Given visual features f_v and true audio a_i, we sample N other audio segments from the same video clip: $a_1, a_2, ..., a_N$, and minimize the contrastive loss [92, 292]:

$$\mathcal{L} = -\log \frac{\exp(S_{av}^{att}(v,a_i))}{\exp(S_{av}^{att}(v,a_i)) + \sum_{j=1}^{N} \exp(S_{av}^{att}(v,a_j))}. \tag{8.4}$$

For the negative examples, we select all audio features (except for the true example) in a temporal window centered on the video frame.

In addition to the synchronization task, we also consider the *correspondence* task of Arandjelović and Zisserman [22], which chooses negatives audio samples from random video clips. Since this problem can be solved with even a single frame, it results in a model that is less sensitive to motion.

8.4 Applications of audio-visual object embeddings

We use our learned audio-visual objects for a variety of applications.

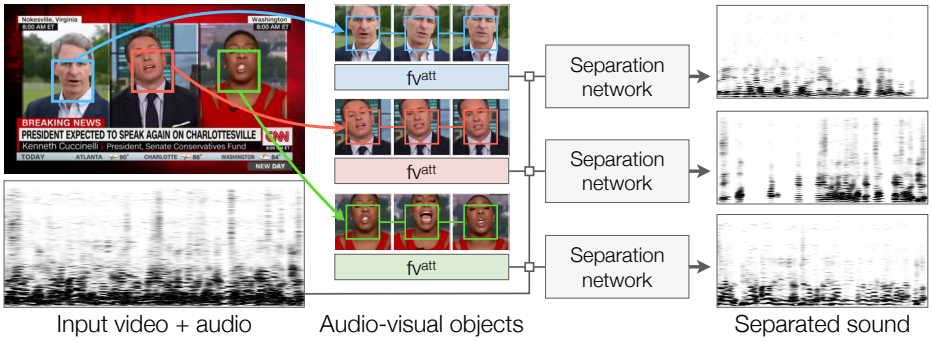


Figure 8.4: Multi-speaker separation. We isolate the sound of each speaker's voice by combining our audio-visual objects with a network similar to [3]. Given a spectrogram of a noisy sound mixture, the network isolates the voice of each speaker, using the visual features provided by their audio-visual object.

8.4.1 Audio-visual object detection and tracking

We can use our model for spatially localizing speakers. To do this, we use the tracked location of an audio-visual object in each frame.

8.4.2 Active speaker detection

For every frame in our video, our model can locate potential speakers and decide whether or not they are speaking. In our setting, this can be viewed as deciding whether an audio-visual object has strong evidence of synchronization in a given frame. For every tracked audio-visual object, we extract the visual features $f_v^{att}(t)$ (Sec. 8.3.2) for each frame t. We then obtain a score that indicates how strong the audio-visual correlation for frame t is, by computing the dot product: $f_v^{att}(t) \cdot f_a(t)$. Following previous work [88], we threshold the result to make a binary decision (active speaker or not).

8.4.3 Multi-speaker source separation

Our audio-visual objects can also be used for separating the voices of speakers in a video. We consider the *multi-speaker* separation problem [3, 120]: given a video with multiple people speaking on-screen (*e.g.*, a television debate show), we isolate the sound of each speaker's voice from the audio stream. We note that this problem is distinct from on/off-screen audio separation [293], which requires only a single speaker to be on-screen.

We train an additional network that, given a waveform containing an audio mixture and an audio-visual object, isolates the speaker's voice (Figure 8.4, full details in the the arXiv version of the paper). We use an architecture that is similar to [3], but conditions on our self-supervised representations instead of detections from a face detector. More specifically, the method of [3] runs a face detection and tracking system on a video, computes CNN features on each crop, and then feeds those to a source separation network. We, instead, simply provide the same separation network with the embedding features $f_v^{att}(t)$.

8.4.4 Correcting audio-visual misalignment

We can also use our model to correct misaligned audio-visual data — a problem that often occurs in the recording and television broadcast process. We follow the problem formulation proposed by Chung and Zisserman [88]. While this is a problem that is typically solved using supervised face detection [88, 92], we instead tackle it with our learned model. During inference, we are given a video with unsynchronized audio and video tracks, and we shift the audio to discover the offset $\hat{\Delta}t$ that maximizes the audio-visual evidence:

$$\hat{\Delta}t = \underset{\Delta t}{\operatorname{argmax}} \frac{1}{T} \sum_{t=1}^{T} S_{\Delta t}^{att}(t), \tag{8.5}$$

where $S_{\Delta t}^{att}(t)$ is the synchronization score of frame t after shifting the audio by Δt. This can be estimated efficiently by recomputing the dot products in Eq. 8.1.

In addition to treating this alignment procedure as a stand-alone application, we also use it as a preprocessing step for our other applications (a common practice in other speech analysis work [3]). When given a test video, we first compute the optimal offset $\hat{\Delta}t$, and use it to shift the audio accordingly. We then recompute $S_{av}(t)$ from the synchronized embeddings.

8.5 Experiments

8.5.1 Datasets

Human speech. We evaluate our model on the Lip Reading Sentences (LRS2 and LRS3) datasets and the Columbia active speaker dataset. LRS2 [2] and LRS3 [5] are audio-visual speech datasets containing 224 and 475 hours of videos respectively, along with ground truth face tracks of the speakers. The Columbia dataset [62] contains footage from an 86-minute panel discussion, where multiple individuals take turns in speaking, and contains approximate bounding boxes and active speaker labels, *i.e.* whether a visible face is speaking at a given point in time. All datasets provide (pseudo-)ground truth bounding boxes obtained via face detection, which we use for evaluation. We resample all videos to a resolution of $H \times W = 270 \times 480$ pixels before feeding them to our model, which outputs $h \times w = 18 \times 31$ attention maps. We train all models on LRS2, and use LRS3 and Columbia only for evaluation.

Non-human speakers To evaluate our method on non-human speakers, we collected television footage from *The Simpsons* and *Sesame Street* shows (Table 8.5). For testing, we obtained ASD and speaker localization labels, using the VIA tool [113]: we asked human annotators to label frames that they believed to contain an active speaker and to localize them. For every dataset, we create a *single-head* and a *multi-head* set, where clips are constrained to contain a single active speaker or multiple heads (talking or not) respectively. We provide dataset statistics in Table 8.5 and more details in the the arXiv version of the paper.

8.5.2 Training details

Audio-visual object detection training. To make training easier, we follow [221] and use a simple learning curriculum. At the beginning of training, we sample negatives from random video clips, then switch to shifted audio tracks later in training. To speed up training, we also begin by taking the mean dot product (Eq. 8.3), and then switch to the maximum. We set ρ to 100 pixels.

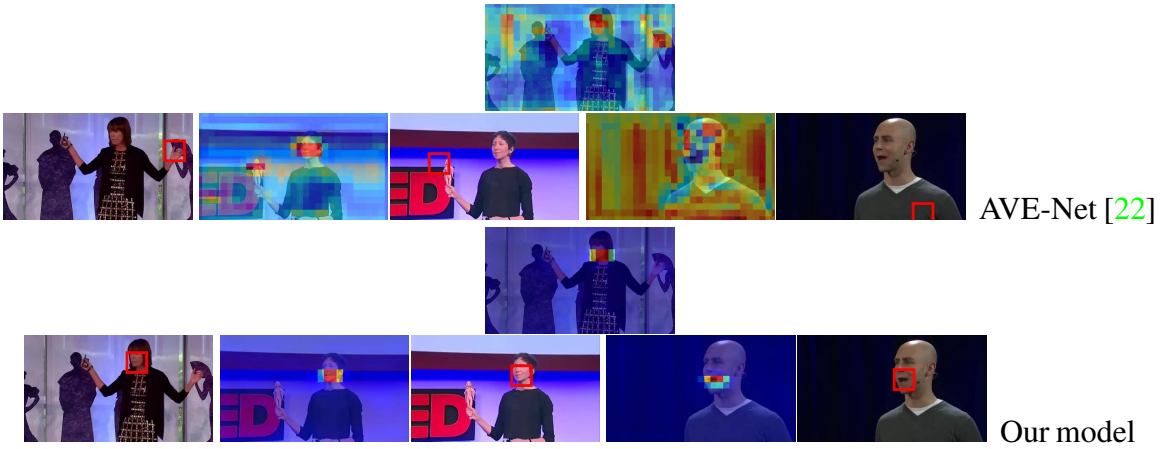


Figure 8.5: Talking head detection and tracking on LRS3 datasets. For each of the 4 examples, we show the audio-visual attention score on every spatial location for the depicted frame, and a bounding box centered on the largest value, indicating the speaker location. Please see our webpage for video results.

Figure 8.6: Handling motion: Talking head detection and tracking on continuous scenes from the validation set of LRS2. Despite the significant movement of the speakers and the camera, our method accurately tracks them.

Source separation training Training takes place in two steps: we first train our model to produce audio-visual objects by solving a synchronization problem. Then, we train the multi-speaker separation network on top of these learned representations. We follow previous work [3, 120] and use a mix-and-separate learning procedure. We create synthetic videos containing multiple talking speakers by 1) selecting two or three videos at random from the training set, depending on the experiment, 2) summing their waveforms together, and 3) vertically concatenating the video frames together. The model is then tasked with extracting a number of talking heads equal to the number of mixed videos and predicting an original corresponding waveform for each.

Non-human model training We fine-tune the best model from LRS2 separately on each of the two datasets with non-human speakers. The lip motion for non-human speakers, such as the motion of a puppet's mouth, is only loosely correlated with speech, suggesting that there is less of an advantage to obtaining our negative examples from temporally shifted audio.

Figure 8.7: Active speaker detection on the Columbia dataset, and an example from the *Friends* TV show. We show active speakers in **blue** and inactive speakers in **red**. The corresponding detection scores are noted above the boxes (the threshold has been subtracted so that positive scores indicate active speakers).

We therefore sample our negative audio examples from other video clips rather than from misaligned audio (Section 8.3.3) when computing attention maps.

8.5.3 Results

1. Talking head detection and tracking. We evaluate how well our model is able to localize speakers, i.e. talking heads (Table 8.1). First, we evaluate two simple baselines: the *random* one, which selects a random pixel in each frame and the *center* one, which always selects the center pixel. Next, we compared with two recent sound source localization methods: Owens and Efros [293] and AVE-Net [22]. Since these methods require input videos that are longer than most of the videos in the test set of LRS2, we only evaluate them on LRS3. We also perform several ablations of our model: To evaluate the benefit of integrating the audio-visual evidence over flow trajectories, we create a variation of our model called *No flow* that, instead, computes the attention S^{tr}_{av} by globally pooling over time throughout the video. Finally, we also consider a variation of this model that uses a larger NMS window ($\rho = 150$).

We found that our method obtains very high accuracy, and that it significantly outperforms all other methods. AVE-Net solves a correspondence task that doesn't require motion information, and uses a single video frame as input. Consequently, it does not take advantage of informative motion, such as moving lips. As can be seen in Figure 8.5, the localization maps produced by AVE-Net [22] are less precise, as it only loosely associates appearance of a person to speech, and won't consistently focus on the same region. Owens and Efros [293], by contrast, has a large temporal receptive field, which results in temporally imprecise predictions, causing very large errors when the subjects are moving. The *No flow* baseline fails to track the talking head

Figure 8.8: Active speaker detection for non-human speakers. We show the top 2 highest-scoring audio-visual objects in each scene, along with the aggregated attention map. Please see our webpage for video results.

Method	LRS2	LRS3
Random	2.8%	2.9%
Center	23.9%	25.9%
Owens & Efros [293]	-	24.8%
AVE-Net [22]	-	58.1%
No flow	98.4%	94.2%
No flow + large NMS	98.8%	97.2%
Full model	**99.6%**	**99.7%**

Table 8.1: Talking head detection and tracking accuracy. A detection is considered correct if it lies within the true bounding box.

well outside the NMS area, and its accuracy is consequently lower on LRS3. Enlarging the NMS window partially alleviates this issue, but the accuracy is still lower than that of our model. We note that the LRS2 test set contains very short clips (usually 1-2 seconds long) with predominantly static speakers, which explains why using flow does not provide an advantage. We show some challenging examples with significant speaker and camera motion in Figure 8.6. Please refer to the the arXiv version of the paper for further analysis of camera and speaker motion.

2. Active speaker detection. Next, we ask how well our model can determine *which* speaker is talking. Following previous work that uses supervised face detection [89, 342], we evaluate our method on the Columbia dataset [62]. For each video clip, we extract 5 audio-visual objects

Method	Speaker					
	Bell	Boll	Lieb	Long	Sick	Avg.
Chakravarty [62]	82.9	65.8	73.6	86.9	81.8	80.2
Shahid [342]	87.3	96.4	92.2	83.0	87.2	89.2
SyncNet [88]	93.7	83.4	86.8	97.7	86.1	89.5
Ours	92.6	82.4	88.7	94.4	95.9	**90.8**

Table 8.2: Active speaker detection accuracy on the Columbia dataset [62]. F1 Scores (%) for each speaker, and the overall average.

(an upper bound on the number of speakers), each of which has an ASD score indicating the likelihood that it is a sound source (Section 8.4.2). We then associate each ground truth bounding box with the audio-visual object whose trajectory follows it the closest. For comparison with existing work, we report the F1 measure (the standard for this dataset) per individual speaker as well as averaged over all speakers. For calculating the F1 we set the ASD threshold to the one that yields the Equal Error Rate (EER) for the pretext task on the LRS2 validation set. As shown in Table 8.2, our model outperforms all previously reported results on this dataset, even though (unlike other methods) it does not use labeled face bounding boxes for training.

3. Multi-speaker source separation. To evaluate our model on speaker separation, we follow the protocol of [3]. We create synthetic examples from the test set of LRS2, using only videos that are between $2-5$ seconds long, and evaluate performance using Signal-to-Distortion-Ratio (SDR) [128] and Perceptual Evaluation of Speech Quality (PESQ, varies between 0 and 4.5) [323] (higher is better for both). We also assess the intelligibility of the output by computing the Word Error Rate (WER, lower is better) between the transcriptions obtained with the Google Cloud speech recognition system. Following [5], we train and evaluate separate models for 2 and 3 speakers, though we note that if the number of speakers were unknown, it could be estimated using active speaker detection.

For comparison, we implement the model of Afouras *et al.* [3], and train it on the same data. For extracting visual features to serve as its input, we use a state-of-the-art audio-visual synchronization model [92], rather than the lip-reading features from Afouras *et al.* [6]. We refer to this model as *Conversation-Sync*. This model uses bounding boxes from a well-engineered face detection system, and thus represents an approximate upper limit on the

Table 8.3: Source separation on LRS2. #Spk indicates the number of speakers. The WER on the ground truth signal is 20.0%.

		SDR		PESQ		WER %	
Method \ # Spk.		2	3	2	3	2	3
Mixed input		-0.3	-3.4	1.7	1.5	91.0	97.2
Conv.-Sync [3]		11.3	7.5	3.0	2.5	30.3	43.5
Ours	Frozen	10.7	7.0	3.0	2.5	30.7	44.2
	Oracle-BB	10.8	7.1	2.9	2.5	30.9	44.9
	Small-NMS	10.6	6.8	3.0	2.5	31.2	44.7
	Full	10.8	7.2	3.0	2.6	30.4	42.0

Table 8.4: Audio-visual synchronization accuracy (%) evaluation for a given number of input frames.

	Input frames					
Method	5	7	9	11	13	15
SyncNet [88]	75.8	82.3	87.6	91.8	94.5	96.1
PM [92]	88.1	93.8	96.4	97.9	98.7	99.1
Ours	78.8	87.1	92.1	94.8	96.3	97.3

performance of our self-supervised model. Our main model for this experiment is trained end-to-end and uses $\rho = 150$. We also performed a number of ablations: a model that freezes the pretrained audio-visual features and a model with a smaller $\rho = 100$.

We observed (Table 8.3) that our self-supervised model obtains results close to those of [3], which is based on supervised face detection. We also asked how much error is introduced by lack of face detection. In this direction we extract the local visual descriptors using tracks obtained with face detectors instead of our audio-visual object tracks. This model, *Oracle-BB*, obtains results similar to ours, suggesting that the quality of our face localization is high.

4. Correcting misaligned visual and audio data. We use the same metric as [92] to evaluate on LRS2. The task is to determine the correct audio-to-visual offset within a ± 15 frame window. An offset is considered correct if it is within 1 video frame from the ground truth. The distances are averaged over 5 to 15 frames. We compare our method to two state-of-the-art

Table 8.5: Label statistics for non-human test sets. S is *single head* and M *multi-head.*

Source	Type	Clips	Frames
The Simpsons	S	41	87
The Simpsons	M	582	251
Sesame Street	S	57	120
Sesame Street	M	143	424

synchronization methods: SyncNet [88] and the state-of-the-art Perfect Match [92]. We note that [92] represents an approximate upper limit to what we would expect our method to achieve, since we are using a similar network and training objective; the major difference is that we use our audio-visual objects instead of image crops from a face detector. The results (Table 8.4) show that our self-supervised model obtains comparable accuracy to these supervised methods.

5. Generalization to non-human speakers. We evaluate the LWTNet model's generalization to non-human speakers using the *Simpsons* and *Sesame Street* datasets described in Section 8.5.1. The results of our evaluation are summarized in Table 8.6. Since supervised speech analysis methods are often based on face detection systems, we compare our method's performance to off-the-shelf face detectors, using the *single-head* subset. As a face detector baseline, we use the state-of-the-art RetinaFace [105] detector, with both the MobileNet and ResNet-50 backbones. We report localization accuracy (as in Table 8.1) and Average Precision (AP). It is clear that our model outperforms the face detectors in both localization and retrieval performance for both datasets.

The second evaluation setting is detecting active speakers in videos from the *multi-head* test set. As expected, our model's performance decreases in this more challenging scenario; however, the AP for both datasets indicates that our method can be useful for retrieving the speaker in this entirely new domain. We show qualitative examples of ASD on the *multi-head* test sets in Figure 8.8.

Table 8.6: Non-human speaker evaluation for ASD and localization tasks on *Simpsons* and *Sesame Street*. MN: MobileNet; RN: ResNet50.

	Loc. Acc		ASD AP			
	Single-head		Single-head		Multi-head	
Method	Simp.	Ses.	Simp.	Ses.	Simp.	Ses.
Random	8.7	16.0	-	-	-	-
Center	62.0	80.1	-	-	-	-
RetinaFace RN	47.7	61.2	40.0	46.8	-	-
RetinaFace MN	72.1	70.2	60.4	52.4	-	-
Ours	**98.8**	**81.0**	**98.7**	**72.2**	**85.5**	**55.6**

8.6 Conclusion

In this paper, we have proposed a unified model that learns from raw video to detect and track speakers. The embeddings learned by the model are effective for many downstream speech analysis tasks, such as source separation and active speaker detection, that in previous work required supervised face detection.

Appendices

Appendices, qualitative videos and code for this chapter can be found online. [1]

Statement of authorship

A statement of authorship for this paper is provided in Appendix A.

[1] https://www.robots.ox.ac.uk/~vgg/research/avobjects/

9 | Self-supervised Object Detection From Audio-visual Correspondence

Triantafyllos Afouras[1*†] Yuki M. Asano[1*] Francois Fagan[2]
Andrea Vedaldi[2] Florian Metze[2]

[1]Visual Geometry Group, Oxford [2] Facebook AI
* Equal Contribution † Work done during an internship at FAIR.

Abstract

We tackle the problem of learning object detectors without supervision. Differently from weakly-supervised object detection, we do not assume image-level class labels. Instead, we extract a supervisory signal from audio-visual data, using the audio component to "teach" the object detector. While this problem is related to sound source localisation, it is considerably harder because the detector must classify the objects by type, enumerate each instance of the object, and do so even when the object is silent. We tackle this problem by first designing a self-supervised framework with a contrastive objective that jointly learns to classify and localise objects. Then, without using any supervision, we simply use these self-supervised labels and boxes to train an image-based object detector. With this, we outperform previous unsupervised and weakly-supervised detectors for the task of object detection and sound source localization. We also show that we can align this detector to ground-truth classes with as little as one label per pseudo-class, and show how our method can learn to detect generic objects that go beyond instruments, such as airplanes and cats.

To be published in the proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR) 2022.

9.1 Introduction

While recent progress in learning image and video representations has been substantial [72, 162, 174, 408], this has not yet translated into an ability to learn interpretable and actionable concepts automatically. By that, we mean that some manual labels are still required in order to map unsupervised representations to useful concepts such as image classes or object detections. In this paper, we thus consider the problem of learning interpretable concepts without any manual

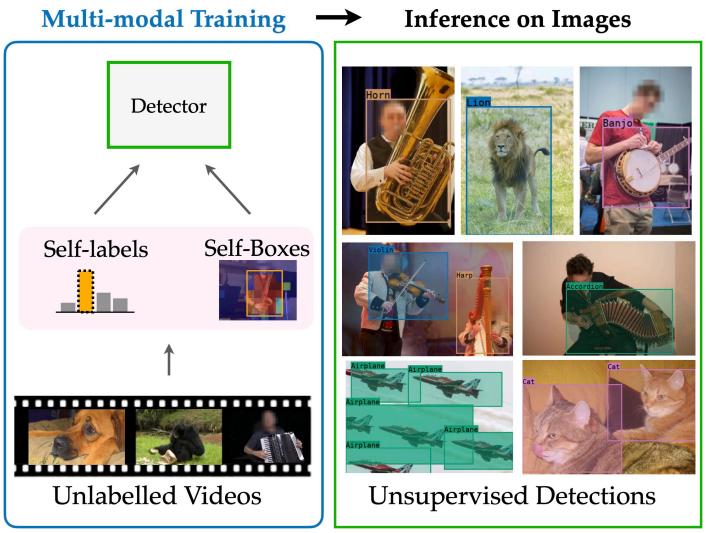


Figure 9.1: We train an object detector simply by watching videos. Without using any manual annotations, we learn to detect different objects in images, by first self-labelling boxes and object categories and then using those as targets to teach a detector. The detection results shown are outputs from our trained model; for visualisation purposes we show Hungarian-matched labels.

supervision. In particular, we focus on a problem that has not been explored extensively in the literature: learning to simultaneously detect and classify objects with no manual labels at all.

This problem is related to weakly supervised object detection (WSOD [39, 284]), with the difference that, in WSOD, the learning algorithm is given image-level labels telling it whether the image contains an occurrence of a given object type or not. Inspired by recent work in self-supervised learning, we seek to replace this source of external supervision with an internal supervisory signal extracted from the observation of video data. Videos are far richer than images, for example because they contain motion. Here, we focus on the multi-modal aspect of videos and use sound as a weak and noisy cue to learn about objects in the visual component of the data.

The power of multi-modal self-supervision has been demonstrated before in self-supervised representation learning, and, closely related, in *video clustering* [24]. However, while video clustering can provide an interpretation of the data in terms of discrete classes, it does not provide any information about the location of the relevant objects in images. On the other hand, *sound source localisation* [22, 27, 207, 294] has considered precisely the problem of localizing the source of sounds in images. It is therefore tempting to trivially combine image classification and sound source localisation in the hope of learning the type and location of objects automatically.

Unfortunately, such an approach does *not* lead to a satisfactory object detector. To understand why, it is important to note that the goal of sound source localisation is to *localize the sound while it is being heard*. This is insufficient for a detector because many objects emit sounds only occasionally and they become invisible to source localisation when they are silent. Instead, a detector that works in the visual domain should be responsive even when the object cannot be heard. Furthermore, source localisation methods generally only extract a heatmap giving the distribution of possible object locations; in contrast, a detector solves the much harder problem of enumerating all individual object instance that occur in an image by outputting instance-specific bounding boxes.

In order to solve these issues, we should treat the sound component as a useful cue to *learn* an object detector, but not as a cue which is *necessary for detection*. Instead, we consider the problem of taking as input a collection of raw videos and producing a list of object classes and locations, in order to train an image-based detector.

On a high level, our method is based on the following observation: we can use a sound source localisation network to learn about possible locations of sounding objects in videos. From this, we can extract a collection of bounding box pseudo-annotations for the objects and use those to learn a standard object detector. Because the latter only uses the visual modality, it immediately transfers to the detection of objects even when no relevant sound is present.

However, one challenge is that sound source localisation does not provide the necessary class information to train class-specific detectors, effectively resulting in only learning a region proposal network for generic objects, with high rates of false positives. To this end, we note that most sound source localizers are based on noise-contrastive formulations that, together with clustering-based approaches, occupy the current state-of-the-art in self-supervised representation learning. Leveraging this, we derive a joint formulation that can simultaneously benefit from and learn to localize sound sources and classify them without *any* supervision. The resulting output can then be used to train any off-the-shelf object detector such as a Faster-RCNN [316] to obtain a completely unsupervisedly learned object detector, as shown in Fig. 9.1.

Empirically, we test our method by training and testing on VGGSound [69] and AudioSet [149], as well as testing only on a subset of OpenImages [230].

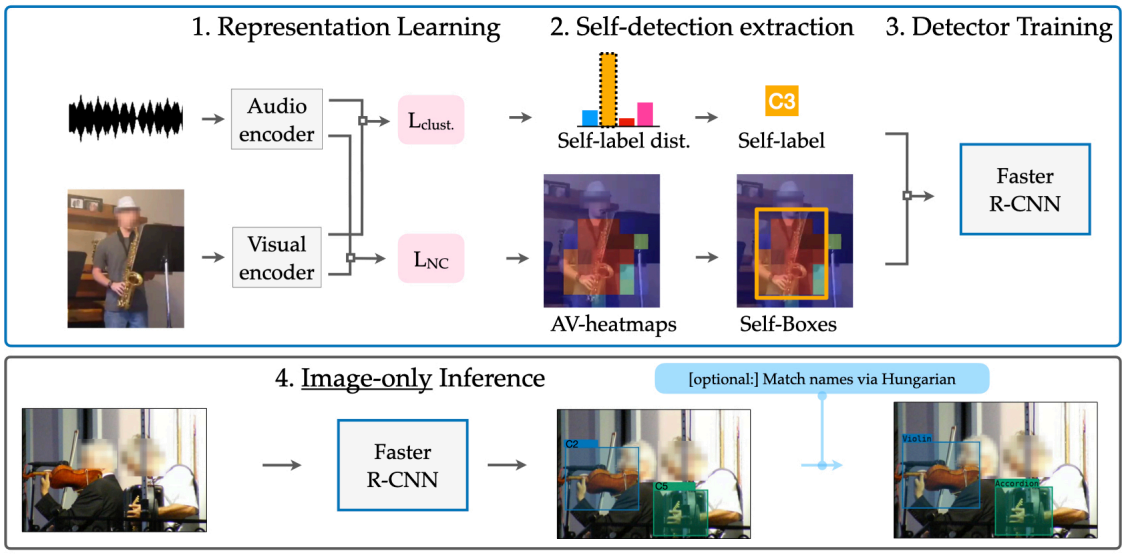


Figure 9.2: Self-supervised object detection from audio-visual correspondence: We combine noise-constrastive and clustering-based self-supervised learning to generate self-detections (boxes and labels) and use those as targets to train a detector. The trained detector can be used to detect objects from many categories on images without requiring audio.

9.2 Related Work

Audio-Visual Sound Source Localisation. Early work in sound-source localisation includes probabilistic models for localisation [130, 180, 207] and segmentation [194], but more recently the focus has shifted to dual-stream neural networks. For example, [22, 171, 341] propose a contrastive learning approach that matches the visual and audio components of the data. The work of [189, 191] instead clusters visual and audio features, associating to them centroids by means of a contrastive loss. Other works [9, 293] learn heatmaps by exploiting audio-visual synchronization in the same video, used previously for lip-to-mouth synchronization and active-speaker detection [87, 257], or by leveraging explicit attention modules [206]. Zhao et al. [440, 441] learn to associate pixels with audio sources by training with a mix-and-separate objective. Others [312] combine activation maps learned from class labels [67, 338] with a contrastive objective, use different levels of supervision and fusion techniques [314], or improve heatmaps by mining hard negative locations [68].

The work most similar to ours is [190], who first train a source localisation model with a contrastive objective and then use the learned heatmaps to extract object representations that are clustered using K-means. The cluster assignments are then used to train classifiers on top of the audio and video encoders. The paper proceeds to use these learned representations to discriminatively localize sound sources while suppressing quiet objects in 'cocktail party' scenarios.

Compared to our work, none of the above can detect and thus enumerate individual object occurrences because they produce heatmaps. Furthermore, they all require audio during inference, and therefore cannot be used on individual images or to detect silent objects.

Audio-visual category discovery. Learning visual categories is usually cast as image clustering, for which there is abundant prior work, such as recent 'deep clustering' methods [25, 60, 197, 381, 410, 417], or clustering with segmentation [382]. However, there is less work for clustering audio-visual data. In [16] the authors extend Deep Cluster [60] to the video domain by constructing two sets of labels from opposing modalities, which are used for cross-modal representation learning. The work of [327] combines clustering with audio-visual co-segmentation achieving combined audio-visual source separation. In [24], the authors extend the self-labelling method of [25] to multi-modal data by learning a shared set of labels between the two modalities. This work builds on the latter to complement and boost sound source localisation in a joint learning framework.

Weakly Supervised Object Detection (WSOD). Weakly supervised detection uses (manual) image-level category labels without bounding box annotations. Many approaches are based on a form of multiple-instance learning [40, 155, 373, 389, 414, 419, 431, 436], or proposal clustering [372]. Recent works in the area [198, 317] combine a variety of ideas, such as self-training [448] and spatial dropout [399] or explore the use of mixed annotations [318]. Other works obtain improvements by adding curriculum learning [434], using motion cues in videos [353], adversarial training [346], combining segmentation and detection [145, 242, 345], or modelling the uncertainty of object locations [23].

