

Imperial College London
Department of Computing

Leveraging Audio-visual Speech Effectively via Deep Learning

Rodrigo Schönbürg Carrillo de Mira

Supervised by Prof. Maja Pantic and Prof. Björn Schuller

Submitted in part fulfilment of the requirements for the degree of
Doctor of Philosophy in Computing and the Diploma of Imperial College

February 2022

Declaration of originality

I hereby declare that, to the best of my knowledge, this thesis and the research it describes are my own work, and all research by other authors has been appropriately and explicitly referenced. This thesis has not been submitted for any other purpose.

Copyright declaration

The copyright of this thesis rests with the author. Unless otherwise indicated, its contents are licensed under a Creative Commons Attribution-NonCommercial 4.0 International Licence (CC BY-NC).

Under this license, you may copy and redistribute the material in any medium or format. You may also create and distribute modified versions of the work. This is on the condition that: you credit the author and do not use it, or any derivative works, for a commercial purpose.

When reusing or sharing this work, ensure you make the license terms clear to others by naming the license and linking to the license text. Where a work has been adapted, you should indicate that the work has been changed and describe those changes.

Please seek permission from the copyright holder for uses of this work that are not included in this license or permitted under UK Copyright Law.

Abstract

The rising popularity of neural networks, combined with the recent proliferation of online audio-visual media, has led to a revolution in the way machines encode, recognize, and generate acoustic and visual speech. Despite the ubiquity of naturally paired audio-visual data, only a limited number of works have applied recent advances in deep learning to leverage the duality between audio and video within this domain. This thesis considers the use of neural networks to learn from large unlabelled datasets of audio-visual speech to enable new practical applications. We begin by training a visual speech encoder that predicts latent features extracted from the corresponding audio on a large unlabelled audio-visual corpus. We apply the trained visual encoder to improve performance on lip reading in real-world scenarios. Following this, we extend the idea of video learning from audio by training a model to synthesize raw speech directly from raw video, without the need for text transcriptions. Remarkably, we find that this framework is capable of reconstructing intelligible audio from videos of new, previously unseen speakers. We also experiment with a separate speech reconstruction framework, which leverages recent advances in sequence modeling and spectrogram inversion to improve the realism of the generated speech. We then apply our research in video-to-speech synthesis to advance the state-of-the-art in audio-visual speech enhancement, by proposing a new vocoder-based model that performs particularly well under extremely noisy scenarios. Lastly, we aim to fully realize the potential of paired audio-visual data by proposing two novel frameworks that leverage acoustic and visual speech to train two encoders that learn from each other simultaneously. We leverage these pre-trained encoders for deepfake detection, speech recognition, and lip reading, and find that they consistently yield improvements over training from scratch.

Acknowledgements

I genuinely tried to spare the reader from the cliché of writing overly emotional acknowledgments in theses that, due to their scientific nature, are otherwise exceedingly dry. Unfortunately, to roughly paraphrase David Foster Wallace’s “This Is Water”, these tiresome clichés often turn out to be rooted in such profound truths that they become inescapable. So here we are.

The heterogeneity of the past four years makes them difficult to summarize, particularly since half of them were spent in Portugal and half in London. It is certainly true that I have achieved many things during this time - this whole thesis is effectively dedicated to the pursuit of self-promotion, so I won’t go into detail there¹. More importantly, however, it is not an exaggeration to say that these have been the most challenging years of my life, having gone through a variety of professional and personal dilemmas. Therefore, I’d like to dedicate the next paragraphs to the people that allowed me to overcome those challenges.

First, I’d like to thank my mother and father for going through the harrowing process of raising me and blessing me with the kind of unconditional love and overwhelming warmth that only parents can give. The praise I give here is obviously partially self-serving, but I think they did a formidable job. I’d also like to thank my siblings Alexandra and Filipe, who shaped me into the person I am today. It has become abundantly clear to me in the past few years that I would be a less capable, more narcissistic person if I had not received their care and guidance. I’d also like to thank my dearest friends Simas, Dani, Fred, Vasco, and Mota, who so often kept me company from afar, and were my refuge from the discomforts and tragedies of academia and life in general.

I’d like to thank my friends Viet, David, Sacha, Pierre, Miguel, Manon, Dominika, Alex, Nihir, Harry, Carles, Andreas, Miriam, Antoni, Akis, and Abhinav who, in more ways than one, put a friendly face on the word ‘Imperial’, as it exists in my mind. On that note, a huge thank you to Teresa, Lloyd, Geoff, Duncan, and everyone at CSG, who on so many occasions selflessly took time out of their day to help me with all sorts of matters. I’d also like to thank my collaborators Alex, Dino, and Pingchuan, to whom I owe all that I know about being a good researcher, and my colleagues at Meta Jacob, Buye, Anurag, and Vamsi, who opened my eyes to new ways of doing research, and gave me such a warm welcome when we met at CVPR and Interspeech.

¹The irony of attempting to display my humility here after having written a 200-page dissertation about myself does not escape me.

Finally, I'd like to thank Stavros, Maja, and Björn for supervising and supporting me during this time - without their invaluable help, completing this Ph.D. would not have been possible. I'd like to include a special mention to those who housed and fed me on multiple occasions, and therefore directly contributed to the existence of this thesis by guaranteeing my survival: mom, dad, Simas, Sebas, Viet, Manon, Pierre, Sacha, and David. Regarding my feelings, there is too much to say, so it is likely best to say very little. Above all else, I have learned that sometimes we must *choose* to be happy, despite what other plans we may have.

‘Quem tem alma não tem calma.’

Fernando Pessoa

Para ti, mãe

CONTENTS

Abstract	iii
Acknowledgements	v
List of Tables	xvii
List of Figures	xxi
1 Introduction	1
1.1 Motivation	1
1.2 Questions and Objectives	8
1.3 Contributions	10
1.4 Publications	12
2 Background	14
2.1 Datasets	15
2.1.1 Studio-recorded	16

2.1.2	In-the-wild	18
2.2	Data Processing	21
2.2.1	Audio	22
2.2.2	Video	25
2.3	Modelling	28
2.3.1	Linear layers	28
2.3.2	Convolutional layers	30
2.3.3	Sequence modelling	35
2.3.4	Generative Adversarial Networks	44
2.4	Evaluation metrics	50
2.4.1	Speech quality metrics	50
2.4.2	Speech intelligibility metrics	52
2.4.3	Other metrics	54
3	Learning Visual Representations from Speech	56
3.1	Introduction	57
3.2	Methodology	60
3.2.1	Learning visual representations from speech	60
3.2.2	Lip reading	61
3.3	Experimental Setup	61
3.4	Results	63
3.4.1	Word-level lip-reading	63
3.4.2	Sentence-level lip-reading	65

3.5	Conclusion	66
4	Synthesizing Speech from Silent Video End-to-End	67
4.1	Introduction	69
4.2	Related Work	71
4.2.1	Lip reading	72
4.2.2	Speech Synthesis	72
4.2.3	Reconstructing audio from visual speech	73
4.2.4	Reconstructing audio from multi-view visual speech	75
4.2.5	Audio reconstruction from video in other applications	76
4.3	Video-Driven Speech Reconstruction	77
4.3.1	Generator	77
4.3.2	Critics	79
4.3.3	Losses	80
4.3.4	Training details	83
4.4	Datasets	84
4.5	Evaluation Metrics	85
4.6	Results on Seen Speakers	85
4.6.1	Ablation Study	86
4.6.2	Comparison with Other Works	89
4.6.3	Performance as a Function of Training Set Size	91
4.7	Results on Unseen Speakers	92
4.7.1	Ablation Study	92

4.7.2	Comparison with Other Works	93
4.7.3	Additional Experiments	93
4.8	Results in the Wild	94
4.9	Conclusion	96
5	Synthesizing Speech from Silent Video using Neural Vocoders	98
5.1	Introduction	99
5.2	Methodology	100
5.2.1	Video-to-spectrogram model	100
5.2.2	Spectrogram-to-waveform	102
5.3	Experimental setup	103
5.3.1	Datasets	103
5.3.2	Data pre-processing and augmentation	104
5.3.3	Training details	104
5.3.4	Evaluation metrics	105
5.4	Results	105
5.4.1	Experiments	105
5.4.2	Ablations	106
5.5	Conclusion	108
6	Audio-visual Speech Enhancement	109
6.1	Introduction	110
6.2	Methodology	113
6.2.1	Audio-visual spectral enhancement	113

6.2.2	Waveform synthesis	114
6.3	Experimental Setup	115
6.3.1	Datasets	115
6.3.2	Pre-processing, augmentation, and evaluation	116
6.3.3	Comparison models	117
6.3.4	Training details	117
6.4	Results	118
6.4.1	Comparison with other works	118
6.4.2	Noise and interference study	119
6.4.3	Spectrogram inversion comparison	121
6.5	Conclusion	122
7	Learning from Both Modalities Simultaneously	123
7.1	Introduction	125
7.2	RealForensics	126
7.2.1	Methodology	126
7.2.2	Experimental Setup	127
7.2.3	Results	129
7.3	RAVE _n	131
7.3.1	Methodology	132
7.3.2	Experimental Setup	134
7.3.3	Results	135
7.4	Conclusion	137

8 Conclusion 138

8.1 Summary 138

8.2 Applications 140

8.3 Ethical Considerations 141

8.4 Future Work 143

LIST OF TABLES

2.1	Summary of the contents of each audio-visual dataset we describe in this section.	16
2.2	Summary of the technical details of each audio-visual dataset we describe in this section.	17
3.1	Comparison of LiRA with previous supervised and self-supervised lip reading methods on LRW.	62
3.2	Lip reading performance comparison of LiRA with other supervised and unsupervised works on LRS2.	64
4.1	Number of speech clips and total number of hours of speech for each dataset used in our study.	84
4.2	Ablation study for our end-to-end video-to-speech model performed on GRID (4 speakers, seen speakers).	88
4.3	Comparison between our model and previous works, using GRID (4 speakers, seen speakers).	91
4.4	Comparison between our model and Lip2Wav, using TCD-Timit (3 lip speakers) with a seen speaker split.	91

4.5	Study on the performance of our model using varying training set sizes, using GRID (33 speakers, seen speakers)	92
4.6	Ablation study for our model performed on GRID (33 speakers, unseen speakers).	93
4.7	Comparison between our current and previous model, using GRID (33 speakers, unseen speakers)	94
4.8	Comparison between our model and Lip2Wav, using the full LRW dataset.	95
4.9	Study on the performance of our speech reconstruction model for different subsets of LRW.	96
5.1	Summary of our proposed SVTS architectures.	102
5.2	Summary of SVTS's results compared to previous works on GRID, LRW, and LRS3.	107
5.3	Vocoder ablation for SVTS on GRID (seen speakers).	107
5.4	Loss ablation for SVTS on GRID (seen speakers).	107
6.1	Comparison between LA-VocE and other speech enhancement methods for different noise conditions.	119
6.2	LA-VocE's performance for different SNR / SIR conditions with 3 background noises and 2 interfering speakers.	120
6.3	LA-VocE's performance for different numbers of background noises and interfering speakers (-5 dB SNR / SIR).	120
6.4	Comparison between different spectrogram inversion methods for LA-VocE (noise condition 2).	121
7.1	Cross-dataset forgery detection performance (Area Under Curve (AUC) in %) of RealForensics compared to previous works.	129
7.2	Ablation study for RealForensics on the downstream deepfake detection task.	130

7.3	Comparison of RealForensics with other self-supervised approaches.	130
7.4	Comparison of RAVEn with other supervised and self-supervised works on LRS3 using only 30 hours of labeled training data.	133
7.5	Ablation study on the prediction tasks used in RAVEn.	136

LIST OF FIGURES

2.1	Illustration of the step-by-step transformation from waveform to dB-scaled mel-frequency spectrogram.	21
2.2	Comparison between Hertz scale and Mel scale.	23
2.3	Conversion from waveform to MFCC.	24
2.4	Face detection example using RetinaFace.	25
2.5	68-point landmark estimation examples using a 2D-FAN.	27
2.6	Examples of a pre-processed cropped face.	28
2.7	A simple illustration of a linear layer.	29
2.8	A simple example of a 2D convolution.	30
2.9	A simple example of a 2D transposed convolution.	33
2.10	Simple example of a residual connection.	34
2.11	A simple illustration of a recurrent neural network (RNN).	36
2.12	Long short-term memory (LSTM) cell.	36
2.13	Gated Recurrent Unit (GRU).	37

2.14	The transformer encoder.	39
2.15	A brief summary of self-attention and multi-head self-attention.	41
2.16	The Conformer.	43
2.17	The convolution module from the Conformer.	44
2.18	Gradients of an optimal discriminator (GAN) and critic (WGAN) trained to discriminate between two Gaussian distributions.	47
2.19	Original GAN's cross-entropy loss function and LSGAN's loss function.	49
2.20	Graphical illustration of the PESQ pipeline.	51
2.21	Graphical illustration of the original ViSQOL algorithm.	52
2.22	Graphical illustration of the STOI pipeline.	53
2.23	Illustration of the area under the receiver operating characteristic curve (AUC).	55
3.1	Overview of LiRA's training procedure.	60
3.2	Lip reading accuracy on LRW using features extracted from different layers of Lira-Frozen.	62
3.3	LiRA's lip reading performance for varying fractions of the labeled training data.	63
4.1	Architecture of the generator, critics, and losses presented in this chapter.	77
4.2	Description of the encoder layers in our end-to-end video-to-speech generator.	78
4.3	Description of the decoder layers in our end-to-end video-to-speech generator.	80
4.4	Description of the layers in the waveform critic and power critic used to train our model.	83
4.5	Mel-frequency spectrograms taken from the audio reconstructed with our seen speaker ablation models.	87

4.6	Mel-frequency spectrograms and waveforms generated by our model, compared to the real audio and previous works.	90
4.7	Spectrogram and waveform produced by our model for a video of a silent speaker.	94
5.1	Summary of our video-to-speech synthesis approach during training and inference.	101
6.1	Summary of LA-VocE’s two-stage training approach and inference procedure. . .	111
7.1	Summary of the RealForensics self-supervised training procedure.	126
7.2	Summary of the RealForensics downstream training procedure, after self-supervised pre-training.	128
7.3	Summary of RAVEn’s self-supervised training procedure.	132
7.4	Summarized overview of RAVEn’s experimental configurations featured in our ablation study.	136

CHAPTER 1

INTRODUCTION

Contents

1.1	Motivation	1
1.2	Questions and Objectives	8
1.3	Contributions	10
1.4	Publications	12

1.1 Motivation

The development of spoken language has played a central role in the development of human society, particularly during its infancy, by allowing individuals to collaborate efficiently via a consistent, widely adopted means of communication. More recently, modern technology has augmented and amplified the role of speech by allowing long-distance transmission of acoustic speech (audio only), via the telephone and radio, and audio-visual speech (audio and video), via the television. This has transformed the nature of speech, which went from being a means of

communication between a local group of individuals to a flexible form of media that is constantly being produced and shared on a worldwide basis. The ubiquity and newfound universality of this medium, together with the accelerating advances in computing, have drawn researchers to study its contents using newly emerging computational techniques.

The intersection between speech and computing has had four primary research focuses in recent years: transmission, feature extraction, recognition, and synthesis. Research on transmission has largely focused on improving the speed and efficiency of audio/video transmission via the Internet. Disregarding hardware improvements, which lie outside the field of software, transmission can be improved by, for example, developing new codecs, i. e., new forms of speech compression. Recently, the introduction of neural codecs for acoustic speech (Polyak *et al.*, 2021) and audio-visual speech (Yang *et al.*, 2022) has drawn substantial attention to this research issue. For the most part, however, this thesis focuses on the three latter topics: feature extraction, recognition, and synthesis.

Historically, the most common method for extracting features from speech is the Short Time Fourier Transform (STFT), which decomposes the raw waveform into frequency and phase. This form of spectral analysis has been applied to produce a variety of acoustic features, the most popular of which are the spectrogram, the mel-spectrogram, and the Mel-Frequency Cepstrum Coefficients (MFCC). These are often leveraged for speech-related tasks such as speech recognition (Ittichaichareon *et al.*, 2012) and speaker identification (Kamruzzaman *et al.*, 2010). For visual speech (video only), traditional feature extraction approaches (Gurban and Thiran, 2009) include compression-based methods, such as the discrete cosine transform (DCT) (Strang, 1999), linear discriminant analysis (LDA) (Tharwat *et al.*, 2017) and principal components analysis (PCA) (Shlens, 2014), and mouth geometry methods, which attempt to extract the lip movements from the images using simple computer vision techniques (e. g., color thresholding (Kumar *et al.*, 2007)).

Recently, however, newly developed neural architectures such as PASE (Pascual *et al.*, 2019a) and PASE+ (Ravanelli *et al.*, 2020a) have been shown to greatly outperform the traditional acoustic features presented above (e. g., STFT, MFCC) for various tasks, including speech and

emotion recognition. Indeed, while spectrograms continue to be used in many state-of-the-art models, they are generally fed into neural encoders which encode them into useful intermediate features (Gulati *et al.*, 2020; Shen *et al.*, 2018). Furthermore, many recent works entirely forego the use of manually extracted speech features and achieve state-of-the-art results in highly competitive tasks such as speech recognition (Zeghidour *et al.*, 2018) and text-to-speech synthesis (Donahue *et al.*, 2021).

On the visual end, the field of computer vision has been revolutionized by the development of powerful visual encoders such as the ResNet (He *et al.*, 2016), EfficientNet (Tan and Le, 2019) and Vision Transformer (Dosovitskiy *et al.*, 2021), which have dominated the ImageNet image classification benchmark (Russakovsky *et al.*, 2015) for many years. In particular, the ResNet-18 (He *et al.*, 2016) (often combined with a 3D convolutional stem, originally proposed in (Stafylakis and Tzimiropoulos, 2017)) has become the standard visual speech encoder in state-of-the-art models for lip reading (Afouras *et al.*, 2018d), audio-visual speech recognition (Afouras *et al.*, 2018c), audio-visual speech enhancement/separation (Afouras *et al.*, 2018e) and video-to-speech synthesis (Mira *et al.*, 2022a).

Speech recognition, also known as speech-to-text, has been one of the most influential tasks in speech research for many decades, particularly due to its evidently impactful applications (e.g., automatic captioning). From the 1970s to the 2000s, speech recognition research (Baker *et al.*, 2009; Kumar *et al.*, 2018a) was largely dominated by Hidden Markov Models (HMMs) (Rabiner and Juang, 1986), often combined with Gaussian Mixture Models (GMMs) (Reynolds, 2008) and trained via the Expectation Maximization (EM) algorithm (Moon, 1996) or the Baum-Welch algorithm (Baum *et al.*, 1970). Other training techniques include Maximum Mutual Information (MMI), Maximum A-posteriori Probability (MAP) estimation, and Maximum Likelihood Linear Regression (MLLR), among others. These methods could not perform recognition on raw audio and therefore relied on handcrafted features (as presented above), such as MFCCs. The methods used for visual speech recognition, also known as lip reading, were similarly based on HMMs applied to handcrafted visual features (Fenghour *et al.*, 2021). HMMs have severe limitations, including a lack of expressive power due to their Markovian nature and difficulties in adapting to noisy inputs or substantial mismatches between training

and testing data. GMMs suffer from similar problems, as they also have difficulties in adapting to unseen data and can be particularly sensitive to outliers and irregular distributions since they make strong assumptions about the input features (e.g., they are assumed to be normally distributed and the clusters are assumed to be elliptical). These issues, combined with the limited generalisability of traditional feature extraction techniques, meant that speech recognition and lip reading models were difficult to train and often failed to adapt to new scenarios for which they were not explicitly trained (Kumar *et al.*, 2018a).

The introduction of deep learning has substantially changed the research landscape in speech recognition. By transitioning to end-to-end deep learning models and foregoing the use of any of the previously established methods, speech recognition models have become simpler, easier to train, and far more adaptable to challenging scenarios. Common architectures include convolutional layers, recurrent layers, attention mechanisms, or any combination of the three, and are typically trained using a simple Connectionist Temporal Classification (CTC) Loss (Graves *et al.*, 2006). Apart from being easier to apply, these techniques have proven to be substantially superior to HMM-based models, and have vastly outperformed them for virtually every corpora available (Kumar *et al.*, 2018a).

Indeed, while previous research focused mostly on studio-recorded datasets such as TIMIT (Baker *et al.*, 2009; Garofolo *et al.*, 1993), new models are able to achieve very high accuracies for audio recorded 'in the wild', e.g., Switchboard (Godfrey *et al.*, 1992; Tüske *et al.*, 2021). This has dramatically changed the state-of-the-art in speech recognition, which went from models that were difficult to deploy and scale in real-world applications, to versatile deep learning models that are extremely accurate even within noisy environments and have been successfully deployed in a variety of existing consumer products (Hoy, 2018). Lip reading has followed a similar trend by combining the visual encoders mentioned above with emerging speech recognition techniques to greatly surpass signal processing approaches (Assael *et al.*, 2016). Similarly, state-of-the-art methods have quickly transitioned from studio-recorded datasets such as GRID (Assael *et al.*, 2016; Cooke *et al.*, 2006a), to challenging 'in-the-wild' datasets such as LRS2 (Afouras *et al.*, 2018a) and LRS3 (Afouras *et al.*, 2018b; Ma *et al.*, 2021c).

The last major research topic in recent speech literature has been synthesis. Before the advent of deep learning, speech synthesis research was divided into four fields (Tabet and Boughazi, 2011; Tan *et al.*, 2021): articulatory synthesis, which aims to directly imitate the physical processes in human speech, e.g., tongue movements; formant synthesis, which attempts to synthesize formants (i.e., word segments) based on manually programmed rules; concatenative synthesis, which concatenates existing speech recordings (from a very large corpus of short clips) to form each word; and statistical parametric synthesis, which predicts a specific set of acoustic parameters from text, and translates these into waveform using a vocoder. On the other hand, visual speech synthesis, also referred to as facial animation, was generally achieved by combining HMMs with signal processing techniques such as vector quantization (Simons and Cox, 1990; Yamamoto *et al.*, 1998). In both modalities, these methods often produced reasonably impressive results on a small scale but were either difficult to scale (e.g., concatenative synthesis) or produced clearly robotic results (e.g., statistical parametric synthesis).