Other methods use technique such as CAM or analogous techniques [36, 67, 131, 338, 352, 446] as a form of weakly supervised saliency or localisation maps. Recent works have suggested that saliency methods can also be applied to self-supervised networks [29, 166], e.g. for object co-localisation [29].

Self-supervised multi-modal learning. Our work is also related to methods that use multiple modalities for representation learning [16, 21, 25, 27, 272, 295, 301] and synchronization [88, 222, 293]. A number of recent papers have leveraged speech as a weak supervisory signal to train video representations [240, 265, 281, 363, 364] whereas [11] uses speech, audio and video. Some works distil knowledge learned from one modality into another [7, 12, 138, 443]. Other works incorporate optical flow and other modalities [169, 170, 309, 440] to learn representations. For instance, the work of [376] learns to temporally localize audio events through audio-visual attention. CMC [377] learns representations that are invariant to multiple views of the data such as different color channels. Multi-modal self-supervision is also used to learn sound source separation in [142], albeit they assume to have pre-trained detectors.

9.3 Method

Our goal is to learn object detectors using only unlabeled videos, simultaneously learning to enumerate, localize and classify objects. Our approach consists of three stages summarized in Fig. 9.2: first, we learn useful representations using clustering and contrastive learning; second, we extract bounding boxes and class categories by combining the trained localisation and classification networks; third, we train an off-the-shelf object detector by using these self-extracted labels and boxes as targets.

Next, we explain each stage and refer the reader to the the arXiv version of the paperfor further architecture and training details.

9.3.1 Representation Learning

Sound source spatial localisation. Our method starts by training a sound source localisation network (SSLN) using a contrastive learning formulation inspired by [22]. The SSLN is learned from pairs (v,a), where $v \in \mathbb{R}^{3\times H\times W}$ is a video frame (i.e., a still image) and $a \in \mathbb{R}^{T\times F}$ is the spectrogram of the audio captured in a temporal window centered at that particular video frame.

We consider a pair of deep neural networks. The first network $f^v(v) \in \mathbb{R}^{C\times h\times w}$ extracts from the video frame a field of C-dimensional feature vectors, one per spatial location. We use the symbol $f^v_u(v) \in \mathbb{R}^C$ to denote the feature vector associated to location $u \in \psi = \{1,...,h\}\times\{1,...,w\}$ Here $h\times w$ is the resolution at which the spatial features are computed and is generally a fraction of the video frame resolution $H\times W$. The second network $f^a(a) \in \mathbb{R}^C$ extracts instead a feature vector for the audio signal.

Importantly, the spatial and audio features share the same C-dimensional embedding space and can thus be contrasted. We further assume that the vectors $f^v_u(v)$ and $f^a(a)$ are L^2 normalized (this is obtained by adding a normalization layer at the end of the corresponding networks). The cosine similarity of the two feature vectors is then used to compute a heatmap of spatial locations, with the expectation that objects that are correlated with the sounds would respond more strongly. This heatmap is given by:

$$h_u(v,a) = \langle f^v_u(v), f^v(a)\rangle/\rho, \quad u \in \psi,$$

where ρ, is a learnable temperature parameter.

For the multi-modal contrastive learning formulation [92, 292, 301], the heatmap is converted in an overall score that the video v and audio a are in correspondence. This is done by taking the maximum of the response:

$$S(v,a) = \max_{u\in\psi} h_u(v,a).$$

The contrastive learning objective is defined by considering videos $(v,a) \in \mathcal{B}$ in a batch $\mathcal{B}$. This comprises two terms. The first tests how well a video frame matches with its specific

audio among the ones available in the batch:

$$\mathcal{L}_{a\to v}(\mathcal{B}) = -\frac{1}{|\mathcal{B}|}\sum_{(v,a)\in\mathcal{B}} \log \frac{\exp S(v,a)}{\sum_{(v',a')\in\mathcal{B}} \exp S(v,a')}.$$

The second is analogous, testing how well an audio matches with its specific video frame:

$$\mathcal{L}_{v\to a}(\mathcal{B}) = -\frac{1}{|\mathcal{B}|}\sum_{(v,a)\in\mathcal{B}} \log \frac{\exp S(v,a)}{\sum_{(v',a')\in\mathcal{B}} \exp S(v',a)}.$$

These two losses are averaged in the *noise-contrastive* loss:

$$\mathcal{L}_{\text{NC}}(\mathcal{B}) = (\mathcal{L}_{a\to v}(\mathcal{B}) + \mathcal{L}_{v\to a}(\mathcal{B}))/2 \tag{9.1}$$

Category self-labeling. Spatial localisation does not provide any class information, whereas our goal is to also associate 'names' to the different objects in the dataset. To this end, we consider the self-labelling approach of [24]. To briefly explain the formulation, let $y(v,a) \in \mathcal{Y} = \{1,...,K\}$ be a label associated to the training pair (v,a). We also consider two classification networks. The first maps a video v to class scores $g^v(v) \in \mathbb{R}^K$ and is optimized by minimizing the standard cross-entropy loss:

$$\mathcal{L}_v(\mathcal{B}|y) = -\frac{1}{|\mathcal{B}|}\sum_{(v,a)\in\mathcal{B}} \text{logsoftmax}(y(v,a)|g^v(v)).$$

Note that this classification loss is equivalent to a contrastive loss on the cluster indices (as opposed to image indices) without normalization: As the last classification layer can be viewed as computing dot-products with each corresponding cluster's feature, it pushes the representation towards the feature of the corresponding cluster and away from the other clusters. The other network $g^a(a)$ is analogous, but uses the audio signal:

$$\mathcal{L}_a(\mathcal{B}|y) = -\frac{1}{|\mathcal{B}|}\sum_{(v,a)\in\mathcal{B}} \text{logsoftmax}(y(v,a)|g^a(a)).$$

As noted in [24], the crucial link between the two losses is that the labels y are shared between modalities. This is obtained by averaging the two losses:

$$\mathcal{L}_{\text{clust}}(\mathcal{B}|y) = (\mathcal{L}_v(\mathcal{B}|y) + \mathcal{L}_a(\mathcal{B}|y))/2. \tag{9.2}$$

Note that the labels y are unknown; following [24] these are learned in an alternate fashion with the classification networks, minimizing the same loss (9.2). In order to avoid degenerate solutions, the labels' marginal distribution must be specified, e.g. using a simple equipartitioning constraint:

$$\frac{1}{|\mathcal{D}|}\sum_{(v,a)\in\mathcal{D}} 1_{\{y(v,a)=k\}} = \frac{1}{K} \quad \text{for all } k=1,\ldots,K \tag{9.3}$$

where $\mathcal{D}$ denotes the entire dataset (union of all batches). Optimizing y can be done efficiently by using the SK algorithm as in [24].

Joint training. To summarize, given the dataset $\mathcal{D}$, we optimize stochastically w.r.t. random batches $\mathcal{B}$ the loss:

$$\mathcal{L}(\mathcal{B}|y) = \lambda\mathcal{L}_{\text{NC}}(\mathcal{B}) + (1-\lambda)\mathcal{L}_{\text{clust}}(\mathcal{B}|y) \tag{9.4}$$

where λ is a balancing hyperparameter.

The loss is optimized with respect to the localisation networks f^v and f^a and the classification networks g^v and g^a. These networks share common backbones q^v and q^a and differ only in their heads, so they can be written as $f^v = \hat{f}^v \circ q^v$, $g^v = \hat{g}^v \circ q^v$, $f^a = \hat{f}^a \circ q^a$ and $g^a = \hat{g}^a \circ q^a$.

The model is trained by alternating between updating the labels y with eq. (9.2) under constraint (9.3) and updating the networks by optimizing eq. (9.4).

9.3.2 Extraction of Self-labels for Detection

Once the localisation and classification networks have been trained, they can be used to extract self-annotations for training a detector. This is done in two steps: extracting object bounding boxes and finding their class labels.

Box extraction. To obtain "self-bounding boxes" for the objects, we use the simple heuristic suggested by [446]: the heatmap $h(v,a)$ is thresholded at a value $\epsilon(h)$, the largest connected component is identified, and a tight bounding box $t^*(v,a) \in \Omega^2$ around that component is

extracted (the notation means that the box is specified by the location of the top-left and bottom-right corners).

The threshold is determined dynamically as a convex combination of the maximum and average responses of the heatmap, controlled by hyperparameter β:

$$\epsilon(h) = \beta \max_{u \in \psi} h_u + (1-\beta) \frac{1}{|\psi|} \sum_{u \in \psi} h_u. \tag{9.5}$$

Class labelling. As noted above, we only extract a single object from each frame for the purpose of training the detector. Likewise, we only need to extract a single class label for the frame. This is done by taking the maximum response of the visual and audio-based classification networks:

$$y^*(v,a) = \operatorname*{argmax}_{y \in \mathcal{Y}} [g^v_y(v) + g^a_y(a)]. \tag{9.6}$$

Filtering the annotations. The assumption that frames contain a dominant object introduces noise but simplifies the problem and gives us the ability to use the audio to obtain purer clusters. Notably, we do not require the method above to work for *all* frames but instead rely on our detector to smooth over the specific and noisy self-annotations to learn a holistic detection.

9.3.3 Training the Object Detector

The process described above results in a shortlist of training triplets $(v, t^*, y^*) \in \mathcal{D}_{\text{det}}$, where v is a video frame (an image), t^* is the extracted bounding box and y^* is its class label. We use this dataset to train an off-the-shelf detector, in particular Faster R-CNN [316] for its good compromise between speed and quality.

Recall that, given an image v, Faster R-CNN detector considers a set of bounding box proposals $m \in M(v) \subset \Omega^2$. It then trains networks $y(m) = f^{\text{cls}}_{\text{det}}(m|v) \in \{1, \dots, K, \text{bkg}\}$ and $t(m) = f^{\text{loc}}_{\text{det}}(m|v) \in \mathbb{R}^4$ inferring, respectively, the class label $y(m)$ and a refined full-resolution bounding box $t(m)$ for the box proposal m. The label space is extended to also include a *background class* bkg, which is required as most proposals do not land on any object.

The detector is trained by finding an association between proposals and annotations. To this end, if $m^* = \text{argmax}_{m \in M(v)} \text{IoU}(m, t^*)$ is the proposal that matches the pseudo-ground truth bounding box t^* the best, one optimizes:

$$\mathcal{L}_{\text{det}}(v, t^*, y^*) = \mathcal{L}_{\text{reg}}(t(m^*), t^*) + \mathcal{L}_{\text{cls}}(y(m^*), y^*) + \sum_{m \in M(v): \text{IoU}(m,t^*) < \tau} \mathcal{L}_{\text{cls}}(y(m), \text{bkg}).$$

Here $\mathcal{L}_{\text{reg}}$ is the L^1 loss for the bounding box corner coordinates and $\mathcal{L}_{\text{cls}}$ the standard cross-entropy loss. Intuitively, this loss requires the best proposal m^* to match the pseudo-ground truth class y^* and bounding box t^* of, while mapping proposal m that are a bad match ($\tau \leq 0.7$) to class bkg. Further details, including how the region-proposal network that generates the proposals is trained, are given in the the arXiv version of the paper

Discussion. Training a detector is obviously necessary to solve the problem we set out to address. However, it can also be seen as a way of extracting 'clean' information from the noisy self-annotations. Specifically: (i) the noise in individual annotations is smoothed over the entire dataset; (ii) because of the built-in NMS step, the detector still learns to extract multiple objects per image even though a single self-annotation is given for each training image; (iii) by learning to reject a large number of false bounding box proposals, the detector learns to be more precise than the self-annotations are.

9.4 Experiments

We first introduce the datasets, experimental setup and relevant baselines; we then test our method against those, analyse it further via ablations and its capacity to generalize.

9.4.1 Datasets

AudioSet-Instrument. AudioSet [149] is a large scale audio-visual dataset consisting of 10-second video clips originally from YouTube. For training we use the *AudioSet-Instruments* [22] subset of the "unbalanced" split, containing 110 sound source classes as well as its more

constrained subset used by [190] spanning 13 instrument classes. Following previous work we use the "balanced" subset for evaluation on the annotations provided by [190].

VGGSound. VGGSound contains over 200K 10-second clips from YouTube spanning 309 categories of objects where there is some degree of correlation between the audio and the video. We create one subset by keeping only the 50 musical instrument categories yielding around 54K training videos, and one other subset, by keeping from those only the 39 categories that can be roughly mapped to the test-set annotations (details in the arXiv version of the paper). For VGGSound pseudo-ground truth test-set annotations are obtained using a supervised detector from [142], following [190].

OpenImages. For evaluation, we also use the subset of the OpenImages [230] dataset containing musical instruments, which spans 15 classes.

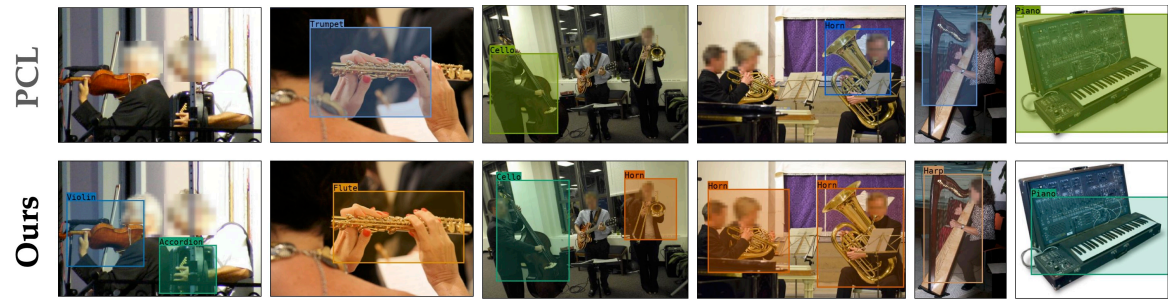


Figure 9.3: Qualitative results and comparison with a weakly supervised object detection method, PCL [372], on the OpenImages test set. Our method accurately detects objects, capturing their boundaries, even though it has been trained without *any* supervision. For visualisation purposes, we show the labels obtained from matching with the Hungarian method. More qualitative results provided in the the arXiv version of the paper.

9.4.2 Baselines

We are not aware of any prior work that learns an object detector for multiple object classes without any supervision. Instead, we compare against weakly-supervised detectors (hence using image-level labels) and unsupervised localisation methods that only produce heatmaps (not detections).

Weakly-supervised detection. For weakly-supervised detection, we consider PCL [372], the strongest such baseline for which we could find an implementation. Image-level labels are obtained from the corresponding dataset: for AudioSet the 13 labels of the training set are used, and for VGGSound we manually merge the 39 sub-classes to equivalent 15 classes (e.g., electric guitar, acoustic guitar to "Guitar" etc.); see the arXiv version of the paperfor full details. However, we have found that training PCL directly on the same data as our method (i.e., random clips from the VGGSound and AudioSet-Instrument subsets) does not work, likely due to the high amount of noise present in the labels (e.g., several videos are be labelled with an instrument, which is however not visible at all). To avoid this issue, we further preprocess the training data with a supervised instrument detector [142] and only retain frames where at least one relevant detection is found. This of course gives an "unfair" advantage to the baseline, but it is necessary to be able to use it at all.

Heatmap-based localisation. For our second baseline, we consider localisation methods that, similarly to us, use cross-modal self-supervised learning. The state-of-the art DSOL method of [190] is the most relevant, as it produces a heatmap roughly localizing the objects and produces class pseudolabels. While DSOL does not use image-level labels like PCL, it does use audio during inference, and thus strictly more information than our method (which performs localisation only in the visual domain).

Region proposals. Finally, we also compare against other baselines such as simply predicting a large centered box and class-agnostic region proposal methods such as selective search, and using a RPN obtained from supervised training on COCO [245].

9.4.3 Implementation Details.

Assessing class pseudolabels. Since class pseudolabels do not come with the "name" of the class (they are just cluster indices), they must be put in correspondence with human-labelled classes for evaluation. Following prior work in unsupervised image clustering [25, 35, 197, 381], we apply Hungarian matching [228] to the learned clusters and the ground truth classes.

Method	No labels?	VGGSound			Audioset			OpenImages		
		mAP_{30}	mAP_{50}	$mAP_{[50:95:5]}$	mAP_{30}	mAP_{50}	$mAP_{[50:95:5]}$	mAP_{30}	mAP_{50}	$mAP_{[50:95:5]}$
PCL (WSOD) [372]	✗	54.9	27.7	7.6	39.0	17.5	4.4	37.9	14.5	3.5
Ours - weak sup.	✗	67.6	42.9	14.2	50.6	30.9	10.3	48.9	33.7	9.5
Center Box*	✓	29.6	5.6	1.5	15.1	3.5	0.7	20.7	4.2	0.8
Selective Search* [379]	✓	5.2	1.1	0.4	2.8	0.4	0.1	7.4	2.1	0.7
COCO-trained RPN*	✗	33.4	7.5	1.6	19.0	4.1	0.8	24.4	11.1	2.6
Ours - self-boxes*	✓	48.1	29.6	10.0	27.8	14.1	4.8	NA	NA	NA
Ours - full	✓	**52.3**	**39.4**	**14.7**	**44.3**	**28.0**	**9.6**	**39.9**	**28.5**	**7.6**

Table 9.1: Self-supervised object detection. We report object detection metrics across three test datasets and find our method is far superior to other unsupervised approaches and outperforms even the weakly supervised baseline in most metrics. For methods denoted by *, we report class-agnostic evaluation numbers. *Center Box* denotes a simple baseline predicting random sized boxes in the middle of the frame. *Ours-weak sup.* is a variant of our model trained with the video-level category annotations in combination with our self-extracted boxes. The class-agnostic performance of the self-boxes that are used to train the detector reveals that the latter greatly outperforms them, which highlights the benefit of our approach.

Importantly, the matching is done *after* the detector is trained and only done for assessment; meaning that the detector does not use any manual label.

Training resolutions. We train our method on random square crops of 224 pixels after resizing to 256 pixels. During the training of the detector, we take random 224 crops and obtain the the self-supervised bounding boxes on-the-fly from our pretrained model, which are scaled and used to train the detector at the larger detector resolution.

Detector warm-up. We warm-up the Faster R-CNN detector by training in a class-agnostic manner for 20 epochs. This gives the RPN (which is randomly initialized) an opportunity to learn sufficiently stable bounding box proposals; we then switch to full class-aware supervision. We found that this leads to more robust convergence compared to training with pseudo-labels from the start.

Backbone pretraining. We also found it beneficial to pretrain the localizer backbone using only the localisation loss on the full AudioSet-Instruments dataset, and the detector backbone using self-supervised SimCLR [72] on ImageNet (note that the DSOL baseline uses instead *supervised* ImageNet pretraining for the backbone).

Method	single-instr. IoU-0.5	single-instr. AUC	multi-instr. cIoU-0.3
Sound of pixels [441]	38.2	40.6	39.8
Object t. Sound [22]	32.7	39.5	27.1
Attention [341]	36.5	39.5	29.9
DMC [189]	32.8	38.2	32.0
DSOL [190]	38.9	40.9	48.7
Ours	**50.6**	**47.5**	**52.4**

Table 9.2: Comparison to sound localisation methods. Since our detector does not require audio, we obtain detections on the video frames directly. Our model outperforms the baselines. Baselines numbers taken from [190].

Detector training. If not stated otherwise, the localizer and detector are trained on VGGSound and AudioSet whereas OpenImages are only used for evaluation. We do not have any information on the number of instruments in VGGSound and use all videos with no single/multi-object curation. For a fair comparison with DSOL, and only for the relevant experiment in Table 9.2, we train on AudioSet using the single-instrument subset for learning the localizer.

Number of clusters. For VGGSound training we use $K = 39$ if not stated otherwise, matching the 39 object types in the training set. Since the dataset is roughly balanced, uniform marginals are used as described in [24]. For AudioSet training we use $K = 30$ and Gaussian marginals.

9.4.4 Results

Self-supervised object detection. We summarise the results of our evaluations on the three test sets that we consider in Table 9.1. Following the image object detection literature, we use mAP at different IOU thresholds as the evaluation metric.

Our method clearly outperforms the PCL baseline even though it uses no manual annotations at all during training. PCL outperforms our approach in some of the datasets only if the IoU threshold used for mAP computation is relaxed substantially (0.3 IoU). However, for stricter thresholds our approach works better, which suggests that our detections have a relatively high spatial accuracy.

Dataset	**mAP**$_{50}$	Accordion	Cello	Drum	Flute	Horn	Guitar	Harp	Piano	Saxophone	Violin	Banjo	Trombone	Trumpet	Oboe
OpenImages	28.5	75.3	30.2	6.6	6.5	15.0	14.5	80.4	28.8	22.5	28.8	57.0	9.7	18.1	6.3
Audioset	28.0	41.3	44.9	0.9	5.5	21.7	39.5	82.6	52.7	2.5	17.4	46.7	8.0	-	-
VGGSound	39.4	88.6	39.4	1.8	50.0	3.4	34.9	95.6	50.2	14.4	56.3	100.0	2.2	11.0	3.8

Table 9.3: Per-class mAP breakdown For entries with '–' the test set does not contain any samples for that class.

To understand the impact of the noisy class self-labels, we also train and test a detector (Ours - weak sup.) with the bounding box labels from our localisation network, but utilising the ground truth video categories. The resulting performance difference is modest, resulting in a 3% AP50 drop in VGGSound and AudioSet. This further demonstrates the accuracy of our class self-labels, but also shows that our method has also the potential to leverage weak supervision if available.

Per-class performance breakdown. To better understand the strengths and weaknesses of our method, we report a performance breakdown by object class in Table 9.3. We observe that the model obtains good results consistently for classes of large objects with a distinctive appearance (e.g. accordions and harps), while it is weaker for smaller objects such as oboes, or for objects that appear closely in numbers, like drums.

Comparison to audio-visual heatmaps. In Table 9.2 we compare the performance of our method trained on AudioSet to state-of-the-art sound source localisation methods. For a fair comparison to these methods, we convert the union of our predicted bounding boxes with confidence above a set threshold into a binary map, and use the latter as a pseudo-heatmap to use the same evaluation code. Our approach outperform others for both class-agnostic single object localisation and for class-aware multi-object localisation, *without* using audio signals during inference.

We note however that cIOU is not a very reliable metric for evaluating a detector (or even sound localizer) as it favours high recall over precision: by averaging this metric over all classes the most frequent ones (e.g. drums, guitars, pianos) dominate the metric. We therefore

		mAP$_{50}$	
β	#boxes	VGGS	O.Images
0.7	single	39.4	28.5
0.8	single	36.6	29.4
0.9	single	35.8	28.8
0.7-0.9	single	37.5	**29.4**
0.7-0.9	multi	**38.0**	29.3

Table 9.4: Ablation of hyperparameter β, which controls the relative width of the bounding box; *multi* denotes the use of multiple self-labelled boxes per sample. Overall the method is fairly stable with respect to the choice of this parameter, however sampling β from a range (0.7-0.9) obtains a better balanced performance. Moreover we do not observe any substantial improvement from using multiple boxes.

		mAP$_{50}$	
# GT-cls.	K	VGGS	O.Images
39	20	34.4	24.4
39	30	35.1	25.1
39	39	39.4	**28.5**
39	50	**41.0**	27.5

Table 9.5: Number of clusters K. Our method is relatively robust ($< 5\%$ decrease in detection AP) to the number of clusters used for self-labelling.

	mAP$_{50}$	
Matching	VGGS	O.Images
Hung.	39.4	28.5
Argmax	39.6	**30.1**
Manual	**41.0**	29.5
1-shot	36.4	25.1
10-shot	37.1	25.8

Table 9.6: Matching strategies. Even with as little as 39 labels, our method can detect and classify objects accurately.

propose to the research community – and report in this paper – mean average precision (mAP) values as a more indicative metric.

Ablation: Thresholding parameter. In Table 9.4 we investigate the influence of the hyper-parameter β and the number of extracted target boxes we use for training the detector. From Eq. (9.5), a smaller β makes the heatmaps more focused and specific. With regards to this parameter, we find somewhat inverse trends for VGGSound vs OpenImages, where smaller β yields better results for the former and larger β for the latter. We find that a good balance in terms of performance can be achieved by sampling β randomly from a range, as over-specific boxes for some images and under-specific ones are successfully combined during detector training. Even when extracting multiple boxes for training the detector, we find our method performs similarly well to when only extracting a single box.

Ablation: Number of clusters K. In Table 9.5, we perform an experiment varying the number of clusters, and as a consequence the number of object categories that the detector learns, while keeping the test-set (containing 15 classes) fixed. We observe that out method achieves reasonable performance for a wide range of the number of clusters. The performance is fairly stable when using more clusters than the ground truth classes, and gradually decreases when fewer clusters are used.

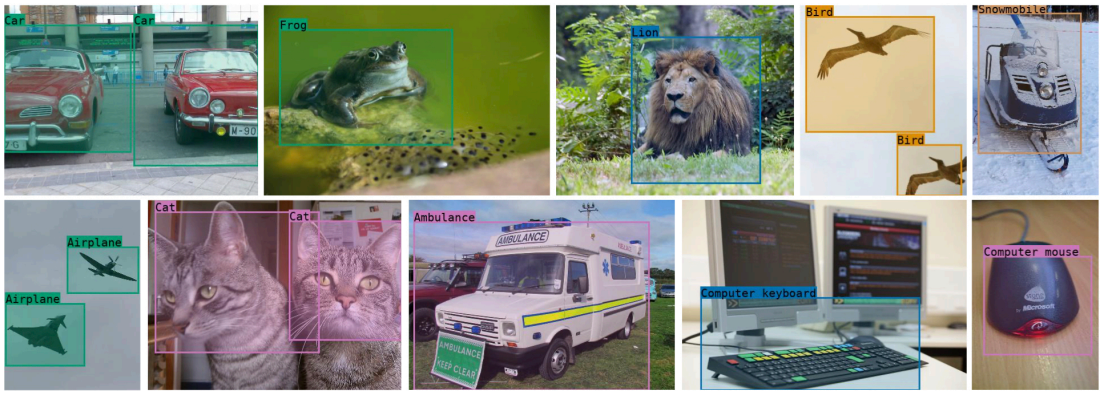


Figure 9.4: Object detection beyond musical instruments. Our proposed method can learn to accurately detect objects from more general categories, as long as they can be associated with a characteristic sound. The results shown here are from a model trained without labels directly on the full VGGSound dataset which includes 309 different video classes. Our method successfully learns to detect non-instrument objects, even in difficult multi-instance cases.

Data-efficient detector alignment. In Table 9.6 we conduct an investigation into the matching of the clusters to the ground truth labels. First, we compare Hungarian matching to simply taking the argmax of the ground truth class per self-label (i.e. assigning to the most frequent class). We find this yields almost the same results for VGGSound and a gain of 1.6% on OpenImages. By refining the Hungarian assignments via manually grouping similar classes and mapping each group to one of the test classes (for example mapping 'piano', 'electronic organ' and 'Hammond organ' all to 'piano'; see the arXiv version of the paperfor details), we find another small additional gain can be realized. While the Hungarian and argmax are common evaluation methods for the self-supervised clustering domain, we note that they are unsatisfactory as they still implicitly require a large set of labels. To alleviate this, we devise a "data-efficient" class-matching procedure as follows: Per pseudo-class, obtain the m videos which have the highest response for being in that particular pseudo-class (averaged across audio and video) and

obtain the class-label for these m samples. The pseudo-class is then assigned the most frequent ground truth class among these m. Overall, this reduces the number of labels required down to Km, making it a more realistic and scalable evaluation method for self-supervised approaches. Following this procedure, we find that even by just using $m = 1$ (i.e., a *total* number of 39 annotations), our method still achieves high performances of 37.1% and 25.3% on VGGSound and OpenImages. This 3% drop compared to the Hungarian can be further decreased to around 2.3% by using 10 labels per pseudo-class, for a total labelling budget of 390 images.

Qualitative analysis. We show examples of successfully detected objects in challenging images in Fig. 9.3, where we also include the outputs of the PCL baseline. Although our model has not been manually shown any objects boundaries during training we see that it can learn very accurate boxes around them and that it can successfully identify multiple objects in complicated scenes. We provide further examples in the arXiv version of the paper.

9.4.5 Towards general object detection

The results presented thus far have focused on subsets of common datasets with instruments solely to ensure comparability with prior works. Since one main goal of self-supervised learning is to leverage the vast amount of unlabelled data, we wish to investigate how general and robust our proposed method when applied on a far larger scale. For this, we increase our pretraining dataset by approximately $10\times$, simply by taking the whole of the VGGSound dataset, without any filtering. We set the number of learned clusters K to 300 and keep all training parameters the same; the result is an unsupervisedly trained object detector that can classify 300 pseudo-classes. As before, we match these to the VGGSound labels with the Hungarian algorithm and out of these take ten categories for which we have annotations in the OpenImages dataset (details in the Appendix).

In Fig. 9.4 we show qualitative results of some detections on OpenImages. The numerical results are given in Table 9.7. We find that even for objects that are deformable, such as cats, we get high AP_{30} values of 67.7% and that even objects that vary in shape, such as airplanes (see Fig. 9.4, bottom-right), we achieve a good performances 62.7%. While the results for the

Class	AP_{30}	AP_{50}	$AP_{[50:95:5]}$
Mean	45.6	24.4	6.5
Airplane	62.7	27.0	6,5
Ambulance	56.9	30.9	7.1
Bird	26.5	15.8	3.7
Car	29.8	18.4	5.1
Cat	67.7	28.0	7.7
Comp. Keyboard.	53.3	42.6	12.9
Comp. Mouse	35.9	25.4	8.8
Frog	43.5	19.5	4.7
Lion	34.1	22.2	4.9
Snowmobile	64.3	14.3	3.5

Table 9.7: Results on general object categories.

$AP_{50:95:5}$ metric indicate that there is still room for improvement, these initial results show that leveraging larger and more diverse video datasets for self-supervisedly learning object detectors is a promising avenue. We note that, since minimal curation is performed on the training data, and we use a large number of different object categories in a noisy dataset, this training setting is very challenging. These results further highlight the potential of our proposed method.

9.5 Conclusion

We have presented a method for training strong object detectors purely with self-supervision by watching unlabelled videos. We demonstrated that our best models perform better than a weakly supervised baseline, even after curating the dataset to filter out noisy samples for training the latter. Our method also outperforms heatmap-based methods in music instruments localisation, while having the ability to detect objects in images directly without requiring audio. We have also addressed one short-coming of using the Hungarian for evaluation by showing that data-efficient alignment of self-supervised detectors is possible with as little as one image per pseudo-label. Finally we applied our method to domains beyond musical instruments and found that it can learn reasonable detectors in this much less curated setting, paving the way to general self-supervised object detection.

Sound is a great natural source of supervision for training detectors; we believe our method will be the first of many to explore this exciting new direction.

Appendices

Appendices for this chapter can be found in the online version of the paper. [1]

Statement of authorship

A statement of authorship for this paper is provided in Appendix A.

[1] https://arxiv.org/pdf/2104.06401.pdf

Part IV

Sign language recognition

10 | Read and Attend: Temporal Localisation in Sign Language Videos

Gül Varol[1,2*] Liliane Momeni[1*] Samuel Albanie[1*]

Triantafyllos Afouras[1*] Andrew Zisserman[1]

[1]Visual Geometry Group, Oxford

[2] LIGM, École des Ponts, Univ Gustave Eiffel, CNRS, France

(* Equal Contribution)

Abstract

The objective of this work is to annotate sign instances across a broad vocabulary in continuous sign language. We train a Transformer model to ingest a continuous signing stream and output a sequence of written tokens on a large-scale collection of signing footage with weakly-aligned subtitles. We show that through this training it acquires the ability to attend to a large vocabulary of sign instances in the input sequence, enabling their localisation. Our contributions are as follows: (1) we demonstrate the ability to leverage large quantities of continuous signing videos with weakly-aligned subtitles to localise signs in continuous sign language; (2) we employ the learned attention to *automatically* generate hundreds of thousands of annotations for a large sign vocabulary; (3) we collect a set of 37K *manually verified* sign instances across a vocabulary of 950 sign classes to support our study of sign language recognition; (4) by training on the newly annotated data from our method, we outperform the prior state of the art on the BSL-1K sign language recognition benchmark.

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

10.1 Introduction

Sign languages are visual languages that, for deaf communities, represent the natural means of communication [366]. Our goal in this paper is to identify and temporally localise instances of signs among sequences of continuous sign language. Achieving automatic sign localisation enables a diverse range of practical applications: construction of sign language dictionaries

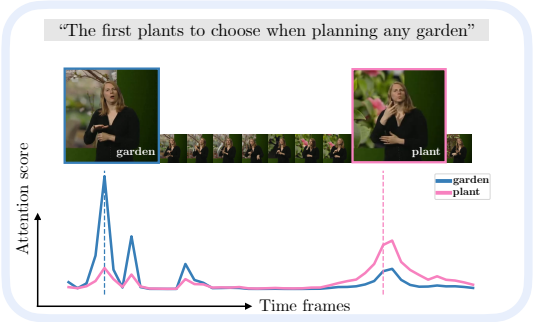


Figure 10.1: Sign localisation emerges from sequence prediction. In this work, we show that the ability to localise instances of signs emerges naturally by training a Transformer model [384] to perform a sequence prediction task on hundreds of hours of continuous signing videos with weakly-aligned subtitles.

to support language learners, indexing of signing content to enable efficient search and "intelligent fast-forward" to topics of interest, automatic sign language dataset construction, "wake-word" recognition for signers [324] and tools to assist linguistic analysis of large-scale signing corpora.

In recent years, there has been a great deal of progress in temporally localising human actions within video streams [350, 442] and spotting words in spoken languages through aural [96] and visual [270, 360] keyword spotting methods. In both cases, a key driver of progress has been the availability of large-scale annotated datasets, enabling the powerful representation learning abilities of convolutional neural networks to be brought to bear on the task.

By contrast, annotated datasets for sign language are limited in scale and typically orders of magnitude smaller than their spoken counterparts [43]. Widely used datasets such as RWTH-PHOENIX [51, 216] and the CSL dataset [192] provide continuous sign annotations in the form of *glosses*[1] or free-form sentences, but lack precise temporal annotations and are limited in content diversity, vocabulary, and scale. Large-scale collections of continuous signing videos exist, but are limited to sparse annotation coverage [14, 335].

In the absence of large-scale annotated training data, in this work we turn to a readily available and large-scale source: sign-interpreted TV broadcast footage together with subtitles of the corresponding speech in English. We propose to annotate this data with signs by training a Transformer [384] to predict, given input streams of continuous signing, the

[1] Glosses are atomic lexical units used to annotate sign languages.

corresponding subtitles, and then using its trained attention mechanism to perform alignment from English words to signs.

This is a very challenging task: first, subtitles are only *weakly aligned* to the signing content—a sign may appear several seconds before or after its corresponding translated word appears in the subtitles, thus subtitles provide a relatively imprecise cue about the temporal location of a sign. Second, sign interpreters produce a *translation* of the speech that appears in subtitles, rather than a *transcription*—words in the subtitle may not correspond directly to individual signs produced by interpreters, and vice versa. Third, grammatical structures between sign languages and spoken languages differ considerably [366], and consequently the *ordering* of words in the subtitle is typically not preserved in the signing.

The core hypothesis motivating this approach is that *in order to solve the sequence prediction task, the attention mechanism of the Transformer must be capable of localising sign instances*. We demonstrate that by employing recent sign spotting techniques [14, 271] to coarsely align subtitles, sequence prediction is rendered tractable. One of the primary findings of this work is that, when performed at large scale (across hundreds of hours of continuous signing content), the ability to localise signs indeed emerges from the attention patterns of the sequence prediction model (Fig. 10.1).

We make the following four contributions: (1) by training on an appropriate sequence prediction task, we show that the attention mechanism of the Transformer learns to attend to specific signs, enabling their *localisation*; (2) we employ the learned attention to *automatically* generate hundreds of thousands of annotations for a large sign vocabulary; (3) we collect a set of 37K *manually verified* sign instances across a vocabulary of 950 sign classes to support our study of sign language recognition; (4) by training on the newly annotated data from our method, we outperform the prior state of the art on the BSL-1K sign language recognition benchmark.

10.2 Related Work

Our approach relates to prior work on sign language recognition, translation, spotting, and in particular automatic annotation of sign language data. We present a discussion of these,

followed by a brief overview of Transformers in natural language processing (NLP) and works in other domains using attention mechanisms for localisation.

Sign language recognition and translation. The computer vision community has a long history of efforts to develop systems for sign language recognition, reaching back to the 1980s [371]. Initial work focused on hand-crafting features [129, 371] to model discriminative shape and motion cues and explored their usage in combination with Hidden-Markov Models [362, 387]. These works were followed by approaches that employed pose estimation as a basis for recognition [289, 307]. The community later transitioned to employing convolutional neural networks (CNNs) for appearance modelling [52]. In particular, the I3D architecture, originally developed for action recognition [61], has proven to be effective for sign recognition [13, 201, 235, 237, 270]—we similarly employ this model in our work.

Continuous sign language recognition entails important challenges compared to *isolated* sign recognition, including epenthesis effects and co-articulation [43] as well as the non-trivial definition of temporal boundaries between signs [45]. Towards dealing with these problems, [73] uses the CTC loss [159] to infer an alignment between sequence-level annotations and visual input and introduces an auxiliary loss to use the alignments as pseudolabels; while [48] proposes a graph convolutional network to automatically segment large sign language video sequences into short sentences, aligned with their subtitle transcription.

Recent works have applied sequence-to-sequence models to sign language translation. Camgöz et al. [51] use a two-stage pipeline that translates a video into gloss sequences then those into spoken language. Subsequent work [56] replaces this framework with a Transformer model trained on frame-level features jointly for recognition and translation, while [55] combines multiple articulators including face and upper body pose to train a translation system without gloss annotations. These approaches [51, 55, 56] have shown improvements towards translation in the restricted domain of discourse of the RWTH-PHOENIX-Weather-2014T German Sign Language (DGS) dataset [51]. Ko et al. [214] train a sequence-to-sequence model using keypoint features on Korean Sign Language translation. Although these methods show promising results in constrained conditions, open-vocabulary sign language translation in the wild remains largely unsolved.

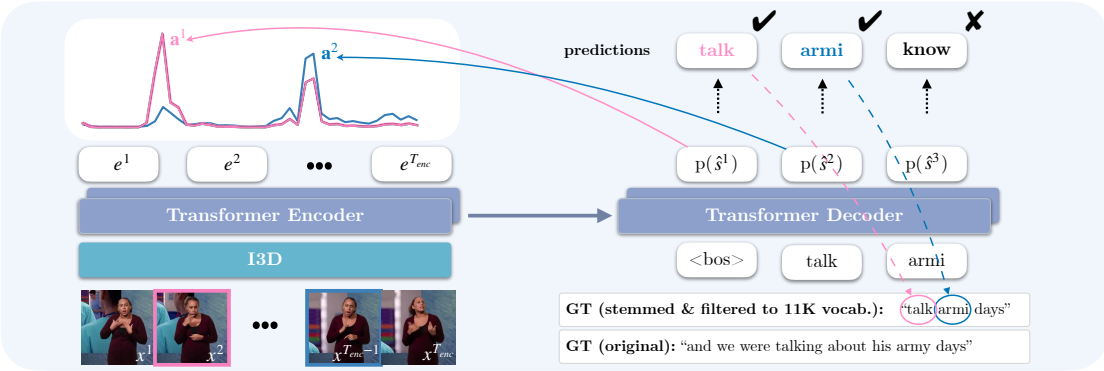


Figure 10.2: Pipeline: We use an I3D model pretrained on sign classification to extract spatio-temporal visual features by using a sliding window. We then train a 2-layer Transformer model to predict stemmed subtitles from the input video feature sequence. We use the learned model's attention vectors to spot new instances of signs by checking which words in the predicted hypothesis overlap with the stemmed subtitle. For example, here the tokens "talk" and "armi", found in the model's hypothesis, also appear in the subtitle and are therefore retained, while "know" does not and is hence discarded. The location of a new spotting is determined by the index at which the corresponding encoder-decoder attention peaks. Note: we omit the sample index, subscript i, shared by all variables (described in Sec. 10.3).

Automatic annotation of sign language data. Sign language datasets either offer isolated gloss-level annotations of single signs, e.g., MSASL [201], WLASL [235], or are heavily constrained in visual domain and vocabulary, e.g., RWTH-PHOENIX [51, 216], KETI [214] (only 105 sentences). Large-scale continuous sign language datasets, on the other hand, are not exhaustively annotated [14, 334]. The recent efforts of Albanie et al. [14] scale up the automatic annotation of sign language data, and construct the BSL-1K dataset with the help of a visual keyword spotter [270, 361] trained on lip reading to detect instances of mouthed words as a proxy for spotting signs. *Sign spotting* refers to a specialised form of sign language recognition in which the objective is to find whether and where a given sign has occurred within a sequence of signing. It has emerged as an intermediate step to collect more annotated sign language data. With this goal, Momeni et al. [271] use dictionary lookups in subtitled videos and improve low-shot sign spotting. Other automatic annotation approaches include an automatic pipeline for active signer detection and sign language diarisation [13]. While these previous methods are *context-free*, in this work, we introduce a *context-aware* approach that can be used to localise signs automatically. In fact, while we profit from annotations obtained in prior works using mouthing cues [14] and dictionaries [271], our approach differs considerably from theirs in method—we define the supervision directly on subtitles and

formulate the problem as a sequence-to-sequence prediction task. We demonstrate the benefits of our approach empirically in Sec. 10.4.

Transformers in NLP. Incorporating an attention mechanism into encoder-decoder architectures led to a revolution in neural machine translation [31] by reducing dependency on strong text alignment. Vaswani et al. [384] further extended this approach by replacing all recurrent and convolutional components of a sequence-to-sequence model with self-attention. Even though such methods implicitly model source-to-target alignment with attention, their primary focus is on translation performance, rather than word-alignment. [144] further studies how to simultaneously optimise for accurate word-alignment without sacrificing translation performance—we investigate a variant of their approach in Sec. 10.4.

Attention mechanisms for localisation. Cross-modal attention has been employed in the literature for various localisation problems such as visual grounding in videos [70, 248, 413, 427] or images [104, 425], keyword spotting in audio [344] or visual speech [270, 361] and audio-visual sound source localisation [22, 171, 341]. However, to the best of our knowledge, our work is the first to apply these ideas at large-scale to sign localisation from weakly-aligned subtitles.

10.3 Sign Localisation with Attention

In this section, we describe how we train a Transformer model on a weakly-supervised sign language sequence-to-sequence task and then use the trained model to perform sign localisation (see Fig. 10.2 for an overview).

Let $\mathcal{X}_{\mathfrak{L}}$ denote the space of sign language video segments $\mathfrak{L}$, and $\mathcal{T}$ denote the space of subtitle sentences. Further, let $\mathcal{V}_{\mathfrak{L}} = \{1,...,V\}$ represent the *vocabulary* (an enumeration of spoken language tokens that correspond to signs that can be performed in $\mathfrak{L}$) and let $\mathcal{S}$ denote a subtitled collection of I videos containing continuous signing, $\mathcal{S} = \{(x_i, s_i) : i \in \{1,...,I\}, x_i \in \mathcal{X}_{\mathfrak{L}}, s_i \in \mathcal{T}\}$. Our objective is to localise potential occurrences of signs in $\mathcal{S}$.

Transformer training with subtitled videos. To address this task, we propose to train a sequence-to-sequence model with attention. Given a video-subtitle pair $(x_i,s_i)\in\mathcal{S}$, we train a Transformer [385] to predict the target text sequence $s_i=(s_i^1,s_i^2\cdots,s_i^{T_{dec}})$ from the source video sequence $x_i=(x_i^1,x_i^2,\cdots x_i^{T_{enc}})$, one token at a time. Specifically, the Transformer's encoder transforms x_i into an encoded sequence $enc(x_i)=(e_i^1,e_i^2,\cdots e_i^{T_{enc}})$. The decoder then attends on the encoded sequence and predicts the output sequence $\hat{s}_i=(\hat{s}_i^1,\hat{s}_i^2,\cdots\hat{s}_i^{T_{dec}})$ auto-regressively, factorising its joint probability into a product of individual conditionals:

$$p(\hat{s}_i|x_i)=\prod_{t=1}^{T_{dec}}p(\hat{s}_i^t|\hat{s}_i^1,\hat{s}_i^2\cdots\hat{s}_i^{t-1},enc(x_i)). \tag{10.1}$$

Using the target subtitles s_i as the ground truth output sequences, we train the model to maximise their log likelihoods by minimising the following loss:

$$\mathcal{L}=-\mathbb{E}_{(x_i,s_i)\in\mathcal{S}}\log p(s_i|x_i) \tag{10.2}$$

Note that we assume access to a sparse collection of automatic sign annotations, $\mathcal{N}=\{(x_k,v_k):k\in\{1,\ldots,K\},v_k\in\mathcal{V}_{\mathfrak{L}},x_k\in\mathcal{X}_{\mathfrak{L}},\exists(x_i,s_i)\in\mathcal{S}\,s.t.\,x_k\subseteq x_i\}$, using mouthing cues [14] and dictionaries [271]. In practice, we restrict the Transformer training on a subset of videos $\mathcal{S}_A\subseteq\mathcal{S}$, containing at least one of these annotations within the subtitle timestamps, formally $\mathcal{S}_{\mathcal{A}}=\{(x_a,s_a):a\in\{1,\ldots,A\},x_a\in\mathcal{X}_{\mathfrak{L}},\exists(x_k,s_k)\in\mathcal{N}\,s.t.x_k\subseteq x_a\}$. This ensures approximate alignment between the source video and target subtitle. For arbitrary sequences in $\mathcal{S}$ this is not guaranteed due to imperfect synchronisation between subtitles (corresponding to audio) and sign language interpretation. The goal of our training is therefore to exploit the knowledge of the unannotated words in the subtitles in $\mathcal{S}_A$ in order to discover a new collection of (x,v) sign-video pairs (that is not included in $\mathcal{N}$) in the entire set $\mathcal{S}$.

Localising new sign instances with attention. Next, we describe how we use the Transformer model to look for new sign instances (see Fig. 10.2). After inputting the video sequence x_i into the trained model, we use a decoding strategy (e.g., greedy) to predict the output sequence $\hat{s}_i$ and corresponding attention vectors $a_i=(\mathbf{a}_i^1,\mathbf{a}_i^2,\cdots\mathbf{a}_i^{T_{dec}})\in R^{T_{dec}\times T_{enc}}$. We iterate over the predicted sequence $\hat{s}_i$ and localise new sign instances *only* for the tokens predicted correctly (i.e., appearing in subtitle s_i); the video location is determined by the index at which the

corresponding attention vector is maximised, to yield sets of (location, sign) pairs of the form: $\{(\text{argmax}_{j\in\{1,2\cdots T_{enc}\}}\mathbf{a}_i^t(j), s_i^t) : \hat{s}_i^t = s_i^t, t \in \{1, 2\cdots T_{dec}\}\}$.

Implementation details. We represent the input video x_i with features extracted using a pretrained spatio-temporal convolutional neural network model, applied in a sliding window manner with a 4-frame stride. In particular, we train an I3D architecture [61] on an extended set of automatic annotations $\mathcal{N}$ that we obtain by combining the methods of [14] and [271], to spot signs via mouthing cues and sign language dictionaries, respectively. We train with a single-sign classification objective and follow the same hyperparameters (e.g., 16-frame inputs) of the sign language recognition models in [14]. The 1024-dimensional video features from I3D are used as input to the Transformer encoder.

To construct ground-truth text labels for our Transformer training, we stem the words in every subtitle under the assumption that variations of a written word could map to the same sign. We note that the many-to-many mapping between words and signs is a complex problem, which we do not explicitly deal with in this work. To establish a tractable problem, we define a vocabulary of 11,515 stems based on their frequency and occurrence within the automatic annotations $\mathcal{N}$. This is reduced from an original set of 40K words appearing in the full set of subtitles S. We further remove stop words for which there is often no sign correspondence. This approach resembles *glossing* sign language data, i.e., representing sign sequences with word sequences, without spoken language grammar.

Following common practice in the sequence-to-sequence literature [385], we train the model with teacher forcing [405], i.e. at every decoding step we provide the previous-step's ground truth as input to the decoder. During inference we experiment with three different decoding strategies: auto-regressive greedy decoding, left-to-right beam search, and teacher forcing. With greedy decoding, we iterate over the available sequences and for each one, we select as new spottings all the words in the predicted hypothesis that appear in the reference subtitle. For beam search, we iterate over the predictions which overlap with the reference from the multiple returned hypotheses, and select for each predicted word the location with maximum attention score. We show results for another variant of beam search where we choose the hypothesis with the highest recall in the Appendix. With teacher forcing, we do not use the token predictions of

the model, but only the attention scores, which we associate with the next ground-truth word in the subtitle at every decoding step. Since we consider all words in the subtitles, this strategy provides good yield but no notion of the model's confidence. In order to obtain a confidence score we use the following heuristic: For every sequence, a word found in the subtitle is automatically annotated if the attention peak for the corresponding decoding step is higher than a threshold τ.

When using Transformers with multiple attention heads, we obtain single attention scores by averaging the attention vectors of the individual heads. In Sec. 10.4.3 we discuss results on combining attention from different decoder layers.

10.4 Experiments

This section is structured as follows: We first present the datasets used as well as the various training and evaluation protocols that we follow in our experiments (Sec. 10.4.1). Next, we show how we choose our pretrained input video features (Sec. 10.4.2). Then, we evaluate our Transformer models trained with these features and discuss different strategies for mining new instances to obtain an automatically annotated training set (Sec. 10.4.3). We show that, when adding our newly mined training samples, we outperform the previous state of the art on sign language recognition (Sec. 10.4.4). Finally, we provide qualitative results on two datasets (Sec. 10.4.5) and discuss limitations (Sec. 10.4.6).

10.4.1 Data and evaluation protocols

Datasets. We use BSL-1K [14], a large-scale, subtitled and sparsely annotated dataset (for a vocabulary of 1,064 signs) of more than 1000 hours of continuous signing from sign language interpreted BBC television broadcasts. The programs cover a wide range of genres: from medical dramas and nature documentaries to cooking shows. In Sec. 10.4.5, we show qualitative examples on the RWTH-PHOENIX [51] dataset, which is significantly smaller in size and from weather broadcasts only, restricting the domain of discourse.

		$\text{Test}^{\text{Rec}}_{2K}$ [14]				$\text{Test}^{\text{Rec}}_{37K}$			
		2K inst. / 334 cls.				37K inst. / 950 cls.			
		per-instance		per-class		per-instance		per-class	
Training	#ann.	top-1	top-5	top-1	top-5	top-1	top-5	top-1	top-5
M [14]§	169K	76.6	89.2	54.6	71.8	26.4	41.3	19.4	33.2
D	510K	70.8	84.9	52.7	68.1	60.9	80.3	34.7	53.5
M+D	678K	**80.8**	**92.1**	**60.5**	**79.9**	**62.3**	**81.3**	**40.2**	**60.1**

Table 10.1: A new recognition test set $\text{Test}^{\text{Rec}}_{37K}$ and an improved I3D model: We employ the method of [271] to find signs via automatic dictionary spotting (D), significantly expanding the training and testing data obtained from mouthing cues by [14] (M). We also significantly expand the test set by manually verifying these new automatic annotations from the test partition ($\text{Test}^{\text{Rec}}_{2K}$ vs $\text{Test}^{\text{Rec}}_{37K}$). By training on the extended M+D data, we obtain state-of-the-art results, outperforming the previous work of [14] and providing strong I3D features for the subsequent steps of our method. §The slight improvement in the performance of [14] over the original results reported in that work is due to our denser test-time averaging when applying sliding windows (8-frame vs 1-frame stride).

Transformer training and evaluation on $\text{Test}^{\text{Loc}}_{7K}$. To form the video-subtitle training data pairs, we sample 183K ($\mathcal{S}_A$) out of 685K subtitles from the BSL-1K training set ($\mathcal{S}$), in which there exists at least 1 automatic annotation (with a confidence score above 0.7) from the annotations collection $\mathcal{N}$. $\mathcal{N}$ is formed by applying the method of [14] on a large vocabulary of words beyond 1K to find signs via mouthing cues and applying the method of [271] to find signs via automatic dictionary spotting. See the Appendix for details on this step. Subtitles originally contain 9.8 words from the initial 40K words vocabulary on average, which is reduced to 4.4 words per subtitle from the 11K stems vocabulary after stemming and filtering. Corresponding videos are tightly extracted according to the subtitle timestamps, and are on average 3.52 seconds long.

For evaluating the localisation capability of the proposed method, we use the automatic annotations $\mathcal{N}$ in the BSL-1K test set whose confidence scores are above 0.9, resulting in 7497 subtitle-video pairs with a total of 7661 annotations, referred to as $\text{Test}^{\text{Loc}}_{7K}$. We measure the localisation accuracy for the annotated words in each subtitle and only on the correct predictions: we consider a correct prediction to be also correctly localised if its predicted location lies within 8 frames of the annotation time. We also report recall and precision of the model's predictions for each sequence by measuring the percentage of words in the subtitle

that are predicted (recall) and the percentage of predicted words which appear in the subtitle (precision). For all three metrics, we report the average over all sequences in the test set.

Single-sign recognition benchmark. In order to justify the value of our automatic annotation approach with the Transformer model, we evaluate on the proxy task of single-sign recognition on trimmed videos by using our localised sign instances from the training set as labels for classification training. Similar to [14, 201, 235], we adopt top-1 and top-5 accuracy metrics reported with and without class-balancing.

We use the BSL-1K manually verified recognition test set with 2K samples [14], which we denote with $\text{Test}^{\text{Rec}}_{2K}$, and significantly extend it to 37K samples as $\text{Test}^{\text{Rec}}_{37K}$. We do this by collecting new annotations from human annotators using the VIA tool [114] with a verification task as in [14]. This extended test set reduces the bias towards signs with easily spotted mouthing cues (since we also include dictionary spottings [270]) and spans a larger fraction of the training vocabulary, i.e. 950 out of 1064 sign classes (vs 334 classes in the original benchmark $\text{Test}^{\text{Rec}}_{2K}$ of [14]).

10.4.2 Comparison of video features

We first conduct experiments to determine which I3D video features are best suited as input to the Transformer model as described in Sec. 10.3. In Tab. 10.1, we demonstrate the benefits of combining annotations from both mouthing (M) [14] and dictionary spottings (D) [271]. We show that our sign classification training using 678K automatic annotations obtains state-of-the-art performance on $\text{Test}^{\text{Rec}}_{2K}$, as well as our new and more challenging test set $\text{Test}^{\text{Rec}}_{37K}$. We therefore use this M+D model for the rest of our experiments. Note that all three models in Tab. 10.1 (M, D, M+D) are pretrained on Kinetics [61], followed by video pose distillation as described in [14]. We observed no improvements when initialising M+D training from M-only pretraining.

10.4.3 Mining training examples through attention

Next, we ablate different design choices for the Transformer model.

			Loc. Acc. (GD)	Loc. Acc. (TF)
Tr.	Recall	Prec.	Att. layer 1/2/3 [avg]	Att. layer 1/2/3 [avg]
1L	15.8	36.4	65.9 [65.9]	44.8 [44.8]
2L	**16.5**	**37.2**	63.9/57.8 [**66.1**]	51.1/37.6 [44.5]
3L	**16.5**	36.9	62.5/60.8/16.4 [65.3]	**51.4**/38.4/15.7 [46.4]

Table 10.2: Localisation performance of attention layers. We evaluate the performance of Transformers on $\text{Test}_{7K}^{\text{Loc}}$ for different number of encoder/decoder layers in the training (different rows). We report the localisation accuracy for the encoder-decoder attention scores from every layer, as well as the average over layers, for both teacher forcing (TF) and greedy decoding (GD) modes.

Which attention layer for sign-video alignment? Similarly to [144], we conduct an investigation into which decoder layer gives attention scores that are more useful for localising signs. We train three models, with 1, 2 and 3 encoder and decoder layers and report the localisation accuracy when using the attention from each layer separately, or an average of all layers. The results on $\text{Test}_{7K}^{\text{Loc}}$ in Tab. 10.2 suggest that averaging the attention scores over all layers gives the best localisation when using greedy auto-regressive decoding, while using the attention scores from the first decoder layer works best with teacher forcing. We note that this finding stands in contrast to those of [144] which concluded that the penultimate layer works better for word alignment in a machine translation task. We conjecture that the difference results from the different nature of the two domains, i.e., video versus text inputs. In terms of precision and recall, all three models perform similarly with rates at 37% and 16%, respectively. We continue with a 2-layer Transformer model for the rest of the experiments and given the observations in Tab. 10.2, we use the layer-averaged attention with greedy decoding and the first layer attention with teacher forcing.

Incorporating sparse annotations. As explained in Sec. 10.3, we make use of the available sparse annotations $\mathcal{N}$ to restrict the training subtitles to those with at least 1 annotation. When removing this constraint, the model does not train as well, and reaches a recall of only 6.8% (vs 16.5%).

Here, we also report some of our findings by employing three additional strategies to improve the Transformer training using the sparse annotations $\mathcal{N}$. In all three cases, we observe no or minor gains (on $\text{Test}_{7K}^{\text{Loc}}$), at the cost of a more complex method and the need for

annotations. Therefore, we do not integrate them in our final model and provide detailed results in the Appendix.

Alignment loss on sparse annotations: We investigate whether the sparse annotations $\mathcal{N}$ could be used for supervising the sign-video alignment explicitly (similar to [144] in NLP). To this end, we define an additional loss that operates on the encoder-decoder attention to enforce a high response whenever there is known location information. We achieve this via an additional L2 loss term between a 1D gaussian centered around the annotated time frame and the corresponding attention vector. While the localisation performance with teacher-forcing increases (58.7% vs 51.1%), it still remains lower compared to the corresponding greedy decoding result and we observe no significant gains for other metrics measured on the predictions.

Curriculum learning with sparse annotations: To provide warmup for the model training, we start by temporally trimmed video inputs around known sign locations $\mathcal{N}$. We gradually increase the number of annotations from 1 to 3, before we fully input the subtitle duration to the Transformer. We only observe minor improvements: 16.0% vs 15.8% recall with the 1-layer architecture.

Subtitle alignment through active signer detection and sparse annotations: To overcome the alignment noise present in the data, we apply an algorithm that combines a pose-based active signer detection [13] and the knowledge of sparse annotations $\mathcal{N}$. Specifically, we apply temporal shifts to subtitles such that their temporal midpoint aligns with the average time of any annotated signs they contain. We then apply affine transformations to the subtitles without annotations such that they fill the regions between those with annotations, subject to the hard constraint that the expansions do not overlap periods of inactive signing. This approach increases the amount of training subtitles with annotations to 230K; however, training with this new set does not improve recall (15.4% vs 16.5% with 2-layers).