In the past decade, the introduction of neural speech synthesis has dramatically outperformed previous techniques. This began with WaveNet (Oord *et al.*, 2016), which introduced stacks of dilated causal convolutions as a way to synthesize waveforms directly from linguistic features and achieved a level of realism far greater than what had previously been achieved. This work set the trend for modern speech synthesis and was followed by new approaches such as WaveGAN (Donahue *et al.*, 2019), Parallel WaveGAN (Yamamoto *et al.*, 2019) and HiFi-GAN (Kong *et al.*, 2020a), which substantially increased inference speed while simultaneously decreasing model size. Similarly, remarkable progress was seen in facial animation with the introduction of an end-to-end speech-to-video model in (Vougioukas *et al.*, 2018). This paper proved the viability of producing realistic facial animations and spawned the field of speech-driven facial animation using neural networks, which has become increasingly competitive in recent years (Das *et al.*, 2020; Vougioukas *et al.*, 2021). Generative adversarial networks (GANs) (Goodfellow *et al.*, 2014) have been generally dominant in both modalities and have played a central role in increasing the realism of synthesized speech.

In summary, it is clear that the advent of deep learning has revolutionized the main research topics in acoustic and visual speech. Tasks such as speech recognition, which were viable

yet difficult to scale to real-world scenarios, became easy to train and achieved great leaps in performance due to the introduction of end-to-end neural architectures. On the other hand, tasks such as speech-driven facial animation, which had generally failed to achieve impressive results, managed to finally produce consistently realistic results through the use of generative adversarial networks. In short, acoustic and visual speech have been the focus of substantial research developments in the past decade, particularly due to the advances in deep learning literature. However, much of this research tends to ignore a fundamental aspect of these two modalities: they are two parts of the same whole.

Indeed, only a minority of works focus on the combination of acoustic and visual speech, commonly referred to as audio-visual speech. In recent years, some attention has been drawn to two tasks that have effectively leveraged audio-visual speech: audio-visual speech recognition and audio-visual speech enhancement. Audio-visual speech recognition aims to transcribe text from speech by using not only acoustic speech but visual speech as well (in most cases, video of the cropped mouth). Most modern approaches effectively add a visual backbone (e.g., ResNet-18 (Stafylakis and Tzimiropoulos, 2017)) to existing speech recognition pipelines (Petridis *et al.*, 2018a; Ma *et al.*, 2021d) (e.g., conformer (Gulati *et al.*, 2020)). This approach tends to yield slight improvements over audio-only speech recognition, but generally requires a larger, slower model, and therefore continues to be significantly less popular than audio-only speech recognition.

Similarly, audio-visual speech enhancement aims to extract clean speech (in waveform format) from a noisy stream of audio and a video of the speaker’s facial movements. This task attempts to improve the quality of the enhanced speech compared to traditional audio-only speech enhancement, which does not receive video as input and relies solely on the noisy audio stream. When the speech is clearly discernible from the original audio stream, audio-visual models achieve only minor improvements over their audio-only counterparts. However, audio-visual approaches yield increasing improvements as the conditions become noisier and the audio supervision becomes unreliable (Hou *et al.*, 2016; Gogate *et al.*, 2021). Furthermore, audio-visual methods excel at the removal of background speech, which is virtually impossible for single-channel acoustic models, as they have no way of distinguishing background speech from target

speech (Ephrat *et al.*, 2018; Gao and Grauman, 2021).

While these works have managed to combine acoustic and visual speech to outperform audio-only approaches, it is clear that more can be learned from audio-visual speech. Indeed, within the context of speech research, audio, and video are not only two parts of the same whole: they are complementary modalities that share some information i. e., verbal content, but also have unique attributes that are effectively absent from the other modality. For example, the pitch of someone’s voice can hardly be estimated from their mouth movements alone, but can easily be extracted from audio. On the other hand, someone’s emotional cues may be hard to estimate from audio, but can often be easily visible in a video of their face.

Recently, self-supervised learning has become increasingly popular in all areas of machine learning: natural language processing (Devlin *et al.*, 2019; Brown *et al.*, 2020), computer vision (Chen *et al.*, 2020; He *et al.*, 2020) and speech processing (Baevski *et al.*, 2020; Hsu *et al.*, 2021). In short, self-supervised learning aims to learn from data by leveraging the data itself as a supervision target i. e., , using two versions of the same unlabelled image to create an input and target simultaneously. This learning approach is uniquely valuable because it circumvents the main limitation of modern machine learning methods: the lack of labeled data. Remarkably, these self-supervised methods have yielded substantial increases in performance for important applications such as question answering (Devlin *et al.*, 2019) and image recognition (Chen *et al.*, 2020; He *et al.*, 2020).

The duality of audio-visual speech and the recent success of self-supervised learning lead us to a logical question: can acoustic and visual speech learn from each other without additional supervision? This idea differs substantially from the audio-visual tasks mentioned above, as it proposes to use one of the modalities as an objective to learn from the other, rather than simply combining them as the input. Furthermore, the use of one of the modalities as an objective naturally fits the self-supervised methodology. For instance, in Bootstrap Your Own Latent (BYOL) (Grill *et al.*, 2020) or Barlow Twins (Zbontar *et al.*, 2021), researchers are forced to elaborate two different views of the same image which are different enough to present a meaningful learning procedure. Audio-visual speech, on the other hand, presents two substantially

different yet perfectly synchronized (disregarding small discrepancies) views of the same speech sample.

1.2 Questions and Objectives

The discussion presented above poses four main questions:

1. How can visual speech leverage acoustic speech as a learning objective?

It is well known that acoustic speech contains more explicit verbal content than visual speech. For instance, most humans find it much easier to understand what someone is saying by listening to them, rather than trying to read their lips with no sound. Recent research has suggested that this assumption holds for machines as well. Indeed, speech recognition models vastly outperform lip reading models when trained and evaluated on the same corpus, with similar architectures and training procedures (Ma *et al.*, 2021d). This poses the question: can one leverage naturally paired audio-visual data to allow visual encoders to learn verbal information from audio?

2. Is it possible to synthesize realistic speech from silent video only?

As mentioned above, we aim to train a visual encoder that learns from audio. This method will attempt to learn useful information from the audio, including verbal content, intonation, accent, etc. By considering this idea to its full extent, one can consider the following edge case: is it possible to train a visual encoder that can synthesize audio directly from silent video, with no additional supervision? This can be seen as the most challenging task for a visual encoder with an acoustic objective: trying to reproduce its audio counterpart without any audio input.

3. How can both modalities learn from each other simultaneously?

While acoustic speech seems to contain information that may not be as explicit in visual speech (e.g., verbal information), the inverse is also true. Emotion, for example, is generally easier to recognize in someone’s facial movements rather than their voice.

Furthermore, recent research in emotion recognition seems to suggest this assumption also holds for machines (Schoneveld *et al.*, 2021). This leads us to the following question: if visual speech can leverage its counterpart to improve verbal encoding, can acoustic speech also leverage its counterpart to learn valuable information? Naturally, this leads to one of the fundamental questions in this thesis: can each speech modality leverage its counterpart to learn new information from audio-visual speech, simultaneously?

4. How can one leverage these models for practical applications?

As outlined in the previous section, neural networks have recently been applied to improve performance in speech applications such as speech recognition and lip reading. Furthermore, self-supervised learning on unlabelled speech has yielded substantial developments in speech recognition, advancing the state-of-the-art once more. This research direction elicits the following question: can audio-visual self-supervised learning improve performance on established acoustic speech/visual speech tasks? Furthermore, it is also relevant to consider if this learning strategy can lead to new tasks, such as video-to-speech synthesis (as proposed in question 2).

In this thesis, I aim to address these questions by focusing on the following goals:

- 1. Develop a self-supervised visual model which can leverage unlabelled audio-visual data to achieve state-of-the-art results in a visual speech application.**
- 2. Build a state-of-the-art video-to-speech synthesis system.**
- 3. Leverage my video-to-speech research to advance the state-of-the-art in audio-visual speech enhancement.**
- 4. Develop a self-supervised audio-visual model where both modalities can learn from each other simultaneously to achieve state-of-the-art results in their respective applications.**

1.3 Contributions

My work advances the state-of-the-art in self-supervised learning, video-to-speech synthesis, and audio-visual speech enhancement:

1. LiRA: Learning visual speech representations from audio through self-supervision

In Chapter 3, I present LiRA (Ma *et al.*, 2021c), a visual speech encoder that learns by leveraging acoustic speech as its training objective. This self-supervised model is based on a ResNet-18+Conformer architecture (Gulati *et al.*, 2020; He *et al.*, 2016) and learns by predicting acoustic features extracted from raw audio using a pre-trained speech encoder (PASE+ (Ravanelli *et al.*, 2020a)). The trained visual encoder is then fine-tuned by my collaborator Pingchuan Ma to perform lip reading on multiple 'in-the-wild' datasets. This approach achieves state-of-the-art lip reading results on LRS2 (Afouras *et al.*, 2018a) and outperforms previous self-supervised approaches on LRW (Chung and Zisserman, 2016a).

2. End-to-end video-to-speech synthesis

In Chapter 4, I present a new video-to-speech synthesis framework (Mira *et al.*, 2022a) that trains a generative adversarial network to reconstruct waveform speech from silent video end to end. The model is based on a convolutional encoder-decoder structure, combined with a recurrent temporal encoder. The model is trained via two separate discriminators which operate on the time/frequency domain, as well as three comparative losses on the resulting waveform. I train this model on various established datasets, achieving state-of-the-art results on multiple evaluation metrics, and study the importance of its components via several ablations studies. I also present experiments analyzing the model's ability to adapt to different vocabulary ranges and present an experiment with silent speakers.

3. SVTS: Scalable video-to-speech synthesis

In Chapter 5, I present SVTS (Mira *et al.*, 2022b): a video-to-speech synthesis model based on neural vocoders, which focuses on architectural simplicity and scalability. I apply a ResNet-18+Conformer (Gulati *et al.*, 2020; He *et al.*, 2016) architecture which

predicts spectrograms from video of the speaker’s lip movements, as well as a speaker embedding extracted using a pre-trained speaker encoder (Jia *et al.*, 2018). The predicted spectrograms are then converted into waveform using a pre-trained neural vocoder - Parallel WaveGAN (Yamamoto *et al.*, 2020). I present three versions of this model (small, medium, and large), and apply them to datasets of varying sizes. Experiments show that SVTS substantially outperforms previous works on small and medium datasets (GRID (Cooke *et al.*, 2006a) and LRW (Chung and Zisserman, 2016a)), and is the first to produce intelligible results for LRS3 (Afouras *et al.*, 2018b) - a very large dataset with an unconstrained vocabulary and thousands of speakers. I also study the importance of the neural vocoder in this framework by comparing Parallel WaveGAN with other pre-trained vocoders, as well as the established Griffin-Lim algorithm (Griffin and Lim, 1983).

4. LA-VocE: Low-SNR audio-visual speech enhancement using neural vocoders

In Chapter 6, I present LA-VocE (Mira *et al.*, 2023), a new two-stage approach that predicts mel-spectrograms from noisy audio-visual speech via a transformer-based architecture, and then converts them into waveform audio using a neural vocoder (HiFi-GAN (Kong *et al.*, 2020a)). I train and evaluate this framework on thousands of speakers and 11+ different languages, and study this model’s ability to adapt to different levels of background noise and speech interference. Experiments show that LA-VocE outperforms existing methods according to multiple metrics, particularly under very noisy scenarios.

5. RealForensics: Leveraging real talking faces via self-supervision for robust forgery detection

In Chapter 7, I collaborate with my colleague Alexandros Haliassos to design, implement and train the self-supervised audio-visual framework in RealForensics (Haliassos *et al.*, 2022) (stage 1). This framework is composed of two speech encoders (one for visual speech and one for acoustic speech) which learn meaningful representation by predicting each other’s features. Specifically, the student from one modality learns to maximize the cosine similarity between its predicted features and the features extracted from the other modality’s teacher, which is a slow-moving copy of the student. We trained these models on unlabelled audio-visual data from LRW and VoxCeleb2. Alex Haliassos then

fine-tuned these trained models for deepfake detection, achieving state-of-the-art results for multiple established datasets.

6. RAVEn: Jointly learning visual and auditory speech representations from raw data

In Chapter 7, I collaborate with my colleague Alexandros Haliassos to design, implement and train RAVEn (Haliassos *et al.*, 2023), a new audio-visual self-supervised learning approach that operates on raw waveform and video. As in RealForensics (Haliassos *et al.*, 2022), this framework features student networks that learn by predicting the features from the other modality’s teacher (slow-moving copy of the student). We present many additions and improvements to the original structure, the most important of which are: **(1)** Inspired by (He *et al.*, 2022), we experiment with masked inputs for the students on each modality and find that this improves performance **(2)** Inspired by recent successes in audio-only self-supervised learning (Baevski *et al.*, 2020; Hsu *et al.*, 2021), the audio student now predicts not only the video teacher’s features but also his own teacher’s features. **(3)** We remove the projector from each encoder, as we find this yields better results. My collaborators Alexandros Haliassos and Pingchuan Ma then fine-tune these models for speech recognition and lip reading, and achieve state-of-the-art performance on multiple datasets, outperforming previous self-supervised methods.

1.4 Publications

1. LiRA: Learning Visual Speech Representations from Audio through Self-supervision

Pingchuan Ma*, **Rodrigo Mira***, Stavros Petridis, Björn W. Schuller, Maja Pantic
Interspeech 2021

* *denotes equal contribution*

2. End-to-end Video-to-speech Synthesis using Generative Adversarial Networks

Rodrigo Mira, Konstantinos Vougioukas, Pingchuan Ma, Stavros Petridis, Björn W.

Schuller, Maja Pantic

IEEE Transactions on Cybernetics

3. SVTS: Scalable Video-to-Speech Synthesis

Rodrigo Mira, Alexandros Haliassos, Stavros Petridis, Björn W. Schuller, Maja Pantic

Interspeech 2022

4. LA-VocE: Low-SNR audio-visual speech enhancement using neural vocoders

Rodrigo Mira, Buye Xu, Jacob Donley, Anurag Kumar, Stavros Petridis, Vamsi Krishna Itapu, Maja Pantic

International Conference on Acoustics, Speech, & Signal Processing (ICASSP) 2023

5. Leveraging Real Talking Faces via Self-Supervision for Robust Forgery Detection

Alexandros Haliassos, **Rodrigo Mira**, Stavros Petridis, Maja Pantic

Computer Vision and Pattern Recognition (CVPR) 2022

6. Jointly Learning Visual and Auditory Speech Representations from Raw Data

Alexandros Haliassos, Pingchuan Ma, **Rodrigo Mira**, Stavros Petridis, Maja Pantic

International Conference on Learning Representations (ICLR) 2023

CHAPTER 2

BACKGROUND

Contents

2.1	Datasets	15
2.1.1	Studio-recorded	16
2.1.2	In-the-wild	18
2.2	Data Processing	21
2.2.1	Audio	22
2.2.2	Video	25
2.3	Modelling	28
2.3.1	Linear layers	28
2.3.2	Convolutional layers	30
2.3.3	Sequence modelling	35
2.3.4	Generative Adversarial Networks	44
2.4	Evaluation metrics	50
2.4.1	Speech quality metrics	50

2.4.2	Speech intelligibility metrics	52
2.4.3	Other metrics	54

This chapter aims to introduce the fundamental concepts and methods used for the remainder of this thesis. We begin by describing the main audio-visual datasets we use to train and evaluate our models. We omit audio-only datasets, as well as deepfake detection datasets (described in Chapter 7), as they are not central to the approaches we present. We then go on to describe our data processing tools for acoustic and visual speech. We describe the main layers that compose our models, exploring how/why they were designed and what purpose they fulfill. Finally, we describe every evaluation metric used to evaluate the synthesized speech.

2.1 Datasets

In this section, we describe the audio-visual datasets (i.e., which contain acoustic and visual speech) used in the remaining chapters. We divide these datasets into two types: 'studio-recorded' datasets and 'in-the-wild' datasets. Studio-recorded datasets are recorded in optimal conditions, with high-quality video and audio streams and minimal acoustic/visual noise. Apart from generally being cleaner, the samples in these datasets are recorded by the same researchers, using the same equipment, in the same location. This consistency makes for a homogeneous corpus, which is useful in research scenarios. In-the-wild datasets, on the other hand, are gathered from real-world recordings (e.g., Youtube videos) and are generally lower in quality. They often feature many speakers recorded using different equipment in vastly different environments. This heterogeneity yields a much noisier corpus, which is more difficult to study and evaluate. However, these have two major advantages over studio-recorded datasets:

1. They are easier to scale

AVSpeech, for example, features roughly 4700 hours of audio-visual speech clips. These are automatically extracted from Youtube videos using a data collection pipeline that involves minimal human feedback (Ephrat *et al.*, 2018). Conversely, creating a studio-recorded dataset of this dimension would require an unreasonably large amount of effort

Table 2.1: Summary of the contents of each audio-visual dataset we describe in this section.

Name	# Languages	# Clips	# Hours	# Words	# Speakers
Studio-recorded					
GRID	1 (English)	33,000	27	51	33
TCD-TIMIT	1 (English)	13,826	7	N/A	62
In-the-wild					
LRW	1 (English)	>500,000	173	500	>1,000
LRS2	1 (English)	144,482	225	>40,000	N/A
LRS3	1 (English)	151,819	438	>51,000	>5,000
Voxceleb2	1 (English)	1,128,246	2442	N/A	6,112
AVSpeech	11+	>2,000,000	~4700	N/A	>150,000

and investment, as each clip would have to be manually recorded. This is why all large-scale audio-visual speech datasets available today are in-the-wild.

2. They are closer to the 'real world'

While studio-recorded datasets present a consistent set of samples that may be used to evaluate a research idea (e. g., is it possible to synthesize speech from silent video only), they are mostly useful for proofs of concept. This is because, in real-world applications, models must learn to adapt to different scenarios, e. g., if a lip reading model only works in perfect studio conditions, its applications will be severely limited. With this in mind, training and evaluating our models on in-the-wild datasets presents a much more realistic challenge and allows us to consider broader applications.

The datasets presented below are summarized in Tables 2.1 and 2.2.

2.1.1 Studio-recorded

GRID

GRID (Cooke *et al.*, 2006a) contains roughly 33,000 3-second long clips, amounting to around 27 hours of audio-visual speech. The 33 participants are told to utter short sentences which are formed using a sequence of 6 simple words sampled from a small vocabulary (51 words in total),

Table 2.2: Summary of the technical details of each audio-visual dataset we describe in this section. In the first row, “fps” stands for frames per second, and “sr” stands for sampling rate.

Name	Video resolution	Video fps	Audio sr (Hz)	Source
Studio-recorded				
GRID	720×576	25	50,000	Studio
TCD-TIMIT	1920×1080	30	48,000	Studio
In-the-wild				
LRW	256×256	25	16,000	TV (BBC)
LRS2	160×160	25	16,000	TV (BBC)
LRS3	224×224	25	16,000	Youtube (TED talks)
Voxceleb2	224×224	25	16,000	Youtube
AVSpeech	Variable	25	16,000	Youtube

e. g., “Bin red in B 9 now.”. All samples are recorded in the same acoustically isolated booth using professional equipment (microphone and camera). The video is recorded with 720×576 resolution and 25 frames per second (fps), and the audio is sampled at 50,000 Hz. The lighting, camera position, microphone position, and head pose were kept consistent throughout the entire recording procedure.

TCD-TIMIT

TCD-TIMIT (Harte and Gillen, 2015) contains 13,826 clips of varying lengths, amounting to around 7 hours of audio-visual speech. The corpus features 59 volunteers, each uttering 98 different sentences, and 3 professional lip speakers, each uttering 377 sentences. The sentences are obtained from the original TIMIT dataset, which features an unconstrained vocabulary and is designed to include as many phoneme pairs as possible. Each speaker is recorded from two different angles: frontal, and 30 degrees to the speaker’s right. The video and audio were recorded in an office with no soundproofing, using professional cameras and microphones. Both video streams were recorded with 1920×1080 resolution and 30 fps, and the audio stream was sampled at 48,000 Hz. As in GRID, lightning, camera position, microphone position, and head pose were regularly checked and kept consistent during the recording procedure.

2.1.2 In-the-wild

LRW

Lip Reading in the Wild (LRW) (Chung and Zisserman, 2016a) contains roughly 500,000 1.16-second-long clips (29 frames each, featuring a single word), amounting to more than 150 hours of audio-visual speech. It features a vocabulary of 500 words (e. g., “Already”, “Westminster”), with approximately 1,000 utterances per word uttered by hundreds of different speakers. The clips are extracted from BBC TV broadcasts of news programs and debates. For each clip, the face is detected, cropped from each frame, and resized to 256×256 resolution (25 fps). The quality of each video is highly dependent on the size of the face in the original video, as well as the recording equipment that was used. Overall, the video quality is moderately high, with clearly discernible facial movements. The audio is sampled at 16,000 Hz and is also inconsistent, as it is recorded in a wide range of different environments, including outdoors. The video often features jittery movements centered around the face, which is a consequence of the landmark extraction process. The audio, on the other hand, frequently contains some background noise from the recording environment but is consistently intelligible in virtually every sample. The lighting, camera position, and microphone position vary greatly from clip to clip, but the head pose is generally constrained to be near-frontal. The corpus is split into “train” (training), “val” (validation), and “test” (testing) sets, which contain 800-1000, 50, and 50 utterances per word, respectively.

LRS2

Lip Reading Sentences 2 (LRS2) (Afouras *et al.*, 2018c) contains 144,482 clips of varying lengths, amounting to around 225 hours of audio-visual speech. The sentences are clipped to 100 characters or 10 seconds in length, feature a vocabulary of more than 40,000 words, and are uttered by thousands of different speakers. The videos are extracted from a variety of programs from BBC TV broadcasts. Similarly to LRW, the face is extracted from each video using an automatic preprocessing pipeline and resized to 160×160 resolution (25 fps). The audio is

sampled at 16,000 Hz. The video and audio are of similar to LRW and suffer largely from the same jitter/background noise problems. Despite this, the facial movements are generally well-preserved and the audio is intelligible. As explained above, recording conditions vary substantially between clips, as they are from different programs recorded at different times in different environments. The corpus is split into “pre-train” (pre-training), “train”, “val”, and “test” sets, which contain 96,318, 45,839, 1,082, and 1,243 utterances, respectively.