Which decoding mechanism? To form a new annotated set for sign recognition training, we apply the trained Transformer models on the whole 685K training video-subtitle pairs of the BSL-1K dataset. In Tab. 10.3 we summarise and compare the yield of new training samples mined with the different decoding strategies we discussed in Sec. 10.3. We report

Spotting mode	#subtitles unannot.	#ann. 11K	#ann. 1K	top-1 per-inst	top-1 per-cls
TF ($\geq .2$)	114K	290K	97K	22.2	4.7
TF ($\geq .1$)	408K	1.7M	545K	37.3	13.4
TF ($\geq .05$)	457K	2.3M	754K	38.7	14.4
TF ($\geq .05$) (align. loss)	457K	2.3M	757K	38.8	14.6
BS (10 best)	109K	329K	166K	49.6	22.7
GD (no subtitle filtering)	480K	1.4M	910K	50.6	22.6
GD (align. loss)	53K	188K	108K	53.6	**24.8**
GD	53K	188K	107K	**53.9**	<u>24.7</u>

Table 10.3: Automatically annotating the training data: We show the yield obtained from various decoding strategies in terms of number of additional annotations (left). Training models only with these annotations, we evaluate the recognition accuracy on $\text{Test}^{\text{Rec}}_{37K}$. Greedy decoding (GD) obtains better results than teacher forcing (TF) even when not filtering the predictions against the ground-truth subtitles. Neither including 10 best predictions from beam search (BS) nor using the model trained with the alignment loss influences the recognition evaluation significantly.

the number of previously unannotated subtitles, for which the attention mechanism is able to localise signs, to demonstrate the benefits of our approach. We also report the amount of new annotations for both the full 11K vocabulary and the 1064-subset which is used for the proxy recognition evaluation. We observe that a significant number of new automatic sign annotations are obtained with our approach.

To compare the different decoding strategies, we train recognition models on the resulting training sets containing the new annotations and evaluate them on the proxy sign recognition task. Note that for faster training, we learn a 4-layer MLP architecture on top of the pre-extracted I3D video features (architecture and optimisation details are given in the Appendix.

We observe that greedy decoding with the simple filtering mechanism (checking against ground truth) gives best downstream recognition performance on $\text{Test}^{\text{Rec}}_{37K}$. Teacher forcing, beam search and no filtering all yield larger but noisier training sets that result in lower performance. However, we note that the "no subtitle filtering" experiment assumes no access to ground-truth subtitles during annotation mining and uses all the predictions, while providing competitive recognition performance (50.6% vs 53.9%).

Training	#ann.	per-instance top-1	per-instance top-5	per-class top-1	per-class top-5
A	107K	$54.0^{\pm 0.08}$	$67.9^{\pm 0.10}$	$24.8^{\pm 0.10}$	$35.5^{\pm 0.20}$
M [14]†	169K	$40.8^{\pm 0.17}$	$62.2^{\pm 0.07}$	$21.7^{\pm 0.19}$	$38.5^{\pm 0.29}$
M+A	276K	$58.5^{\pm 0.17}$	$75.5^{\pm 0.02}$	$30.4^{\pm 0.04}$	$45.9^{\pm 0.26}$
D [271]†	510K	$62.1^{\pm 0.24}$	$80.8^{\pm 0.10}$	$35.1^{\pm 0.38}$	$54.3^{\pm 0.11}$
D+A	276K	$64.2^{\pm 0.08}$	$81.7^{\pm 0.07}$	$36.0^{\pm 0.26}$	$54.0^{\pm 0.32}$
M+D	678K	$63.5^{\pm 0.28}$	$82.1^{\pm 0.04}$	$37.2^{\pm 0.12}$	$\mathbf{56.4}^{\pm 0.17}$
M+D+A	786K	$\mathbf{65.0}^{\pm 0.14}$	$\mathbf{82.6}^{\pm 0.02}$	$\mathbf{37.9}^{\pm 0.07}$	$56.3^{\pm 0.02}$

Table 10.4: Sign recognition on BSL-1K Test$^{\text{Rec}}_{37K}$: We evaluate our 4-layer MLP classification models trained on video feature inputs for 1064-sign recognition for various training label sets: mouthing (M), dictionary (D), and our proposed attention (A) spottings. We obtain state-of-the-art results, by consistently improving over previous works when including our attention localisations. †The results are obtained from our MLP trained with the annotations from [14] and our application of [271].

10.4.4 Comparison with other automatic annotations

In this section, we train for sign recognition on BSL-1K [14] on various label sets, comparing different automatic annotation methods and showing that our new sign instances are complementary when added to training data, achieving state of the art. As in the previous experiments, we use the MLP architecture on frozen I3D features to compare the different annotation sets. This time we perform 3 trainings per model with different random seeds and report the average and standard deviation.

Tab. 10.4 summarises the results on Test$^{\text{Rec}}_{37K}$. We first note that the MLP performance of M+D annotations matches and slightly outperforms that of I3D from Tab. 10.1 (63.5% vs 62.3%), validating the suitability of MLP for efficiently comparing annotation set quality. When compared to the visual keyword spotting through mouthing (M) [14], our automatic attention localisations (A) show significant improvements. Furthermore, we observe consistent improvements when combining our new annotations with either the mouthing (M+A) or dictionary (D+A) annotations. Combining all available annotations (M+D+A), we achieve state-of-the-art performance (65%) outperforming previous work of [14] (M: 40.8%), as well as a new much stronger baseline (D: 62.1%) that we establish in this work, which uses the new annotations obtained using sign language dictionaries for sign spotting [271]. Our final

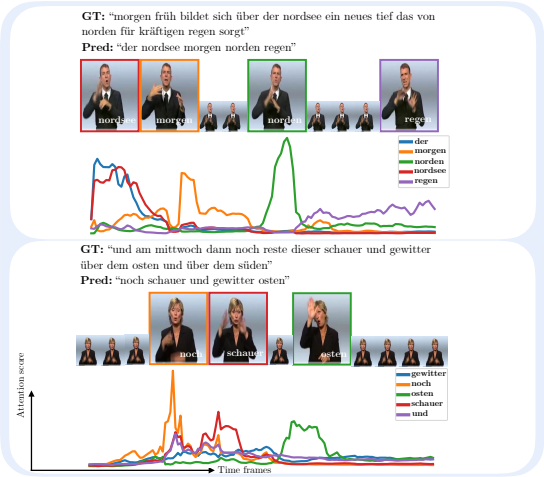


Figure 10.3: Qualitative analysis on the RWTH-PHOENIX: We show example sign localisation results on the test set of RWTH-PHOENIX 2014T. For each video clip, we show the ground-truth sentence as well as the predicted words from the Transformer model of [56] which overlap with the target sentence. We plot attention scores over time frames for these predicted words and show the frame index at which the corresponding attention vector is maximised for a subset of the correctly predicted words.

recognition model can be interpreted as distilling information from multiple sources (mouthing, dictionary, attention), each of which has access to a large training set.

We also evaluate the performance of our MLP trained on M+D+A annotations on the BSL-1K sign spotting benchmark proposed by [14], following their protocol, and achieve a score of 0.174 mAP, outperforming the previous state-of-the-art performance of 0.170 mAP [271] and 0.159 mAP [14].

10.4.5 Qualitative analysis

We demonstrate the potential of our Transformer model to localise sign instances through its attention mechanism. Fig. 10.4 shows qualitative examples of localising multiple signs, by plotting attention scores over video time frames for predicted words that occur in corresponding subtitles of the BSL-1K test set ($\text{Test}^{\text{Loc}}_{7K}$). We observe close alignment with the automatic annotations $\mathcal{N}$. One potential limitation of this approach for localisation is that the attention vector does not peak only at the corresponding sign location, but also on other signs suggesting that the predictions use context (e.g., "smell" and "sweet" in Fig. 10.4, top-left).

We also investigate whether this localisation ability extends to other datasets. In particular,

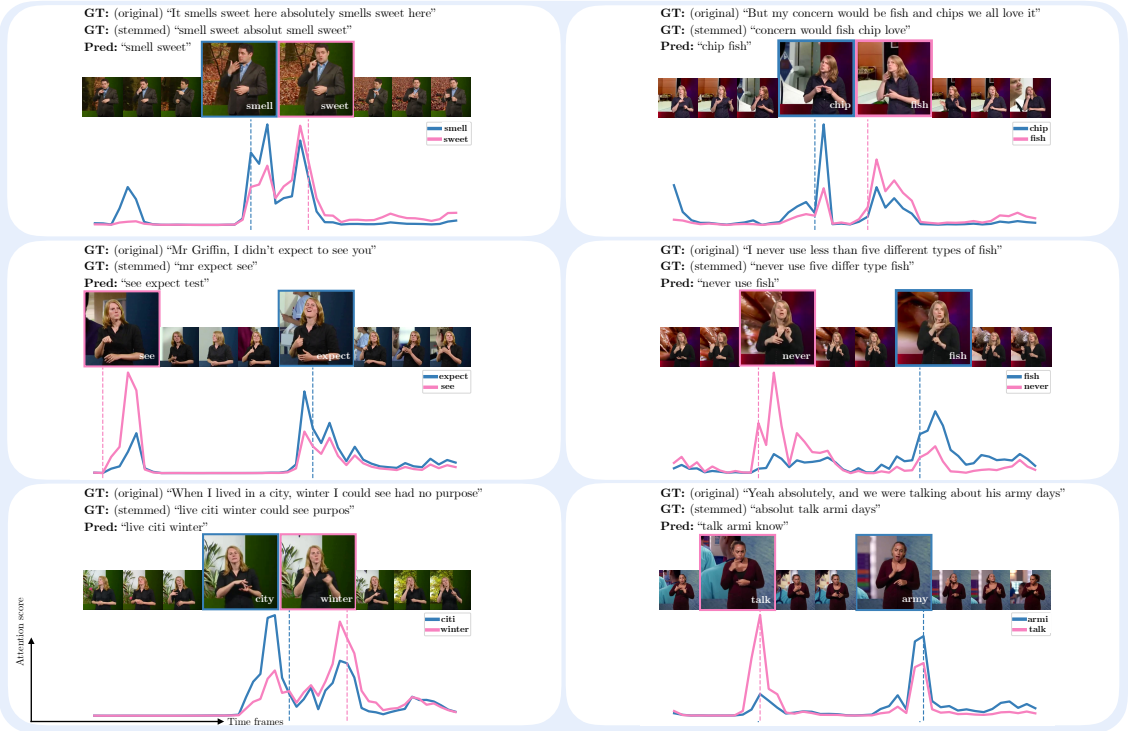


Figure 10.4: Qualitative analysis on BSL-1K: We show example sign localisation results on the BSL-1K test set ($\text{Test}^{\text{Loc}}_{7K}$). For each video clip, we show the original subtitle, the ground-truth stemmed and filtered to 11K vocabulary version, and the prediction of our Transformer model. We plot attention scores over time frames for the predicted words which overlap with the subtitle and for which we have annotated sign times in $\mathcal{N}$ (shown by vertical dashed lines). We highlight the frame at which the corresponding attention vector is maximised.

we reproduce the translation method of Camgöz et al. [56] on RWTH-PHOENIX 2014T [51] and similarly to [51], we visualise the attention score plots for predicted words in Fig. 10.3. We are unable to compute the localisation accuracy as sign annotation times are not available for RWTH-PHOENIX 2014T; however, we observe correct signs when indexing the frame at which the corresponding attention vector is maximised. This suggests that alignment emerges from the attention mechanism also for a full translation system.

10.4.6 Discussion

From our investigations in this work, we believe there are important and challenging problems to be solved before achieving large-vocabulary sign language *translation* from videos to spoken language. First, significantly expanding the coverage of the *vocabulary* of both languages is necessary, and the current state of the art only covers about 3K spoken language and 1K sign lan-

guage vocabularies [56]. In preliminary experiments, we found that a direct application of [56] to translation on the significantly broader vocabulary of 40K contained within the subtitles of BSL-1K failed to converge to meaningful results (for more details see the Appendix). In this work, we have extended to an 11K spoken language vocabulary, but the NLP literature typically works with much larger vocabularies (e.g. a few hundred thousand words [99]). Our attempts to move to 40K words did not obtain sufficient-quality results. Second, the *alignment* between text and video is far from perfect in large-scale sign language datasets which inserts significant amount of noise in training. Our automatic alignment attempts in this work did not obtain improvements. Relying on sparse annotations for approximate alignments limits the amount of data. Third, most of the works, including ours, focus on *interpreted* data, which has certain biases. In fact, the act of interpreting can cause a simplification in signing style and vocabulary, and even lead to a reduction in speed for comprehension [43]. Datasets of native signers should be built to train strong, robust models that generalise at scale and in the wild. Given these observations, we believe that future work that specifically targets translation systems will benefit from addressing these challenges. We refer to the Appendix for a discussion of broader impact.

10.5 Conclusions

We have presented an approach to localise signs in continuous sign language videos with weakly-supervised subtitles by leveraging the attention mechanism of a Transformer model trained on a video-to-text sequence prediction task. We find that state-of-the-art translation models have very low recall on a large-vocabulary dataset, but a satisfactory localisation accuracy through attention that allows us to annotate sign timings. We automatically annotate hundreds of thousands of new signing instances through our learned attention and validate their quality by using them to train a sign language recognition model that surpasses the state of the art on the BSL-1K benchmark as well as a more robust sign language benchmark which is 18 times larger. Future work can leverage our automatic annotations and recognition model for large-vocabulary sign language translation.

Appendices

Appendices for this chapter can be found in the online version of the paper. [2]

Statement of authorship

A statement of authorship for this paper is provided in Appendix A.

[2] https://www.robots.ox.ac.uk/~vgg/research/bslattend/

11 | Aligning Subtitles in Sign Language Videos

Hannah Bull[1*] Triantafyllos Afouras[2*] Gül Varol[2,3]
Samuel Albanie[2] Liliane Momeni[2] Andrew Zisserman[2]

[1] LISN, Univ Paris-Saclay, CNRS, France
[2]Visual Geometry Group, Oxford
[3] LIGM, École des Ponts, Univ Gustave Eiffel, CNRS, France
(* Equal Contribution)

Abstract

The goal of this work is to temporally align asynchronous subtitles in sign language videos. In particular, we focus on sign-language interpreted TV broadcast data comprising (i) a video of continuous signing, and (ii) subtitles corresponding to the audio content. Previous work exploiting such weakly-aligned data only considered finding keyword-sign correspondences, whereas we aim to localise a complete subtitle text in continuous signing. We propose a Transformer architecture tailored for this task, which we train on manually annotated alignments covering over 15K subtitles that span 17.7 hours of video. We use BERT subtitle embeddings and CNN video representations learned for sign recognition to encode the two signals, which interact through a series of attention layers. Our model outputs frame-level predictions, i.e., for each video frame, whether it belongs to the queried subtitle or not. Through extensive evaluations, we show substantial improvements over existing alignment baselines that do not make use of subtitle text embeddings for learning. Our automatic alignment model opens up possibilities for advancing machine translation of sign languages via providing continuously synchronized video-text data.

Published in the proceedings of the International Conference on Computer Vision (ICCV) 2021.

11.1 Introduction

Sign languages constitute a key form of communication for Deaf communities [366]. Our goal in this paper is to temporally localise subtitles in continuous signing video. Automatic alignment of subtitle text to signing content has great potential for a wide range of applications including assistive tools for education and translation, indexing of sign language video corpora,

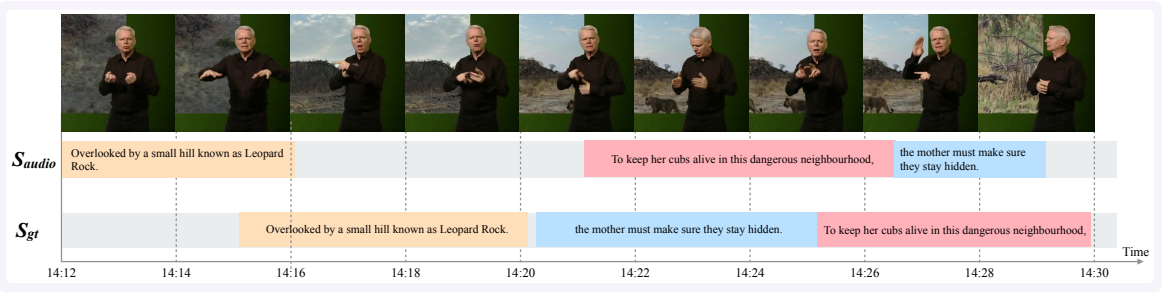


Figure 11.1: Subtitle alignment: We study the task of aligning subtitles to continuous signing in sign language interpreted TV broadcast data. The subtitles in such settings usually correspond to and are aligned with the audio content (top: audio subtitles, S_{audio}) but are unaligned with the accompanying signing (bottom: Ground Truth annotation of the signing corresponding to the subtitle, S_{gt}). This is a *very challenging* task as (i) the *order* of subtitles varies between spoken and sign languages, (ii) the *duration* of a subtitle differs considerably between signing and speech, and (iii) the signing corresponds to a *translation* of the speech as opposed to a transcription.

efficient subtitling technology for signing vloggers[1], and automatic construction of large-scale sign language datasets that support computer vision and linguistic research.

Despite recent advances in computer vision, machine translation between continuous signing and written language remains largely unsolved [43]. Recent works [54, 56] have shown promising translation results, but to date these have been achieved only in *constrained* settings where continuous signing is *manually pre-segmented* into clips, with each clip associated to a written sentence from a *limited vocabulary*. Two key bottlenecks for scaling up translation to continuous signing depicting unconstrained vocabularies are (i) the segmentation of signing into sentence-like units, and (ii) the availability of large-scale sign language training data.

Manual alignment of subtitles to sign language video is tedious – an expert fluent in sign language takes approximately 10-15 hours to align subtitles to 1 hour of continuous sign language video. In this work, we focus on the task of aligning a particular known subtitle within a given temporal signing window. We explore this task in the context of sign language interpreted TV broadcast footage – a readily available and large-scale source of data – where the subtitles are synchronised with the audio, but the corresponding sign language translations are largely unaligned due to differences between spoken and sign languages as well as lags from the live interpretation.

Subtitle alignment to continuous signing remains a *very challenging* task. First, sign languages have grammatical structures that vary considerably from those of spoken languages [366],

[1]Unlike spoken vlogs that benefit from automatic closed captioning on sites such as YouTube, signing vlog creators who wish to provide written subtitles must both translate *and* align their subtitles manually.

and as a result the *ordering* of words within a subtitle as well as the subtitles themselves is often not maintained in the signing (see Fig. 11.1). Second, the *duration* of a subtitle varies considerably between signing and speech due to differences in speed and grammar. Third, the signing corresponds to a *translation* of the speech that appears in the subtitles as opposed to a transcription: there is no direct one-to-one mapping between subtitle words and signs produced by interpreters, and entire subtitles may not be signed.

Previous work exploiting such weakly-aligned data has mainly focused on finding sparse correspondences between keywords in the subtitle and individual signs [14, 271, 383], as opposed to localising the start and end times of a complete subtitle text in continuous signing. Though, as we show, localising isolated signs identified by keyword spotting nevertheless forms a useful pretraining task for full subtitle alignment. Most closely related to our work, Bull et al. [48] consider the task of segmenting a continuous signing video into subtitle units purely based on body keypoints. In fact, similarly to speech which can be segmented based on prosodic cues such as pauses, sign sentence boundaries can *to an extent* be detected through visual cues such as lowering the hands, head movement, pauses, and facial expressions [127]. However, as shown in our evaluations in Sec. 11.4, such approaches based on prosody-only perform poorly in our setting, where subtitles do not necessarily correspond to complete sign sentences with clear visual boundaries.

In this paper, we instead propose to use *the subtitle text as an additional signal* for better alignment. We make the following three contributions: (1) we show that encoding the subtitle text as input to the alignment model significantly improves the temporal localisation quality as opposed to only relying on visual cues to segment continuous sign language videos into subtitle units; (2) we design a novel formulation for the subtitle alignment task based on Transformers; and (3) we present a comprehensive study ablating our design choices and provide promising results for this new task when evaluating on unseen signers and content.

11.2 Related Work

For a recent comprehensive survey about sign language recognition and translation, see [215]. Here, we review relevant works on temporal localisation at the levels of individual signs and sequences, in addition to more general temporal alignment methods from the literature.

Temporal localisation of individual signs. A rich body of work has considered the task of localising sparse sign instances in continuous signing, often referred to as "sign spotting". Early efforts using signing gloves [243] were followed by methods employing hand-crafted visual features to represent the hands, face and motion that were integrated with CRFs [416, 418], HMMs [332] and HSP Trees [291]. Several studies have sought to employ subtitles as weak supervision for learning to localise and classify signs, using apriori mining [95] and multiple-instance learning [46, 47, 306]. More recent work has leveraged cues such as mouthings [14] and visual dictionaries [271] and by making use of deep neural network features with sliding window classifiers [237] and attention learned via a proxy translation task [383]. In deviation from these works, our objective is to localise complete subtitle units, rather than individual signs.

Temporal localisation of sign sequences. The alignment of subtitles to continuous signing was considered in creative early work by combining cues from multiple sparse correspondences [124], but under the assumption that ordering of words in subtitles are preserved in the signing (which does not hold in our problem setting). Other sequence-level sign language temporal localisation tasks that have received attention in the literature include category-agnostic sign segmentation [123, 319], active signer detection [42, 75, 273, 349] and diarisation [13, 147, 148]—each considers a temporal granularity that differs from subtitle units. Most closely related to our work, Bull et al. [48] employ a keypoint-based model to segment continuous signing into sentence-like units without knowledge of the written subtitles during inference. Our approach relaxes this assumption and considers instead the practical scenario in which we assume access to the written subtitle to be aligned. We compare our approach with theirs in Sec. 11.4.

Continuous sign language recognition. Hybrid models coupling CNNs with HMMs [219, 220], attention mechanisms [192] and CTC losses [53, 73] have been studied for continuous

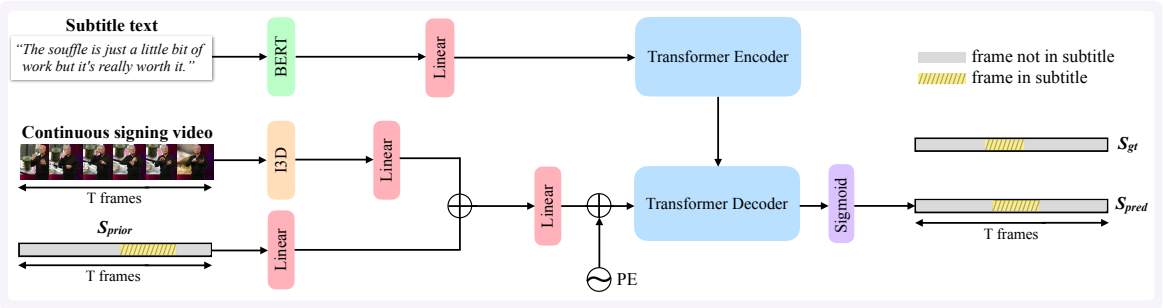


Figure 11.2: SAT model overview: We input to our model (i) token embeddings of the subtitle text we wish to align, (ii) a sequence of video features extracted from a continuous sign language video segment and (iii) the shifted temporal boundaries of the audio-aligned subtitle, S_{prior}. Using these inputs, the model outputs a vector of values between 0 and 1 of length T. Its first and last values above a threshold τ delimit the predicted temporal boundaries for the query subtitle. The location of the subtitle with respect to the window is represented in dashed yellow.

sign language recognition, with recent extensions to sequence-to-sequence models [54] and Transformers [56, 236] to tackle the task of sign language translation. These models produce either implicit or explicit alignments over a signing sequence corresponding to a sentence. However, these approaches have only been demonstrated to work on *pre-segmented* sentences of signing [54].

Aligning bodies of text to video. The Dynamic Time Warping (DTW) algorithm [278] has been applied to the problem of aligning sequences of movies to transcripts [121, 331] and plots synopses [374] using cues such as character recognition and subtitle content. It has also been successfully applied to the problem of aligning generic text descriptions against untrimmed video [41]. While effective, these methods require the preservation of sequence ordering across modalities, which does not hold in our problem setting. We nevertheless show in Sec. 11.3 how DTW can be used as a secondary stage of processing that resolves conflicting local alignments on the re-ordered subtitle prediction timings via a global objective. The fixed ordering assumption is relaxed by the work of [375], which aligns book chapters to video scenes. Their approach, however, which works through matching sparse character identifications against specific shots, is not applicable in our setting where shot boundaries do not provide a natural segmentation of the signing content.

Natural language grounding in videos. Our work is also related to the task of natural language grounding, which aims to locate a temporal segment within an untrimmed video sequence

corresponding to a given natural language query. Existing methods have considered two-stage *propose and rank* approaches [139, 179, 248, 413], iterative grounding agents trained with reinforcement learning [173, 395] and single-stage regression models [70, 151, 427, 430]. Our proposed subtitle alignment task differs from natural language grounding in three ways: (i) The signing content is more *fine-grained*—the visual appearance of a signing sequence remains very similar across frames, necessitating nuanced recognition of body dynamics; (ii) Differently from language grounding, each subtitle to be aligned comes with its own reference location, providing an instance-specific prior over the start time and duration. As we show in Sec. 11.4, our effective use of this reference is important to achieving good performance, and our model is specifically designed to take advantage of this cue; (iii) Subtitles occupy mutually exclusive temporal regions, a property that we further exploit to improve alignment quality, but that does not hold in general for natural language grounding.

11.3 Method

In this section, we describe our Transformer-based subtitle alignment model operating on a single subtitle and a short video segment (Sec. 11.3.1), our pretraining on sparse sign spottings (Sec. 11.3.2), and our final step that globally adjusts multiple subtitles in a long video using DTW (Sec. 11.3.3).

Problem formulation. As inputs to the model, we provide (i) token embeddings of the subtitle text we wish to align to signing, (ii) a sequence of video features extracted from a continuous sign language video segment, as well as (iii) prior estimates of the temporal boundaries for the given query, which we refer to as S_{prior}. The latter is provided as an approximate location and duration cue of the signing-aligned subtitle. Using these inputs, we predict a binary vector of the same length as the video features, where a consecutive sequence of 1s denotes the temporal location of the subtitle.

11.3.1 Subtitle Aligner Transformer

The core of our model is a Transformer [386], as shown in Fig. 11.2, which we refer to as Subtitle Aligner Transformer (SAT). In contrast to the common approach of feeding video frames as input to the encoder [59, 106], we input the video frames to the *decoder* side in order for the model to learn the association between the frame-level features and the output vector of the same duration. We first describe the structure of the Transformer, and then the text and video feature extraction. Additional implementation details are provided in the Appendix.

Encoder. The input to the encoder is a sequence of text embeddings corresponding to the subtitle we wish to align. Positional encodings are not used on the encoder side of the Transformer since the text embeddings (see below) already contain positional information. The encoder is a stack of Transformer layers, each containing a multi-head attention mechanism followed by a feedforward network and embedding dimensionalities of size d_{model}.

Decoder. The decoder is a stack of Transformer layers that attend on the encoded sequence.[2] The input to the decoder consists of a sequence of video features encoding the visual signing information from the video, as well as a binary vector representing a prior estimation of the location of the signing-aligned subtitle (S_{prior}). Positional encodings are added to the decoder input in order for the model to exploit the temporal ordering of the signing. The final layer of the model is a linear layer with a sigmoid activation which outputs T predictions in the range $[0,1]$ one for each video frame. Values of this output vector, S_{pred}, that are above a threshold τ correspond to the predicted temporal location of the queried subtitle text.

Text features. Each subtitle is encoded using a BERT [107] model pretrained on a large text corpus with a masked language modelling task, to produce a sequence of 768-dimensional vectors, one for each token in the sentence. To match the input dimension of the encoder Transformer, these embeddings are first linearly projected to d_{model}.

Video features. The visual features are 1024-dimensional embeddings extracted from the I3D [61] sign classification model made publicly available by the authors of [383]. The

[2]Note: There is no auto-regression.

features are pre-extracted over sign language video segments. A visual feature sequence of length T is used as input to the model.

Prior position encoding. Besides the video features, the input to the decoder also includes a subtitle timing estimate as a prior position and duration cue. This prior estimate is encoded as a binary vector of length T, where 1 indicates that the associated video frame is within the temporal boundaries of the subtitle, and 0 otherwise. The video and prior inputs are fused via concatenation before being passed as input to the decoder. Before the concatenation both inputs are linearly projected to the same dimension. The fusion output is finally projected to d_{model} in order to be input to the Transformer decoder.

Training objective. The model is trained with a binary cross entropy loss between the predicted vector and the ground truth S_{gt} of the signing-aligned subtitle within the video segment:

$$\mathcal{L} = -\frac{1}{T}\sum_{t=1}^{T} S_{gt}^{t} \log S_{pred}^{t} + (1 - S_{gt}^{t}) \log(1 - S_{pred}^{t}).$$

11.3.2 Word pretraining with individual sign locations

SAT is designed for alignment of subtitles to video signing streams. However, the same architecture can be used without any alterations to align smaller text units, e.g. single words. Given that we have access to sparse sign annotations from mouthings [14] and dictionary exemplars [271], we can use these to initialise the model weights and incorporate this knowledge via a potentially easier single-sign spotting task. We obtain timings of the sparse word-level annotations and assume a fixed single-second width as the precise sign boundaries are not available. The model is then trained to spot the single sign occurrence within a video window of size T. In our experiments, we demonstrate the advantages of such a pretraining strategy.

11.3.3 Global alignment with DTW

Our model does not take into account global information from the length of the video (e.g. 1-hour), rather it looks for signing associated to a given subtitle within a short temporal window T (e.g. 20-seconds). Hence, there may be overlaps between predictions for different subtitles; we

resolve these overlap conflicts using DTW [278]. We find an order-preserving global alignment from all elements of a sequence of video frames to all elements of sequence of subtitles, maximising the sum of sigmoid outputs of our model in our cost function for each subtitle query.

As DTW aligns all frames in a video sequence to subtitles, we select all frames of the signing video which are likely to be associated with subtitle queries. Specifically, we select all frames associated to an output score over τ_{dtw}. In the case where our model outputs only values below τ_{dtw} for a particular subtitle, we instead select all frames within the prior location S_{prior}.

We order the subtitles by the mid-point of their predicted temporal location. This allows the predicted subtitles to follow a different order to the original subtitles, because the order of phrases in the sign language interpretation does not necessarily follow the order of phrases of the written English subtitles (see the Appendix for further details).

We construct a cost matrix of dimension (i) the number of frames by (ii) the number of subtitles, and with entries of $1 - p_{ij}$, where p_{ij} is the sigmoid output corresponding to frame i with subtitle j as the encoder input. We apply the DTW algorithm to this cost matrix of aligning video frames to subtitles. This maximises the overall sum of the sigmoid outputs of the model under the ordering and allocation constraints of DTW.

If not otherwise mentioned, our full SAT model uses DTW postprocessing.

11.4 Experiments

In this section, we first give implementation details (Sec. 11.4.1) and describe the datasets and evaluation metrics used in this work (Sec. 11.4.2). We then compare the results of the proposed SAT model against strong baselines (Sec. 11.4.3) and present a series of ablation studies (Sec. 11.4.4). Next, we demonstrate the performance of our model on an additional dataset (Sec. 11.4.5). Finally, we provide qualitative results and discuss limitations (Sec. 11.4.6).

11.4.1 Implementation details

Architecture. For both the encoder and the decoder we use 2 identical Transformer layers with 2 heads and size $d_{model} = 512$ each.

Backbone pretraining. The I3D model is pretrained to perform 1064-way classification across the sign spotting instances with mouthings [14] and dictionary exemplars [271] (further details can be found in [383]). The model is then frozen and used to densely pre-extract visual features with stride 1 over the clips of the datasets.

Prior input selection. As the prior estimate input S_{prior} we use the temporal location of the audio-aligned subtitle S_{audio} shifted by +3.2 seconds. This value, which we denote with S^{+}_{audio}, corresponds to the average temporal shift between the audio-aligned subtitles S_{audio} and the ground truth subtitles S_{gt} in our training data (see Fig. 11.3a).

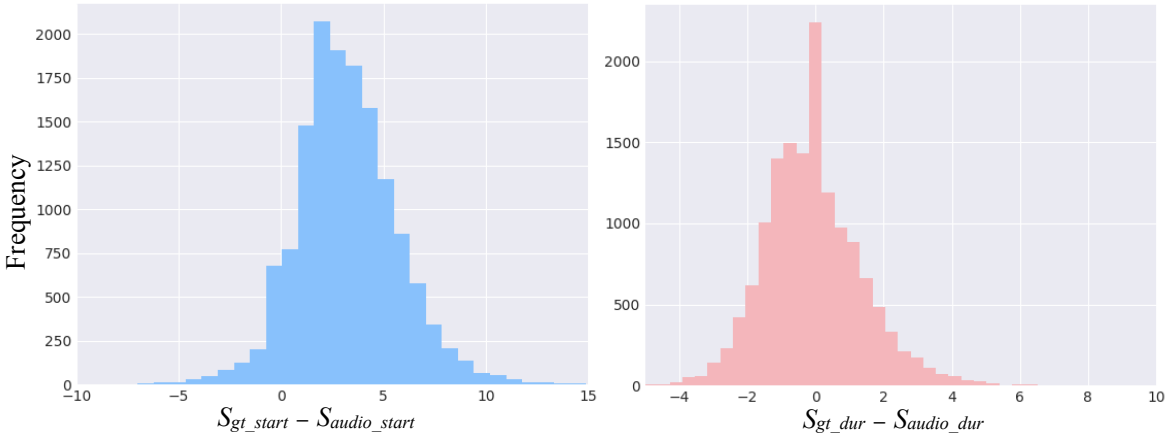
Search windows. During training, we randomly select a search window of 20 seconds around the location of the ground truth subtitle S_{gt}, select the densely extracted video features for this window, and temporally subsample them by a factor of 4. All videos are sampled at 25 FPS, therefore this results in $T = 125$ frames. During testing, we select a search window of the same length centered around the shifted subtitle location S^{+}_{audio}.

Text augmentation. During training, we augment the text query inputs randomly to reduce overfitting: For 50% of the samples, we shuffle the word order and add or delete up to two words.

Hyper-parameters. We set thresholds τ to 0.5, τ_{dtw} to 0.4. Further details are provided in the Appendix.

11.4.2 Data and evaluation metrics

BSL-1K$_{aligned}$ is a subset of BSL-1K [14] which we manually annotated for subtitle alignment. The subset contains 24 episodes covering a number of different television programmes (cooking, nature, travel and reality shows), corresponding to 17.7 hours of BSL content of 3 different signers with 16K subtitles. The subtitles were originally aligned to the audio,



(a) **(b)**

Figure 11.3: S_{gt} vs. S_{audio}: We plot the distribution of temporal shifts between ground-truth (S_{gt}) and audio-aligned (S_{audio}) subtitles on the training split of the BSL-1K$_{aligned}$ dataset by showing the differences in subtitle (a) start times and (b) duration. We observe the difficulty of the subtitle alignment task: (i) there is no fixed shift between ground-truth and audio-aligned subtitle timings, and (ii) the subtitle duration varies between spoken and signed languages.

	#vids.	#hours	#subs	#inst.	Vocab.	OOV
Train	20	14.4	13.8K	128.1K	8.6K	\
Test (total)	4	3.3	2.0K	18.6K	2.8K	726
signer$_{seen}$, genre$_{seen}$	1	0.7	648	6.1K	1.3K	188
signer$_{seen}$, genre$_{unseen}$	1	0.9	465	4.1K	1.0K	233
signer$_{unseen}$, genre$_{seen}$	1	0.7	506	5.6K	1.1K	99
signer$_{unseen}$, genre$_{unseen}$	1	1.0	360	2.8K	882	234

Table 11.1: BSL-1K$_{aligned}$: number of videos, hours, subtitles, word instances, vocabulary size and number of out-of-vocabulary (OOV) words.

	#vids.	#hours	#subs	#inst.	Vocab.	OOV
Train	191	22.9	33.7K	261.5K	7.5K	\
Val	15	1.5	2.6K	18.1K	1.8K	196
Test	21	2.6	3.8K	27.3K	2.4K	369

Table 11.2: BSL Corpus: number of videos, hours, subtitles, word instances, vocabulary size and number of out-of-vocabulary (OOV) words in the dataset's splits.

but we have manually aligned them to the signing. The unaligned subtitles (i.e. those that are synchronised with the audio track, rather than the signing) differ from the signing-aligned subtitles in both start time and duration. In particular, Fig. 11.3, shows that there is no fixed shift or temporal scaling that can be applied to transform audio-synchronised subtitles to their signing-aligned counterparts. We note that the differences exhibit an approximately Gaussian

distribution, with the exception of an accentuated peak at 0 in Fig. 11.3b—if the duration of the subtitle is approximately correct, annotators tend not to further refine the boundaries. The subtitles cover a total of 147K word instances for a vocabulary size of 9.4K in spoken English. We divide the data into 20 training episodes and 4 test episodes. The test episodes are chosen to evaluate the alignment model in different settings: seen/unseen signer and seen/unseen programme genre (which affects the number of out-of-vocabulary words) as shown in Tab. 11.1. The manual alignment of subtitles to signing content for the 24 episodes was performed over approximately 200 hours by native BSL annotators using the open-source VIA tool [114].

BSL Corpus [334, 335] is a public dataset of videos of deaf signers gathered from several regions across the UK and accompanied by a variety of linguistic annotations. For our task, we employ the *FreeTranslation* annotation tier, which provides written English subtitles to accompany portions of the *Conversation* and *Interview* subsets of the corpus. In total, the annotations cover a total of 227 videos after cropping to include a single signer. Of these, 141 are sourced from the *Interview* subset and 86 videos are sourced from the *Conversation* subset. For consistency with prior work, we follow the train, validation and test partition employed by [14, 319]. However, since this partition does not fully span the dataset, we add any dataset instances that were not present in the partition to the training set. Dataset statistics on the resulting train, validation and test partition, including the total number of hours, subtitles and vocabulary spanned by the data, are given in Tab. 11.2. Unlike BSL-1K, the subtitles in this dataset are aligned to signing, and the translation direction is from sign language to English. We therefore simulate unaligned data by perturbing the subtitle locations in our experiments.

Evaluation metrics. We consider two main evaluation metrics: (i) frame-level accuracy, and (ii) $F1$-score. For the $F1$-score, hits and misses of subtitle alignment to sign language video are counted under three temporal overlap thresholds (IoU $\in \{0.1, 0.25, 0.50\}$) between predicted S_{pred} and manually aligned S_{gt} subtitles, denoted as F1@.10, F1@.25, F1@.50, respectively.

11.4.3 Comparison to baselines

Simple temporal shift baseline (S^{+}_{audio}). As a first baseline we use the shifted audio-aligned subtitles S^{+}_{audio}.

Prosodic cues baseline (Bull et al. [48]). We compare to the state of the art on subtitle-unit segmentation, which is a model based on 2D body keypoints. In contrast to our framework, this method only uses visual prosodic cues and does not use semantic information from the query subtitle. It has been trained on a large-scale sign language corpus with aligned subtitles, and the pretrained model is public. The model consists of ST-GCN [415] and BiLSTM layers and segments sign language video into subtitle units. However, this is a different task than alignment, i.e. segments have no correspondence to subtitles. To obtain an association from each predicted segment to a subtitle, we align the shifted subtitles S^{+}_{audio} to a subtitle-unit segmentation of [48] using DTW, where the cost of alignment is the temporal distance.

Heuristic baseline based on sparse sign spottings. Inspired by previous works that approached the alignment task through sparse correspondences [124], we implement a heuristic approach to align the subtitles using a combination of sign spotting and active signer detection. Sign spotting, performed by [14, 271], searches in the temporal vicinity of each audio-synchronised subtitle (the search window is constructed by padding the original subtitle by four seconds at each end) for individual sign instances corresponding to words that appear in the subtitle. From these sparse sign localisations, we perform subtitle alignment in four stages. First, we segment the episode into sequences that contain active signing, following [13]. Second, for any subtitle containing words that were spotted in the signing (assigned a posterior probability of 0.8 or greater by the model of [271]), we shift the subtitle such that its centre falls on the mean position of the spotted signs. Third, we transform all subtitles without spottings by affine transformations such that they fall within the "gaps" between those subtitles that contained spotted signs, while preserving ordering (we use one such transformation per gap). Finally, we expand the duration of subtitles locally (applying a single scaling factor to each subtitle) in left to right ordering, such that they maximally fill the active signing segments predicted by the first stage.

Method	frame-acc	F1@.10	F1@.25	F1@.50
S_{audio}	44.67	45.82	30.51	12.57
S^{+}_{audio}	60.76	71.69	60.74	36.10
Sign-spotting heuristics	61.71	69.23	59.60	36.04
Bull et al. [48]	62.14	73.93	64.25	38.16
SAT (random subtitle)	65.52	70.30	60.36	40.04
SAT w/out DTW	65.81	74.32	64.69	41.27
SAT	**68.72**	**77.80**	**69.29**	**48.15**

Table 11.3: Comparison to baselines: We show significant improvements by training a Subtitle Aligner Transformer (SAT) over several baselines. Moreover, randomly shuffling subtitles obtains poor performance, demonstrating that our model does indeed rely on token embedding, and does not simply learn prosodic cues to align the subtitles. We obtain a further boost by correcting the overlaps of our predicted subtitles using DTW.

A comparison of our model to the above baselines is given in Tab. 11.3. The simple temporal shift baseline and the heuristic baseline based on sparse sign spottings perform similarly, but are a significant improvement over the non-shifted subtitles S_{audio}. Using prosodic cues through the model of [48] results in a slight improvement over these two baselines. Our model significantly outperforms all baselines by exploiting the subtitle text to find the associated video segment. Indeed, when providing random subtitle text during training, our model fails to outperform baseline F1 scores. Using DTW to resolve overlaps in predicted subtitles boosts our model performance.

A breakdown of our results by test episode is provided in Tab. 11.4. Our model tends to result in larger improvements over the S^{+}_{audio} baseline for signers seen in the training episodes, but still outperforms the S^{+}_{audio} baseline for unseen signers in unseen genres. More training data would be needed to better generalise to unseen signers.

11.4.4 Ablation study

We ablate the effects of inputting the prior estimate $S_{prior} = S^{+}_{audio}$ to the model, the size of the search window, modifying the text input to the encoder, pretraining on sign localisation and alternative model formulations. Some additional ablations are presented in the Appendix.

Test episode signer	genre	Method	frame-acc	F1@.10	F1@.25	F1@.50
seen	*seen*	S^+_{audio}	45.48	66.92	55.02	31.84
		SAT	**60.23**	**77.74**	**68.47**	**49.00**
seen	*unseen*	S^+_{audio}	64.31	74.84	64.73	34.19
		SAT	**72.56**	**81.29**	**74.19**	**52.47**
unseen	*seen*	S^+_{audio}	56.30	**80.79**	69.70	44.95
		SAT	**63.68**	80.32	**72.40**	**52.82**
unseen	*unseen*	S^+_{audio}	71.84	63.29	53.16	33.76
		SAT	**74.93**	**69.76**	**59.92**	**34.32**

Table 11.4: Performance breakdown by test episode: Our model improves upon the S^+_{audio} baseline for all the combinations of seen/unseen for signer and genre. The improvements however are greater in the test episodes where the signer has been seen during training.

Knowledge of S_{prior}. We experiment with several versions of inputs as additional information to the alignment task. Tab. 11.5 summarises the results. We first observe a significant drop in performance when S_{prior} is not provided (48.15 vs 30.66 F1@.50), suggesting that the position and duration of the corresponding audio content allows an approximate localisation cue, enabling the model to refine this via a series of attention layers. Inputting the 3.2 seconds shifted subtitle timings ($S_{prior} = S^+_{audio}$) performs better than inputting the audio-aligned subtitle timings ($S_{prior} = S_{audio}$). Moreover, we carry out two additional experiments to investigate whether this cue provides a position prior or a duration prior. First, we always input the subtitle timing centred with respect to the search window. The poor performance of this model suggests the importance of the position. Second, we preserve the shifted location, but randomly change the input subtitle duration at training time by up to 2s. This slightly reduces the performance, therefore duration cues seem less essential for the model than location cues.

Size of the search window T. In Tab. 11.6, we report the performance against different choices for input duration T We conclude that larger search windows generally improve performance, at the cost of computational complexity. This might be due to increased supervision, since with larger windows the training sees more negative examples, as well as due to better coverage at test time. A too short window size inhibits recovery of the correct location, if the correct location falls outside of the window boundaries. In all our experiments, we use 20-second windows.

Additional input	frame-acc	F1@.10	F1@.25	F1@.50
w/out S_{audio}	61.37	59.03	49.35	30.66
w/ S_{audio}	67.81	74.69	66.53	45.10
w/ S^{+}_{audio} 3.2-sec shift	**68.72**	**77.80**	**69.29**	**48.15**
w/ S_{audio} centre position	61.40	58.07	51.13	35.01
w/ S^{+}_{audio} rand. duration	68.61	75.10	66.84	46.72

Table 11.5: Inputting S_{prior} variants: Without information on the approximate position and duration of the subtitle, our model fails to improve upon our baseline methods. In particular, when setting the input S_{prior} to be systematically in the centre of the search window and with the duration of S_{audio}, model performance is poor. When using S^{+}_{audio} in its correct location in the search window, but varying the duration randomly of up to 2s, performance is relatively high. This suggests the position is a stronger cue than duration.

Window size	frame-acc	F1@.10	F1@.25	F1@.50
8 sec	66.98	73.12	64.66	44.13
12 sec	68.63	75.52	67.56	47.29
16 sec	68.51	76.18	68.63	48.10
20 sec	**68.72**	**77.80**	**69.29**	**48.15**

Table 11.6: Search window size T: We vary T between 50 and 125 frames (corresponding to 8- and 20-second inputs, respectively). Larger windows tend to perform better, possibly due to increased contextual information and the fact that the difference between S_{audio} and the aligned subtitle S_{gt} can be in the order of 10s.

Effect of text input to the encoder. We perform a series of ablations regarding the text encoding, including: no text augmentations, adding extra positional encodings to the BERT text features (as described in the Appendix), and using the sentence embedding only (the output embedding corresponding to the BERT “CLS" token) instead of the sequence of individual token embeddings. Tab. 11.7 presents the results on BSL-1K$_{aligned}$ with these text ablations. Augmenting the subtitle text improves performance, while adding extra positional encodings or using the sentence embedding degrades performance.

Effect of sign localisation pretraining. As explained in Sec. 11.3.2, we initially pretrain our model for temporal localisation of individual signs. In Tab. 11.8, we measure the effect of this pretraining on a large set of word-video training pairs, and conclude that it provides a good initialisation for finetuning on long subtitles.

Method	frame-acc	F1@.10	F1@.25	F1@.50
w/o augmentations	67.35	75.72	66.85	45.31
w/ augmentations	**68.72**	**77.80**	**69.29**	**48.15**
w/ aug. + positional enc.	68.21	74.89	67.14	46.36
w/ aug. sentence emb.	66.18	72.99	63.71	41.71

Table 11.7: Text ablations: As a data augmentation step during training, we shuffle the words in 50% of the subtitles and add or delete up to 2 words in the subtitle. This results in a large performance gain. Adding positional encodings to the BERT text features does not improve our model. Using sentence embeddings instead of token embeddings for the subtitle query degrades performance.

Pretraining	frame-acc	F1@.10	F1@.25	F1@.50
w/o word pretraining	67.26	76.18	66.19	42.47
w/ word pretraining	**68.72**	**77.80**	**69.29**	**48.15**

Table 11.8: Pretraining for sign localisation: By pretraining our model to locate individual words within a given temporal window, we boost performance of subtitle alignment.

Model formulation. We consider an alternative version of the Transformer model, inspired by the DETR model in [59] for object detection in images. This model inputs image features into the Transformer encoder and text query into the Transformer decoder. Similarly, we input the sign language video features into the Transformer encoder. On the decoder side, we input the subtitle text features as well as either (i) the start and end times or (ii) the shift and scale of the shifted subtitles S^{+}_{audio} relative to the temporal window. We then consider the problem of subtitle alignment as a regression problem, and aim to predict (i) the start and end times or (ii) the shift and scale of the subtitle relative to the temporal window. As a further ablation, we also consider the same model architecture (with subtitle features and the start and end times as decoder input), but outputting a fixed binary vector of length T, which we train with a binary classification objective (as in SAT).

The results in Tab. 11.9 suggest that our proposed approach with video features as input to the Transformer decoder enables significantly better learning, perhaps by providing a one-to-one mapping between video inputs and the frame-wise outputs. Another possible explanation for our proposed model's superiority is that it outputs alignment scores between subtitles and individual frames which allows for better conflict resolution strategies for overlapping subtitle predictions.

Prior input	Loss	frame-acc	F1@.10	F1@.25	F1@.50
shift/scale	shift/scale regress.	59.23	70.55	59.00	33.71
start/end	start/end regress.	60.04	72.20	60.41	34.33
start/end	binary classif.	60.48	74.05	62.75	35.07
binary	binary classif. (SAT)	**68.72**	**77.80**	**69.29**	**48.15**

Table 11.9: Model formulation: We present an ablation where we experiment with a DETR-style Transformer model [59]. Video features are inputs to the Transformer encoder, and the subtitle query is fed to the Transformer decoder. Moreover, on the decoder side, we input either the start and end times or the shift and scale of the shifted subtitles S^{+}_{audio} relative to the temporal window, and use a regression model to predict the true values. This model fails to produce satisfactory results. Changing the regression model to a classification one by instead predicting a binary vector of length T (as in the SAT model) results in a small improvement; however SAT outperforms all the alternative models with a large margin.

Figure 11.4: Qualitative results: This figure shows short time windows of 5s (left) or 7s (right) with shifted audio-aligned subtitles (S^{+}_{audio}), ground truth signing-aligned subtitles (S_{gt}) and our predicted signing-aligned subtitles (S_{pred}). In practice, we input 20 seconds of video during training and testing as our search window.

11.4.5 Performance on a different dataset

We demonstrate our model's performance on the BSL Corpus [334, 335]. The subtitles in this dataset are aligned to the sign language, and so we randomly shift and scale the subtitles in order to create artificial training data. We then train our SAT model to learn the correct alignment of subtitles to video in the BSL Corpus. We train the model (i) without any pretraining, (ii) with only word pretraining (on BSL-1K) and (iii) with SAT pretraining on BSL-1K$_{aligned}$. We report results in Tab. 11.10.

Rand. perturb. (σ_{pos}, σ_{dur})	Method	frame-acc	F1@.10	F1@.25	F1@.50
(3.5s, 1.5s)	Rand. shift & scale	63.24	37.13	26.54	12.47
	SAT w/out pretrain.	73.73	51.51	43.33	27.98
	SAT pretrain.	75.77	55.55	47.45	32.57
	SAT w/ word pretrain.	**76.29**	**57.65**	**50.35**	**34.54**
(4.5s, 1.5s)	Rand. shift & scale	60.18	29.52	20.61	10.00
	SAT pretrain.	73.69	48.41	41.34	28.06
	SAT w/ word pretrain.	**74.29**	**51.33**	**44.37**	**30.13**
(3.5s, 2s)	Rand. shift & scale	62.62	37.47	26.82	11.87
	SAT pretrain.	75.79	55.31	47.24	32.89
	SAT w/ word pretrain.	**76.00**	**57.86**	**50.43**	**33.79**

Table 11.10: BSL Corpus: We show results on another dataset [334, 335] with subtitles aligned to signing. We randomly shift and scale the correctly aligned subtitles in BSL Corpus to simulate unaligned data and then use our SAT model to recover the original correct alignments. Position is randomly shifted following a normal distribution with standard deviation σ_{pos} and duration is randomly changed according to a normal distribution with standard deviation σ_{dur}. Our model is capable of learning to align subtitles on this data. Word pretraining on BSL-1K increases performance, but pretraining the SAT model on BSL-1K$_{aligned}$ (SAT pretrain.) does not result in further gains.

At each subtitle, we apply a random shift following a normal distribution with standard deviation σ_{pos} and a random change of duration of the subtitle also following a normal distribution with standard deviation σ_{dur}. Tab. 11.10 shows that our model is able to partially recover the correct original alignment. Larger shifts make it more difficult for our model to recover the correct original alignment, but random changes in subtitle duration seems to have less effect. This is consistent with the results in Tab. 11.5, where changing the duration of S^{+}_{audio} does not greatly impact results. Word pretraining on BSL-1K helps the model, but SAT pretraining on BSL-1K$_{aligned}$ does not. Word pretraining may help the SAT model recognise certain signs in BSL, but domain difference between BSL Corpus and BSL-1K$_{aligned}$ subtitles may explain why SAT pretraining on BSL-1K$_{aligned}$ does not lead to any significant gains on BSL Corpus.

11.4.6 Qualitative analysis

Fig. 11.4 illustrates several test examples on BSL-1K$_{aligned}$. The timeline shows the ground truth alignment (S_{gt}), our prediction (S_{pred}), as well as the S^{+}_{audio} baseline, alongside a sample

of video frames and the query subtitle text. While the shifted baseline S^{+}_{audio} provides an approximate position, it is largely unaligned. Our model effectively learns to attend to both visual and textual cues. A typical failure mode happens when the prior position encoding is significantly far from the ground truth (see Fig. 11.4 bottom right). For additional qualitative examples on BSL Corpus, we refer to the Appendix.

11.5 Conclusion

We presented a Transformer-based approach to synchronise subtitles with sign language video content in interpreted data. We showed that knowledge of subtitle content is essential to effectively align subtitles to signing. We hope that our work will be a stepping stone to obtain video-subtitle pairs that allow training of unconstrained machine translation systems for sign languages. Furthermore, our approach is potentially applicable to other domains, such as temporal grounding of sentences. We refer to the Appendix for a discussion on the broader impact on the community.

Appendices

Appendices for this chapter can be found in the online version of the paper. [3]

Statement of authorship

A statement of authorship for this paper is provided in Appendix A.

[3] `https://www.robots.ox.ac.uk/~vgg/research/bslalign/`

12 | Discussion

In this chapter we summarize some of the impact that the work included in this thesis (Section 12.2.2) has had so far, discuss ethical and privacy aspects (Section 12.2), and highlight potential directions for future work (Section 12.3) .

12.1 Impact

Lip Reading and AVSR. LRS2 (Chapter 2) and LRS3([5]) have become the default benchmarks for sentence-level lip reading. Their wide embrace by the research community is reflected by the fact that they have been downloaded over 1000 times in total. Moreover they have been used for training and evaluation on other applications such as lip-sync [311], lip-to-speech synthesis [288], Active Speaker Detection (ASD) [92], and speech enhancement and separation [143, 177].

The TM-seq2seq lip reading model that we have introduced in Section 2 is the base architecture used by many follow-up works to build upon and improve the state-of-the-art in sentence-level lip reading. For example, [435] extend the TM-seq2seq architecture by inserting convolution blocks within the Transformer layers, obtaining improvements in performance. Moreover it serves as the common benchmark architecture to compare new architectures against [412, 435].

To assist the community and enable further research we have open-sourced the code implementation and pretrained models for TM-seq2seq. Indeed, the pretrained lip representations have helped bootstrap various works. For example, several lip reading works have used our pretrained weights as initialization to accelerate their research in lip reading [247, 422, 445] or ASD [438]. These representations were also the basis for work on keyword spotting [270] that in turn facilitated sign spotting from mouthings that enabled a series of works on sign language [13, 14, 271].

Finally, our sequence-to-sequence training curriculum, has been adopted in more general tasks such as learning 3D human dynamics from videos [203, 433].

AV speech enhancement and separation. Our work in audio-visual speech enhancement and separation, described in Chapter 6, has spurred various followup works that extend and improve our proposed models, follow our framework and evaluation protocol, or compare with our models as baselines [163, 193, 202, 234]. For instance Chung *et al.* [91], proposed solving the speech separation task by conditioning on the speakers' visual identities instead of lip motions, while Gao *et al.*[143] took this direction further by adding visual identity cues, learned as cross-modal speaker embeddings, to a lip-conditioned separation model. In a different direction, Yu *et al.* [424] build on our work to create an integrated framework for audio-visual recognition of overlapped speech.

Our speech separation models were also used for the creation of VoxConverse[83], the largest scale freely available audio-visual diarization dataset. Due to its large size and accessibility, VoxConverse is becoming a standard dataset and benchmark for training and evaluating speaker diarization models [231, 232, 394, 409].

Full-frame ASD, synchronization and separation . Our work presented in Chapter 8 resulted in an all-in-one-model solution for a variety of speech related audio-visual tasks – including ASD, audio-visual synchronization and speaker separation – that can be used directly on the frame level, without requiring any prior preprocessing. We have open sourced the code to facilitate research on audio-visual learning and also as a simple alternative to complicated preprocessing pipelines that is easier to use by researchers in less technical fields. For example, it has since been used as a benchmark for active speaker detection [343].

Self-supervised object detection. In Chapter 9 we designed and implemented a method for training object detectors without manual supervision, based on audio-visual correspondence in videos. To the best of our knowledge we are the first to have proposed an entirely self-supervised method for training a full object detector. We expect this line of work to inspire and encourage further research in this exciting new direction.

12.2 Responsible AI

In this Section we summarise and discuss broader privacy and ethical aspects associated with the contributions of this thesis. In particular we discuss (i) the trade-off between the benefits of applications that are of service to society and the risk of malign uses, and (ii) ways to limit the potential for harmful uses, while preserving individual privacy.

We note that the suggestions proposed here are only indicative and that this brief analysis is by no means a comprehensive study; a thorough examination of this important aspect is outside the scope of this thesis and should be extensively conducted in future work.

12.2.1 Trade-off between potential risks and benefits

Risks. The great potential of the methods proposed in this thesis inadvertently raises privacy concerns and creates some risk for malicious applications.

The most commonly raised issue related to lip reading is the potential for malign surveillance, by using video recordings of public places (e.g. CCTV footage) to eavesdrop on private civilian conversations. Similarly, audio-visual speech enhancement could in theory be used for creating enhanced recordings of naturally obfuscated speech in similar scenarios (*e.g.* people discussing in a noisy cafe). Similar risks are also relevant to deaf communities, as further advance in automatic sign language understanding could enable the surveillance of conversations conducted in sign language.

Benefits. Although the above concerns are valid, they should be considered in the context of all the good impact that this research can result in.

In particular, we reiterate that lip reading can be used for enabling speech-impaired individuals (*e.g.* people with aphonia or ALS patients) to improve their communication, either directly with other humans or through human-computer interaction. Audio-visual speech enhancement can be used to facilitate communication with devices (e.g. phones or smart speakers) in

noisy environments such as a car, or enable Zoom calls in crowded places such as train stations and airports.

Audio-visual speech recognition, that is robust to noise, can be used for enabling automatic subtitling in noisy videos, which is important for improving the video understanding experience of the hearing-impaired. Moreover, lip reading and audio-visual speech enhancement could both contribute to the development of even smarter hearing aids (*e.g.* by incorporation into devices such as smart glasses), beyond just removing the distracting noise of the background or other speakers; it could even transcribe the speech to text, and then use text-to-speech to speak it back clearly and at a desired volume to the hard-of-hearing or deaf.

Finally, sign-language recognition has a large range of practical applications, with potential to greatly improve the daily life of deaf and hard-of-hearing individuals. Most importantly it could allow the deaf community to use signing (their own language) to directly communicate with people that do not know how to sign. Other applications that may be made possible include human-computer-interaction through sign language (e.g. for communication with virtual assistants such as Siri and Alexa), the development of sign-language avatars for automatic interpretation of speech into signs, the efficient indexing and searchability of sign-language videos (e.g. on YouTube), or the development of subtitling tools that help signing vloggers align subtitles to their videos.