LRS3

Lip Reading Sentences 3 (LRS3) (Afouras *et al.*, 2018a) contains 151,819 clips of varying lengths, amounting to around 438 hours of audio-visual speech. As in LRS2, the sentences have a very large vocabulary (more than 50,000 words) and are uttered by thousands of speakers. The videos are extracted from YouTube videos of TED/TEDx talks available online. As presented above, the face is extracted from each clip using an automatic preprocessing pipeline and resized to 224×224 resolution (25 fps). The audio is sampled at 16,000 Hz. In this dataset, the video quality is more inconsistent compared to LRW and LRS2, to the point where, in some videos, the speaker’s lip movements are difficult to recognize. The audio, on the other hand, is consistently clear and contains minimal noise, which is unsurprising given the TED talk format. The head poses also appear to have a larger variance - some videos contain segments with near-profile head poses. These adverse conditions make this dataset more challenging (e. g., for lip-reading (Ma *et al.*, 2021d)) than LRW or LRS2. The corpus is split into “pre-train”, “trainval”, and “test” sets, which contain 118,516, 31,982, and 1,321 utterances, respectively. The sentences from the “trainval”, and “test” sets are clipped to 100 characters or 6 seconds, whereas the ones from the “pre-train” set are often longer.

VoxCeleb2

VoxCeleb2 (Chung *et al.*, 2018b) contains 1,128,246 clips of varying lengths, amounting to 2,442 hours of audio-visual speech. It features an unconstrained vocabulary and more than 6,000 speakers. The clips are extracted from videos of celebrities downloaded from YouTube.

Similarly to the other in-the-wild datasets, the faces are detected and cropped from each video using an automatic preprocessing pipeline and resized to 224×224 resolution (25 fps). The audio is sampled at 16,000 Hz. As in LRS3, the video of the speaker’s cropped face is occasionally low in quality, meaning that their facial movements are more difficult to understand. Unlike LRS3, the audio quality is also inconsistent, with some samples featuring substantial background noise and static. The head poses and lighting conditions also vary greatly throughout the dataset. Overall, VoxCeleb2 is one of the largest audio-visual datasets available online today and is quite heterogeneous in its contents. The corpus is split into “dev”, and “test” sets, which contain 1,092,009 and 36,237 utterances, featuring 5,994 and 118 speakers, respectively.

AVSpeech

AVSpeech (Ephrat *et al.*, 2018) contains more than 1 million 3- to 10-second long clips, amounting to roughly 4,700 hours of audio-visual speech. It features an unconstrained vocabulary and approximately 150,000 different speakers. Notably, unlike all datasets presented above, which exclusively contain speech in English, AVSpeech features 11+ different languages including Spanish, Japanese, and Russian. The clips are extracted from 290,000 YouTube videos (ranging from video lectures to TED talks) using an automatic data collection pipeline. This preprocessing discards videos which are blurry, poorly lit, or near-profile, as well as audio clips with a low signal-to-noise ratio (SNR). The videos are uncropped and vary greatly in resolution. Fps and audio sampling rate are also unspecified, although the authors choose to resample all videos to 25 fps and audio to 16,000 Hz in their application (Ephrat *et al.*, 2018). As in VoxCeleb2, the video quality can at times be quite deteriorated, and, despite the SNR filter in preprocessing, the audio often contains substantial background noise and reverberation. The head poses and lightning conditions also vary greatly but are generally adequate thanks to the filtering during data collection. AVSpeech is today the largest publicly available audio-visual speech dataset and therefore plays a central role in modern audio-visual deep learning research.

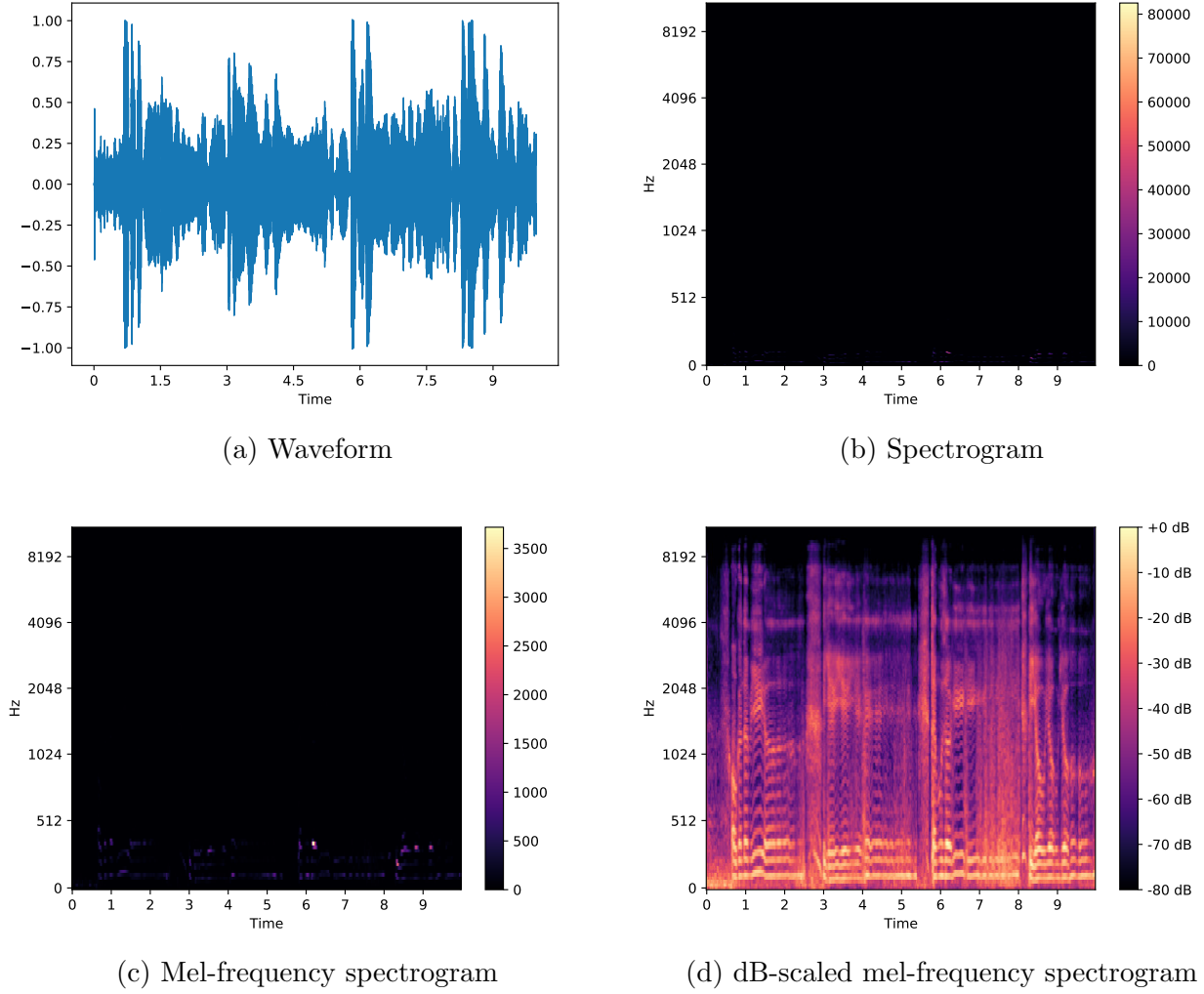


Figure 2.1: Illustration of the step-by-step transformation from waveform (a) to dB-scaled mel-frequency spectrogram (d).

2.2 Data Processing

In this section, we describe the data processing techniques that are used throughout this thesis to transform raw audio and video into representations that we can more easily exploit via our deep learning models.

2.2.1 Audio

Spectrogram

In computing, audio files are universally stored, played, and shared in the waveform domain, via many formats ranging from *.wav* to *.mp3*. Although this format is convenient for playing and recording audio on digital devices, it is not particularly helpful when trying to analyze its contents. This is because humans' perception of sound depends on the frequencies that this waveform contains, which are not explicitly represented in the waveform format. For this reason, researchers use the discrete-time Short-Time Fourier Transform (STFT) (Gold *et al.*, 2011), which allows us to visualize the frequency and phase of the signal via the following equation:

$$STFT(x, f, m) = \sum_{k=0}^{w_{len}-1} w[k]x[mh+k]e^{-i\frac{2\pi fk}{w_{len}}} \quad (2.1)$$

where f is the frequency we are analyzing, m is the index of the sliding window over which we are computing the Fourier transform, which we refer to as 'time-step', w and w_{len} are the window function (typically Hann window) and its length, respectively, h is the hop size, and x is the input waveform. We can then compute the spectrogram by computing the squared magnitude of the Fourier transform:

$$S(x, f, m) = |STFT(x, f, m)|^2 \quad (2.2)$$

where S is the spectrogram function. Finally, this allows us to visualize the magnitude of each frequency at each time step, as presented in Figure 2.1b. There are then two adjustments that can be performed to make the spectrogram more useful. The first is to convert it into the Mel scale, which scales the frequencies based on human pitch perception (Gold *et al.*, 2011; Huang *et al.*, 2001) via the following formula:

$$f_{mel} = 2595 \log_{10} \left(1 + \frac{700}{f} \right) \quad (2.3)$$

where f is the frequency in Hertz (Hz), and f_{mel} is the frequency in mel scale. Although

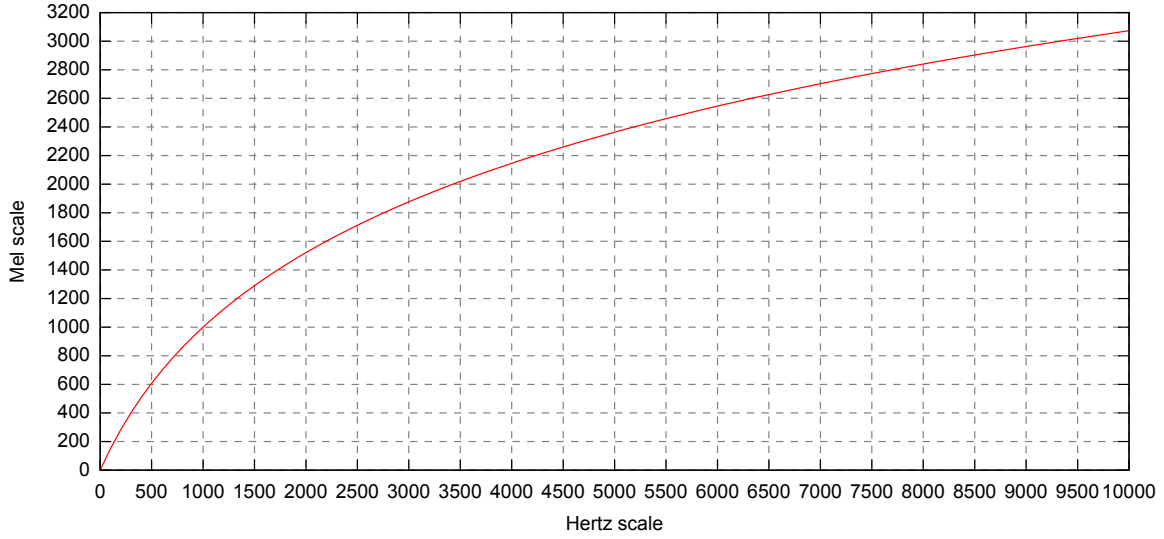


Figure 2.2: Comparison between Hertz scale and Mel scale. Source: https://en.wikipedia.org/wiki/Mel_scale.

other formulas have been proposed (Gold *et al.*, 2011), this is the most widely used and is implemented in well-known packages such as PyTorch (Paszke *et al.*, 2017) and Librosa (McFee *et al.*, 2022). For these reasons, this is the formula we will be using for the remainder of this thesis. A comparison between the Hertz scale and the mel scale is shown in Figure 2.2. In practice, this conversion is done by applying a set of m mel scale triangular filter banks which convert the original spectrogram into a mel frequency spectrogram (Gold *et al.*, 2011; Huang *et al.*, 2001). The second adjustment is to convert the power spectrogram (amplitude squared), as it is computed in Equation X, into the decibel (dB) scale, by scaling them logarithmically:

$$S_{dB}(x, f, m) = 10 \log_{10}(S(x, f, m)) \quad (2.4)$$

where S_{dB} is the dB-scaled spectrogram. The decibel scale is widely used in many signal processing applications and allows us to visualize the power of each frequency within a narrower range. The step-by-step conversion process from raw waveform to dB-scaled mel-frequency spectrogram is illustrated in Figure 2.1.

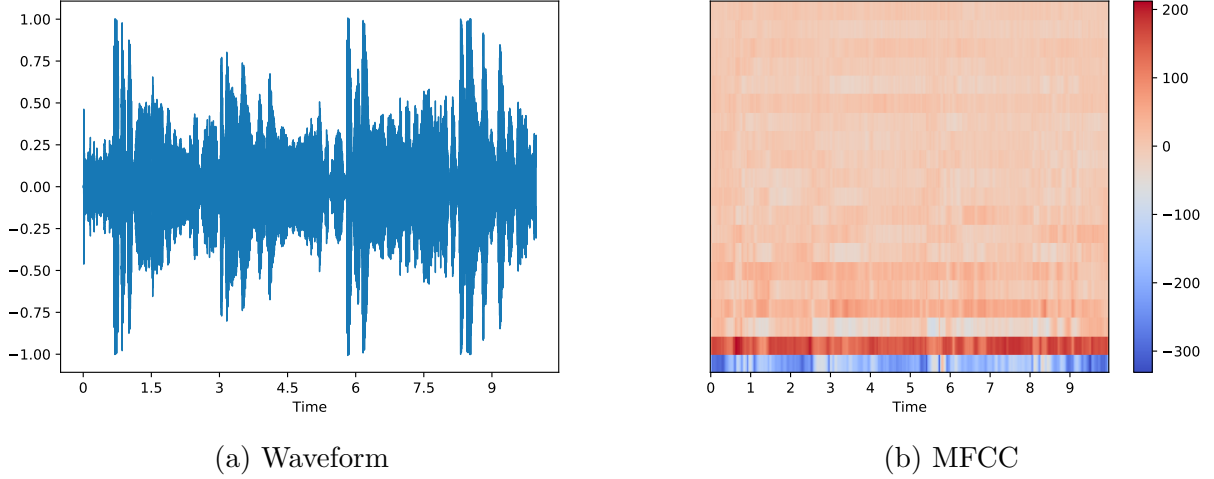


Figure 2.3: Conversion from waveform (a) to MFCC (b).

MFCC

The Mel-Frequency Cepstrum Coefficients (MFCC) are a more elaborate speech representation that was originally shown to be effective in speech recognition (Huang *et al.*, 2001; Davis and Mermelstein, 1980). Since then, MFCCs have been widely used as an auxiliary representation in many impactful tasks including speaker identification (Kamruzzaman *et al.*, 2010), speech recognition (Ittichaichareon *et al.*, 2012), and emotion recognition (Kishore and Satish, 2013). It is obtained by computing the discrete cosine transform (DCT) of the dB-scaled mel-frequency cepstrum (as presented above):

$$MFCC(x, n) = \sum_{m=0}^{M-1} S_{mel, dB}[m] \cos\left(\frac{\pi n(m + 0.5)}{M}\right) \quad (2.5)$$

where $MFCC$ is the MFCC function, x is the input signal, M is the number of mel-scale filter banks and $S_{mel, dB}$ is the dB-scaled mel-spectrogram. n varies from 0 to N_{MFCC} , where N_{MFCC} is the number of coefficients we are computing (this is typically set to 13 in most speech-related applications (Huang *et al.*, 2001)). An example of the conversion from waveform to MFCC is presented in Figure 2.3.



Figure 2.4: Face detection example using RetinaFace. Source: <https://github.com/serengil/retinaface>

2.2.2 Video

For the purposes of this thesis, we are interested in videos of people’s facial movements while they are talking, also known as visual speech. As mentioned in Section 2.1, the audio-visual datasets we approach in this thesis are carefully designed to include videos of people talking where the face and lip movements are clearly visible, the quality/resolution of the video is reasonable, the audio is intelligible and does not contain major noise sources, and the audio and video streams are synchronized. However, even with all these aspects accounted for, it is still necessary to extract a consistently aligned face from each frame of each video, as the original is typically misaligned and contains other visual cues which are irrelevant for visual speech analysis, e.g., the objects in the background. We extract the face via a pre-processing pipeline with three major steps: face detection, landmark detection, and alignment. We may also choose to crop the mouth at the end if we are only interested in lip movements.

Face detection

We begin by detecting the face in the video, which in practice is done by estimating a bounding box around the speaker’s face in each video frame. Throughout the chapters of this thesis, two different methods are applied for face detection: dlib (King, 2009) and RetinaFace (Deng *et al.*, 2020). dlib is a well-established C++ toolkit that encapsulates a wide variety of methods that can be useful for practical applications, such as face detection. dlib’s default face detection function (which is used in Chapters 3 and 4) is based on a combination of Histograms of Oriented Gradients (HOG) (Dalal and Triggs, 2005) and Linear Support Vector Machines (SVM) (Hearst *et al.*, 1998). Despite not depending on any neural networks, this method is reasonably accurate, fast, and efficient.

RetinaFace is a convolutional model trained for face detection via a multi-task framework that performs face classification, face box regression, facial landmark regression, and dense face regression simultaneously. This method is trained and evaluated on a very large annotated dataset (WIDER FACE (Yang *et al.*, 2016)), where it achieves state-of-the-art performance. RetinaFace has since become an established face detection framework that is widely used for many applications, including Chapters 5, 6, and 7. For our work, it is assumed that there is only one speaker in each video - if two speakers are detected, the video is discarded. An example of face detection using RetinaFace is presented in Figure 2.4.

Landmark estimation

We proceed by estimating the facial landmarks on the detected face, which are a set of 68 points on the speaker’s face (examples are pictured in Figure 2.5) that will be used for alignment and cropping later. Throughout this thesis, two methods are used for landmark estimation: dlib (King, 2009) (used in Chapters 3 and 4) and 2D-Face Alignment Network (FAN) (Bulat and Tzimiropoulos, 2017a) (used in Chapters 5, 6, and 7). dlib’s landmark estimation model is based on an Ensemble of Regression Trees (ERT), as proposed in (Kazemi and Sullivan, 2014). Again, despite not relying on deep learning, this method is shown to be extremely fast and



Figure 2.5: 68-point landmark estimation examples using a 2D-FAN. Source: <https://github.com/1adrianb/face-alignment>

generally provides accurate landmark estimates in real-world scenarios.

2D-FAN, on the other hand, is a model composed of a stack of four HourGlass Networks (Newell *et al.*, 2016) that predicts the 2D coordinates of each landmark. This network is trained and evaluated on a very large, synthetically augmented landmark dataset (300-W-LP (Zhu *et al.*, 2016)) via a simple Mean Squared Error (MSE) loss, and substantially outperforms other approaches in terms of area under the receiver operating curve (AUC - this metric is described in Section 2.4.3). Due to its impressive performance and open-source implementation¹, FAN is commonly used to extract facial landmarks in many research projects, including Chapters 5, 6, and 7.

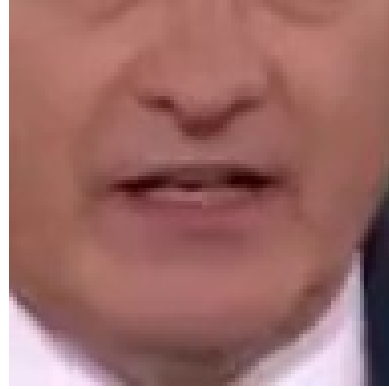
Smoothing, alignment, and cropping

After computing the 2D landmarks for each frame of each video, we must perform smoothing to reduce motion jitter in the resulting video. We do this by computing the average landmarks over time using a 12-frame sliding window. Then, we align each frame of each video with the mean face by performing an affine transformation, which ensures all faces in all videos will be aligned with the same reference. In our case, the mean face is computed by averaging the landmarks from every video in LRW (Chung and Zisserman, 2016a). Finally, the aligned face, together with the estimated landmarks, can be used to crop out our region of interest (ROI), which will be either the full face or the mouth, depending on the application. Examples of the

¹<https://github.com/1adrianb/face-alignment>



(a) Cropped face



(b) Cropped mouth

Figure 2.6: Examples of a pre-processed cropped face from GRID (Cooke *et al.*, 2006a) and cropped mouth from LRW (Chung and Zisserman, 2016a).

cropped face and mouth are shown in Figure 2.6.

2.3 Modelling

In this section, we will focus on the components that make up our models in each of the chapters below. Fundamentally, our neural networks feature four types of layers: linear layers, convolutional layers, recurrent layers, and attention layers.

2.3.1 Linear layers

The linear layer (also known as the fully connected layer) is the most basic neural layer described in this thesis. In short, this layer features two vectors of neurons that are fully connected, i. e., each neuron on each vector is connected to every neuron on the other vector, such that each output neuron depends on every input neuron, as shown in Figure 2.7. Each connection is modeled via trainable weights and biases such that:

$$y_j = w_{i,j}x_i + b_j \quad (2.6)$$

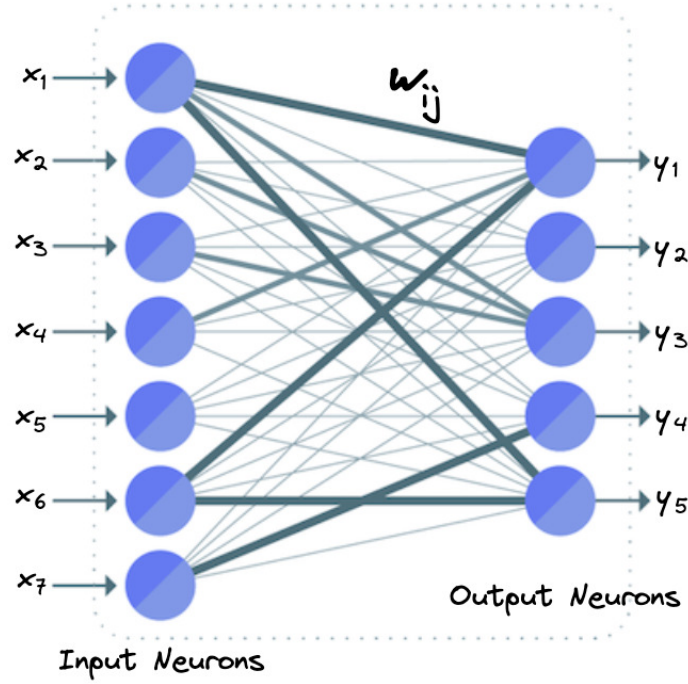


Figure 2.7: A simple illustration of a linear layer. Source: <https://www.baeldung.com/cs/neural-networks-conv-fc-layers>

where x_i is the i -th input neuron, y_j is the j -th output neuron, $w_{i,j}$ is the weight between the two neurons, and b_j is the bias associated with the output neuron. It is worth noting that the linear layer can have different numbers of input and output neurons, meaning that it can connect two tensors with different latent dimensions. Linear layers may be combined sequentially to form what is commonly referred to as a multilayer perceptron (MLP) (Heaton, 2018), which is effectively a stack of linear layers.

For the purposes of this thesis, linear layers are generally used as projectors, i.e., they are used to project some output features into a new desired dimensionality. A good example is the linear projection layer in Chapter 5, Figure 5.1. In this case, the conformer’s output feature has a hidden dimension of 256, but we would like to generate 4 spectrogram frames, which would be a hidden dimension of 320. To project our hidden size into the desired dimension, we use a linear layer with an input size of 256 and an output size of 320, which is a simple and lightweight solution. Linear layers are also used heavily in transformers, as we will discuss later in Section 2.3.3.

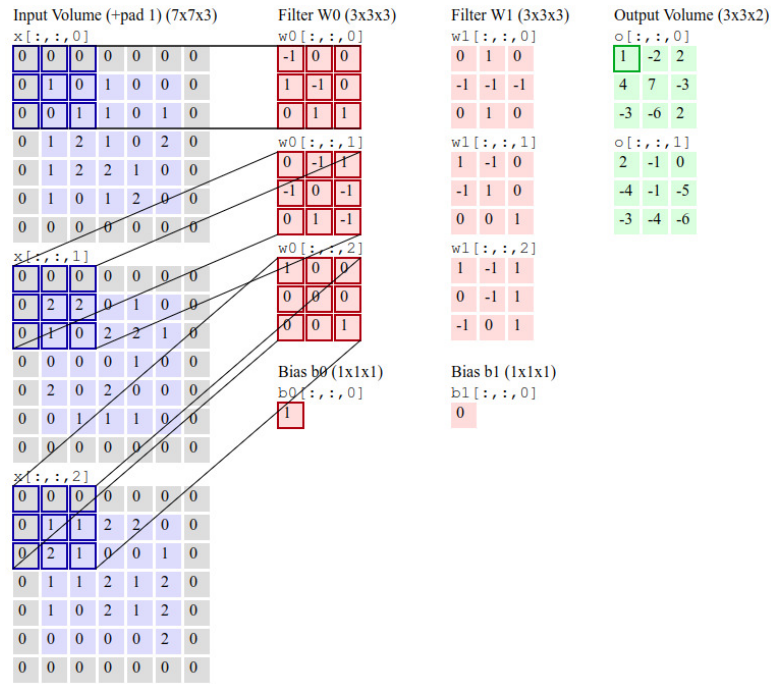


Figure 2.8: A simple example of a 2D convolution. In this case, our input has 2 dimensions (height and width) and 3 channels, and there are 2 kernels of size 3×3 . The stride is set to 2 and the padding is set to 1, so the output is of size 3×3 . Source: <https://cs231n.github.io/convolutional-networks/>

2.3.2 Convolutional layers

While linear layers generally work well for low-dimensional inputs, such as latent vectors, our work in this thesis adopts modalities with high dimensionality such as waveforms, spectrograms (which, from a modeling perspective, can be treated as 2D images), and video. Therefore, we need spatial models that can efficiently encode these modalities into useful features, as well as decode these features into larger dimensions.

Convolutions

In the context of deep learning (Heaton, 2018), the convolution is an operation that applies a sliding window of trainable weights and biases (known as the *kernel*) on an input tensor, resulting in a new set of features. Although this method can be applied for any number of dimensions, we can proceed by looking at a 2D image example, as this was the first application for convolutional neural networks (CNNs) (LeCun *et al.*, 1989). If we have a 64×64 image and

apply a kernel of size 3×3 with the stride set to 1, this will result in a set of features of the same dimension (64×64), assuming we choose to add a padding of 1 to the input image, which is an adjustable parameter in the convolution. In the resulting set of features, each value will depend on the corresponding pixel in the input image, as well as its 8 neighboring pixels - this is known as the receptive field. An example of a 2D convolution is shown in Figure 2.8.

If we increase the kernel size to 5×5 , each feature point will depend on a larger receptive field of 25 pixels, which means that each feature will take into account a larger spatial context, and therefore is likely to focus on higher level visual cues. We may also adjust how the sliding window will move through the input, by adjusting the stride. If we choose a stride of 1, the kernel will move 1 pixel to the right (or down, if it reaches the end of the row) at each step, resulting in a set of features of size 64×64 (if padded accordingly). However, if we set the stride to 2 (and adjust the padding accordingly), the kernel will slide by 2 pixels each step, resulting in a set of features of size 32×32 . In short, the convolution allows us not only to produce features that contextualize each pixel with its surrounding receptive field but also allows us to produce a compressed set of features with lower dimensionality.

Each convolution layer can apply multiple trainable kernels, which will result in multiple sets of features. For example, if we apply a convolutional layer with stride 1 and N kernels on an RGB image of size $3 \times 64 \times 64$, this will result in a set of features of size $N \times 64 \times 64$. These layers may then be stacked sequentially, as in the MLP presented above, to form powerful deep visual encoders. After propagating through this deep network, the resulting features will yield a very large receptive field, as they will depend on, e. g., 3×3 features from the previous layer, each of which depends on 3×3 features from the previous layer, etc. Convolutional layers are typically followed by non-linear activation functions such as ReLU (Nair and Hinton, 2010) or Swish (Ramachandran *et al.*, 2018). Many convolutional architectures (such as ResNet (He *et al.*, 2016)) also apply batch normalization (Ioffe and Szegedy, 2015) after each layer, which normalizes the features over the batch dimension. This has been shown to improve training stability and yield better performance on the downstream task.

In this thesis, convolutional layers are useful as feature encoders for audio, e. g., , 1D convo-

lutions in the waveform discriminator in Chapter 4 and in the audio waveform encoder (1D ResNet-18) in Chapter 7; image, e.g., 2D convolutions in the power critic in Chapter 4 and in the spectrogram encoder in Chapter 7; and video, e.g., 3D front-end convolutions in the video encoders in Chapters 4, 5, 3, 7 and 6. In short, convolutional layers serve as effective feature encoders for all modalities because:

1. They can easily be adapted for different dimensions (1D, 2D, or 3D).
2. When stacked, convolutions produce features with a very wide receptive field, which allows them to take into account large contextual windows in the input.
3. They are able to encode high-dimensional inputs into compressed low-dimensional latent features, which can more easily be fed into other models, such as transformers.
4. They enable weight sharing (i.e., each kernel's weights are used to process the entire input, and are therefore 'shared' between many sets of inputs), which leads to highly efficient architectures.

Transposed convolutions

Transposed convolutions can be seen as the inverse of the convolutions presented above. We apply a sliding kernel, as before, but instead of encoding features using this kernel as a receptive field on the input, it projects each feature in the input into a new wider set of features, as presented in Figure 2.9. The kernel size, stride, and padding function as expected having the inverse effect of the convolution, e.g., a 2D transposed convolution with kernel size 3×3 , stride 2 and output padding 1, when applied on an input of size 64×64 will output features of size 128×128 .

In this thesis, 1D-transposed convolutions are used in the decoder in Chapter 4 and in the neural vocoder in Chapter 6. In short, transposed convolutions are effective at decoding latent features into higher dimensional modalities (waveform audio, in this case), mirroring the convolution's feature encoding capabilities.

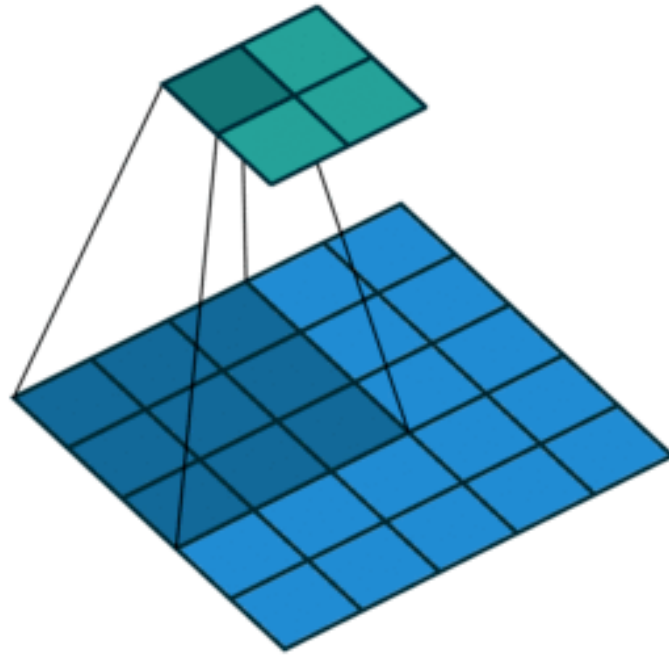


Figure 2.9: A simple example of a 2D transposed convolution. In this case, our input is of size 2×2 , and there is one kernel of size 3×3 . The stride is set to 2, so the output is of size 5×5 . Source: <https://blog.paperspace.com/transpose-convolution/>

AlexNet, ResNet and beyond

After their inception in 1989 (LeCun *et al.*, 1989), convolutional neural networks became an active research field within the computer vision community. However, it was only with the introduction of AlexNet (Krizhevsky *et al.*, 2017), which set a new state-of-the-art on the ImageNet Large Scale Visual Recognition Challenge (ILSCRC) (Russakovsky *et al.*, 2015), that CNNs began to dominate the image classification landscape. The next years saw successive iterations of similar architectures, such as GoogleNet (Szegedy *et al.*, 2015) and VGG (Simonyan and Zisserman, 2015), which substantially outperformed AlexNet. These models built upon the fundamental ideas of LeNet (LeCun *et al.*, 1989) (convolutional layers intertwined with pooling layers) by adding new activation functions such as ReLU, among other tweaks.

In 2015, ResNets became the new state-of-the-art on ImageNet by proposing one simple development which had a dramatic impact on the future of deep neural network modeling: residual connections. In short, instead of propagating the signal (and backpropagating the gradients)

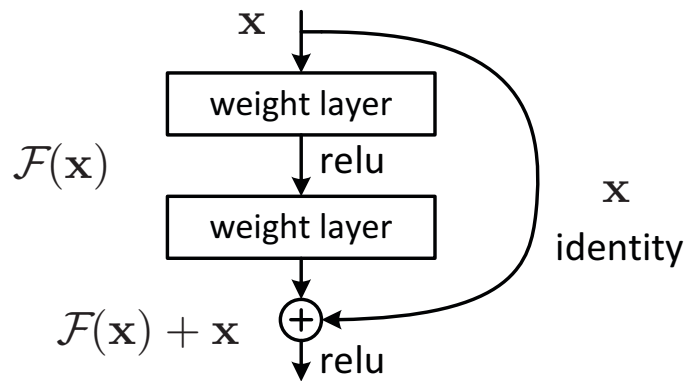


Figure 2.10: Simple example of a residual connection. Source: He *et al.* (2016)

through the model in a single line from input to output, as previous models had done, He *et al.* (2016) proposed to add residual connections between layers. Specifically, ResNets are made of residual blocks - sets of convolutional layers which, apart from being connected sequentially, as usual, have a residual connection that connects the beginning (before the first layer) to the end (after the last layer) of the block, as pictured in Figure 2.10.

He *et al.* (2016) propose ResNets of varying sizes (as in Simonyan and Zisserman (2015)), ranging from the 18-layer ResNet-18 to the 152-layer ResNet-152. ResNets were shown to outperform all previous methods on ImageNet classification and went on to win first place in ILSRC 2015. Furthermore, residual connections are shown to be one of the main factors behind the ResNet’s superior performance, via multiple ablation studies that compare ResNets to their ‘plain’ counterparts without residual connections.

He *et al.* (2016) suggest that residual connections are a way to combat the degradation of the signal that is being propagated through deep networks by adding identity mappings from one block of layers to the next. Despite this convincing motivation, there has been extensive discussion around the exact reasons why the residual connection is so successful. For instance, Li *et al.* (2018) clearly show that residual connection creates a loss landscape that is easier to optimize but do not provide conclusions on why this is the case. On the other hand, Veit *et al.* (2016) claim that ResNets effectively behave like ensembles of smaller networks, and solve the issue of vanishing gradients through the shorter paths that are created by the residual

connections.

Since their introduction in 2015 (He *et al.*, 2016), ResNets have become standard deep learning models for speech recognition (1D) (Saon *et al.*, 2017), image classification (2D) (Kolesnikov *et al.*, 2020) and action (3D) recognition (Xie *et al.*, 2017), among many other tasks. They have been followed by many successors including ResNeXt, EfficientNet, and ConvNeXt, which borrow from the original ResNets, featuring blocks of convolutions with residual connections. New transformer-based vision models have also been proposed, such as the original ViT (Dosovitskiy *et al.*, 2021) and the recent Swin Transformer (Liu *et al.*, 2022). Despite these developments, state-of-the-art lip reading models continue to use a combination of a 3D front-end layer with a 2D ResNet-18 (Stafylakis and Tzimiropoulos, 2017) as their visual backbone (Ma *et al.*, 2021a, 2022b). Given its success in lip reading and relatively low number of parameters, the 3D+2D ResNet-18 is also widely adopted as the preferred visual speech encoder throughout the course of this thesis.

2.3.3 Sequence modelling

In this thesis, we mainly work with audio and video - two modalities with important temporal aspects. In order to model audio-visual speech accurately, we require strong temporal encoders that can model the temporal correlations in long sequences effectively. In this section, we discuss the shift from recurrent neural networks (RNNs), which are used in Chapter 4, to transformers, which are used in the remaining chapters.

Recurrent neural networks

Recurrent neural networks were originally proposed by Hopfield (1982) with the aim of recognizing patterns in sequences. In short, RNNs are neural networks that have an explicit connection to the next time-step, rather than operating on a single static input. In general, RNNs model sequences by receiving one token of the input sequence x_t each iteration and keeping a hidden ‘state’ h_t that is updated every time-step, as shown in Figure 2.11. Following this

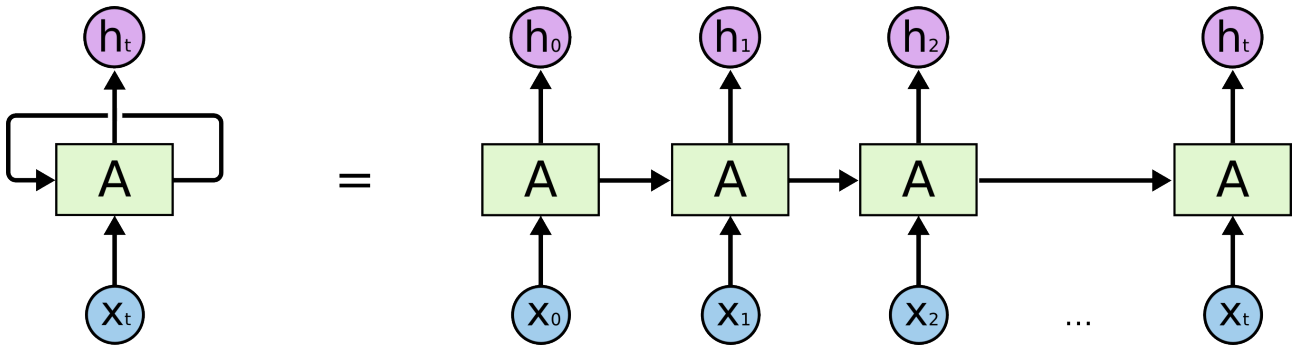


Figure 2.11: A simple illustration of a recurrent neural network (RNN). Source: <https://colah.github.io/posts/2015-08-Understanding-LSTMs/>

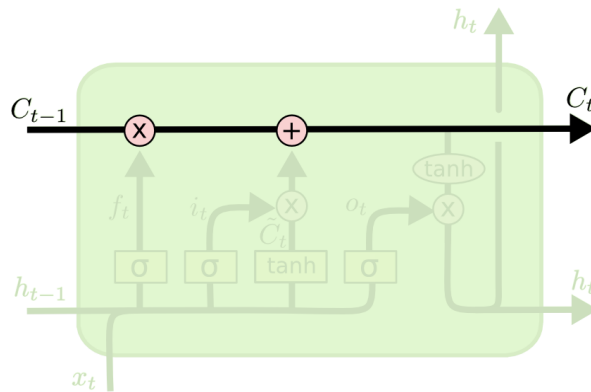
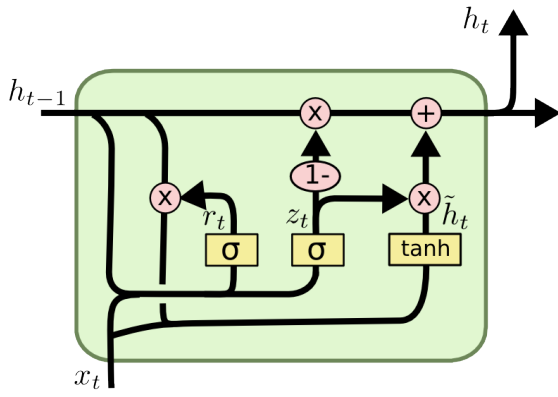


Figure 2.12: Long short-term memory (LSTM) cell. Source: <https://colah.github.io/posts/2015-08-Understanding-LSTMs/>

abstract definition, many attempts were made to create RNNs that were efficient and easy to train. However, early networks suffered from vanishing and exploding gradients during back-propagation, which made training unstable (Lipton, 2015), and struggled to model long-term dependencies effectively (Kolen and Kremer, 2001).

The first major development in RNNs came with the proposal of the Long Short-Term Memory (LSTM) model (Hochreiter and Schmidhuber, 1997). This work proposed to replace the hidden layers in traditional RNNs with memory cells, as described in Figure 2.12. The LSTM cell differentiates itself from previous RNN hidden layers by having not only a hidden state but also a cell state, which flows from time step to time step with minimal changes. This is the key idea of LSTMs: to have an internal state that can flow through each step of the sequence while avoiding the vanishing and exploding gradients in the traditional hidden state.

The LSTM cell has three major components presented from left to right in Figure 2.12: the



$$z_t = \sigma(W_z \cdot [h_{t-1}, x_t])$$

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t])$$

$$\tilde{h}_t = \tanh(W \cdot [r_t * h_{t-1}, x_t])$$

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t$$

Figure 2.13: Gated Recurrent Unit (GRU). Source: <https://colah.github.io/posts/2015-08-Understanding-LSTMs/>

forget gate f_t , introduced by Gers *et al.* (2000), which controls how much of the previous time step's cell state C_{t-1} is kept and how much is forgotten via an element-wise multiplication; the input gate i_t , which computes an embedding based on the previous hidden state h_{t-1} and current input x_t , and adds it to the cell state C_t ; and an output gate o_t , which combines the current input x_t , previous hidden state h_{t-1} and current cell state C_t to produce the current hidden state h_t , while the cell state C_t flows to the next time-step.

Due to their ability to model long-term dependencies while avoiding the training instabilities of previous RNNs, LSTM models quickly became popular in sequence modeling fields such as Natural Language Processing (NLP) (Otter *et al.*, 2021) and speech recognition (Graves *et al.*, 2013). In practice, most works opted for bi-directional LSTM networks, as proposed by Schuster and Paliwal (1997), which leverages two LSTM layers: one that models the forward dependencies (from $t-1$ to t), as in the original RNN, and another one that models the backward dependencies (from t to $t-1$). This yields a network that can model the temporal dependencies from past inputs as well as future inputs. This can be useful in machine translation, for example, where the translation of a word at time t can be dependent on the input word at time $t+1$.

A more recent variant of the RNN came with the introduction of the Gated Recurrent Unit (GRU) (Cho *et al.*, 2014; Chung *et al.*, 2014), shown in figure 2.13. The GRU differs fundamentally from the LSTM by foregoing the idea of the cell state, and instead keeping only the hidden state at each time step. However, it borrows the idea of gating by having a reset gate r_t , which allows the network to reset the hidden state h_{t-1} (which can also be seen as forgetting the

previous state), and an update gate z_t , which updates the new state h_t based on the previous state h_{t-1} and new input x_t .

For the purposes of this thesis, we use a two-layer bidirectional GRU model as our temporal encoder in the generator in Chapter 4. We chose this model since we needed to model the complex temporal dependencies between the visual features from each video frame, which can be related to past and future frames. We also use a pre-trained LSTM-based model as our speaker encoder in Chapter 5, which was originally trained to perform speaker verification from spectrograms (Jia *et al.*, 2018).

Despite being substantially more stable than traditional RNNs, LSTMs, and GRUs still suffer from severe issues that stem from their auto-regressive nature (i.e., each output o_t depends on the previous output o_{t-1}). Firstly, although the issue of vanishing and exploding gradients is alleviated in these new variants, it is not completely eliminated (Le and Zuidema, 2016), which makes for an unreliable training procedure. Secondly, although substantial efforts have been made to improve long-term dependence modeling, it has been shown that LSTMs still struggle to consistently display long-term memory (Zhao *et al.*, 2020a). Intuitively, there are limits to the long-term dependencies that can be modeled via a single continuously changing hidden state, particularly when compared to the explicit connections present in, e.g., CNNs.

Thirdly and most importantly, the RNN’s autoregressive training procedure, where we must compute each time-step sequentially since h_t depends on h_{t-1} , is not easily parallelizable. This is unlike CNNs, for example, where the sliding kernel can be applied on many segments of the input in parallel, rather than one by one. This makes RNNs slow to train even on high-end GPUs, which is a substantial drawback for practitioners.