Trade-off in favour of good uses. Overall we believe that the benefits of potential good uses of our research greatly outweights the risks of malicious ones. Ideally it is desirable to develop such applications, embracing the full potential of these technologies, while simultaneously limiting the risk of malicious exploitation. To that end we suggest a number of privacy preserving practices in the following Section.

12.2.2 Privacy preserving practices

Ensuring adequate user awareness. Techniques for maintaining the privacy of the individuals who do not wish for their communications to be captured and processed, can be borrowed from

common practices in other privacy-sensitive computer vision applications (*e.g.* facial recognition). In the spirit of "notice and consent", users should be informed of the presence of cameras and given the option to opt out of any audio-visual processing of their communications (speech, lip movements, signs). A simple and non-intrusive suggestion in that direction is to require any applications to clearly display a light on the recording device that indicates camera recording and any simultaneous audio-visual processing. A more advanced step is for any application (lip reading, speech enhancement or sign language recognition) to ensure that all the individuals that are being monitored are aware of it and are co-operating, by requiring an active initiation input from users. This could for example be achieved through the use of a wake-word (or phrase) such as "Hey Siri". Depending on the application, this input could be aural, visual or audio-visual. Another way is to limit the functionality of the applications so that they would only work on speakers that are clearly on the foreground, (*i.e.* not background faces), or to require a minimum size for the face of a speaker in order for them to be considered for processing. Those methods are of course not fool-proof but could provide a good starting point for ensuring user privacy.

Further research into understanding limitations. We have mentioned in Chapter 3, that the risk of using CCTV for surveillance using our proposed methods is limited due to the low resolution and frame rate of the cameras usually employed in those systems. Moreover for most applications frontal views from speakers that are aware of being filmed and co-operate is important; deviating from these conditions either completely blocks functionality or greatly hinders performance. In most cases however, we have only qualitatively observed the performance drop that comes from deviating from perfect conditions, and no extensive study to determine the exact extent has been conducted. Therefore, another important step towards privacy is is gaining a better understanding of the limitations of the proposed methods and the conditions under they may function. This will become increasingly important in the future, as high-resolution and fps video footage of public spaces become more easily accessible (e.g. from widespread adoption of smart-glasses usage).

Controlled distribution of code and models. Similar technologies to the ones we have developed are already available to a small handful of corporations (*e.g.*AVSR and lipread-

ing [256], speech separation and enhancement [120]), that have access to enough data and compute resources for training.

We believe that open access is important in order to accelerate progress in the field; in that sense, open release of code and models leads to democratisation of research. However, although we believe that transparent research into this field should be continued and encouraged by the community, we also acknowledge that completely open release can result with those powerful new technologies in the hands of malign users.

One solution to this is constraining the availability of the technologies for research purposes only. For example this is the practice we have followed for data releases (*e.g.*LRS2 and LRS3 datasets). On the one hand this makes the malign applications less likely. On the other hand this may prohibit commercial use that is often necessary for deployment of research output into real products. As this is a broader problem with complicated trade-offs, we believe that the community should reach consensus and eventually propose conventions for the distribution of research.

12.3 Future Work

We conclude this thesis by outlining possible avenues for future works. This includes some open-ended questions on broader objectives and more abstract ideas that might be ripe for exploration in the near future.

Joint cross-modal self-learning for AVSR. In Chapter 4 we presented a method for using a trained ASR systems to exploit the abundance of unannotated speech videos available online, in order to improve lip reading performance.

Although this is a promising framework, it still relies on strong supervision for learning the ASR teacher model. Self-supervised learning has recently had success in pre-training phonetic representations without supervision, which when used for initializing ASR models achieved remarkable performance with very little data[30]. A natural step is therefore to investigate ways of using cross-modal self-supervision to jointly learn audio and video speech representations, suitable for recognition, either with each modality separately, or using both modalities together.

Generalized sound source separation. In Chapter 6 and 7 we proposed a method for separating voices of simultaneous speakers in cocktail party scenarios. This application can be viewed as part of a wider family of methods that perform audio-visual sound source separation. Indeed related works have recently achieved remarkable results for separating the sounds of musical instrument by conditioning on visual input [140, 142, 440, 441]. However generalizing those frameworks to a larger repertoire of object categories and in in-the-wild conditions is a challenging task, yet to be solved. To achieve this it is necessary to learn to associate the appearance and motion of a broad range of objects with their sounds, as well as to model dynamic information for disambiguating between different instances of the same class.

Learning with less dataset curation. We have achieved remarkable results by relying solely self-supervision. For instance in Chapter 8 we showed that it is possible to train complete object detectors without manual labels of any kind, relying on audio-visual correspondence. Under the hood however, those methods were trained on datasets that have been heavily curated. For example, for the creation of the AudioSet and VGGSound datasets, complex pipelines involving audio classifiers and object detectors were used, to clean up the noise and balance the class distribution. On the way towards truly self-supervised training frameworks that operate on in-the-wild videos, rendering those methods more robust to noise is a key challenge.

Discover new connections between appearance, motion, audio and text in videos.. We have seen in Chapters 8 and 9 how audio and speech can be used for obtaining saliency heatmaps on videos or for joint semantic clustering enabling detection. However can we go further than learning about objects individually? For example, when observing dynamic scenes, humans are able to learn a great deal about relationships between objects from motion and sound. Is it possible to teach machines in a similar manner? For instance, is it possible to enable them to learn about relationships between different objects, or about how humans and animals interact with objects, just by watching videos? Some examples of this are illustrated in Figure 12.1. We postulate that multi-modality is again the key to achieve this at a large scale and with minimal supervision.

Another strand to investigate is learning more about the underlying relationships between objects and sounds that temporally co-occur with their appearance or movement. Are there underlying causal relationships or is the co-occurrence a result of spurious correlations? Some

Figure 12.1: Can we learn about the interactions of humans and animals with objects by watching unlabeled videos? Examples of this include (i) learning about the physical properties of the racket and the tennis ball from watching a player hit it, (ii) that the "chopping wood" sound is generated when the axe smashing the log, (iii) that a peculiar sound is the result of a dog chewing on a squeaky toy, or (iv) that a particular sound comes from a woodpecker pecking on a trunk.

examples of different kinds of relationships are shown on Figure 12.1. Humans can easily reason about a situation and infer those relationships because they have prior knowledge about the world, physics, etc. Injecting these priors into the models could perhaps improve performance on current tasks, and vice versa, learning about the world by observing these multi-modal interactions would be even more interesting.

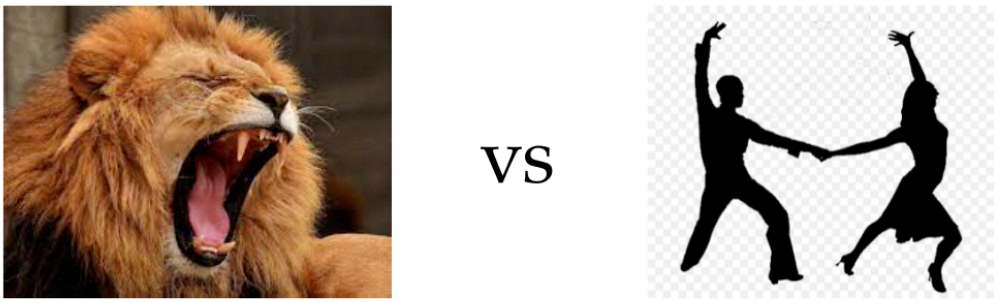


Figure 12.2: Relationship between objects and sounds: causation or co-occurrence? A lion roaring is an example of a causal relationship between object and sound. Dancing is an example of an object (i.e. human) reactively synchronising to a sound.

12.4 Conclusion

In this thesis we developed methods that exploit the natural co-occurrence of audio-visual data in videos for a large range of applications. In particular we studied lip reading and AVSR in Chapters 2 to 5, audio-visual speech enhancement and separation in Chapters 6 and 7, audio-visual object localization and detection in Chapters 8 and 9, and finally sign language recognition in Chapters 10 and 11. Overall, we built on and improved prior methods, pushed the boundaries on existing tasks and set exciting new directions for audio-visual machine learning.

Bibliography

[1] Martın Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*, 2016.

[2] Triantafyllos Afouras, Joon Son Chung, Andrew Senior, Oriol Vinyals, and Andrew Zisserman. Deep audio-visual speech recognition. *IEEE PAMI*, 2019.

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

[4] Triantafyllos Afouras, Joon Son Chung, and Andrew Zisserman. Deep lip reading: a comparison of models and an online application. In *INTERSPEECH*, 2018.

[5] Triantafyllos Afouras, Joon Son Chung, and Andrew Zisserman. LRS3-TED: a large-scale dataset for visual speech recognition. In *arXiv preprint arXiv:1809.00496*, 2018.

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

[7] 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*, 2020.

[8] Triantafyllos Afouras, Joon Son Chung, and Andrew Zisserman. Now you're speaking my language: Visual language identification. In *INTERSPEECH*, 2020.

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

[10] David Alais and David Burr. The ventriloquist effect results from near-optimal bimodal integration. *Current biology*, 14(3):257–262, 2004.

[11] Jean-Baptiste Alayrac, Adrià Recasens, Rosalia Schneider, Relja Arandjelovic, Jason Ramapuram, Jeffrey De Fauw, Lucas Smaira, Sander Dieleman, and Andrew Zisserman. Self-supervised multimodal versatile networks. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

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

[13] Samuel Albanie, Gül Varol, Liliane Momeni, Triantafyllos Afouras, Andrew Brown, Chuhan Zhang, Ernesto Coto, Necati Cihan Camgöz, Ben Saunders, Abhishek Dutta, Neil Fox, Richard Bowden, Bencie Woll, and Andrew Zisserman. Signer diarisation in the wild. *Technical Report*, 2021.

[14] 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 *Proc. ECCV*, 2020.

[15] Ibrahim Almajai and Ben P. Milner. Effective visually-derived wiener filtering for audio-visual speech processing. In *AVSP*, 2009.

[16] Humam Alwassel, Dhruv Mahajan, Bruno Korbar, Lorenzo Torresani, Bernard Ghanem, and Du Tran. Self-supervised learning by cross-modal audio-video clustering. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[17] Dario Amodei, Sundaram Ananthanarayanan, Rishita Anubhai, Jingliang Bai, Eric Battenberg, Carl Case, Jared Casper, Bryan Catanzaro, Qiang Cheng, Guoliang Chen, Jie Chen, Jingdong Chen, Zhijie Chen, Mike Chrzanowski, Adam Coates, Greg Diamos, Ke Ding, Niandong Du, Erich Elsen, and Zhenyao Zhu. Deep speech 2: End-to-end speech recognition in english and mandarin. 12 2015.

[18] MA Anusuya and Shriniwas K Katti. Speech recognition by machine, a review. *arXiv preprint arXiv:1001.2267*, 2010.

[19] R. Arandjelović, P. Gronat, A. Torii, T. Pajdla, and J. Sivic. NetVLAD: CNN architecture for weakly supervised place recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[20] R. Arandjelović and A. Zisserman. Look, listen and learn. In *Proc. ICCV*, 2017.

[21] Relja Arandjelovic and Andrew Zisserman. Look, listen and learn. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2017.

[22] Relja Arandjelovic and Andrew Zisserman. Objects that sound. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.

[23] Aditya Arun, C. V. Jawahar, and M. Pawan Kumar. Dissimilarity coefficient based weakly supervised object detection. In *CVPR*, 2019.

[24] Yuki M. Asano, Mandela Patrick, Christian Rupprecht, and Andrea Vedaldi. Labelling unlabelled videos from scratch with multi-modal self-supervision. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[25] Yuki M. Asano, Christian Rupprecht, and Andrea Vedaldi. Self-labelling via simultaneous clustering and representation learning. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2020.

[26] Yannis M. Assael, Brendan Shillingford, Shimon Whiteson, and Nando de Freitas. Lipnet: Sentence-level lipreading. *arXiv:1611.01599*, 2016.

[27] Yusuf Aytar, Carl Vondrick, and Antonio Torralba. Soundnet: Learning sound representations from unlabeled video. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2016.

[28] Jimmy Ba and Rich Caruana. Do deep nets really need to be deep? In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, pages 2654–2662, 2014.

[29] Kyungjune Baek, Minhyun Lee, and Hyunjung Shim. Psynet: Self-supervised approach to object localization using point symmetric transformation. In *Proc. AAAI*, 2020.

[30] Alexei Baevski, Henry Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations. *arXiv preprint arXiv:2006.11477*, 2020.

[31] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *Proc. ICLR*, 2015.

[32] Shaojie Bai, J. Zico Kolter, and Vladlen Koltun. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271*, 2018.

[33] Z. Barzelay and Y. Y. Schechner. Harmony in motion. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

[34] Timo Baumann, Arne Köhn, and Felix Hennig. The Spoken Wikipedia Corpus collection: Harvesting, alignment and an application to hyperlistening. *Language Resources and Evaluation*, 2018.

[35] Miguel A Bautista, Artsiom Sanakoyeu, Ekaterina Tikhoncheva, and Bjorn Ommer. Cliquecnn: Deep unsupervised exemplar learning. In *NeurIPS*, pages 3846–3854, 2016.

[36] Loris Bazzani, Alessandra Bergamo, Dragomir Anguelov, and Lorenzo Torresani. Self-taught object localization with deep networks. In *2016 IEEE winter conference on applications of computer vision (WACV)*, pages 1–9. IEEE, 2016.

[37] Yonatan Belinkov and Yonatan Bisk. Synthetic and natural noise both break neural machine translation. In *International Conference on Learning Representations*, 2018.

[38] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? In *Proc. ICML*, 2021.

[39] Hakan Bilen, Vinay P. Namboodiri, and Luc Van Gool. Object and action classification with latent variables. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2011.

[40] Hakan Bilen and Andrea Vedaldi. Weakly supervised deep detection networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[41] P. Bojanowski, Rémi Lajugie, E. Grave, Francis R. Bach, I. Laptev, J. Ponce, and C. Schmid. Weakly-supervised alignment of video with text. In *ICCV*, 2015.

[42] M. Borg and K. P. Camilleri. Sign language detection "in the wild" with recurrent neural networks. *ICASSP*, 2019.

[43] Danielle Bragg, Oscar Koller, Mary Bellard, Larwan Berke, Patrick Boudreault, Annelies Braffort, Naomi Caselli, Matt Huenerfauth, Hernisa Kacorri, Tessa Verhoef, Christian Vogler, and Meredith Ringel Morris. Sign language recognition, generation, and translation: An interdisciplinary perspective. In *ACM SIGACCESS*, 2019.

[44] Aparna Brahme and Umesh Bhadade. Lip detection and lip geometric feature extraction using constrained local model for spoken language identification using visual speech recognition. *Indian Journal of Science and Technology*, 9(32):1–7, 2016.

[45] Diane Brentari. Effects of language modality on word segmentation: An experimental study of phonological factors in a sign language. *Papers in laboratory phonology, vol.8*, pages 155–164, 2009.

[46] Patrick Buehler, Mark Everingham, and Andrew Zisserman. Learning sign language by watching TV (using weakly aligned subtitles). In *Proc. CVPR*, 2009.

[47] Patrick Buehler, Mark Everingham, and Andrew Zisserman. Employing signed TV

broadcasts for automated learning of British sign language. In *Workshop on the Representation and Processing of Sign Languages*, 2010.

[48] Hannah Bull, Michèle Gouiffès, and Annelies Braffort. Automatic segmentation of sign language into subtitle-units. In *ECCVW, Sign Language Recognition, Translation and Production (SLRTP)*, 2020.

[49] Weicheng Cai, Jinkun Chen, and Ming Li. Exploring the encoding layer and loss function in end-to-end speaker and language recognition system. In *Speaker Odyssey*, 2018.

[50] Weicheng Cai, Cai Danwei, Shen Huang, and Ming Li. Utterance-level end-to-end language identification using attention-based cnn-blstm. In *ICASSP*, pages 5991–5995, 05 2019.

[51] N. C. Camgoz, S. Hadfield, O. Koller, H. Ney, and R. Bowden. Neural sign language translation. In *CVPR*, 2018.

[52] Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, and Richard Bowden. Subunets: End-to-end hand shape and continuous sign language recognition. In *Proc. ICCV*, pages 22–27, Oct 2017.

[53] Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, and Richard Bowden. SubUNets: End-to-end hand shape and continuous sign language recognition. In *ICCV*, 2017.

[54] Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, Hermann Ney, and Richard Bowden. Neural sign language translation. In *CVPR*, 2018.

[55] Necati Cihan Camgoz, Oscar Koller, Simon Hadfield, and Richard Bowden. Multi-channel transformers for multi-articulatory sign language translation. In *ECCVW*, 2020.

[56] Necati Cihan Camgoz, Oscar Koller, Simon Hadfield, and Richard Bowden. Sign language transformers: Joint end-to-end sign language recognition and translation. In *CVPR*, 2020.

[57] William M Campbell, Joseph P Campbell, Douglas A Reynolds, Elliot Singer, and Pedro A Torres-Carrasquillo. Support vector machines for speaker and language recognition. *Computer Speech & Language*, 20(2-3):210–229, 2006.

[58] Qiong Cao, Li Shen, Weidi Xie, Omkar M. Parkhi, and Andrew Zisserman. VGGFace2: A dataset for recognising faces across pose and age. In *Proc. Int. Conf. Autom. Face and Gesture Recog.*, 2018.

[59] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020.

[60] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *ECCV*, 2018.

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

[62] Punarjay Chakravarty and Tinne Tuytelaars. Cross-modal supervision for learning active speaker detection in video. *arXiv preprint arXiv:1603.08907*, 2016.

[63] William Chan, Navdeep Jaitly, Quoc V. Le, and Oriol Vinyals. Listen, attend and spell.

arXiv.cs, abs/1508.01211, 2015.

[64] William Chan, Navdeep Jaitly, Quoc V. Le, and Oriol Vinyals. Listen, attend and spell: A neural network for large vocabulary conversational speech recognition. In *ICASSP*, 2016.

[65] V. Chandrasekhar, M. Emre Sargin, and D. A. Ross. Automatic language identification in music videos with low level audio and visual features. In *Proc. ICASSP*, 2011.

[66] Ken Chatfield, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Return of the devil in the details: Delving deep into convolutional nets. *arXiv preprint arXiv:1405.3531*, 2014.

[67] Aditya Chattopadhay, Anirban Sarkar, Prantik Howlader, and Vineeth N Balasubramanian. Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 839–847. IEEE, 2018.

[68] Honglie Chen, Weidi Xie, Triantafyllos Afouras, Arsha Nagrani, Andrea Vedaldi, and Andrew Zisserman. Localizing visual sounds the hard way, 2021.

[69] Honglie Chen, Weidi Xie, Andrea Vedaldi, and Andrew Zisserman. Vggsound: A large-scale audio-visual dataset. In *International Conference on Acoustics, Speech, and Signal Processing*, 2020.

[70] Jingyuan Chen, Xinpeng Chen, Lin Ma, Zequn Jie, and Tat-Seng Chua. Temporally grounding natural sentence in video. In *EMNLP*, 2018.

[71] Jinkun Chen, Weicheng Cai, Danwei Cai, Zexin Cai, Haibin Zhong, and Ming Li. End-to-end Language Identification using NetFV and NetVLAD. In *International Symposium on Chinese Spoken Language Processing*, pages 319–323. IEEE, 2018.

[72] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *ICML*, 2020.

[73] Ka Leong Cheng, Zhaoyang Yang, Qifeng Chen, and Yu-Wing Tai. Fully convolutional networks for continuous sign language recognition. In *ECCV*, 2020.

[74] Yong Cheng, Lu Jiang, Wolfgang Macherey, and Jacob Eisenstein. AdvAug: Robust adversarial augmentation for neural machine translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5961–5970, Online, Jul 2020. Association for Computational Linguistics.

[75] N. Cherniavsky, R. E. Ladner, and E. A. Riskin. Activity detection in conversational sign language video for mobile telecommunication. In *IEEE International Conference on Automatic Face Gesture Recognition*, 2008.

[76] Chung-Cheng Chiu, Tara N. Sainath, Yonghui Wu, Rohit Prabhavalkar, Patrick Nguyen, Zhifeng Chen, Anjuli Kannan, Ron J. Weiss, Kanishka Rao, Katya Gonina, Navdeep Jaitly, Bo Li, Jan Chorowski, and Michiel Bacchiani. State-of-the-art speech recognition with sequence-to-sequence models. *CoRR*, abs/1712.01769, 2017.

[77] Kyunghyun Cho, Bart van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder–decoder approaches. In *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Transla-*

tion, pages 103–111, Doha, Qatar, Oct 2014. Association for Computational Linguistics.

[78] 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 *EMNLP*, 2014.

[79] Francois Chollet. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[80] Jan Chorowski, Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. End-to-end continuous speech recognition using attention-based recurrent nn: first results. In *NIPS 2014 Workshop on Deep Learning*, 2014.

[81] Jan K Chorowski, Dzmitry Bahdanau, Dmitriy Serdyuk, Kyunghyun Cho, and Yoshua Bengio. Attention-based models for speech recognition. In *NeurIPS*, pages 577–585, 2015.

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

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

[84] Joon Son Chung, Bong-Jin Lee, and Icksang Han. Who said that?: Audio-visual speaker diarisation of real-world meetings. In *Interspeech*, 2019.

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

[86] Joon Son Chung, Andrew Senior, Oriol Vinyals, and Andrew Zisserman. Lip reading sentences in the wild. In *Proc. CVPR*, 2017.

[87] Joon Son Chung and Andrew Zisserman. Lip reading in the wild. In *Proc. ACCV*, 2016.

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

[89] 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*, 2016.

[90] Joon Son Chung and Andrew Zisserman. Lip reading in profile. In *Proc. BMVC*, 2017.

[91] Soo-Whan Chung, Soyeon Choe, Joon Son Chung, and Hong-Goo Kang. Facefilter: Audio-visual speech separation using still images. *arXiv preprint arXiv:2005.07074*, 2020.

[92] Soo-Whan Chung, Joon Son Chung, and Hong-Goo Kang. Perfect match: Improved cross-modal embeddings for audio-visual synchronisation. In *Proc. ICASSP*, pages 3965–3969. IEEE, 2019.

[93] Ronan Collobert, Christian Puhrsch, and Gabriel Synnaeve. Wav2letter: An end-to-end ConvNet-based speech recognition system. *CoRR*, abs/1609.03193, 2016.

[94] Martin Cooke, Jon Barker, Stuart Cunningham, and Xu Shao. An audio-visual corpus

for speech perception and automatic speech recognition. *The Journal of the Acoustical Society of America*, 120(5):2421–2424, 2006.

[95] Helen Cooper and Richard Bowden. Learning signs from subtitles: A weakly supervised approach to sign language recognition. In *CVPR*, 2009.

[96] Alice Coucke, Mohammed Chlieh, Thibault Gisselbrecht, David Leroy, Mathieu Poumeyrol, and Thibaut Lavril. Efficient keyword spotting using dilated convolutions and gating. In *ICASSP*, 2019.

[97] Ross Cutler and Larry Davis. Look who's talking: Speaker detection using video and audio correlation. In *2000 IEEE International Conference on Multimedia and Expo. ICME2000. Proceedings. Latest Advances in the Fast Changing World of Multimedia (Cat. No. 00TH8532)*, volume 3, pages 1589–1592. IEEE, 2000.

[98] Andrzej Czyzewski, Bozena Kostek, Piotr Bratoszewski, Jozef Kotus, and Marcin Szykulski. An audio-visual corpus for multimodal automatic speech recognition. *Journal of Intelligent Information Systems*, pages 1–26, 2017.

[99] 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 *ACL*, 2019.

[100] Virginia R de Sa. Learning classification with unlabeled data. In *NeurIPS*, pages 112–119, 1994.

[101] Najim Dehak, Pedro A Torres-Carrasquillo, Douglas Reynolds, and Reda Dehak. Language recognition via i-vectors and dimensionality reduction. In *Interspeech*, 2011.

[102] J. Delhumeau, P.-H. Gosselin, H. Jégou, and P. Pérez. Revisiting the VLAD image representation. In *Proc. ACMM*, 2013.

[103] Sabine Deligne, Gerasimos Potamianos, and Chalapathy Neti. Audio-visual speech enhancement with avcdcn (audio-visual codebook dependent cepstral normalization). In *Sensor Array and Multichannel Signal Processing Workshop Proceedings*, 2002.

[104] Chaorui Deng, Qi Wu, Qingyao Wu, Fuyuan Hu, Fan Lyu, and Mingkui Tan. Visual grounding via accumulated attention. In *CVPR*, 2018.

[105] Jiankang Deng, Jia Guo, Zhou Yuxiang, Jinke Yu, Irene Kotsia, and Stefanos Zafeiriou. Retinaface: Single-stage dense face localisation in the wild. In *arxiv*, 2019.

[106] Karan Desai and Justin Johnson. VirTex: Learning visual representations from textual annotations. *arXiv:2006.06666*, 2021.

[107] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv:1810.04805*, 2019.

[108] Terrance DeVries and Graham W. Taylor. Improved regularization of convolutional neural networks with cutout. In *arXiv preprint arXiv:1708.04552*, 2017.

[109] Haisong Ding, Kai Chen, and Qiang Huo. Compression of CTC-Trained Acoustic Models by Dynamic Frame-Wise Distillation or Segment-Wise N-Best Hypotheses Imitation. In *INTERSPEECH*, pages 3218–3222, 2019.

[110] Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised visual representation learning by context prediction. In *Proc. ICCV*, pages 1422–1430, 2015.

[111] 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.

[112] Mohit L. Dubey, Garrett T. Kenyon, Nils Carlson, and Austin Thresher. Does phase matter for monaural source separation? *CoRR*, abs/1711.00913, 2017.

[113] Abhishek Dutta and Andrew Zisserman. The VIA annotation software for images, audio and video. In *Proceedings of the 27th ACM International Conference on Multimedia*, MM '19, New York, NY, USA, 2019. ACM.

[114] Abhishek Dutta and Andrew Zisserman. The via annotation software for images, audio and video. In *Proc. ACMM*, volume 27 of *MM 19*, New York, USA, Oct 2019. ACM, ACM. to appear in Proceedings of the 27th ACM International Conference on Multimedia (MM 19).

[115] Javid Ebrahimi, Daniel Lowd, and Dejing Dou. On adversarial examples for character-level neural machine translation. In *arXiv preprint arXiv:1806.09030*, 2018.

[116] G. Edelman and J. Gally. Reentry: a key mechanism for integration of brain function. *Frontiers in Integrative Neuroscience*, 7, 2013.

[117] Gerald M Edelman. *Neural Darwinism: The theory of neuronal group selection.* Basic books, 1987.

[118] Valentin Emiya, Emmanuel Vincent, Niklas Harlander, and Volker Hohmann. Subjective and objective quality assessment of audio source separation. *IEEE Transactions on Audio, Speech and Language Processing*, 2011.

[119] Ariel Ephrat, Tavi Halperin, and Shmuel Peleg. Improved speech reconstruction from silent video. *ICCV Workshop on Computer Vision for Audio-Visual Media*, 2017.

[120] 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. *SIGGRAPH*, 2018.

[121] M. Everingham, Josef Sivic, and Andrew Zisserman. Hello! my name is... buffy" – automatic naming of characters in tv video. In *BMVC*, 2006.

[122] Johannes Fahringer, Tobias Schrank, Johannes Stahl, Pejman Mowlaee, and Franz Pernkopf. Phase-aware signal processing for automatic speech recognition. In *INTERSPEECH*, 2016.

[123] Iva Farag and Heike Brock. Learning motion disfluencies for automatic sign language segmentation. In *ICASSP*, 2019.

[124] Ali Farhadi and David Forsyth. Aligning ASL for statistical translation using a discriminative word model. In *CVPR*, 2006.

[125] Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman. Convolutional two-stream

network fusion for video action recognition. In *Proc. CVPR*, 2016.

[126] Souheil Fenghour, Daqing Chen, Kun Guo, and Perry Xiao. Lip reading sentences using deep learning with only visual cues. *IEEE Access*, 8:215516–215530, 2020.

[127] J. Fenlon. *Seeing sentence boundaries: the production and perception of visual markers signalling boundaries in signed languages*. PhD thesis, UCL, 2010.

[128] C. Févotte, R. Gribonval, and E. Vincent. BSS EVAL toolbox user guide. *IRISA Technical Report 1706. http://www.irisa.fr/metiss/bss eval/.*, 2005.

[129] Holger Fillbrandt, Suat Akyol, and K-F Kraiss. Extraction of 3D hand shape and posture from image sequences for sign language recognition. In *SOI*, 2003.

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

[131] Ruth Fong and Andrea Vedaldi. Explanations for attributing deep neural network predictions. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2019.

[132] Szu-Wei Fu, Ting-Yao Hu, Yu Tsao, and Xugang Lu. Complex spectrogram enhancement by convolutional neural network with multi-metrics learning. *arXiv preprint arXiv:1704.08504*, 2017.

[133] Aviv Gabbay, Ariel Ephrat, Tavi Halperin, and Shmuel Peleg. Seeing through noise: Visually driven speaker separation and enhancement. 2018.

[134] Aviv Gabbay, Asaph Shamir, and Shmuel Peleg. Visual Speech Enhancement using Noise-Invariant Training. *arXiv preprint arXiv:1711.08789*, 2017.

[135] Raghudeep Gadde, Varun Jampani, and Peter V. Gehler. Semantic video cnns through representation warping. In *Proceedings of the International Conference on Computer Vision (ICCV)*, pages 4463–4472, 2017.

[136] Philip Gage. A new algorithm for data compression. *C Users J.*, 12(2):23–38, Feb 1994.

[137] Georgios Galatas, Gerasimos Potamianos, and Fillia Makedon. Audio-visual speech recognition incorporating facial depth information captured by the kinect. In *Signal Processing Conference (EUSIPCO), 2012 Proceedings of the 20th European*, pages 2714–2717. IEEE, 2012.