Transformers

Transformers (Vaswani *et al.*, 2017) successfully solve the issues faced by RNNs via one simple design decision: non-autoregressive modeling. Instead of receiving one input token x_t for each step, transformers process the entire sequence in parallel. Each element of the sequence is

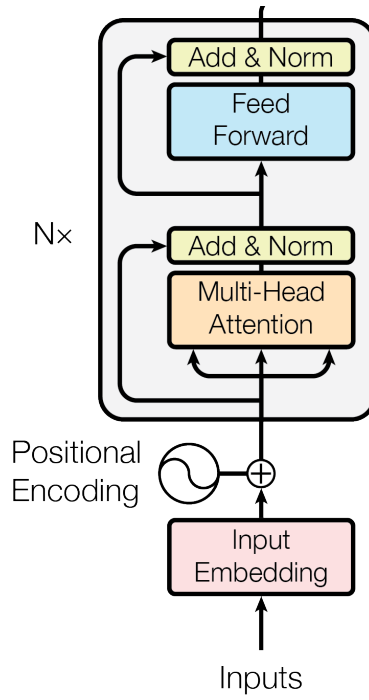


Figure 2.14: The transformer encoder. Source: Vaswani *et al.* (2017)

associated with a time-step t via positional encoding, which is added to the input embedding:

$$p(t, 2i) = \sin\left(\frac{t}{10,000^{\frac{2i}{d}}}\right) \quad (2.7)$$

$$p(t, 2i + 1) = \cos\left(\frac{t}{10,000^{\frac{2i}{d}}}\right) \quad (2.8)$$

where $p(t, 2i)$ and $p(t, 2i + 1)$ are the positional encodings at time t for input dimensions $2i$ and $2i + 1$, respectively, and d is the input's total dimension. This mix of sine and cosine functions ensures that each input token is associated with a unique positional encoding that identifies time-step t , and can adapt to sequences of variable lengths. Positional encodings allow us to simply feed the entire sequence into the model and, more importantly, compute the outputs in parallel.

Vaswani *et al.* (2017) introduce an encoder-decoder structure that is proposed for machine translation (encoding the input in one language and decoding the output in another language). However, for the purposes of this thesis, we are only interested in the transformer encoder for sequence modeling and therefore disregard the transformer decoder for the remainder of this section. The transformer encoder (shown in Figure 2.14) stacks multiple encoder layers,

each of which contains two main components: a multi-head attention layer and a feedforward layer. We will focus on describing the multi-head attention layer, as the feedforward layer is simply composed of two fully connected linear layers, as described in Section 2.3.1. Attention mechanisms were originally proposed for machine translation in (Bahdanau *et al.*, 2015). The general idea was to allow the output translation to attend to different parts of the input with different weights via a context vector:

$$c_j = \sum_{i=0}^T w_{ij} h_i \quad (2.9)$$

where c_j is the context vector for the j -th output embedding, h_i is the i -th input embedding (computed via an RNN), w_{ij} is the attention coefficient for the i -th input and j -th output, which is computed via a learnable function, and T is the total number of input embeddings. Effectively, this allowed the model to explicitly model the dependencies between inputs and outputs, rather than solely relying on recurrent mechanisms.

This fundamental idea was then extended and reformulated by following works (Niu *et al.*, 2021), eventually converging into self-attention as proposed by Vaswani *et al.* (2017). In the transformer, attention layers receive three vectors as input: queries Q , keys K , and values V . These are computed from the input embedding using a set of trainable weights w - one for each of the three vectors:

$$Q = w_Q X \quad (2.10)$$

$$K = w_K X \quad (2.11)$$

$$V = w_V X \quad (2.12)$$

where X is the input embedding. Borrowing from the nomenclature presented above for the original attention mechanisms, the attention coefficient for the i -th input and j -th output w_{ij} will be calculated via the dot product between the j -th query Q_j and the i -th key K_i . This coefficient is then scaled down by dividing by key's dimension $\sqrt{d_K}$ (this prevents the coefficient values from becoming too large before the softmax, which can create issues during

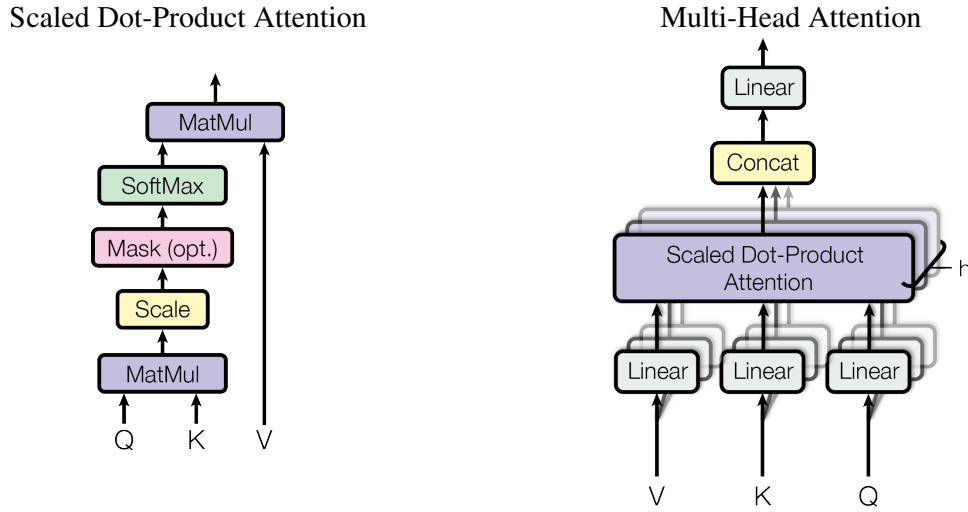


Figure 2.15: A brief summary of self-attention and multi-head self-attention. Source: Vaswani *et al.* (2017)

training (Vaswani *et al.*, 2017)), and the softmax function is applied. Finally, as in the original attention mechanisms, the attention coefficient is multiplied with the i -th value V_i to obtain the final attention value for the i -th input and j -th output. These are then summed for all inputs i to compute the final output for index j . This can be computed in matrix form via the following equation:

$$A = \text{softmax}\left(\frac{QK^T}{\sqrt{d_K}}\right)V \quad (2.13)$$

where A is the output of the attention layer. This formula allows for very efficient computation since it computes the output of the attention layer via a set of highly parallelizable matrix multiplications. The authors then proposed multi-head attention as a wider, more powerful version of this self-attention. Multi-head self-attention with N heads is performed by projecting the queries, keys, and values via a set of N linear layers, passing those projections through a set of N attention layers (denoted here as 'heads'), concatenating the outputs and projecting these once more via a linear layer. This can be seen as applying self-attention on multiple 'views' of the input, which again can efficiently be computed in parallel. A brief summary of multi-head self-attention is shown in Figure 2.15.

By abandoning the autoregressive modeling proposed by traditional RNNs, transformers can

more efficiently be trained in parallel on GPUs and are able to explicitly model the dependencies between different time steps of the input sequence, rather than relying on a recurrent hidden state. After their introduction by Vaswani *et al.* (2017), transformers quickly overtook RNNs in virtually every NLP-related task (Devlin *et al.*, 2019; Niu *et al.*, 2021). Transformers were also later extended with a new kind of positional embedding (Dai *et al.*, 2019), which solved the original transformer’s fixed length limitation by adding recurrent mechanisms.

Transformers have also become popular in image processing, especially for self-supervised learning (Baevski *et al.*, 2020; Hsu *et al.*, 2021), and image classification (Dosovitskiy *et al.*, 2021), where the state-of-the-art has been dominated by transformer-based approaches, particularly due to their ability to leverage transfer learning effectively (Yu *et al.*, 2022; Zhai *et al.*, 2022). It is worth noting, however, that, when training from scratch, smaller vision encoders still benefit from having a convolutional front-end, as shown in the original Vision Transformer (ViT) (Dosovitskiy *et al.*, 2021). Transformers have also been widely used in video-related tasks such as action recognition (Tong *et al.*, 2022) and lip reading (Shi *et al.*, 2022a), where they have obtained state-of-the-art results by similarly leveraging self-supervised learning.

In this thesis, transformer encoders (with the positional embeddings from (Dai *et al.*, 2019)) are used for sequence modeling as powerful audio-visual encoders in Chapters 6 and 7. On the video branch, these transformers are preceded by the standard 3D+2D spatiotemporal encoder, resembling an improved version of the 3D+2D ResNet + BGRU video encoder presented in Chapter 4, which was limited by the BGRU in terms of performance and speed. On the audio branch, they are preceded by either a 1D-ResNet 18 for raw audio (Chapter 7) or a simple linear layer for spectrograms (Chapter 6). For both audio and video, the front-end layers merely serve as initial feature extractors, while the transformer is the main component of the network with a much larger number of parameters.

Conformers

During their rise in popularity, there have been many proposals to make lighter/faster versions of transformers (Fournier *et al.*, 2021), including the Reformer (Kitaev *et al.*, 2020), the Lin-

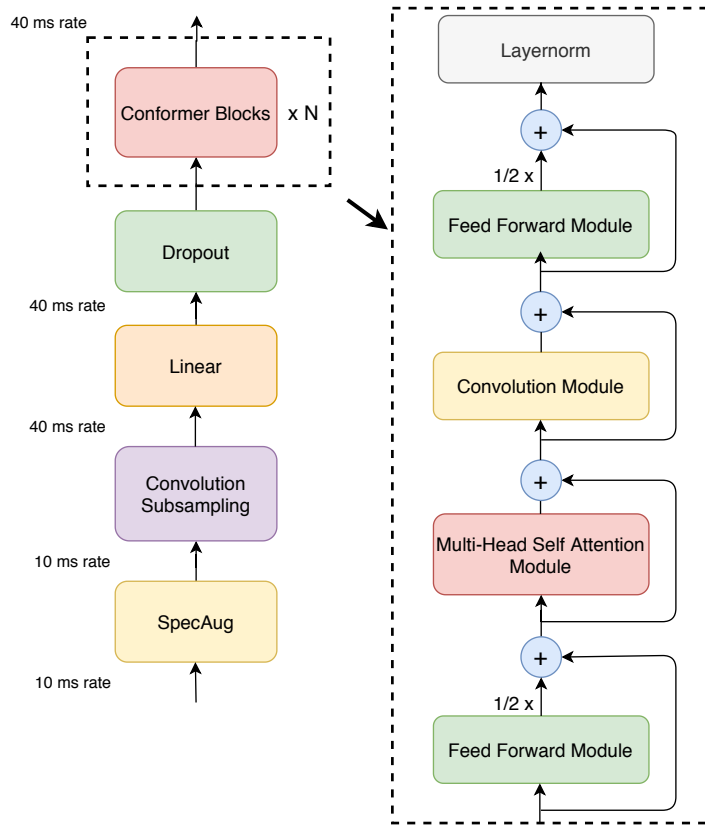


Figure 2.16: The Conformer (Gulati *et al.*, 2020). Source: Gulati *et al.* (2020)

former (Wang *et al.*, 2020b) and the Performer (Choromanski *et al.*, 2021). These models have focused on decreasing the quadratic complexity of the transformer’s self-attention ($O(n^2)$) in an attempt to make the computations faster during training and inference. However, each of these new models has displayed its own set of limitations and challenges, and therefore none has succeeded in becoming as dominant in sequence modeling as the original transformer.

On the other hand, other models have focused on augmenting the transformer to improve its performance, disregarding the pursuit of lower computation overhead. One such model has been the Conformer (Gulati *et al.*, 2020), which aims to outperform the original transformer by combining self-attention with convolutions. In other words, the fundamental idea is that self-attention can model the global dependencies between the different elements of the input sequence, while convolutions can model local dependencies more effectively.

The Conformer encoder is composed of a front-end convolutional layer followed by a set of conformer blocks. Each conformer block (shown in Figure 2.16) mirrors the transformer block

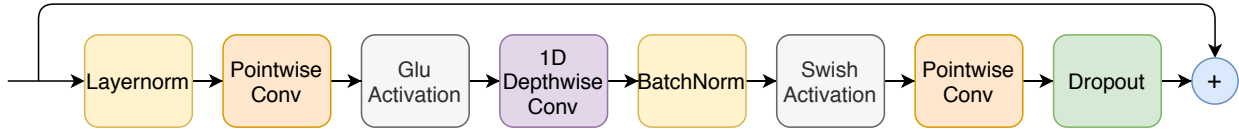


Figure 2.17: The convolution module from the Conformer (Gulati *et al.*, 2020). Source: Gulati *et al.* (2020)

(one multi-head self-attention layer followed by a feedforward module with two linear layers) but features two major changes. The first is that instead of using a single feedforward module after the self-attention layer, the conformer block employs a half-step feed forward module before and after the self-attention layer, inspired by Macaron-Net (Lu *et al.*, 2019). Secondly, a convolutional module is added between the self-attention layer and the second feedforward module. The convolutional module is shown in Figure 2.17. It effectively encapsulates a single 1D depthwise convolution (Chollet, 2017), sandwiched between two pointwise convolutions. Intuitively, the idea is that this convolutional module will capture local dependencies via its sliding kernel (which, in the original conformer, is set to a size of 32).

In the original paper (Gulati *et al.*, 2020), the conformer is proposed for speech recognition from mel-spectrograms, and is shown to outperform the Transducer (Zhang *et al.*, 2020a), as well as other previous transformer-based approaches. Due to its state-of-the-art performance, the conformer became a prevalent model choice in speech recognition (Zhang *et al.*, 2020b; Chan *et al.*, 2021), lip reading (Ma *et al.*, 2022a) and audio-visual speech recognition (Ma *et al.*, 2021e). For the purposes of this thesis, conformer encoders are used as video encoders (preceded by a 3D+2D ResNet-18) in Chapters 3 and 5 due to their exceptional sequence modeling performance and scalability.

2.3.4 Generative Adversarial Networks

The original generative adversarial network (GAN) was proposed by Goodfellow *et al.* (2014). It was designed to produce impressive generative results using deep learning, which had not yet been convincingly achieved despite the existence of variational autoencoders (VAEs) (Kingma and Welling, 2014). Indeed, until the introduction of GANs, neural networks had only achieved

substantial success in discriminative tasks and generally struggled with tasks such as image generation, where models tended to produce blurry images that often contained noticeable artifacts.

GANs aim to generate realistic images by following a simple adversarial technique that involves two main components: the generator and the discriminator. The generator receives random Gaussian noise as input and generates e. g., a 2D image. This module is generally composed of a stack of transposed convolutional layers that gradually decode the input noise into a full-sized image. The discriminator, on the other hand, is designed as a traditional image classifier and aims to classify whether the input it has received is a real image (from the ground truth dataset) or a fabricated image produced by the generator. These two networks are trained within an adversarial framework, where the generator aims to ‘fool’ the discriminator by producing images that are classified as real, and the discriminator aims simply to behave as an accurate classifier that classifies the real and fabricated images correctly. During training, the generator and discriminator losses L_G and L_D , respectively, are given by:

$$L_G = \log(D(G(z))) \quad (2.14)$$

$$L_D = \log(D(x)) + \log(1 - D(G(z))) \quad (2.15)$$

where G is the generator, D is the discriminator, x is a real image and z is random gaussian noise. Despite its simplicity, this approach forces the generator to correct its blurry/noisy outputs and model the real image distribution more accurately. Despite not displaying particularly groundbreaking results in their original proposal (Goodfellow *et al.*, 2014), GANs were quickly adopted by other researchers and became the most influential generative deep learning framework of the 2010s. Conditional GAN (Mirza and Osindero, 2014) adapted this method to receive a condition as input, rather than random Gaussian noise, which led the way for interesting applications such as text-to-image generation (Reed *et al.*, 2016) and image-to-image translation (Isola *et al.*, 2017). Furthermore, DCGAN (Radford *et al.*, 2016) built on the original GAN framework by proposing a fully convolutional generator/discriminator ensemble with a variety of empirically defined architectural tweaks. This model was trained on multiple large

datasets, including ImageNet-1k, and achieved state-of-the-art performance on multiple tasks, further solidifying GANs as the de-facto standard in generative image modeling.

The GAN was also popularized for other domains, such as video and audio. GANs were first introduced for audio synthesis by Donahue *et al.* (2019). This paper proposed two generator/discriminator frameworks - one to generate raw waveform (i.e., a 1D vector) and another to generate spectrograms (i.e., a 2D image), which can be converted to raw audio using the Griffin-Lim algorithm (Griffin and Lim, 1983). Both models were trained using a variant of the GAN loss presented below (the Wasserstein GAN) on various different domains, including instruments (e.g., drums, piano) and speech (e.g., TIMIT (Harte and Gillen, 2015)). Despite initially underwhelming results, particularly when compared with the well-established WaveNet (Oord *et al.*, 2016), WaveGAN became a pioneering work in audio synthesis. In the following years, researchers gradually succeeded in leveraging adversarial speech synthesis to design lightweight speech synthesizers that outperform WaveNet while having fewer parameters and faster inference, such as Parallel WaveGAN (Yamamoto *et al.*, 2019) and HiFi-GAN (Kong *et al.*, 2020a).

Wasserstein GAN

While GANs are very effective under the right conditions, they are notoriously difficult to train and are known to have unstable training patterns. These issues have led to the development of new GAN variants, which tweak the original GAN loss in an attempt to yield improved results, or simply stabilize training. One of these variants has been the Wasserstein GAN (WGAN), which aims to minimize the Wasserstein distance between the real distribution P_r and the generated distribution P_g . In practice, WGAN is trained using exactly the same generator/discriminator logic, but proposes a new training loss for both:

$$L_G = -C(G(z)) \tag{2.16}$$

$$L_C = C(G(z)) - C(x) \tag{2.17}$$

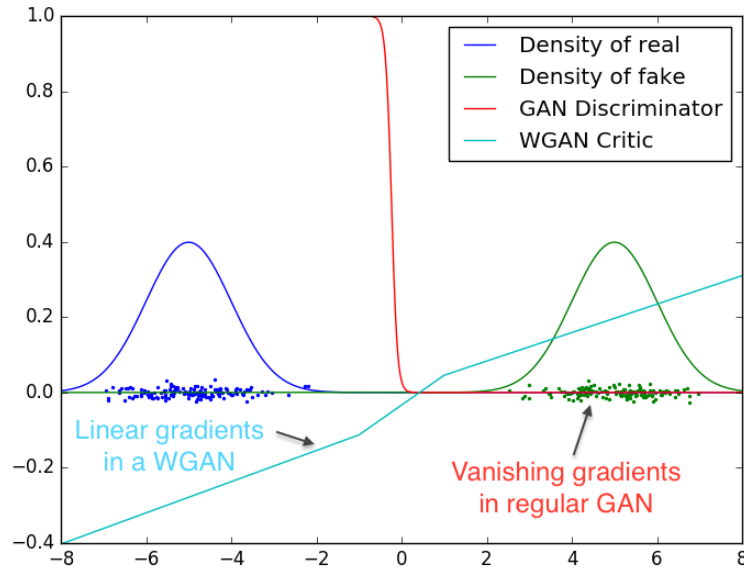


Figure 2.18: Gradients of an optimal discriminator (GAN) and critic (WGAN) trained to discriminate between two gaussian distributions. As we can see, the original GAN suffers from vanishing gradients, while WGAN provides a smooth optimization landscape with reasonably valued gradients at almost every point. Source: Arjovsky *et al.* (2017)

where G is the generator, z is random gaussian noise, and x is an image from the real dataset, as before, and C is the critic. In this GAN variant, we refer to the discriminator as the critic, since it does not aim to discriminate (i.e., classify) the inputs as real or fabricated (1 or 0), but rather it attributes an unbounded positive score (≥ 0) to the input, where a higher score indicates a more realistic input (i.e., closer to the real distribution). This loss configuration leads to a smoother optimization landscape, as shown in Figure 2.18.

The losses presented above refer to the optimization problem we are hoping to solve, which is derived from the highly intractable Wasserstein distance criteria using the Kantorovich-Rubinstein duality, resulting in the following equation:

$$W(P_r, P_g) = \sup_{\|f\|_L \leq 1} \mathbb{E}_{x \sim P_r}[f(x)] - \mathbb{E}_{x \sim P_g}[f(x)] \quad (2.18)$$

where $W(P_r, P_g)$ is the Wasserstein distance between the real and generated distributions, and f is a function that takes inputs from both distributions - in Wasserstein GAN, this role is played by the critic C . Importantly, this means that the critic is constrained to be a K -Lipschitz continuous function at all times (as indicated by the 1-Lipschitz constraint above $\|f\|_L \leq 1$).

In practice, there are two ways of satisfying this constraint:

1. **Weight clipping** - In the original WGAN, the K-Lipschitz continuity is enforced by clipping the critic’s weights to keep them within a very compact range (in this case, $[-0.01, 0.01]$). However, the authors openly claim that “Weight clipping is a clearly terrible way to enforce a Lipschitz constraint.”. In short, whether the gradient clipping parameter is small or large, it can have adverse effects on the optimality and stability of training.
2. **Gradient penalty** - Gulrajani *et al.* (2017) aim to address the issues that arise from gradient clipping by proposing an improved method to enforce the Lipschitz constraint. Through a series of proofs, the authors propose that a 1-Lipschitz function’s gradients should be at most 1 (≤ 1) at all points. In practice, this can be enforced by adding a gradient penalty L_{GP} to the WGAN loss:

$$L_{GP} = \lambda(\|\nabla_{\hat{x}} D(\hat{x})\|_2 - 1)^2 \quad (2.19)$$

where λ is the gradient penalty coefficient, which is an adjustable hyperparameter, and $\hat{x}$ is a point randomly sampled between a real sample x and a fake sample $\tilde{x}$. By leveraging this alternative way to enforce the Lipschitz constraint, the authors find that the training objective becomes easier to optimize, yielding better generative results. This improved version of WGAN is known as WGAN-GP.

For the purposes of this thesis, the WGAN-GP loss is used in Chapter 4 to train our video-to-speech generator.