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

[139] Jiyang Gao, Chen Sun, Zhenheng Yang, and Ram Nevatia. Tall: Temporal activity localization via language query. In *ICCV*, 2017.

[140] Ruohan Gao, Rogério Schmidt Feris, and Kristen Grauman. Learning to separate object sounds by watching unlabeled video. *CoRR*, abs/1804.01665, 2018.

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

[142] Ruohan Gao and Kristen Grauman. Co-separating sounds of visual objects. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2019.

[143] Ruohan Gao and Kristen Grauman. Visualvoice: Audio-visual speech separation with

cross-modal consistency. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[144] Sarthak Garg, Stephan Peitz, Udhyakumar Nallasamy, and Matthias Paulik. Jointly learning to align and translate with transformer models. In *EMNLP*, 2019.

[145] Weifeng Ge, Sibei Yang, and Yizhou Yu. Multi-evidence filtering and fusion for multi-label classification, object detection and semantic segmentation based on weakly supervised learning. In *CVPR*, 2018.

[146] Jack W Gebhard and G Hamilton Mowbray. On discriminating the rate of visual flicker and auditory flutter. *The American journal of psychology*, 72(4):521–529, 1959.

[147] Binyam Gebrekidan Gebre, Peter Wittenburg, Tom Heskes, and Sebastian Drude. Motion history images for online speaker/signer diarization. In *ICASSP*, 2014.

[148] Binyam Gebrekidan Gebre, Peter Wittenburg, and Tom Heskes. Automatic signer diarization-the mover is the signer approach. In *CVPRW*, 2013.

[149] 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 *Proc. ICASSP*, 2017.

[150] Wang Geng, Wenfu Wang, Yuanyuan Zhao, Xinyuan Cai, Bo Xu, Cai Xinyuan, et al. End-to-end language identification using attention-based recurrent neural networks. *INTERSPEECH*, 2016.

[151] S. Ghosh, A. Agarwal, Zarana Parekh, and A. Hauptmann. Excl: Extractive clip localization using natural language descriptions. In *NAACL-HLT*, 2019.

[152] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. *Proceedings of the International Conference on Learning Representations (ICLR)*, 2018.

[153] Laurent Girin, Jean-Luc Schwartz, and Gang Feng. Audio-visual enhancement of speech in noise. *The Journal of the Acoustical Society of America*, 2001.

[154] R. Goecke, G. Potamianos, and C. Neti. Noisy audio feature enhancement using audio-visual speech data. May 2002.

[155] Nicolas Gonthier, Yann Gousseau, Said Ladjal, and Olivier Bonfait. Weakly supervised object detection in artworks. *Computer Vision – ECCV 2018 Workshops*, page 692–709, 2019.

[156] Javier Gonzalez-Dominguez, David Eustis, Ignacio Lopez-Moreno, Andrew Senior, Françoise Beaufays, and Pedro J Moreno. A real-time end-to-end multilingual speech recognition architecture. *IEEE Journal of selected topics in signal processing*, 9(4):749–759, 2014.

[157] John N Gowdy, Amarnag Subramanya, Chris Bartels, and Jeff Bilmes. Dbn based multi-stream models for audio-visual speech recognition. In *2004 IEEE International conference on acoustics, speech, and signal processing*, volume 1, pages I–993. IEEE, 2004.

[158] Alex Graves. Sequence transduction with recurrent neural networks. *arXiv preprint arXiv:1211.3711*, 2012.

[159] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proc. ICML*, pages 369–376. ACM, 2006.

[160] Alex Graves and Navdeep Jaitly. Towards end-to-end speech recognition with recurrent neural networks. In *Proc. ICML*, pages 1764–1772, 2014.

[161] DW Griffin and Jae S. Lim. Signal estimation from modified short-time fourier transform. In *Proc. ICASSP*, 05 1984.

[162] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Ávila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent: A new approach to self-supervised learning. *CoRR*, abs/2006.07733, 2020.

[163] Rongzhi Gu, Shi-Xiong Zhang, Yong Xu, Lianwu Chen, Yuexian Zou, and Dong Yu. Multi-modal multi-channel target speech separation. *IEEE Journal of Selected Topics in Signal Processing*, 14(3):530–541, 2020.

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

[165] Saurabh Gupta, Judy Hoffman, and Jitendra Malik. Cross modal distillation for supervision transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[166] Shir Gur, Ameen Ali, and Lior Wolf. Visualization of supervised and self-supervised neural networks via attribution guided factorization, 2020.

[167] Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 297–304. JMLR Workshop and Conference Proceedings, 2010.

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

[169] Tengda Han, Weidi Xie, and Andrew Zisserman. Memory-augmented dense predictive coding for video representation learning. In *Proc. ECCV*, 2020.

[170] Tengda Han, Weidi Xie, and Andrew Zisserman. Self-supervised co-training for video representation learning. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[171] 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 (ECCV)*, pages 649–665, 2018.

[172] David F. Harwath, Antonio Torralba, and James R. Glass. Unsupervised learning of spoken language with visual context. In *NIPS*, 2016.

[173] D. He, Xiang Zhao, Jizhou Huang, F. Li, Xiao Liu, and Shilei Wen. Read, watch, and move: Reinforcement learning for temporally grounding natural language descriptions in videos. In *AAAI*, 2019.

[174] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. *CVPR*, 2020.

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

[176] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2016.

[177] Sindhu B Hegde, KR Prajwal, Rudrabha Mukhopadhyay, Vinay P Namboodiri, and CV Jawahar. Visual speech enhancement without a real visual stream. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1926–1935, 2021.

[178] Olivier J Hénaff, Aravind Srinivas, Jeffrey De Fauw, Ali Razavi, Carl Doersch, SM Eslami, and Aaron van den Oord. Data-efficient image recognition with contrastive predictive coding. *ICML*, 2020.

[179] Lisa Anne Hendricks, O. Wang, E. Shechtman, Josef Sivic, Trevor Darrell, and Bryan C. Russell. Localizing moments in video with natural language. In *ICCV*, 2017.

[180] J Hershey and JR Movellan. Audio-vision: Locating sounds via audio-visual synchrony. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, volume 12, 1999.

[181] John Hershey, Hagai Attias, Nebojsa Jojic, and Trausti Kristjansson. Audio-visual graphical models for speech processing. In *Proc. ICASSP*, 2004.

[182] John R Hershey and Michael Casey. Audio-visual sound separation via hidden markov models. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2002.

[183] John R Hershey, Zhuo Chen, Jonathan Le Roux, and Shinji Watanabe. Deep clustering: Discriminative embeddings for segmentation and separation. In *Proc. ICASSP*, pages 31–35. IEEE, 2016.

[184] Geoffrey Hinton, Li Deng, Dong Yu, George Dahl, Abdel-Rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Brian Kingsbury, and Tara Sainath. Deep neural networks for acoustic modeling in speech recognition. *IEEE Signal Processing Magazine*, 29:82–97, November 2012.

[185] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. In *NIPS Deep Learning and Representation Learning Workshop*, 2015.

[186] Hans-Günter Hirsch and Michael Gref. On the influence of modifying magnitude and phase spectrum to enhance noisy speech signals. In *INTERSPEECH*, 2017.

[187] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[188] Jen-Cheng Hou, Syu-Siang Wang, Ying-Hui Lai, Yu Tsao, Hsiu-Wen Chang, and Hsin-Min Wang. Audio-Visual Speech Enhancement Using Multimodal Deep Convolutional Neural Networks. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 2018.

[189] Di Hu, Feiping Nie, and Xuelong Li. Deep multimodal clustering for unsupervised audiovisual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[190] Di Hu, Rui Qian, Minyue Jiang, Xiao Tan, Shilei Wen, Errui Ding, Weiyao Lin, and Dejing Dou. Discriminative sounding objects localization via self-supervised audiovisual matching. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[191] Di Hu, Zongge Wang, Haoyi Xiong, Dong Wang, Feiping Nie, and Dejing Dou. Curriculum audiovisual learning. *ArXiv*, abs/2001.09414, 2020.

[192] Jie Huang, Wengang Zhou, Qilin Zhang, Houqiang Li, and Weiping Li. Video-based sign language recognition without temporal segmentation. In *AAAI*, 2018.

[193] Koichiro Ito, Masaaki Yamamoto, and Kenji Nagamatsu. Audio-visual speech enhancement method conditioned in the lip motion and speaker-discriminative embeddings. In *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6668–6672, 2021.

[194] Hamid Izadinia, Imran Saleemi, and Mubarak Shah. Multimodal analysis for identification and segmentation of moving-sounding objects. *IEEE Transactions on Multimedia*, 15(2):378–390, 2012.

[195] Saumya Jetley, Nicholas A Lord, Namhoon Lee, and Philip HS Torr. Learn to pay attention. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2018.

[196] Abhishek Jha, Vinay Namboodiri, and C. Jawahar. Spotting words in silent speech videos: a retrieval-based approach. *Machine Vision and Applications*, 30, 03 2019.

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

[198] Zequn Jie, Yunchao Wei, Xiaojie Jin, Jiashi Feng, and Wei Liu. Deep self-taught learning for weakly supervised object localization. In *CVPR*, 2017.

[199] Zhaozhang Jin and DeLiang Wang. A supervised learning approach to monaural segregation of reverberant speech. *IEEE Transactions on Audio, Speech, and Language Processing*, 2009.

[200] Benjamin Johnston and Philip Chazal. A review of image-based automatic facial landmark identification techniques. *EURASIP Journal on Image and Video Processing*, 2018:86, 09 2018.

[201] Hamid Reza Vaezi Joze and Oscar Koller. MS-ASL: A large-scale data set and benchmark for understanding American Sign Language. In *BMVC*, 2019.

[202] Hamid Reza Vaezi Joze, Amirreza Shaban, Michael L Iuzzolino, and Kazuhito Koishida.

MMTM: multimodal transfer module for CNN fusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13289–13299, 2020.

[203] Angjoo Kanazawa, Jason Y Zhang, Panna Felsen, and Jitendra Malik. Learning 3d human dynamics from video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5614–5623, 2019.

[204] Anjuli Kannan, Yonghui Wu, Patrick Nguyen, Tara N. Sainath, Zhifeng Chen, and Rohit Prabhavalkar. An analysis of incorporating an external language model into a sequence-to-sequence model. In *Proc. ICASSP*, 2018.

[205] Faheem Khan and Ben Milner. Speaker separation using visually-derived binary masks. In *AVSP*, 2013.

[206] Naji Khosravan, Shervin Ardeshir, and Rohit Puri. On attention modules for audio-visual synchronization. *arXiv preprint arXiv:1812.06071*, 1, 2018.

[207] Einat Kidron, Yoav Y Schechner, and Michael Elad. Pixels that sound. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.

[208] Ho-Gyeong Kim, Hwidong Na, Hoshik Lee, Jihyun Lee, Tae Gyoon Kang, Min-Joong Lee, and Young Sang Choi. Knowledge distillation using output errors for self-attention end-to-end models. In *Proc. ICASSP*, pages 6181–6185. IEEE, 2019.

[209] Suyoun Kim, Michael L. Seltzer, Jinyu Li, and Rui Zhao. Improved training for online end-to-end speech recognition systems, 2017.

[210] Yoon Kim and Alexander M Rush. Sequence-level knowledge distillation. In *Proc. EMNLP*, 2016.

[211] Davis E King. Dlib-ml: A machine learning toolkit. *The Journal of Machine Learning Research*, 10:1755–1758, 2009.

[212] Diederik P Kingma and Jimmy Ba. ADAM: A method for stochastic optimization. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2015.

[213] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *Proc. ICLR*, 2014.

[214] Sang-Ki Ko, Chang Jo Kim, Hyedong Jung, and Choongsang Cho. Neural sign language translation based on human keypoint estimation. *Appl. Sci.*, 2019.

[215] Oscar Koller. Quantitative survey of the state of the art in sign language recognition. *arXiv:2008.09918*, 2020.

[216] Oscar Koller, Jens Forster, and Hermann Ney. Continuous sign language recognition: Towards large vocabulary statistical recognition systems handling multiple signers. *Computer Vision and Image Understanding*, 141:108–125, 2015.

[217] Oscar Koller, Hermann Ney, and Richard Bowden. Deep learning of mouth shapes for sign language. In *Proc. CVPR*, pages 85–91, 2015.

[218] Oscar Koller, Hermann Ney, and Richard Bowden. Deep learning of mouth shapes for sign language. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[219] Oscar Koller, O Zargaran, Hermann Ney, and Richard Bowden. Deep sign: Hybrid CNN-HMM for continuous sign language recognition. In *BMVC*, 2016.

[220] Oscar Koller, Sepehr Zargaran, and Hermann Ney. Re-sign: Re-aligned end-to-end sequence modelling with deep recurrent CNN-HMMs. In *CVPR*, 2017.

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

[222] Bruno Korbar, Du Tran, and Lorenzo Torresani. Cooperative learning of audio and video models from self-supervised synchronization. 2018.

[223] Alexandros Koumparoulis, Gerasimos Potamianos, Youssef Mroueh, and Steven J. Rennie. Exploring roi size in deep learning based lipreading. In *Proc. The 14th International Conference on Auditory-Visual Speech Processing*, pages 64–69, 2017.

[224] A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet classification with deep convolutional neural networks. In *NeurIPS*, pages 1106–1114, 2012.

[225] Oleksii Kuchaiev, Boris Ginsburg, Igor Gitman, Vitaly Lavrukhin, Jason Li, Huyen Nguyen, Carl Case, and Paulius Micikevicius. Mixed-Precision Training for NLP and Speech Recognition with OpenSeq2Seq, 2018.

[226] P. Kuhl and A. Meltzoff. The bimodal perception of speech in infancy. *Science*, 218 4577:1138–41, 1982.

[227] Patricia Kuhl. A new view of language acquisition. *Proceedings of the National Academy of Sciences of the United States of America*, 97:11850–7, 11 2000.

[228] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.

[229] Gakuto Kurata and Kartik Audhkhasi. Guiding CTC Posterior Spike Timings for Improved Posterior Fusion and Knowledge Distillation. In *INTERSPEECH*, 2019.

[230] 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.

[231] Youngki Kwon, Hee Soo Heo, Jaesung Huh, Bong-Jin Lee, and Joon Son Chung. Look who's not talking. In *2021 IEEE Spoken Language Technology Workshop (SLT)*, pages 567–573. IEEE, 2021.

[232] Federico Landini, Ondřej Glembek, Pavel Matějka, Johan Rohdin, Lukáš Burget, Mireia Diez, and Anna Silnova. Analysis of the but diarization system for voxconverse challenge. In *arXiv preprint arXiv:2010.11718*, 2021.

[233] Roger Lass. *Phonology: An introduction to basic concepts*. Cambridge University Press, 1984.

[234] Jiyoung Lee, Soo-Whan Chung, Sunok Kim, Hong-Goo Kang, and K. Sohn. Looking into your speech: Learning cross-modal affinity for audio-visual speech separation. *ArXiv*, abs/2104.02775, 2021.

[235] Dongxu Li, Cristian Rodriguez Opazo, Xin Yu, and Hongdong Li. Word-level deep sign

language recognition from video: A new large-scale dataset and methods comparison. In *WACV*, 2019.

[236] Dongxu Li, Chenchen Xu, Xin Yu, K. Zhang, Ben Swift, Hanna Suominen, and H. Li. TSPNet: Hierarchical feature learning via temporal semantic pyramid for sign language translation. *NeurIPS*, 2020.

[237] Dongxu Li, Xin Yu, Chenchen Xu, Lars Petersson, and Hongdong Li. Transferring cross-domain knowledge for video sign language recognition. In *CVPR*, 2020.

[238] Jason Li, Vitaly Lavrukhin, Boris Ginsburg, Ryan Leary, Oleksii Kuchaiev, Jonathan M. Cohen, Huyen Nguyen, and Ravi Teja Gadde. Jasper: An end-to-end convolutional neural acoustic model. In *INTERSPEECH*, 2019.

[239] Jinyu Li, Rui Zhao, Jui-Ting Huang, and Yifan Gong. Learning small-size DNN with output-distribution-based criteria. In *INTERSPEECH*, 2014.

[240] Tianhao Li and Limin Wang. Learning spatiotemporal features via video and text pair discrimination. *arXiv.cs*, abs/2001.05691, 2020.

[241] Wei Li, Sicheng Wang, Ming Lei, Sabato Marco Siniscalchi, and Chin-Hui Lee. Improving audio-visual speech recognition performance with cross-modal student-teacher training. In *Proc. ICASSP*, pages 6560–6564. IEEE, 2019.

[242] Xiaoyan Li, Meina Kan, Shiguang Shan, and Xilin Chen. Weakly supervised object detection with segmentation collaboration. In *cvpr*, 2019.

[243] Rung-Huei Liang and Ming Ouhyoung. A real-time continuous gesture recognition system for sign language. In *Proceedings third IEEE international conference on automatic face and gesture recognition*, 1998.

[244] R. Lienhart. Reliable transition detection in videos: A survey and practitioner's guide. *International Journal of Image and Graphics*, Aug 2001.

[245] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2014.

[246] Vitaliy Liptchinsky, Gabriel Synnaeve, and Ronan Collobert. Letter-based speech recognition with gated ConvNets. *CoRR*, abs/1712.09444, 2017.

[247] Jinglin Liu, Yi Ren, Zhou Zhao, Chen Zhang, Baoxing Huai, and Jing Yuan. Fastlr: Non-autoregressive lipreading model with integrate-and-fire. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 4328–4336, 2020.

[248] Meng Liu, Xiang Wang, Liqiang Nie, Xiangnan He, Baoquan Chen, and Tat-Seng Chua. Attentive moment retrieval in videos. In *ACM SIGIR*, 2018.

[249] Qingju Liu, Wenwu Wang, Philip JB Jackson, Mark Barnard, Josef Kittler, and Jonathon Chambers. Source separation of convolutive and noisy mixtures using audio-visual dictionary learning and probabilistic time-frequency masking. *IEEE Transactions on Signal Processing*, 2013.

[250] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *Proc.*

ECCV, pages 21–37. Springer, 2016.

[251] Karen Livescu, Ozgur Cetin, Mark Hasegawa-Johnson, Simon King, Chris Bartels, Nash Borges, Arthur Kantor, Partha Lal, Lisa Yung, Ari Bezman, et al. Articulatory feature-based methods for acoustic and audio-visual speech recognition: Summary from the 2006 jhu summer workshop. In *2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP'07*, volume 4, pages IV–621. IEEE, 2007.

[252] B. D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proc. of the 7th International Joint Conference on Artificial Intelligence*, pages 674–679, 1981.

[253] Pingchuan Ma, Stavros Petridis, and Maja Pantic. End-to-end audio-visual speech recognition with conformers. In *ICASSP*, 2021.

[254] Andrew L. Maas, Ziang Xie, Dan Jurafsky, and Andrew Y. Ng. Lexicon-free conversational speech recognition with neural networks. In *Proceedings the North American Chapter of the Association for Computational Linguistics*, 2015.

[255] Shoji Makino, Te-Won Lee, and Hiroshi Sawada. *Blind speech separation*. Springer, 2007.

[256] Takaki Makino, Hank Liao, Yannis Assael, Brendan Shillingford, Basilio Garcia, Otavio Braga, and Olivier Siohan. Recurrent neural network transducer for audio-visual speech recognition. In *IEEE Workshop on Automatic Speech Recognition and Understanding*, 2019.

[257] Etienne Marcheret, Gerasimos Potamianos, Josef Vopicka, and Vaibhava Goel. Detecting audio-visual synchrony using deep neural networks. In *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[258] David Martinez, Oldřich Plchot, Lukáš Burget, Ondřej Glembek, and Pavel Matějka. Language recognition in ivectors space. In *Interspeech*, 2011.

[259] Hanna Mazzawi, Xavi Gonzalvo, Aleks Kracun, Prashant Sridhar, Niranjan Subrahmanya, Ignacio Lopez Moreno, Hyun Jin Park, and Patrick Violette. Improving keyword spotting and language identification via neural architecture search at scale. In *INTERSPEECH*, 2019.

[260] Michael McAuliffe, Michaela Socolof, Sarah Mihuc, M. Wagner, and Morgan Sonderegger. Montreal forced aligner: Trainable text-speech alignment using kaldi. In *INTERSPEECH*, 2017.

[261] Harry McGurk and John MacDonald. Hearing lips and seeing voices. *Nature*, 264, 1976.

[262] Michael McTear. Book reviews : Language acquisition in the blind child: Normal and deficient. ed. by a. mills (london: Croom helm, 1983). pp.235. £13.95. isbn 0 7099 1768 6. *First Language*, 5(14):161–165, 1984.

[263] Xiaoxiao Miao, Ian McLoughlin, and Yonghong Yan. A new time-frequency attention mechanism for tdnn and cnn-lstm-tdnn, with application to language identification. *Interspeech*, pages 4080–4084, 2019.

[264] Paul Michel, Xian Li, Graham Neubig, and Juan Miguel Pino. On evaluation of adversarial perturbations for sequence-to-sequence models. In *arXiv preprint*

arXiv:1903.06620, 2019.

[265] Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. End-to-end learning of visual representations from uncurated instructional videos. *arXiv.cs*, abs/1912.06430, 2019.

[266] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. HowTo100M: Learning a text-video embedding by watching hundred million narrated video clips. *arXiv.cs*, abs/1906.03327, 2019.

[267] Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. In *CVPR*, 2020.

[268] Mendelson Mj and Hath Mm. The relation between audition and vision in the human newborn. *Monographs of The Society for Research in Child Development*, 41:1–72, 1976.

[269] Hossein Mobahi, Ronan Collobert, and Jason Weston. Deep learning from temporal coherence in video. In *Proc. ICML*, pages 737–744, 2009.

[270] Liliane Momeni, Triantafyllos Afouras, Themos Stafylakis, Samuel Albanie, and Andrew Zisserman. Seeing wake words: Audio-visual keyword spotting. In *Proc. BMVC*, 2020.

[271] Liliane Momeni, Gül Varol, Samuel Albanie, Triantafyllos Afouras, and Andrew Zisserman. Watch, read and lookup: Learning to spot signs from multiple supervisors. In *Proc. ACCV*, 2020.

[272] Pedro Morgado, Yi Li, and Nuno Vasconcelos. Learning representations from audio-visual spatial alignment. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[273] Amit Moryossef, Ioannis Tsochantaridis, Roee Aharoni, Sarah Ebling, and Srini Narayanan. Real-Time Sign Language Detection using Human Pose Estimation. In *ECCVW, Sign Language Recognition, Translation and Production (SLRTP)*, 2020.

[274] Pejman Mowlaee. On speech intelligibility estimation of phase-aware single-channel speech enhancement. In *ICASSP*, 2015.

[275] Pejman Mowlaee and Josef Kulmer. Phase estimation in single-channel speech enhancement: Limits-potential. *IEEE Transactions on Audio, Speech and Language Processing*, 2015.

[276] Pejman Mowlaee, Rahim Saeidi, and Yannis Stylianou. Advances in phase-aware signal processing in speech communication. *Speech Communication Elsevier*, 2016.

[277] Youssef Mroueh, Etienne Marcheret, and Vaibhava Goel. Deep multimodal learning for audio-visual speech recognition. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2130–2134. IEEE, 2015.

[278] C. Myers and L. Rabiner. A comparative study of several dynamic time-warping algorithms for connected-word recognition. *The Bell System Technical Journal*, 60:1389–1409, 1981.

[279] M. Müller, S. Stüker, and A. Waibel. Neural codes to factor language in multilingual

speech recognition. In *Proc. ICASSP*, 2019.

[280] Arsha Nagrani, Joon Son Chung, Samuel Albanie, and Andrew Zisserman. Disentangled speech embeddings using cross-modal self-supervision. In *Proc. ICASSP*, pages 6829–6833. IEEE, 2020.

[281] 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 (CVPR)*, 2020.

[282] J. L. Newman and S. J. Cox. Language identification using visual features. *IEEE Transactions on Audio, Speech, and Language Processing*, 20(7):1936–1947, 2012.

[283] Jacob Newman and Stephen Cox. Speaker independent visual-only language identification. In *Proc. ICASSP*, pages 5026–5029, 01 2010.

[284] Minh Hoai Nguyen, Lorenzo Torresani, Lorenzo de la Torre, and Carsten Rother. Weakly supervised discriminative localization and classification: a joint learning process. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2009.

[285] Kuniaki Noda, Yuki Yamaguchi, Kazuhiro Nakadai, Hiroshi G Okuno, and Tetsuya Ogata. Lipreading using convolutional neural network. In *INTERSPEECH*, pages 1149–1153, 2014.

[286] Kuniaki Noda, Yuki Yamaguchi, Kazuhiro Nakadai, Hiroshi G Okuno, and Tetsuya Ogata. Audio-visual speech recognition using deep learning. *Applied Intelligence*, 42(4):722–737, 2015.

[287] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *Proc. ECCV*, pages 69–84. Springer, 2016.

[288] Dan Oneata, Adriana Stan, and Horia Cucu. Speaker disentanglement in video-to-speech conversion. *arXiv preprint arXiv:2105.09652*, 2021.

[289] E. Ong, H. Cooper, N. Pugeault, and R. Bowden. Sign language recognition using sequential pattern trees. In *CVPR*, 2012.

[290] Eng-Jon Ong and Richard Bowden. Learning temporal signatures for lip reading. In *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, pages 958–965, 2011.

[291] Eng-Jon Ong, Oscar Koller, Nicolas Pugeault, and Richard Bowden. Sign spotting using hierarchical sequential patterns with temporal intervals. In *CVPR*, 2014.

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

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

[294] 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 (CVPR)*, 2016.

[295] Andrew Owens, Jiajun Wu, Josh H. McDermott, William T. Freeman, and Antonio

Torralba. Ambient sound provides supervision for visual learning. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2016.

[296] Andrew Owens, Jiajun Wu, Josh H. McDermott, William T. Freeman, and Antonio Torralba. Learning sight from sound: Ambient sound provides supervision for visual learning. *International Journal of Computer Vision*, 2018.

[297] Bharat Padi, Anand Mohan, and Sriram Ganapathy. Towards relevance and sequence modeling in language recognition. *arXiv preprint arXiv:2004.01221*, 2020.

[298] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: an ASR corpus based on public domain audio books. In *Proc. ICASSP*, pages 5206–5210. IEEE, 2015.

[299] George Papandreou, Athanassios Katsamanis, Vassilis Pitsikalis, and Petros Maragos. Adaptive multimodal fusion by uncertainty compensation with application to audiovisual speech recognition. *Audio, Speech, and Language Processing, IEEE Transactions on*, 17(3):423–435, 2009.

[300] Daniel S. Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D. Cubuk, and Quoc V. Le. SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition. In *Proc. Interspeech 2019*, pages 2613–2617, 2019.

[301] 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. *arXiv preprint arXiv:2003.04298*, 2020.

[302] Michelle Patterson and Janet Werker. Infants match phonetic information in lips and voice. *Developmental Science*, 6:191 – 196, 04 2003.

[303] S. Petridis and M. Pantic. Deep complementary bottleneck features for visual speech recognition. *ICASSP*, pages 2304–2308, 2016.

[304] Stavros Petridis, Themos Stafylakis, Pingchuan Ma, Feipeng Cai, Georgios Tzimiropoulos, and Maja Pantic. End-to-end audiovisual speech recognition. *CoRR*, abs/1802.06424, 2018.

[305] Stavros Petridis, Themos Stafylakis, Pingchuan Ma, Georgios Tzimiropoulos, and Maja Pantic. Audio-Visual Speech Recognition with a Hybrid CTC/Attention Architecture. In *IEEE Spoken Language Technology Workshop*, pages 513–520, 2018.

[306] Tomas Pfister, James Charles, and Andrew Zisserman. Large-scale learning of sign language by watching TV (using co-occurrences). In *Proc. BMVC*, 2013.

[307] Tomas Pfister, James Charles, and Andrew Zisserman. Domain-adaptive discriminative one-shot learning of gestures. In *ECCV*, 2014.

[308] Tomas Pfister, James Charles, and Andrew Zisserman. Flowing convnets for human pose estimation in videos. In *Proc. ICCV*, 2015.

[309] A. J. Piergiovanni, Anelia Angelova, and Michael S. Ryoo. Evolving losses for unsupervised video representation learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[310] G. Potamianos, C. Neti, G. Gravier, A. Garg, and A.W. Senior. Recent advances

in the automatic recognition of audiovisual speech. *Proceedings of the IEEE*, 91(9):1306–1326, 2003.

[311] 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 1428–1436. Association for Computing Machinery, 2019.

[312] 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 (ECCV)*, 2020.

[313] Mohammad H Radfar and Richard M Dansereau. Single-channel speech separation using soft mask filtering. *IEEE Transactions on Audio, Speech, and Language Processing*, 2007.

[314] Janani Ramaswamy and Sukhendu Das. See the sound, hear the pixels. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, March 2020.

[315] Aarthi M Reddy and Bhiksha Raj. Soft mask methods for single-channel speaker separation. *IEEE Transactions on Audio, Speech, and Language Processing*, 2007.

[316] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN: towards real-time object detection with region proposal networks. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2016.

[317] Zhongzheng Ren, Zhiding Yu, Xiaodong Yang, Ming-Yu Liu, Yong Jae Lee, Alexander G. Schwing, and Jan Kautz. Instance-aware, context-focused, and memory-efficient weakly supervised object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[318] Zhongzheng Ren, Zhiding Yu, Xiaodong Yang, Ming-Yu Liu, Alexander G. Schwing, and Jan Kautz. UFO^2: A unified framework towards omni-supervised object detection. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[319] Katrin Renz, Nicolaj Stache, Samuel Albanie, and Gül Varol. Sign segmentation with temporal convolutional networks. In *International Conference on Acoustics, Speech, and Signal Processing*, 2021.