Least Squares GAN

The Least Squares GAN (LSGAN) (Mao *et al.*, 2017) is designed with the same goals of WGAN, i. e., to improve the stability and effectiveness of adversarial networks by proposing a smoother

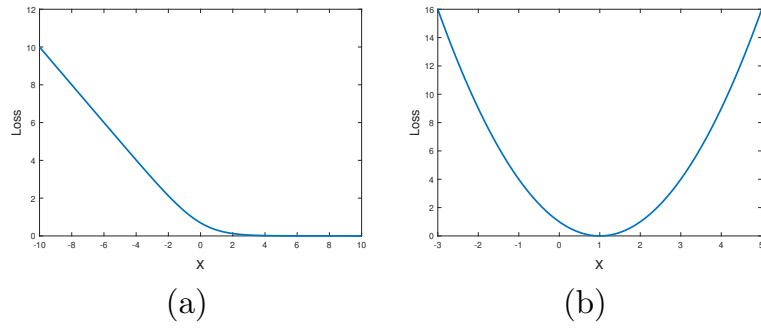


Figure 2.19: (a) Original GAN's cross-entropy loss function (b) LSGAN's loss function. Source: Mao *et al.* (2017)

loss landscape that is easier to optimize. The LSGAN's loss is defined as:

$$L_G = (D(G(z)) - 1)^2 \quad (2.20)$$

$$L_D = (D(x) - 1)^2 + (D(G(z)))^2 \quad (2.21)$$

Although this may seem like a trivial transformation of the original GAN loss which squares each of the loss components, this leads to a vastly different loss curve, as presented in Figure 2.19. Unlike the original GAN's cross-entropy loss, which flattens and saturates around 0 for larger inputs, the LSGAN's loss is a smooth function with only one flat point in the center. The authors argue that this helps prevent the well-known vanishing gradients issue in adversarial training, which makes it difficult to adequately update the generator's weights and yields decreased training stability. The authors train LSGAN on a variety of image generation benchmarks and conclusively find that it produces higher-quality images than its GAN counterpart.

For the purposes of this thesis, the LSGAN loss is used to train Parallel WaveGAN in Chapter 5 and HiFi-GAN in Chapter 6, which play a central role in achieving state-of-the-art results on video-to-speech synthesis and audio-visual speech enhancement.

2.4 Evaluation metrics

In this short section, we describe the metrics that we apply to evaluate our results in the following chapters. Speech assessment metrics typically aim to evaluate one of two aspects of speech: quality and intelligibility (Xu *et al.*, 2017). Quality evaluates how natural and clear the speech sounds for human users i.e., how comfortable the listening experience is, while intelligibility evaluates how intelligible the speech is i.e., how easy it is to understand the words being uttered. Since no metric is perfectly correlated with human perception for either of these two aspects, most speech synthesis approaches, including the ones presented in Chapters 4, 5, and 6 in this thesis, use multiple evaluation metrics for both. This results in a more diverse set of metrics, which are often combined with a user study and/or a set of demo samples to further illustrate the generative model’s performance. Except for word error rate (WER), all of the speech assessment metrics presented below receive two signals: the reference signal, which is the ground truth audio taken from a real dataset, and the degraded signal, which is an estimation of the original reference signal (in our case, generated by a neural network).

2.4.1 Speech quality metrics

PESQ

PESQ (Perceptual Evaluation of Speech Quality) (Rix *et al.*, 2001) is an objective speech quality metric originally proposed for telephony quality assessment. The pipeline for PESQ, which receives two audio waveforms and outputs a speech quality score, is summarized as follows: first, the signals are aligned to a standard listening level, filtered using an FFT-based input filter, and aligned in time; then, they are processed via an auditory transform based on PSQM (Beerends and Stemerdink, 1994), and two distortion parameters are obtained from the difference between the transformed signals; finally, these parameters are aggregated in frequency and time, and are mapped to a MOS (Mean Opinion Score) prediction. The full pipeline is illustrated in Figure 2.20. The authors mention that it correlates very well with subjective listening tests for various types of telephony distortion. However, it is sensitive to loudness

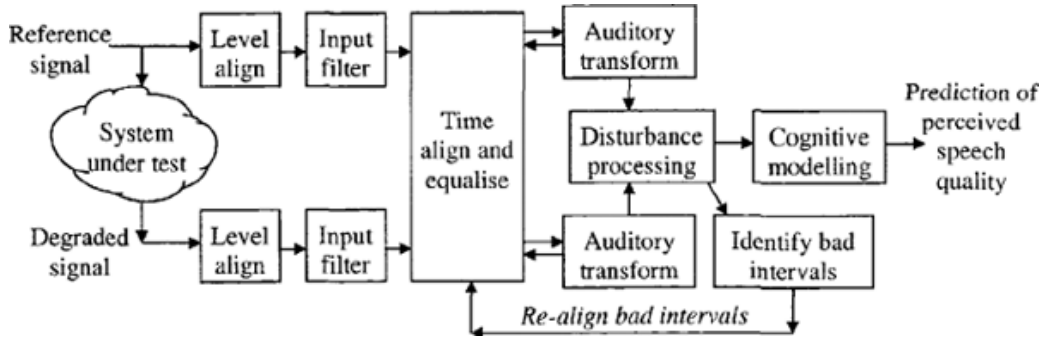


Figure 2.20: Graphical illustration of the PESQ pipeline. Source: Rix *et al.* (2001)

and the speaker’s voice profile (Rix *et al.*, 2001), which evidently hinders its objectivity. We use the narrowband version of PESQ (PESQ-NB), which is designed to assess the quality of narrowband speech (300–3,400 Hz), in Chapters 4 and 5 (for simplicity, we refer to this as “PESQ”, as this is the most commonly-used version of this metric), and the wideband version of PESQ (PESQ-WB), which is designed to assess the quality of wideband speech (50–7,000 Hz), in Chapter 6.

MCD

MCD (Mel-Cepstral Distance) (Kubichek, 1993) is designed to evaluate speech quality based on the cepstrum distance on the Mel scale. In practice, this is calculated as the distance between the MFCCs (as defined in Section 2.2.1) extracted from two signals:

$$MCD(x, y) = \frac{1}{F} \sum_{f=1}^F \sqrt{\sum_{n=1}^{N_{MFCC}} [MFCC(x, n) - MFCC(y, n)]^2} \quad (2.22)$$

where x and y are the reference and degraded signals, respectively, F is the number of audio frames, N_{MFCC} is the number of coefficients for the MFCC. This metric is used in Chapters 4 and 6. We find that it works quite reliably in measuring perceptual quality in our synthesized outputs when compared to the original signals.

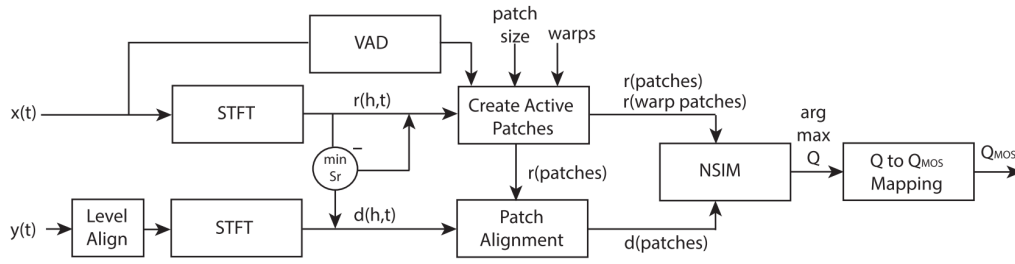


Figure 2.21: Graphical illustration of the original ViSQOL algorithm. Source: Hines *et al.* (2015)

ViSQOL

ViSQOL (Virtual Speech Quality Objective Listener) (Hines *et al.*, 2015; Chinen *et al.*, 2020) is an objective speech quality metric, recently proposed by *Google*. It is designed to measure the quality of human speech by using a spectro-temporal measure of similarity between the real and synthesized audio. The step-by-step process to compute ViSQOL (the original version (Hines *et al.*, 2015)) is shown in Figure 2.21 and briefly summarized as follows: first, the degraded signal is scaled to match the power of the reference signal; then, spectrograms are computed from both signals and active patches are extracted from each using a simple voice activity detector (VAD); the patches are then aligned in time and adjusted to compensate for warping; finally, the similarity between the frames for each signal is computed, averaged, and mapped into a MOS score. ViSQOL is shown to consistently outperform PESQ through multiple experiments while being sensitive to a wide variety of degradations, background noises, and speech enhancement methods. This metric is used in Chapter 5 – we use ViSQOL v3 (Chinen *et al.*, 2020), which makes minor tweaks and additions to the original algorithm, rather than the original version, as it is shown to be more effective.

2.4.2 Speech intelligibility metrics

STOI

STOI (Short-Time Objective Intelligibility measure) (Taal *et al.*, 2011a) aims to measure how intelligible a speech signal is through a comparative approach. The pipeline (shown in Figure

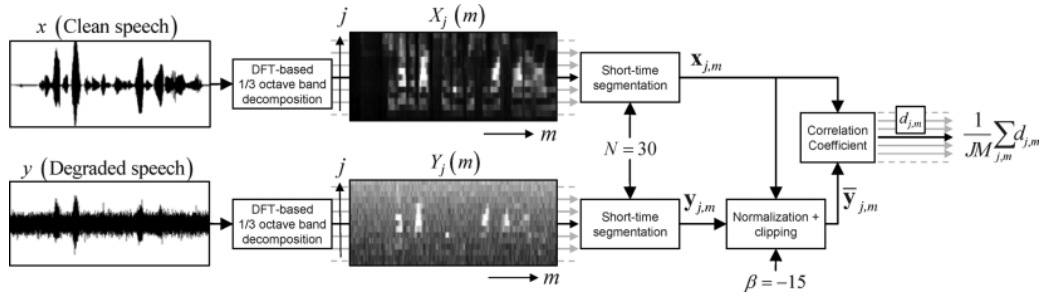


Figure 2.22: Graphical illustration of the STOI pipeline. Source: Taal *et al.* (2011b)

2.22) is as follows: first, the signals are decomposed into DFT-based one-third octave bands; then, short-time segmentation is performed on the resulting signals and the degraded segments are normalized and clipped; following this, the short-term intelligibility measures are computed by comparing the segments between the signals via a correlation coefficient; finally, these measures are averaged to obtain the final intelligibility score. STOI has been shown to correlate closely with human intelligibility scores. This metric is used in Chapters 4, 5, and 6. In our experiments, we find that it correlates well with the intelligibility of the generated signal, compared to the original.

ESTOI

ESTOI (Extended Short-Time Objective Intelligibility measure) (Jensen and Taal, 2016) is proposed as a new version of STOI that can accurately adapt to a wider range of input signals. In practice, the algorithm for ESTOI is almost identical to STOI, but instead of computing short-term correlation coefficients and averaging across frequency, it instead computes spectral correlation coefficients between the complete 400-ms spectrograms. This allows ESTOI to perform better, in particular for temporally highly modulated noise sources. In addition, the clipping presented at the start of the original STOI pipeline is removed in this extended version. This metric is used in Chapters 5 and 6.

Word Error Rate

WER (Word Error Rate) measures the accuracy of a speech recognition system. It is calculated as:

$$WER = \frac{S + D + I}{N}, \quad (2.23)$$

where S is the number of substitutions, D is the number of deletions, I is the number of insertions and N is the total number of words in an utterance. This metric is used to measure the accuracy of our speech recognition/lip reading models in Chapters 3 and 7. In Chapter 4, we apply pre-trained ASR models to measure WER (using the dataset’s manually labeled transcriptions as the ground truth), which serves as an objective intelligibility metric for the reconstructed speech. In Chapter 5, we propose to forego the use of manual text transcriptions and use instead the transcription predicted from the corresponding real audio as the ground truth. This increases the interpretability of the reported numbers, as they are a direct measure of the difference in intelligibility between real and generated audio, and it also removes the requirement for labeled datasets in future work.

2.4.3 Other metrics

Above we have presented all metrics used throughout this thesis for the purposes of evaluating generated speech. However, Chapters 3 and 7 focus instead on lip reading and deepfake detection, which require slightly different metrics. For sentence-level lip reading, where the model aims to predict a sentence/phrase from a speech sample, WER, as presented above, is typically used to measure performance. However, for word-level lip reading, where the model aims to predict a single word, accuracy is used instead, which simply measures the ratio of correct predictions over the total number of predictions (typically expressed in percentage):

$$\text{Accuracy} = \frac{\# \text{ correct predictions}}{\# \text{ total predictions}}, \quad (2.24)$$

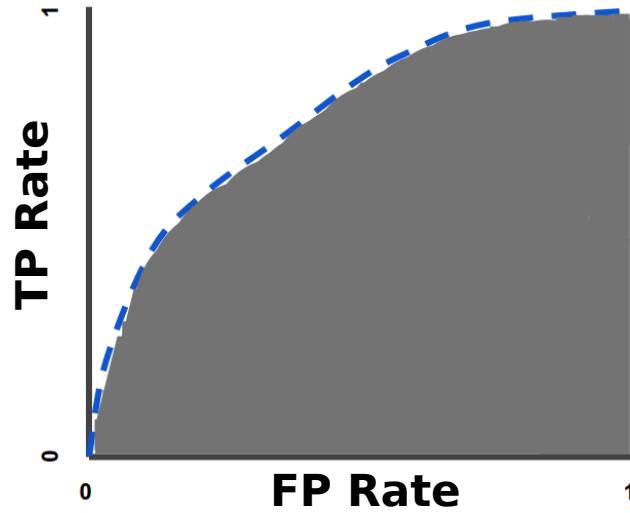


Figure 2.23: Illustration of the area under the receiver operating characteristic curve (AUC). Source: <https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc>

For deepfake detection, the area under the receiver operating curve (AUC) is used in addition to accuracy, in an attempt to better evaluate the reliability of the trained forgery detector. The receiver operating characteristic curve (ROC) measures the accuracy of a binary classifier (in this case, the model aims to predict whether the video is real or forged) for different classification thresholds, as plotted in Figure 2.23. The vertical axis measures the true positive rate (TPR), while the horizontal axis measures the false positive rate (FPR), which are computed as follows:

$$TPR = \frac{TP}{TP + FN}, \quad FPR = \frac{FP}{FP + TN}, \quad (2.25)$$

where TP is the number of true positive, FN is the number of false negatives, FP is the number of false positives, and TN is the number of true negatives. AUC simply measures the area below this curve. In short, a binary classifier such as the one presented in Chapter 7 aims to maximize the AUC, since this would imply that the model achieves a high true positive rate (TPR) across different classification thresholds.

CHAPTER 3

LEARNING VISUAL REPRESENTATIONS FROM SPEECH

Contents

3.1	Introduction	57
3.2	Methodology	60
3.2.1	Learning visual representations from speech	60
3.2.2	Lip reading	61
3.3	Experimental Setup	61
3.4	Results	63
3.4.1	Word-level lip-reading	63
3.4.2	Sentence-level lip-reading	65
3.5	Conclusion	66

This chapter is based on *LiRA: Learning Visual Speech Representations from Audio through Self-supervision*, a conference paper published in *Interspeech 2021*. Pingchuan Ma and I were

joint first authors of the paper. I designed the self-supervised framework, conducted the self-supervised experimental procedure, and wrote the sections of the manuscript that introduced, described, and discussed the self-supervised framework. Pingchuan Ma designed the lip reading models, conducted the downstream lip reading experiments to evaluate the quality of the learned representations, and wrote the sections of the manuscript that refer to lip reading. Stavros Petridis, Björn Schuller, and Maja Pantic supervised the project throughout its entirety, providing helpful suggestions, feedback, and guidance regarding the design of the framework and the writing of the manuscript.

It is important to highlight that the results presented in this chapter are only compared with the works that were available at the time of submission (March 2021). The decision to not include more recent works in the related work and in the comparison with competing models serves to contextualize the work presented in this chapter within the time it was published, which we find is increasingly important in this fast-moving field. In particular, although the LRS2 results presented here were state-of-the-art at the time of publication, they have now been outperformed by newer models developed by some of our co-authors (Ma *et al.*, 2022a, 2023), as well as our newer self-supervised model RAVeN, presented in Chapter 7 (please refer to the appendix of the full paper for the LRS2 results (Haliassos *et al.*, 2023)). Similarly, newer works have further pushed the state-of-the-art for lip reading on LRW (Ma *et al.*, 2022b).

3.1 Introduction

Self-supervised learning can be seen as a subset of unsupervised learning. While unsupervised learning typically leverages machine learning techniques such as K-Means clustering (MacQueen, 1967) to classify unlabelled data, self-supervised learning aims to leverage supervised deep learning techniques for unlabelled data, by extracting a training objective from the data itself. As mentioned in Chapter 1, self-supervised learning has become increasingly popular to improve results in image classification and speech recognition. In summary, recent self-supervised learning methods (for image, audio, video, or any other modality) can generally be

divided into three categories:

- Contrastive - these methods are based on teaching the model to distinguish between matching and non-matching data. A simple example is SimCLR (Chen *et al.*, 2020), which augments the same image in two different ways, e.g., a picture of a dog flipped upside down, and a picture of a dog in grayscale, and teaches the model to minimize the distance between the representations via a cosine similarity loss. On the other hand, it simultaneously teaches the model to maximize the distance between representations from augmented versions of different images. The idea is that the model must look past superficial visual cues and understand the content of the image (e.g., understand what a dog looks like) to make this distinction accurately. This idea has been adapted and re-iterated in numerous works focusing on images (He *et al.*, 2020; Chen *et al.*, 2021) and speech (Baevski *et al.*, 2020).
- Clustering-based - these methods learn by clustering the data via unsupervised learning (typically K-Means) and trying to predict the cluster assignments using a deep learning model. This task can be made more difficult to leverage further learning by increasing the number of clusters in K-Means, effectively leading to a classification task with more classes, or applying aggressive augmentations on the model's input, such as masking. This method originally became popular for visual representations (Caron *et al.*, 2018), and was also later successfully applied for speech (Hsu *et al.*, 2021).
- Predictive - these methods learn by predicting either the input image (typically from a masked (He *et al.*, 2022) or augmented (Zhang *et al.*, 2016; Noroozi and Favaro, 2016) version of the image) or a set of features extracted from the image. Examples of features that are used as objectives include acoustic features, such as spectrograms or MFCC in Problem Agnostic Speech Encoder (PASE) (Pascual *et al.*, 2019b) and PASE+ (Ravanelli *et al.*, 2020b), or features from another visual encoder, as in Bootstrap Your Own Latent (BYOL) (Grill *et al.*, 2020).

Of the three presented approaches, predictive methods are the most computationally intensive, since they require predicting a denser set of outputs, but have become prevalent in recent years

since they offer a more challenging and robust learning experience. Contrastive and clustering-based methods, on the other hand, tend to saturate during the later stages of training, with the objective becoming “too easy” for the network to continue learning.

Similarly to speech recognition, lip reading is a task that can benefit heavily from self-supervised learning, especially since there is more room for improvement due to the difficulties in training a VSR model from scratch. Despite this, very few works have attempted to improve lip reading models by leveraging the audio modality, which is typically paired with the video. Recent self-supervised works leverage audio-visual synchronization with contrastive learning to create new objectives from unlabelled data (Chung *et al.*, 2020a, 2019, 2020b), and experiment with intra-modal and cross-modal prediction (Tellamekala *et al.*, 2020). Recently, Zhao *et al.* (2020b) use a pre-trained speech recognition model to perform distillation, effectively allowing the lip reading model to directly learn from the ASR model. On the other hand, Afouras *et al.* (2020) experiment with semi-supervised learning by using a pre-trained speech recognition model to generate ‘pseudo-labels’ for unlabelled video datasets such as VoxCeleb2, while also performing distillation to accelerate training. While both methods yield improvements over training from scratch, they both require very large labeled speech datasets (LibriSpeech (Panayotov *et al.*, 2015a)) to train the ASR models, so they are not entirely self-supervised.

In this chapter, we propose **Learning Visual Representations from Audio (LiRA)**: a new framework which can leverage unlabelled audio-visual speech to improve lip reading models. The method works by feeding visual speech into a ResNet+Conformer model which is trained to predict audio features from the corresponding acoustic speech. We extract audio features using PASE+ (Ravanelli *et al.*, 2020a), a pre-trained speech encoder trained entirely on unlabelled data to generate problem-agnostic features. This allows the visual model to learn about the content that is shared between video and audio, e.g., verbal information. We then fine-tune this model for lip reading to evaluate the quality of the learned representations. We pre-train LiRA on LRS3 and find that, after being fine-tuned for lip reading, it outperforms previous self-supervised approaches by a substantial margin on LRW, and achieves state-of-the-art results on LRS2.

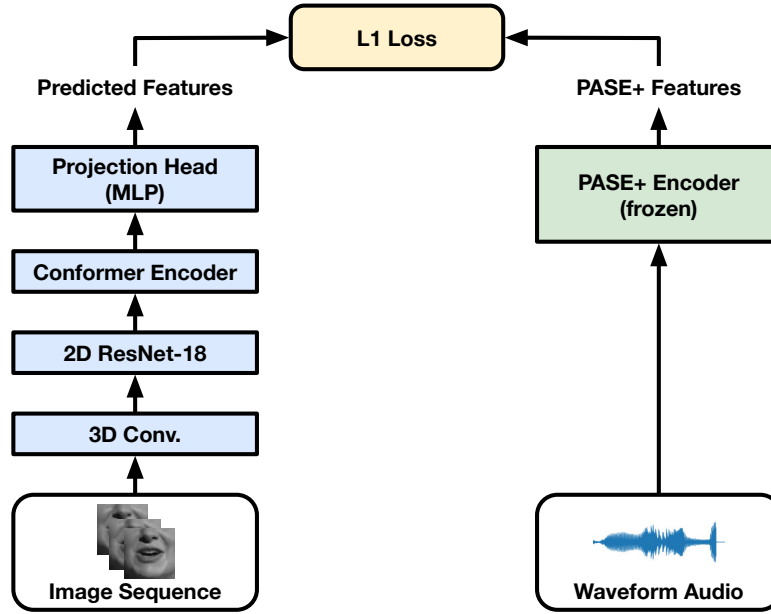


Figure 3.1: Overview of LiRA’s training procedure.

3.2 Methodology

3.2.1 Learning visual representations from speech

Our visual encoder combines a 2D ResNet-18 with a 3D front-end convolutional layer which outputs one embedding per frame, as introduced in Chapter 2 and explained in detail in Chapter 4. This is followed by our temporal encoder, the conformer encoder (described in detail in Chapter 2). It contains an initial linear layer, followed by positional encoding, and 6/12 conformer blocks (for LRW/LRS2) with 4 attention heads, a feedforward dimension of 2048, and an attention dimension of 256. Finally, the resulting features are passed through a linear layer which projects them into the predicted PASE+ features. The ground truth features are extracted using the official pre-trained PASE+ model (Ravanelli *et al.*, 2020a) available online¹. These features (100 fps) are averaged over time to match the visual encoder’s features (25 fps). The model is trained using an L1 loss between these and the predicted features. It is worth noting that we attempted to use features from the original PASE model (Pascual *et al.*, 2019b) for the same purpose, but found that they yielded worse performance.

¹<https://github.com/santi-pdp/pase>

3.2.2 Lip reading

In order to evaluate the quality of our self-supervised visual encoder, we transfer its weights to perform lip reading and measure the improvements over training from scratch. We experiment with three configurations: LiRA-Supervised, which is trained from scratch and serves as our baseline; LiRA-Frozen, which uses the weights from the pre-trained 3D+2D ResNet-18 from our self-supervised training and keeps them frozen during downstream training; and LiRA-FineTuned, which is identical to LiRA-Frozen, but unfreezes the weights in the ResNet, allowing them to be fine-tuned for this new task. For word-level lip reading (on LRW), we use a Multi-Scale Temporal Convolutional Network (MS-TCN) as our temporal model, whereas for sentence-level lip reading (on LRS2) we use a conformer followed by a transformer decoder.

3.3 Experimental Setup

As mentioned above, we use LRS3 for our self-supervised pre-training stage, LRW for word-level lip reading, and LRS2 for sentence-level lip reading. We use the original training/validation/testing file splits for each dataset. For LRS2 we combine the ‘pre-train’ and ‘train’ sets into a single training set, and for LRS3 we split the ‘train-val’ set into training and validation sets using a 90–10 % ratio, and combine the training set with the ‘pre-train’ set. To model the speaker’s lip movements, we crop a 96×96 Region of Interest (ROI) around the mouth using dlib (King, 2009) to detect faces and estimate landmarks, as described in Chapter 2.

Additionally, we employ two data augmentation methods during training. Firstly, we apply random cropping on the input frame, producing a frame with roughly 90 % of the original size. Furthermore, we apply horizontal flipping to each frame with a probability of 50 %. These procedures help make our model more robust and provide regularization. During test time, the same cropping is performed on the center of the frame and no horizontal flipping is performed. We train our model using the Adam optimizer (Kingma and Ba, 2015) with the Noam learning rate scheduler (Vaswani *et al.*, 2017). We use $\beta_1 = 0.9$, $\beta_2 = 0.98$, and a batch size of 32. To simplify the training procedure and avoid the issue of long sentences, we randomly sample 1

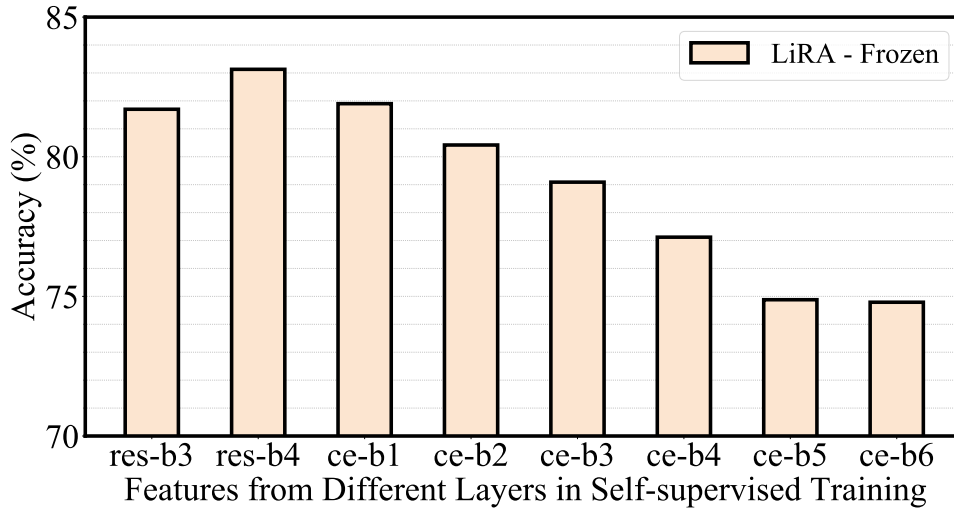


Figure 3.2: Lip reading accuracy on LRW using features extracted from different layers of Lira-Frozen. “res-b3” and “res-b4” refer to the output of blocks 3 and 4 from the ResNet, respectively; and “ce-b2” to “ce-b6” refer to the layers from every two conformer blocks from bottom to top.

Table 3.1: Comparison of LiRA with previous supervised and self-supervised lip reading methods on LRW. In the upper row, “Acc.” stands for accuracy, as described in Chapter 2.

Methods	Strategy	Acc. (%)
ResNet + BLSTM (Stafylakis and Tzimiropoulos, 2017)	Supervised	83.0
Two-stream 3D CNN (Weng and Kitani, 2019)	Supervised	84.1
ResNet + BLSTM (Stafylakis <i>et al.</i> , 2018)	Supervised	84.3
ResNet + DenseTCN (Ma <i>et al.</i> , 2021f)	Supervised	88.4
PerfectMatch (Chung <i>et al.</i> , 2020a)	Self-supervised	71.6
PT-CDDL (Chung <i>et al.</i> , 2020b)	Self-supervised	75.9
AV-PPC (Tellamekala <i>et al.</i> , 2020)	Self-supervised	84.8
LiRA-Supervised (Ma <i>et al.</i> , 2021b)	Supervised	87.4
LiRA-Frozen	Self-supervised	83.1
LiRA-FineTuned	Self-supervised	88.1

second from each clip and feed this into our visual encoder, disregarding utterances that are longer than 1 second.

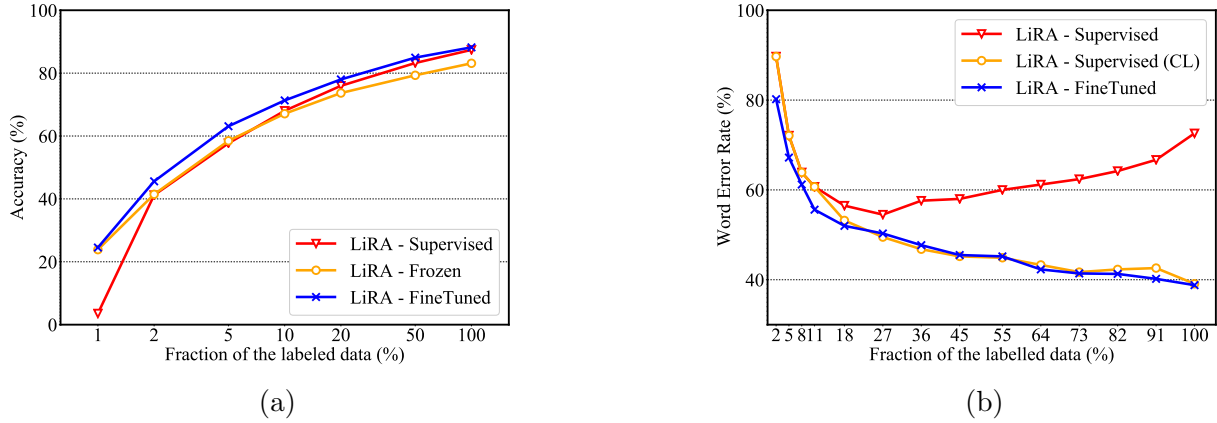


Figure 3.3: LiRA’s lip reading performance for varying fractions of the labeled training data. (a): Accuracy of the lip reading model as a function of the percentage of the training set (on a logarithmic scale) used for training on LRW. (b) WER as a function of the percentage of labeled data used for training on LRS2. “CL” refers to the model being trained using curriculum learning.

3.4 Results

3.4.1 Word-level lip-reading

We begin by comparing the performance of LiRA-Frozen for features extracted from different layers in Figure 3.2. The best lip reading performance is achieved by extracting features from the last layer of the ResNet. Remarkably, the performance consistently drops if we extract the features from deeper layers, which may indicate that these are further tuned towards the pre-training task of predicting PASE+ features and fail to generalize for lip reading. On the other hand, it is relevant to note that the features from the penultimate block of the ResNet also perform worse, as they are likely too low-level to serve as effective visual features. Based on this analysis, in subsequent experiments, we extract features from the last layer of the ResNet when reporting results on LiRA-Frozen.

We then compare our three configurations with other works in Table 3.1. We find that LiRA-Supervised achieves an accuracy of 87.4%, which outperforms many previous supervised works and demonstrates the capacity of our ResNet+MS-TCN architecture. On the other hand, LiRA-Frozen achieves 83.1%, which is an impressive performance considering that it is trained by extracting visual features from a frozen visual encoder which is not trained for this task.

Table 3.2: Lip reading performance comparison of LiRA with other supervised and unsupervised works on LRS2. “CL” refers to Curriculum learning.

Methods	Strategy	WER. (%)
Hyb. CTC/Att. (Petridis <i>et al.</i> , 2018b)	Supervised	63.5
Conv-seq2seq (Zhang <i>et al.</i> , 2019)	Supervised	51.7
TDNN (Yu <i>et al.</i> , 2020)	Supervised	48.9
TM-seq2seq (Afouras <i>et al.</i> , 2018a)	Supervised	48.3
KD-seq2seq (Afouras <i>et al.</i> , 2020)	Unsupervised	51.3
LiRA-Supervised (Ma <i>et al.</i> , 2021e)	Supervised (CL)	39.1
LiRA-FineTuned	Self-supervised	38.8

Finally, LiRA-FineTuned yields a considerable improvement over LiRA-Supervised, achieving an accuracy of 88.1%. This result clearly outperforms previous self-supervised methods and comes very close to the state-of-the-art method (88.4%). Most importantly, the improvement over LiRA-Supervised shows that our self-supervised pre-training strategy works effectively, providing an improvement of 0.7% without additional labels or supervision.

We conclude our word-level lip reading experiments by comparing LiRA’s performance for different fractions of the labeled training data in Figure 3.3a. We observe that both LiRA-Frozen and LiRA-FineTuned are superior LiRA-Supervised when training on 1% of the labeled data, which suggests that our self-supervised learning approach yields significant improvements when very few labels are available. As we scale up our training set, we observe that LiRA-Frozen’s performance drops sharply, eventually being overtaken by LiRA-Supervised, which is unsurprising given that the visual encoder is frozen. In contrast, LiRA-FineTuned continues to outperform LiRA-Supervised at all points, even when using the full training set. This analysis leads us to two main conclusions: **1)** For very small labeled datasets (in this case, 1% of LRW), using a pre-trained visual encoder is always better than training from scratch, even if it is frozen; and **2)** Fine-tuning the self-supervised model is superior to training it from scratch or freezing the pre-trained weights for any proportion of labeled training data on LRW.

3.4.2 Sentence-level lip-reading

We proceed by evaluating LiRA on sentence-level lip reading, which poses a substantially more challenging task. We compare with other works in Table 3.2. As in word-level lip reading, we find that LiRA-FineTuned outperforms LiRA-Supervised in WER, meaning that our self-supervised training procedure yields improvements in sentence-level lip reading as well. Furthermore, we find that LiRA-FineTuned outperforms all previous supervised and unsupervised approaches on this task by an absolute margin of 9.5 %, setting a new state-of-the-art on LRS2.

We then investigate the impact of labeled training on LiRA’s performance in Figure 3.3b, as we did for word-level lip reading. During our experiments, we found that LiRA-Supervised was difficult to train from scratch without a good initialization. As we observe in the plot, the model starts to perform poorly when we increase the size of the training set, and in fact, yields gradually decreasing performance when trained on more labeled data. We hypothesize that this behavior is due to the very long sentences in the LRS2 training set, which create instability during training. We propose curriculum learning (CL) as a solution to this issue - we begin by training on a set composed only of videos that are less than 155 frames long, amounting to 11 % of the total labeled data, and then continue by training the initialized model on the full training set. This curriculum learning strategy is also used in other lip reading works that train from scratch on LRS2, namely Ma *et al.* (2022a) which applies this strategy for LRS2 and LRS3 in order to achieve state-of-the-art results without additional training data.

Overall, we find that LiRA-FineTuned is consistently superior to LiRA-Supervised for any fraction of the training corpus. We also observe that the self-supervised initialization of LiRA-FineTuned generally yields better performance than the CL initialization, including when using 100 % of the training set. Once again, we find that the discrepancy between LiRA-FineTuned and the supervised models is particularly large for very small fractions of labeled data, where the model is more sensitive to its initialization. Finally, we highlight that we are able to outperform the previous state-of-the-art on LRS2 (Afouras *et al.*, 2018a) (48.3 % WER, trained on 1,362 hours of data) *using* $18\times$ *less labeled training data* (36 % of LRS3 $\approx$ 76 hours).

3.5 Conclusion

In this chapter, we present LiRA, a new self-supervised framework that leverages unlabelled audio-visual speech to produce better visual representations. We experiment with different feature extraction techniques and find that the visual features from the last layer of the visual encoder’s ResNet-18 provide the best lip reading performance. After this, we experiment with feature extraction configurations from a frozen encoder, as well as fine-tuning experiments that adapt the pre-trained ResNet encoder for lip reading. We find that the fine-tuned models consistently outperform those trained with random initialization, outperforming previous self-supervised approaches on LRW and setting a new state-of-the-art for LRS2. Finally, we show that the impact of the self-supervised pre-training is particularly large when training lip reading models on smaller sets of labeled data. In the future, we would like to apply similar audio-visual self-supervised methods to improve other speech-related tasks, such as emotion recognition.

CHAPTER 4

SYNTHESIZING SPEECH FROM SILENT VIDEO END-TO-END

Contents

4.1	Introduction	69
4.2	Related Work	71
4.2.1	Lip reading	72
4.2.2	Speech Synthesis	72
4.2.3	Reconstructing audio from visual speech	73
4.2.4	Reconstructing audio from multi-view visual speech	75
4.2.5	Audio reconstruction from video in other applications	76
4.3	Video-Driven Speech Reconstruction	77
4.3.1	Generator	77
4.3.2	Critics	79
4.3.3	Losses	80

4.3.4	Training details	83
4.4	Datasets	84
4.5	Evaluation Metrics	85
4.6	Results on Seen Speakers	85
4.6.1	Ablation Study	86
4.6.2	Comparison with Other Works	89
4.6.3	Performance as a Function of Training Set Size	91
4.7	Results on Unseen Speakers	92
4.7.1	Ablation Study	92
4.7.2	Comparison with Other Works	93
4.7.3	Additional Experiments	93
4.8	Results in the Wild	94
4.9	Conclusion	96

This chapter is based on *End-to-end Video-to-speech Synthesis using Generative Adversarial Networks*, a conference paper published in *IEEE Transactions on Cybernetics* in 2022. I was the first author of this paper, designed the framework, conducted the experimental procedure, and wrote the manuscript. Konstantinos Vougioukas provided the code for his previous video-to-speech project (Vougioukas *et al.*, 2019a), which served as an important baseline and made substantial contributions to discussions surrounding the design and implementation of the model. Pingchuan Ma contributed to our implementation by aiding in the development of our model architecture and trained several pre-trained speech recognition models which were essential for our evaluation procedure. Stavros Petridis, Björn Schuller, and Maja Pantic supervised the project throughout its entirety, providing helpful suggestions, feedback, and guidance regarding the design of the framework and the writing of the manuscript.

As in the previous chapter, it should be noted that, in order to adequately contextualize the work portrayed in this chapter, the related work and comparison with other works presented below only include works presented before the time of submission (December 2020). Since then,

this work has been outperformed on GRID, TCD-TIMIT, and LRW by new video-to-speech approaches (Kim *et al.*, 2023; Hsu *et al.*, 2022), including the one presented in Chapter 5.

4.1 Introduction

The previous chapter focused on lip reading, and how we achieved state-of-the-art performance on this impactful task. However, although the translation from video-to-text can be achieved with remarkable consistency, various applications would benefit from a video-to-audio model, such as videoconferencing in noisy conditions, speech inpainting (Zhou *et al.*, 2019), i. e., filling in audio gaps from video in an audio-visual stream, or generating an artificial voice for people suffering from aphonia (i. e., people who are unable to produce voiced sound). One approach for this task would be to simply combine a lip reading model (such as the one presented in the previous chapter) with a text-to-speech (TTS) model. This approach is especially attractive since state-of-the-art TTS models can now produce realistic speech with considerable efficacy (Shen *et al.*, 2018; Oord *et al.*, 2018).

Combining video-to-text and text-to-speech models to perform video-to-speech has, however, some disadvantages. Firstly, these models require large transcribed datasets, since they are trained with text supervision. This is a sizeable constraint given that generating transcripts is a time-consuming and expensive process. Secondly, generation can only happen as each word is recognized, which imposes a delay on the throughput of the model, jeopardizing the viability of real-time synthesis. Lastly, using text as an intermediate representation removes any intonation and emotion from the spoken statement, which are fundamental for natural-sounding speech.

Given these constraints, some authors have developed end-to-end video-to-speech models which circumvent these issues. The first of these models (Cornu and Milner, 2015) used visual features based on discrete cosine transform (DCT) and active appearance models (AAM) to predict linear predictive coding (LPC) coefficients and mel-filterbank amplitudes. Following works have mostly focused on predicting spectrograms (Ephrat *et al.*, 2017; Akbari *et al.*, 2018; Prajwal *et al.*, 2020a), which is also a common practice in text-to-speech works (Shen *et al.*,

2018). These models achieve intelligible results but are only applied to seen speakers, i.e., there is an exact correspondence between the speakers in the training, validation, and test sets, or choose to focus on single speaker speech reconstruction (Prajwal *et al.*, 2020a). Recently, Michelsanti *et al.* (2020a) have proposed an alternative approach based on predicting WORLD vocoder parameters (Morise *et al.*, 2016) which generates clear speech for unseen speakers as well. However, the reconstructed speech is still not realistic.

It is clear that previous works have avoided synthesizing raw audio, likely due to the lack of a suitable loss function, and have focused on generating intermediate representations which are then used for reconstructing speech. To the best of our knowledge, the only work which directly synthesizes the raw audio waveform from video is by Vougioukas *et al.* (2019a). This work introduces the use of GANs (Goodfellow *et al.*, 2014; Arjovsky *et al.*, 2017), and thanks to the adversarial loss, it can directly reconstruct the audio waveform. This approach also produces realistic utterances for seen speakers and is the first to produce intelligible speech for unseen speakers.

Our approach builds upon the work of Vougioukas *et al.* (2019a) by proposing architectural changes to the model, and to the training procedure. Firstly, we replace the original encoder composed of five stacked convolutional layers with a ResNet-18 (He *et al.*, 2016) composed of a front-end 3D convolutional layer (followed by a max pooling layer), four blocks containing four convolutional layers each and an average pooling layer, as in Chapter 3. Additionally, we replace the GRU (Gated Recurrent Unit) layer following the encoder with two bidirectional GRU layers, increasing the capacity of our temporal model. The adversarial methodology was a major factor in generating intelligible waveform audio in the original model proposed by Vougioukas *et al.* (2019a). Hence, our approach is also based on the Wasserstein GAN (Arjovsky *et al.*, 2017), but we propose a new critic adapted from Kumar *et al.* (2019a). We also propose an additional critic which discriminates real from synthesized spectrograms.

Furthermore, we revise the loss configuration presented by Vougioukas *et al.* (2019a). Firstly, we decide to forego the use of the total variation loss and the L1 loss, as their benefit was minimal. Secondly, we use the recently proposed PASE (Problem Agnostic Speech Encoder) (Pascual

et al., 2019b) as a perceptual feature extractor. Finally, we propose two additional losses: the power loss and the MFCC loss. The power loss is an L1 loss between the (log-scaled) spectrograms of the real and generated waveforms. The MFCC loss is an L1 Loss between the MFCCs (Mel Frequency Cepstral Coefficients) of the real and generated waveforms.

Our contributions for this chapter are described as follows: **1)** We propose a new approach for reconstructing waveform speech directly from video based on GANs without using any intermediate representations. We use two separate critics to discriminate real from synthesized waveforms and spectrograms, respectively, and apply three comparative losses to improve the quality of outputs. **2)** We include a detailed ablation study where we measure the effect of each component on the final model. We also investigate how the type of visual input, size of the training set, and range of vocabulary affect the performance. **3)** We show results on two different datasets (GRID (Cooke *et al.*, 2006b) and TCD-TIMIT (Harte and Gillen, 2015)) for seen speakers. We find that our model substantially outperforms the state-of-the-art for GRID and adapts well to a larger pool of speakers. **4)** We also include results for unseen speakers on two datasets (GRID and LRW (Chung and Zisserman, 2016a)). We show that our model achieves intelligible results, even when applied to utterances recorded ‘in the wild’, and outperforms the state-of-the-art for the corpora we present. **5)** Finally, we study our model’s ability to generalize for videos of silent speakers and discuss our findings.

4.2 Related Work

Video-driven speech reconstruction is effectively the combination of two tasks: lip reading and speech synthesis. As such, we begin by briefly describing the main works in each field and then go on to describe existing approaches for video-to-speech.

4.2.1 Lip reading

Traditional lip reading approaches relied on HMMs (Hidden Markov Models) (Gurban and Thiran, 2009) or SVMs (Support Vector Machines) (Zhao *et al.*, 2009) to transcribe videos from manually extracted features such as DCT (Gurban and Thiran, 2009) or mouth geometry (Kumar *et al.*, 2007). Recently, end-to-end models have attracted attention due to their superior performance over traditional approaches. One of the first end-to-end architectures for lip reading was proposed by Assael *et al.* (2016). This model featured a convolutional encoder as the visual feature extractor and a two-layer BGRU-RNN (Bidirectional GRU recurrent neural network) followed by a linear layer as the classifier, and it achieved state-of-the-art performance for the GRID corpus. This work was followed by Chung and Zisserman (2016b), whose model relied entirely on CNNs (Convolutional Neural Networks) and RNNs, and was successfully applied to spoken utterances recorded in the wild.