[320] Fred Richardson, Douglas Reynolds, and Najim Dehak. Deep neural network approaches to speaker and language recognition. *IEEE signal processing letters*, 22(10):1671–1675, 2015.

[321] Bertrand Rivet, Wenwu Wang, Syed Mohsen Naqvi, and Jonathon A. Chambers. Audiovisual speech source separation: An overview of key methodologies. *IEEE Signal Processing Magazine*, 2014.

[322] A. W. Rix, J. G. Beerends, M. P. Hollier, and A. P. Hekstra. Perceptual evaluation of speech quality (pesq)-a new method for speech quality assessment of telephone networks and codecs. In *ICASSP*, 2001.

[323] Antony W Rix, John G Beerends, Michael P Hollier, and Andries P Hekstra. Perceptual evaluation of speech quality (pesq)-a new method for speech quality assessment of tele-

phone networks and codecs. In *Proc. ICASSP*, volume 2, pages 749–752. IEEE, 2001.

[324] Jason Rodolitz, Evan Gambill, Brittany Willis, Christian Vogler, and Raja Kushalnagar. Accessibility of voice-activated agents for people who are deaf or hard of hearing. *The Journal On Technology and Persons with Disabilities*, 2019.

[325] Rebecca E Ronquest, Susannah V Levi, and David B Pisoni. Language identification from visual-only speech signals. *Attention, Perception, & Psychophysics*, 72(6):1601–1613, 2010.

[326] Joseph Roth, Sourish Chaudhuri, Ondrej Klejch, Radhika Marvin, Andrew Gallagher, Liat Kaver, Sharadh Ramaswamy, Arkadiusz Stopczynski, Cordelia Schmid, Zhonghua Xi, et al. AVA-ActiveSpeaker: An audio-visual dataset for active speaker detection. *arXiv preprint arXiv:1901.01342*, 2019.

[327] Andrew Rouditchenko, Hang Zhao, Chuang Gan, Josh McDermott, and Antonio Torralba. Self-supervised audio-visual co-segmentation. In *Proc. ICASSP*, pages 2357–2361. IEEE, 2019.

[328] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, S. Huang, A. Karpathy, A. Khosla, M. Bernstein, A.C. Berg, and F.F. Li. Imagenet large scale visual recognition challenge. *IJCV*, 2015.

[329] Donald S. Williamson, Yuxuan Wang, and DeLiang Wang. Complex ratio masking for joint enhancement of magnitude and phase. In *Proc. ICASSP*, 2016.

[330] Hasim Sak, Andrew W. Senior, Kanishka Rao, and Françoise Beaufays. Fast and accurate recurrent neural network acoustic models for speech recognition. In *INTERSPEECH*, 2015.

[331] K. P. Sankar, C. Jawahar, and Andrew Zisserman. Subtitle-free movie to script alignment. In *BMVC*, 2009.

[332] P. Santemiz, Oya Aran, M. Saraçlar, and L. Akarun. Automatic sign segmentation from continuous signing via multiple sequence alignment. *ICCVW*, 2009.

[333] CR Scheier. Sound alters visual temporal resolution. *Invest. Ophthalmol. Visual Sci.*, 40(4):4169, 1999.

[334] Adam Schembri, Jordan Fenlon, Ramas Rentelis, and Kearsy Cormier. British Sign Language Corpus Project: A corpus of digital video data and annotations of British Sign Language 2008-2017 (Third Edition), 2017.

[335] Adam Schembri, Jordan Fenlon, Ramas Rentelis, Sally Reynolds, and Kearsy Cormier. Building the british sign language corpus. *Language Documentation & Conservation*, 7:136–154, 2013.

[336] Mike Schuster and K.K. Paliwal. Bidirectional recurrent neural networks. *Trans. Sig. Proc.*, page 2673–2681, Nov 1997.

[337] R. Sekuler, A. Sekuler, and R. Lau. Sound alters visual motion perception. *Nature*, 385:308–308, 1997.

[338] 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.

[339] Andrew Senior, Haşim Sak, Félix de Chaumont Quitry, Tara Sainath, Kanishka Rao, et al. Acoustic modelling with CD-CTC-sMBR LSTM RNNs. In *IEEE Workshop on Automatic Speech Recognition and Understanding*, pages 604–609. IEEE, 2015.

[340] Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany, Aug 2016. Association for Computational Linguistics.

[341] Arda Senocak, Tae-Hyun Oh, Junsik Kim, Ming-Hsuan Yang, and In So Kweon. Learning to localize sound source in visual scenes. In *CVPR*, 2018.

[342] Muhammad Shahid, Cigdem Beyan, and Vittorio Murino. Voice activity detection by upper body motion analysis and unsupervised domain adaptation. In *The IEEE International Conference on Computer Vision (ICCV) Workshops*, Oct 2019.

[343] Muhammad Shahid, Cigdem Beyan, and Vittorio Murino. S-vvad: Visual voice activity detection by motion segmentation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2332–2341, 2021.

[344] Changhao Shan, Junbo Zhang, Yujun Wang, and Lei Xie. Attention-based end-to-end models for small-footprint keyword spotting. *arXiv preprint arXiv:1803.10916*, 2018.

[345] Y. Shen, R. Ji, Y. Wang, Y. Wu, and L. Cao. Cyclic guidance for weakly supervised joint detection and segmentation. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 697–707, 2019.

[346] Y. Shen, R. Ji, S. Zhang, W. Zuo, and Y. Wang. Generative adversarial learning towards fast weakly supervised detection. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5764–5773, 2018.

[347] Brendan Shillingford, Yannis Assael, Matthew W. Hoffman, Thomas Paine, Cían Hughes, Utsav Prabhu, Hank Liao, Hasim Sak, Kanishka Rao, Lorrayne Bennett, Marie Mulville, Ben Coppin, Ben Laurie, Andrew Senior, and Nando de Freitas. Large-Scale Visual Speech Recognition. In *INTERSPEECH*, 2019.

[348] Shinsuke Shimojo and Ladan Shams. Sensory modalities are not separate modalities: Plasticity and interactions. *Current Opinion in Neurobiology*, 11:505–509, 09 2001.

[349] F. Shipman, Satyakiran Duggina, Caio D. D. Monteiro, and R. Gutierrez-Osuna. Speed-accuracy tradeoffs for detecting sign language content in video sharing sites. *Proceedings of the 19th International ACM SIGACCESS Conference on Computers and Accessibility*, 2017.

[350] Zheng Shou, Dongang Wang, and Shih-Fu Chang. Temporal action localization in untrimmed videos via multi-stage CNNs. In *CVPR*, 2016.

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

[352] Krishna Kumar Singh and Yong Jae Lee. Hide-and-seek: Forcing a network to be

meticulous for weakly-supervised object and action localization. In *2017 IEEE international conference on computer vision (ICCV)*, pages 3544–3553. IEEE, 2017.

[353] Krishna Kumar Singh and Yong Jae Lee. You reap what you sow: Using videos to generate high precision object proposals for weakly-supervised object detection. In *CVPR*, 2019.

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

[355] Linda Smith, Alexandra Quittner, Mary Osberger, and Richard Miyamoto. Audition and visual attention: The developmental trajectory in deaf and hearing populations. *Developmental psychology*, 34:840–50, 10 1998.

[356] Salvador Soto-Faraco, Jordi Navarra, Whitney M Weikum, Athena Vouloumanos, Núria Sebastián-Gallés, and Janet F Werker. Discriminating languages by speech-reading. *Perception & Psychophysics*, 69(2):218–231, 2007.

[357] E. Spelke. Perceiving bimodally specified events in infancy. *Developmental Psychology*, 15:626–636, 1979.

[358] Matthias Sperber, J. Niehues, and A. Waibel. Toward robust neural machine translation for noisy input sequences. In *International Workshop on Spoken Language Translation*, 2017.

[359] Radim Špetlík, Jan Čech, Vojtěch Franc, and Jiří Matas. Visual language identification from facial landmarks. In *Scandinavian Conference on Image Analysis*, pages 389–400. Springer, 2017.

[360] Themos Stafylakis and Georgios Tzimiropoulos. Combining residual networks with lstms for lipreading. In *Interspeech*, 2017.

[361] Themos Stafylakis and Georgios Tzimiropoulos. Zero-shot keyword spotting for visual speech recognition in-the-wild. In *ECCV*, 2018.

[362] Thad E Starner. Visual recognition of American Sign Language using hidden Markov models. Technical report, Massachusetts Inst Of Tech Cambridge Dept Of Brain And Cognitive Sciences, 1995.

[363] Chen Sun, Fabien Baradel, Kevin Murphy, and Cordelia Schmid. Contrastive bidirectional transformer for temporal representation learning. *arXiv.cs*, abs/1906.05743, 2019.

[364] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. Videobert: A joint model for video and language representation learning. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2019.

[365] I. Sutskever, O. Vinyals, and Q. Le. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112, 2014.

[366] Rachel Sutton-Spence and Bencie Woll. *The Linguistics of British Sign Language: An Introduction*. Cambridge University Press, 1999.

[367] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper

with convolutions. In *Proc. CVPR*, 2015.

[368] Cees Taal, Richard Hendriks, Richard Heusdens, and Jesper Jensen. An algorithm for intelligibility prediction of time-frequency weighted noisy speech. *IEEE Transactions on Audio, Speech and Language Processing*, 2011.

[369] Ryoichi Takashima, Li Sheng, and Hisashi Kawai. Investigation of Sequence-level Knowledge Distillation Methods for CTC Acoustic Models. In *Proc. ICASSP*, pages 6156–6160. IEEE, 2019.

[370] Satoshi Tamura, Hiroshi Ninomiya, Norihide Kitaoka, Shin Osuga, Yurie Iribe, Kazuya Takeda, and Satoru Hayamizu. Audio-visual speech recognition using deep bottleneck features and high-performance lipreading. In *2015 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, pages 575–582. IEEE, 2015.

[371] Shinichi Tamura and Shingo Kawasaki. Recognition of sign language motion images. *Pattern Recognition*, 21(4):343–353, 1988.

[372] Peng Tang, Xinggang Wang, Song Bai, Wei Shen, Xiang Bai, Wenyu Liu, and Alan L. Yuille. PCL: proposal cluster learning for weakly supervised object detection. *CoRR*, abs/1807.03342, 2018.

[373] Peng Tang, Xinggang Wang, Xiang Bai, and Wenyu Liu. Multiple instance detection network with online instance classifier refinement. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[374] Makarand Tapaswi, M. Bäuml, and R. Stiefelhagen. Story-based video retrieval in tv series using plot synopses. In *Proceedings of International Conference on Multimedia Retrieval*, 2014.

[375] Makarand Tapaswi, M. Bäuml, and R. Stiefelhagen. Book2Movie: Aligning video scenes with book chapters. In *CVPR*, 2015.

[376] 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 (ECCV)*, pages 247–263, 2018.

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

[378] Andrew Titus, Jan Silovsky, Nanxin Chen, Roger Hsiao, Mary Young, and Arnab Ghoshal. Improving language identification for multilingual speakers. *arXiv preprint arXiv:2001.11019*, 2020.

[379] J.R.R. Uijlings, K.E.A. van de Sande, T. Gevers, and A.W.M. Smeulders. Selective search for object recognition. *International Journal of Computer Vision*, 2013.

[380] Aäron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[381] Wouter Van Gansbeke, Simon Vandenhende, Stamatios Georgoulis, Marc Proesmans, and Luc Van Gool. Scan: Learning to classify images without labels. In *Proceedings of the European Conference on Computer Vision*, 2020.

[382] Wouter Van Gansbeke, Simon Vandenhende, Stamatios Georgoulis, and Luc Van Gool. Unsupervised semantic segmentation by contrasting object mask proposals. *arxiv preprint arxiv:2102.06191*, 2021.

[383] Gül Varol, Liliane Momeni, Samuel Albanie, Triantafyllos Afouras, and Andrew Zisserman. Read and attend: Temporal localisation in sign language videos. In *Proc. CVPR*, 2021.

[384] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention Is All You Need. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[385] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, 2017.

[386] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017.

[387] Christian Vogler and Dimitris Metaxas. A framework for recognizing the simultaneous aspects of American Sign Language. *Computer Vision and Image Understanding*, 81(3):358–384, 2001.

[388] Scott K.J. Walker J.T. Auditory-visual conflicts in the perceived duration of lights, tones and gaps. *Journal of Experimental Psychology: Human Perception and Performance*, 1981.

[389] Fang Wan, Chang Liu, Wei Ke, Xiangyang Ji, Jianbin Jiao, and Qixiang Ye. C-mil: Continuation multiple instance learning for weakly supervised object detection. In *cvpr*, 2019.

[390] Li Wan, Prashant Sridhar, Yang Yu, Quan Wang, and Ignacio Lopez Moreno. Tuplemax loss for language identification. In *ICASSP*, pages 5976–5980. IEEE, 2019.

[391] Michael Wand, Jan Koutn, et al. Lipreading with long short-term memory. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6115–6119. IEEE, 2016.

[392] DeLiang Wang and Jitong Chen. Supervised speech separation based on deep learning: an overview. *IEEE Transactions on Audio, Speech and Language Processing*, 2017.

[393] Quan Wang, Hannah Muckenhirn, Kevin Wilson, Prashant Sridhar, Zelin Wu, John Hershey, Rif A Saurous, Ron J Weiss, Ye Jia, and Ignacio Lopez Moreno. Voicefilter: Targeted voice separation by speaker-conditioned spectrogram masking. In *INTERSPEECH*, 2018.

[394] Renyu Wang, Ruilin Tong, Yu Ting Yeung, and Xiao Chen. The huawei speaker diarisation system for the voxceleb speaker diarisation challenge. *arXiv preprint arXiv:2010.11657*, 2020.

[395] Weining Wang, Yan Huang, and Liang Wang. Language-driven temporal activity localization: A semantic matching reinforcement learning model. In *CVPR*, 2019.

[396] Wenwu Wang, Darren Cosker, Yulia Hicks, S Saneit, and Jonathon Chambers. Video assisted speech source separation. In *Proc. ICASSP*, 2005.

[397] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7794–7803, 2018.

[398] Xiaolong Wang and Abhinav Gupta. Unsupervised learning of visual representations using videos. In *Proc. ICCV*, pages 2794–2802, 2015.

[399] Xiaolong Wang, Abhinav Shrivastava, and Abhinav Gupta. A-Fast-RCNN: Hard positive generation via adversary for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[400] Yisen Wang, Xuejiao Deng, Songbai Pu, and Zhiheng Huang. Residual Convolutional CTC Networks for Automatic Speech Recognition. *arXiv preprint arXiv:1702.07793*, 2017.

[401] David H. Warren. *Blindness and Children: An Individual Differences Approach*. Cambridge University Press, 1994.

[402] David H Warren, Robert B Welch, and Timothy J McCarthy. The role of visual-auditory "compellingness" in the ventriloquism effect: Implications for transitivity among the spatial senses. *Perception & Psychophysics*, 30(6):557–564, 1981.

[403] Whitney M Weikum, Athena Vouloumanos, Jordi Navarra, Salvador Soto-Faraco, Núria Sebastián-Gallés, and Janet F Werker. Visual language discrimination in infancy. *Science*, 316(5828):1159–1159, 2007.

[404] Michael Wertheimer. Psychomotor coordination of auditory and visual space at birth. *Science*, 134(3491):1692–1692, 1961.

[405] Ronald J Williams and David Zipser. A learning algorithm for continually running fully recurrent neural networks. *Neural computation*, 1(2):270–280, 1989.

[406] 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.

[407] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Lukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. Google's neural machine translation system: Bridging the gap between human and machine translation. *CoRR*, abs/1609.08144, 2016.

[408] Zhirong Wu, Yuanjun Xiong, Stella X. Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[409] Xiong Xiao, Naoyuki Kanda, Zhuo Chen, Tianyan Zhou, Takuya Yoshioka, Sanyuan Chen, Yong Zhao, Gang Liu, Yu Wu, Jian Wu, Shujie Liu, Jinyu Li, and Yifan Gong. Microsoft speaker diarization system for the voxceleb speaker recognition challenge 2020. In *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5824–5828, 2021.

[410] Junyuan Xie, Ross Girshick, and Ali Farhadi. Unsupervised deep embedding for clustering analysis. In *International conference on machine learning*, pages 478–487, 2016.

[411] 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.

[412] Bo Xu, Cheng Lu, Yandong Guo, and Jacob Wang. Discriminative multi-modality speech recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14433–14442, 2020.

[413] Huijuan Xu, Kun He, Bryan A Plummer, Leonid Sigal, Stan Sclaroff, and Kate Saenko. Multilevel language and vision integration for text-to-clip retrieval. In *AAAI*, 2019.

[414] G. Yan, B. Liu, N. Guo, X. Ye, F. Wan, H. You, and D. Fan. C-midn: Coupled multiple instance detection network with segmentation guidance for weakly supervised object detection. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9833–9842, 2019.

[415] Sijie Yan, Yuanjun Xiong, and Dahua Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *AAAI*, 2018.

[416] Hee-Deok Yang, Stan Sclaroff, and Seong-Whan Lee. Sign language spotting with a threshold model based on conditional random fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2008.

[417] Jianwei Yang, Devi Parikh, and Dhruv Batra. Joint unsupervised learning of deep representations and image clusters. In *CVPR*, pages 5147–5156, 2016.

[418] Ruiduo Yang and Sudeep Sarkar. Detecting coarticulation in sign language using conditional random fields. In *ICPR*, 2006.

[419] Zhenheng Yang, Dhruv Mahajan, Deepti Ghadiyaram, Ram Nevatia, and Vignesh Ramanathan. Activity driven weakly supervised object detection. In *cvpr*, 2019.

[420] 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.

[421] Dong Yu, Morten Kolbæk, Zheng-Hua Tan, and Jesper Jensen. Permutation invariant training of deep models for speaker-independent multi-talker speech separation. In *Proc. ICASSP*, 2017.

[422] Jianwei Yu, Shi-Xiong Zhang, Bo Wu, Shansong Liu, Shoukang Hu, Xunying Liu, Helen M Meng, and Dong Yu. Audio-visual multi-channel integration and recognition of overlapped speech. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2021.

[423] Jianwei Yu, Shi-Xiong Zhang, Jian Wu, Shahram Ghorbani, Bo Wu, Shiyin Kang, Shansong Liu, Xunying Liu, Helen Meng, and Dong Yu. Audio-visual recognition of overlapped speech for the lrs2 dataset. In *ICASSP*, pages 6984–6988, 05 2020.

[424] Jianwei Yu, Shi-Xiong Zhang, Jian Wu, Shahram Ghorbani, Bo Wu, Shiyin Kang, Shansong Liu, Xunying Liu, Helen Meng, and Dong Yu. Audio-visual recognition of overlapped speech for the LRS2 dataset. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6984–6988.

IEEE, 2020.

[425] Licheng Yu, Zhe Lin, Xiaohui Shen, Jimei Yang, Xin Lu, Mohit Bansal, and Tamara L Berg. Mattnet: Modular attention network for referring expression comprehension. In *CVPR*, 2018.

[426] Jiahong Yuan and Mark Liberman. Speaker identification on the scotus corpus. *Journal of the Acoustical Society of America*, 123(5):3878, 2008.

[427] Yitian Yuan, Tao Mei, and Wenwu Zhu. To find where you talk: Temporal sentence localization in video with attention based location regression. In *AAAI*, 2019.

[428] Yuhui Yuan and Jingdong Wang. Ocnet: Object context network for scene parsing. In *arXiv preprint arXiv:1809.00916*, 2018.

[429] Neil Zeghidour, Nicolas Usunier, Iasonas Kokkinos, Thomas Schatz, Gabriel Synnaeve, and Emmanuel Dupoux. Learning filterbanks from raw speech for phone recognition. *CoRR*, abs/1711.01161, 2017.

[430] Runhao Zeng, H. Xu, W. Huang, Peihao Chen, Mingkui Tan, and Chuang Gan. Dense regression network for video grounding. In *CVPR*, 2020.

[431] Zhaoyang Zeng, Bei Liu, Jianlong Fu, Hongyang Chao, and Lei Zhang. Wsod2: Learning bottom-up and top-down objectness distillation for weakly-supervised object detection. In *iccv*, 2019.

[432] Huaao Zhang, Shigui Qiu, Xiangyu Duan, and Min Zhang. Token drop mechanism for neural machine translation. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4298–4303, Barcelona, Spain (Online), 2020. International Committee on Computational Linguistics.

[433] Jason Y Zhang, Panna Felsen, Angjoo Kanazawa, and Jitendra Malik. Predicting 3d human dynamics from video. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7114–7123, 2019.

[434] Xiaopeng Zhang, Jiashi Feng, Hongkai Xiong, and Qi Tian. Zigzag learning for weakly supervised object detection. In *CVPR*, 2018.

[435] Xingxuan Zhang, Feng Cheng, and Shilin Wang. Spatio-temporal fusion based convolutional sequence learning for lip reading. In *Proceedings of the International Conference on Computer Vision (ICCV)*, pages 713–722, 2019.

[436] Y. Zhang, Y. Bai, M. Ding, Y. Li, and B. Ghanem. W2f: A weakly-supervised to fully-supervised framework for object detection. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 928–936, 2018.

[437] Ying Zhang, Mohammad Pezeshki, Philemon Brakel, Saizheng Zhang, César Laurent, Yoshua Bengio, and Aaron C. Courville. Towards end-to-end speech recognition with deep convolutional neural networks. *CoRR*, abs/1701.02720, 2017.

[438] Yuan-Hang Zhang, Jingyun Xiao, Shuang Yang, and Shiguang Shan. Multi-task learning for audio-visual active speaker detection. *The ActivityNet Large-Scale Activity Recognition Challenge*, 2019.

[439] Yuanhang Zhang, Shuang Yang, Jingyun Xiao, Shiguang Shan, and Xilin Chen. Can

We Read Speech Beyond the Lips? Rethinking RoI Selection for Deep Visual Speech Recognition. In *arXiv preprint arXiv:2003.03206*, 2020.

[440] Hang Zhao, Chuang Gan, Wei-Chiu Ma, and Antonio Torralba. The sound of motions. *Proceedings of the International Conference on Computer Vision (ICCV)*, 2019.

[441] Hang Zhao, Chuang Gan, Andrew Rouditchenko, Carl Vondrick, Josh McDermott, and Antonio Torralba. The sound of pixels. *arXiv preprint arXiv:1804.03160*, 2018.

[442] Hang Zhao, Antonio Torralba, Lorenzo Torresani, and Zhicheng Yan. Hacs: Human action clips and segments dataset for recognition and temporal localization. In *ICCV*, 2019.

[443] M. Zhao, T. Li, M. A. Alsheikh, Y. Tian, H. Zhao, A. Torralba, and D. Katabi. Through-wall human pose estimation using radio signals. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7356–7365, 2018.

[444] 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 (CVPR)*, June 2018.

[445] Ya Zhao, Rui Xu, Xinchao Wang, Peng Hou, Haihong Tang, and Mingli Song. Hearing lips: Improving lip reading by distilling speech recognizers. In *arXiv preprint arXiv:1911.11502*, 2019.

[446] Bolei Zhou, Aditya Khosla, Àgata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *CVPR*, 2016.

[447] Ziheng Zhou, Guoying Zhao, Xiaopeng Hong, and Matti Pietikäinen. A review of recent advances in visual speech decoding. *Image and vision computing*, 32(9):590–605, 2014.

[448] Yang Zou, Zhiding Yu, Xiaofeng Liu, B. V. K. Vijaya Kumar, and Jinsong Wang. Confidence regularized self-training. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2019.

Appendices

A | Statements of Authorship

A statement of authorship is provided for each multi-authored apper included in this thesis. The statements describe the candidate's 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 2: Deep Audio-Visual Speech Recognition.

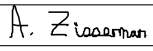
Paper title	Deep Audio-Visual Speech Recognition
Publication status	Published
Authors	**Triantafyllos Afouras***, Joon Son Chung*, Andrew Senior, Oriol Vinyals, Andrew Zisserman. (* denotes equal contribution)
Details	Published in IEEE Transactions on Pattern Analysis and Machine Intelligence, 2019

Student Confirmation

Student Name	Triantafyllos Afouras		
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		Date	04 / 04 / 2022

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	04 / 04 / 2022

Statment of Authorship for multi-authored paper in Chapter 3: Sub-word Level Lip-reading with Visual Attention.

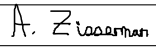
Paper title	Sub-word Level Lip-reading with Visual Attention
Publication status	Published
Authors	Prajwal Kondajji Renukananda, **Triantafyllos Afouras**, Andrew Zisserman.
Details	To be published in the proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR) 2022.

Student Confirmation

Student Name	Triantafyllos Afouras		
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		Date	04 / 04 / 2022

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	04 / 04 / 2022

Statment of Authorship for multi-authored paper in Chapter 4: ASR Is All You Need: Cross-modal Distillation For Lip Reading.

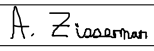
Paper title	ASR is all you need: cross-modal distillation for lip reading
Publication status	Published
Authors	**Triantafyllos Afouras**, Joon Son Chung, Andrew Zisserman.
Details	Published in IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2020, pp. 2143-2147, doi: 10.1109/ICASSP40776.2020.9054253.

Student Confirmation

Student Name	Triantafyllos Afouras		
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		Date	04 / 04 / 2022

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	04 / 04 / 2022

Statment of Authorship for multi-authored paper in Chapter 5: Now You're Speaking My Language: Visual Language IDentification.

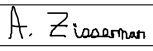
Paper title	Now you're speaking my language: Visual language identification
Publication status	Published
Authors	**Triantafyllos Afouras**, Joon Son Chung, Andrew Zisserman.
Details	Published in the proceedings of INTERSPEECH, 2020, pp. 2402-2406, doi: 10.21437/Interspeech.2020-2921.

Student Confirmation

Student Name	Triantafyllos Afouras		
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		Date	04 / 04 / 2022

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	04 / 04 / 2022

Statment of Authorship for multi-authored paper in Chapter 6: The Conversation: Deep Audio-Visual Speech Enhancement.

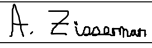
Paper title	The Conversation: Deep Audio-Visual Speech Enhancement
Publication status	Published
Authors	**Triantafyllos Afouras**, Joon Son Chung, Andrew Zisserman.
Details	Published in the proceedings of INTERSPEECH, 2018, pp. 3244-3248, doi: 10.21437/Interspeech.2018-1400.

Student Confirmation

Student Name	Triantafyllos Afouras		
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		Date	04 / 04 / 2022

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	04 / 04 / 2022

Statment of Authorship for multi-authored paper in Chapter 7: My Lips Are Concealed: Audio-visual Speech Enhancement Through Obstructions.

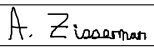
Paper title	My lips are concealed: Audio-visual speech enhancement through obstructions
Publication status	Published
Authors	**Triantafyllos Afouras**, Joon Son Chung, Andrew Zisserman.
Details	Published in the proceedings of INTERSPEECH, 2019, pp. 4295-4299, doi: 10.21437/Interspeech.2019-3114.

Student Confirmation

Student Name	Triantafyllos Afouras		
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		Date	04 / 04 / 2022

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	04 / 04 / 2022

Statment of Authorship for multi-authored paper in Chapter 8: Self-Supervised Learning of Audio-Visual Objects from Video.

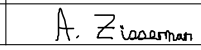
Paper title	Self-Supervised Learning of Audio-Visual Objects from Video
Publication status	Published
Authors	**Triantafyllos Afouras**, Andrew Owens, Joon Son Chung, Andrew Zisserman.
Details	Published in the proceedings of European Conference on Computer Vision (ECCV), 2020, pp. 208-224, doi: :10.1007/978-3-030-58523-5_13.

Student Confirmation

Student Name	Triantafyllos Afouras		
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		Date	04 / 04 / 2022

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	04 / 04 / 2022

Statment of Authorship for multi-authored paper in Chapter 9: Self-supervised Object Detection From Audio-visual Correspondence.

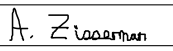
Paper title	Self-supervised object detection from audio-visual correspondence
Publication status	Published
Authors	**Triantafyllos Afouras***, Yuki M. Asano*, Francois Fagan, Andrea Vedalid, Florian Metze.
Details	To be published in the proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR) 2022.

Student Confirmation

Student Name	Triantafyllos Afouras		
Contribution to the paper	1. 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		Date	04 / 04 / 2022

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	04 / 04 / 2022

Statment of Authorship for multi-authored paper in Chapter 10: Read and Attend: Temporal Localisation in Sign Language Videos.

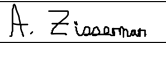
Paper title	Read and Attend: Temporal Localisation in Sign Language Videos
Publication status	Published
Authors	Gul Varol*, Liliane Momeni*, Samuel Albanie*, **Triantafyllos Afouras***, Andrew Zisserman.
Details	Published in the proceedings of Conference on Computer Vision and Pattern Recognition (CVPR), 2021.

Student Confirmation

Student Name	Triantafyllos Afouras		
Contribution to the paper	1. Joint conception of the idea 2. Research of prior work 3. Bug fixes 4. Writing of the paper		
Signature		Date	04 / 04 / 2022

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	04 / 04 / 2022

Statment of Authorship for multi-authored paper in Chapter 11: Aligning Subtitles in Sign Language Videos.

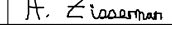
Paper title	Aligning Subtitles in Sign Language Videos
Publication status	Published
Authors	Hannah Bull*, **Triantafyllos Afouras***, Gul Varol, Samuel Albanie, Liliane Momeni, Andrew Zisserman.
Details	Published in the proceedings of the International Conerence on Computer Vision (ICCV) 2021.

Student Confirmation

Student Name	Triantafyllos Afouras		
Contribution to the paper	1. Joint conception of the idea 2. Joint design and implementation of models 3. Running of part of the experiments 4. Writing of the paper		
Signature		Date	04 / 04 / 2022

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	04 / 04 / 2022