Various works have followed which apply end-to-end deep learning models to achieve competitive lip reading performance. Petridis *et al.* (2017a, 2020) propose an encoder composed of fully connected layers and performs classification using LSTMs (as described in Chapter 2). Other works choose to use convolutional encoders (Shillingford *et al.*, 2019), often featuring residual connections (Stafylakis and Tzimiropoulos, 2017), and then apply RNNs to perform classification. Furthermore, these end-to-end architectures have been extended for multi-view lip reading (Petridis *et al.*, 2017b) and audio-visual (Petridis *et al.*, 2018a) speech recognition. Finally, as we have shown in Chapter 3 with LiRA, self-supervised pre-training stages, as well as transformer-based architectures, have also been leveraged to push the state-of-the-art in this field.

4.2.2 Speech Synthesis

One of the most popular speech synthesis models in recent years has been WaveNet (Oord *et al.*, 2016), which proposed dilated causal convolutions to compose waveform audio sample by sample, taking advantage of the large receptive field achieved by stacking these layers. This

model achieved far more realistic results than any artificial synthesizer proposed before then. Another work (Wang *et al.*, 2017b) introduced a vastly different sequence-to-sequence model that predicted linear-scale spectrograms from text, which were then converted into waveform using the Griffin-Lim Algorithm (GLA) (Zhu *et al.*, 2007). This process produced very clear and intelligible audio. In the following years, Shen *et al.* (2018) combined these two methodologies to push the state-of-the-art once more, and Oord *et al.* (2018) accelerated and improved the original WaveNet.

The first model to apply GANs for end-to-end speech synthesis was proposed by Donahue *et al.* (2019), which used simple convolutional networks with large kernels as the generator and discriminator and applied the improved Wasserstein loss (Gulrajani *et al.*, 2017). In a later work (Yamamoto *et al.*, 2019), the original WaveNet vocoder (Oord *et al.*, 2016) has been combined with the adversarial methodology introduced by Donahue *et al.* (2019). This results in a network that has far fewer parameters than the original WaveNet, but remains on par with the latest WaveNet-based models. Recently, the first end-to-end adversarial Text-To-Speech model (Donahue *et al.*, 2020) was also proposed, whose performance is comparable to the state-of-the-art.

4.2.3 Reconstructing audio from visual speech

To the best of the authors' knowledge, the first work to attempt the task of video-to-speech synthesis directly was proposed by Cornu and Milner (2015). The proposed model aims to predict the spectral envelope (LPC or mel-filterbanks) from manually extracted visual features (DCT or AAM) using Gaussian Mixture Models (GMMs) or deep neural networks. These acoustic parameters are then fed into an HMM-based vocoder, together with an estimate of the voicing parameters. Through multiple user studies, the speech reconstructed by this model is shown to have fairly low intelligibility (WER $\approx 50\%$), but shows that this task is indeed achievable. This work was extended by Cornu and Milner (2017), which introduced additional temporal information in the visual features and in the model itself. These improvements yielded an impressive 15% WER for GRID (single speaker), based on user studies.

The next development in this field came with Ephrat and Peleg (2017), who use a deep CNN architecture to predict acoustic features – LPC analysis followed by LSP (Line Spectral Pairs) decomposition, frame by frame – from gray-scale video frames. These are combined with white noise (excitation signal) and fed into a source-filter speech synthesizer which produces unvoiced speech. This model produces intelligible results ($\text{WER} < 20\%$) when trained and tested on a single speaker from GRID, and constitutes a step forward given that it no longer relies on handcrafted visual features as input. An improved version of this model was later introduced (Ephrat *et al.*, 2017), which predicts spectrograms that are then translated into waveform using the Griffin-Lim algorithm. This extension also proposes a new encoder composed of two ResNet-18s followed by a post-processing network that increases temporal resolution. This work is the first to experiment with multiple speakers and achieves much more realistic speech than any previous work for this task.

Lip2Audspec (Akbari *et al.*, 2018) proposes a similar CNN+RNN encoder to predict spectrograms directly from the gray-scale frames of the video. As in the work presented above (Ephrat *et al.*, 2017), the spectrograms are converted to waveform using a phase estimation method. The resulting spectrograms are very close to the original samples, but the reconstructed waveforms sound clearly artificial. Another recent work (Michelsanti *et al.*, 2020a) uses CNNs+RNNs to predict vocoder parameters (aperiodicity and spectral envelope), rather than spectrograms. Additionally, the model is trained to predict the transcription of the speech, in other words performing speech reconstruction and recognition simultaneously in a multi-task fashion. This approach achieves results that are very impressive when measured with objective speech quality metrics (PESQ, STOI), but yields samples that still sound noticeably robotic.

Finally, a recent paper (Prajwal *et al.*, 2020a) proposes an approach based on the Tacotron 2 architecture (Shen *et al.*, 2018), predicting mel-frequency spectrograms from video rather than text. To perform this task, it applies a stack of residual 3D convolutional layers as a spatiotemporal encoder for the video and combines it with an attention-based decoder adapted from Shen *et al.* (2018), which generates the spectrograms. Unlike Tacotron, these spectrograms are decoded into waveform audio using the Griffin-Lim algorithm (Zhu *et al.*, 2007) rather than WaveNet, as the authors claim the generated spectrograms are not as accurate as modern TTS

works, and therefore do not perform well with neural vocoders. This work is able to generate remarkably intelligible audio from visual speech and achieves state-of-the-art performance in all presented metrics. However, it focuses on speaker-specific speech reconstruction, i. e., it is trained and tested on the same speaker.

An aspect that is worth highlighting is that none of these models attempt to generate the waveform end-to-end from video, instead predicting spectrograms or other features which can be translated into waveform. This is likely due to the notoriously arduous task of generating realistic waveforms, which can be attributed to the lack of suitable loss functions. The only model to perform video-to-waveform speech reconstruction without the use of intermediate representations is presented by Vougioukas *et al.* (2019a). This paper introduces the use of conditional generative adversarial networks (Mirza and Osindero, 2014), which have been popularized in image-related generative tasks (Reed *et al.*, 2016; Isola *et al.*, 2017), for the task of video-to-speech synthesis, which can be seen as speech synthesis conditioned on video. The model is based on a convolutional encoder-decoder model (combined with a GRU) which encodes video into visual features and decodes them directly into waveform audio. The generator is trained with an adversarial loss based on a convolutional waveform critic, as well as three other comparative losses. This procedure achieves competitive results for speech reconstruction on both seen and unseen speaker datasets (GRID).

4.2.4 Reconstructing audio from multi-view visual speech

The majority of works in video-driven speech reconstruction use frontal views of the face. In this section, we briefly describe a set of works that instead use multiple views, i. e., videos of the same spoken utterance taken from multiple camera angles to capture different parts of the speaker’s face (typically recorded in studios), to improve the quality of the reconstructed speech.

The first work to use multi-view video for this task was proposed by Kumar *et al.* (2018b). This model is very similar to the work of Ephrat and Peleg (2017) in the sense that it applies a CNN to extract visual features directly from video, which then predicts vocoder parameters (LPC

followed by LSP). This work, however, uses video taken from two different angles for every speaker (Oulu VS2 dataset (Anina *et al.*, 2015)). The results presented in this paper show that the use of multiple views can substantially improve speech reconstruction performance.

This model was later improved (Kumar *et al.*, 2018c) by replacing the LSTM with a BGRU and using more than two views as input. It is shown that the use of three views can yield improvements of 20 % in the quality of reconstructed outputs (measured with PESQ). This has been extended (Kumar *et al.*, 2019b) by including a view classifier to attribute view labels to the input videos and by also generating text transcriptions. The latest work in this field (Uttam *et al.*, 2019) follows the trend seen in single-view speech reconstruction research (Ephrat *et al.*, 2017; Ephrat and Peleg, 2017) and speech synthesis in general (Tabet and Boughazi, 2011; Wang *et al.*, 2017b) by switching from LPC coefficients to spectrograms as the predicted audio representation.

4.2.5 Audio reconstruction from video in other applications

Finally, a set of past works has approached the application of video-to-audio models to domains outside speech (Owens *et al.*, 2016; Zhou *et al.*, 2018; Chen *et al.*, 2017). Namely, these papers have focused on a diverse range of datasets that feature a set of generic sounds such as fireworks and drums (Zhou *et al.*, 2018); different instruments being played (Chen *et al.*, 2017); or even objects composed of different materials being hit with a drumstick (Owens *et al.*, 2016). The methodology applied to reconstruct audio from video is similar to what is seen for video-to-speech systems. CNNs are applied to encode the video frames, followed by RNNs or fully connected layers to produce acoustic features which are decoded into audio using vocoders. While some of these works struggle to reproduce the corresponding audio, Zhou *et al.* (2018) can produce remarkably realistic audio (as proven by its user studies) by combining the extraction of optical flow with a neural network-based vocoder.

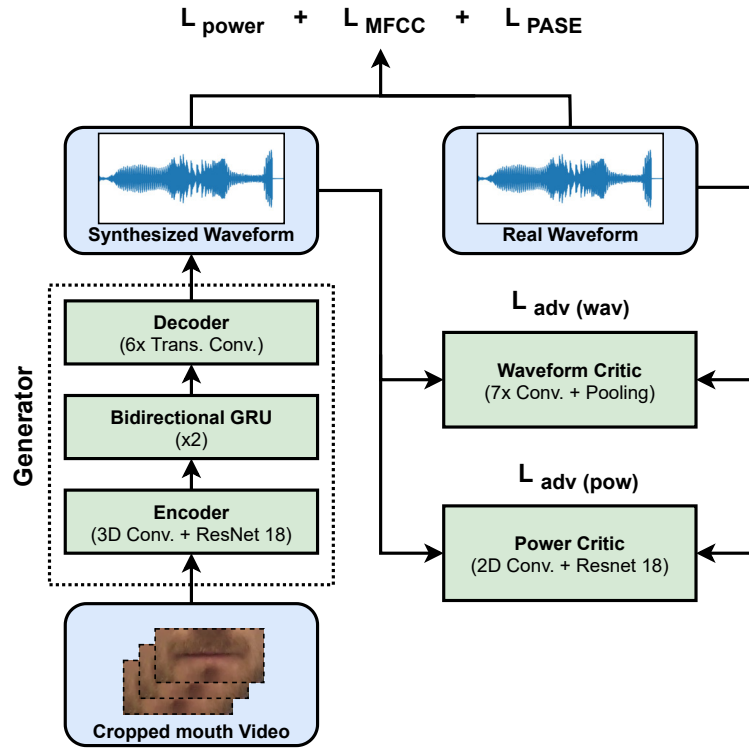


Figure 4.1: Architecture of the generator (encoder, bidirectional GRU, decoder) and critics (waveform critic, power critic) used in this chapter, as well as the losses that are used for training.

4.3 Video-Driven Speech Reconstruction

Our model is composed of a video encoder based on a ResNet-18 combined with a Bidirectional GRU, as well as a convolutional decoder that transforms the visual features into waveform audio. This generator is trained using two separate critics, to maximize the realism of the outputs, as well as three L1 losses to minimize the difference between real and synthesized audio for each video.

4.3.1 Generator

Given that we aim to synthesize speech directly from video, our generator accomplishes two sequential tasks: encoding temporal visual features and decoding them into an audio waveform. Firstly, we encode the frames of the video using a Resnet-18 preceded by a spatiotemporal 3D convolutional layer (combined with a max pooling layer), as in Chapter 3. This initial layer has a receptive field of five frames centered on the frame it will encode meaning that the encoding

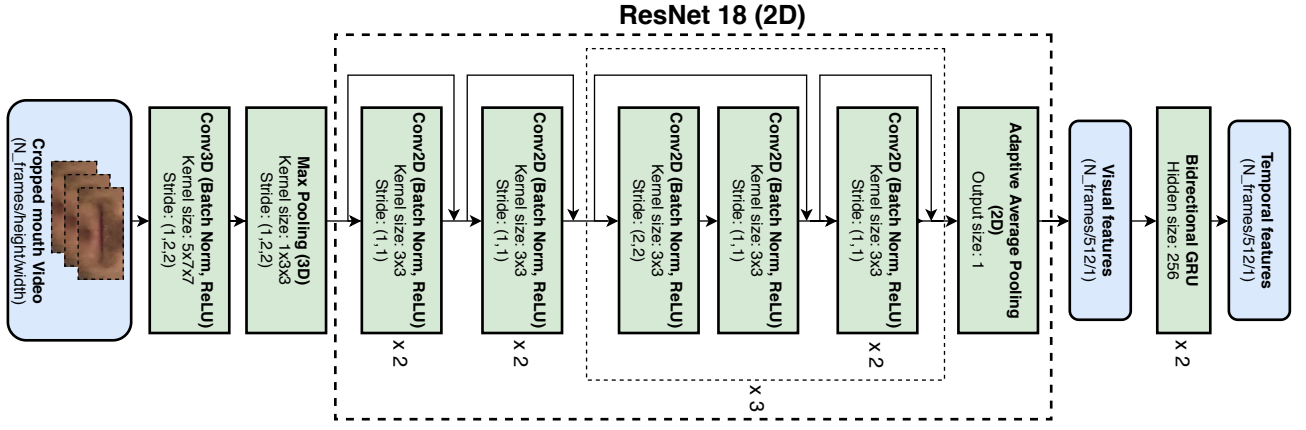


Figure 4.2: Description of the encoder layers in our end-to-end video-to-speech generator.

for each frame will depend on the previous two frames and the following two frames. We experimented with different numbers of frames as input to this layer (3 and 7) but found that this did not considerably affect results. The ResNet-18 is composed of 4 blocks of 4 convolutional layers, each followed by batch normalization and ReLU (Rectified Linear Unit) activation, and an adaptive average pooling layer. The features extracted from the ResNet encoder are then fed into a 2-layer bidirectional GRU (as described in Chapter 2) which temporally correlates the features produced from each set of frames. This encoder is adapted from the audio-visual speech recognition work (Petridis *et al.*, 2018a) mentioned in the previous section. This architecture is described in detail in Figure 4.2.

After this, the decoder upsamples the features from each video frame into a waveform segment of N audio samples. The length of each segment is given by:

$$N = \frac{\text{audio sampling rate}}{\text{video frame rate}}. \quad (4.1)$$

Since we use a sampling rate of 16 kHz and a frame rate of 25 frames per second, N is equal to 640 (corresponding to 40 ms of audio). The decoder is composed of six stacked transposed convolutional layers, each followed by batch normalization and ReLU activation except for the last layer which uses a hyperbolic tangent activation function. In an attempt to alleviate the issue of abrupt frame transitions, we use an overlap of 50% between the generated waveform frames, as proposed by Ephrat and Peleg (2017). The overlapped segments are linearly averaged

sample by sample in order to maintain the original waveform scale. The detailed architecture of the decoder is shown in Figure 4.3. The end-to-end waveform generation formula can therefore be defined as:

$$a = \mathbb{D}(\mathbb{T}(\mathbb{E}(v))), \quad (4.2)$$

where v is the input video, a is the generated audio, $\mathbb{E}$ is the encoder (Figure 4.2), $\mathbb{T}$ is the temporal model ((Figure 4.2)) and $\mathbb{D}$ is the decoder (Figure 4.3). The full generator is shown in Figure 4.2.

4.3.2 Critics

As demonstrated in recent works (Donahue *et al.*, 2019; Yamamoto *et al.*, 2019; Kumar *et al.*, 2019a), the use of a waveform critic can dramatically increase the realism and clarity of synthesized speech. To discriminate the real from the synthesized waveforms, we adapt the critic proposed by Kumar *et al.* (2019a). After experimenting extensively with and without weight normalization for this module, as well as for the generator, we find that weight normalization increases the stability of adversarial training but overall leads to worse results. Therefore, we remove weight normalization from this critic but otherwise keep the original architecture: 7 convolutional layers, each followed by Leaky ReLU activation, as shown in Figure 4.4a.

We did not attempt batch normalization, which worked well for the generator since this interferes with the gradient penalty for our adversarial loss (Gulrajani *et al.*, 2017). We compared this architecture to other convolutional critics similar to the one proposed by Donahue *et al.* (2019) as well as a one-dimensional ResNet-18 and found that this critic produced the best results. Remarkably, this critic has a far smaller receptive field than any of the critics we experimented with. This may indicate that waveform critics work best when focusing on the small scale.

Inspired by SpecGAN (Donahue *et al.*, 2019), we combine the waveform critic, which judges the audio in the temporal domain, with a power critic, which judges the audio in the spectral domain. This module discriminates the spectrograms computed from real and generated audio.

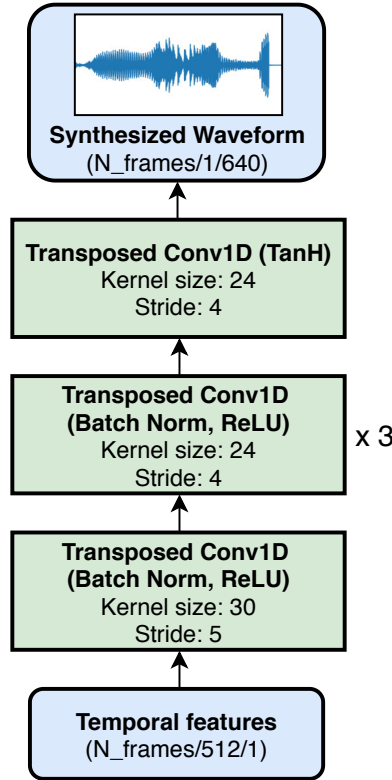


Figure 4.3: Description of the decoder layers in our end-to-end video-to-speech generator.

We first compute the spectrogram from both the real and generated samples using the short-time Fourier transform (STFT) with a window size of 25 ms, a hop size of 10 ms, and frequency bins of size 512. We then compute the natural logarithm of the spectrogram magnitudes, normalize these values to mean 0 and variance 1, clip values outside $[-3, 3]$, and normalize them to $[-1, 1]$, similarly to Donahue *et al.* (2019). In this case, we use a ResNet18 identical to the one presented in our generator, except with a two-dimensional front-end convolutional layer in the beginning, since our input is a single image. As with the waveform critic, we cannot use batch normalization in this module due to the gradient penalty and found that weight normalization did not improve results. The architecture for the power critic is shown in Figure 4.4b.

4.3.3 Losses

To train our network, we apply the Wasserstein GAN loss (Arjovsky *et al.*, 2017), which aims to minimize the Wasserstein Distance between the distributions of real and synthesized data. We also add the gradient penalty (Gulrajani *et al.*, 2017) to satisfy the Lipschitz constraint

in the Wasserstein GAN objective (as explained in detail in Chapter 2). The losses for the generator and respective critic(s) are defined as:

$$L_G = - \mathbb{E}_{\tilde{x} \sim \mathbb{P}_G} [D(\tilde{x})] + \lambda \mathbb{E}_{\hat{x} \sim \mathbb{P}_{\hat{x}}} [(\|\nabla_{\hat{x}} D(\hat{x})\|_2 - 1)^2] \quad (4.3)$$

$$L_D = \mathbb{E}_{\tilde{x} \sim \mathbb{P}_G} [D(\tilde{x})] - \mathbb{E}_{x \sim \mathbb{P}_R} [D(x)], \quad (4.4)$$

where G is the generator, D is the critic, $x \sim \mathbb{P}_R$ are samples from the real distribution, $\tilde{x} \sim \mathbb{P}_G$ are samples from the estimated distribution (produced by the generator) and $\hat{x} \sim \mathbb{P}_{\hat{x}}$ are sampled uniformly between two points from P_G and P_R respectively. In this work, we apply two critics: the waveform critic and the power critic. Each critic is trained with their own losses $L_{D_{wave}}$ and $L_{D_{power}}$, whereas the generator combines the losses from the two critics such that:

$$L_{G_{adv}} = L_{G_{wave}} + L_{G_{power}}, \quad (4.5)$$

where $L_{G_{wave}}$ and $L_{G_{power}}$ are calculated as mentioned in Eq. 4.3. The coefficient for the gradient penalty λ is kept at 10 for both critics, as proposed by Gulrajani *et al.* (2017).

In addition to this adversarial loss, we also apply three other losses to train the generator. The first is a perceptual loss:

$$L_{PASE} = \|\delta(x) - \delta(\tilde{x})\|_1, \quad (4.6)$$

where x is the real waveform, $\tilde{x}$ is the synthesized waveform from the same video and δ is our perceptual feature extractor. In this chapter, we use the pre-trained PASE model (Pascual *et al.*, 2019b) to extract perceptual features $\delta(x)$. PASE has been trained in a self-supervised manner to produce meaningful speech representations. We have also tried using PASE+ (Ravanelli *et al.*, 2020b), which is an improved version of PASE that we used to train our model in Chapter 3. However, no improvement in the speech reconstruction quality was observed. Furthermore, we experimented with multiple ASR models as feature extractors, but we found that they also did not improve results.

The second loss we apply is the Power Loss. This function aims to improve the accuracy of the reconstructed audio by attempting to match it with the real audio in the frequency domain. For this purpose, we use the L1 loss between the STFT magnitudes of the real and synthesized audio (as described in Chapter 2) as follows:

$$L_{power} = \|\log |STFT(x)|^2 - \log |STFT(\tilde{x})|^2\|_1, \quad (4.7)$$

where x is the real waveform, $\tilde{x}$ is the synthesized waveform from the same video and $STFT$ is the Short Time Fourier Transform with a window size of 25 ms, a hop size of 10 ms and frequency bins of size 512 (same parameters used for the power critic). We found that scaling the magnitudes using the natural logarithm and using an L1 Loss rather than the L2 Loss presented by Oord *et al.* (2018) greatly improves training stability and performance.

The third loss we apply is the MFCC Loss:

$$L_{MFCC} = \|MFCC(x) - MFCC(\tilde{x})\|_1, \quad (4.8)$$

where x is the real waveform, $\tilde{x}$ is the synthesized waveform from the same video and $MFCC$ is the MFCC function which extracts 25 mel-frequency cepstral coefficients from the corresponding waveform. The objective of this loss lies in increasing the accuracy and intelligibility of the synthesized speech, given that MFCCs are known to be effective in ASR (Han *et al.*, 2006) and emotion recognition (Krishna Kishore and Krishna Satish, 2013). We adapt the function provided on an open-source repository¹.

Finally, the loss for the generator is described based on the losses mentioned above as:

$$L_G = \alpha_1 L_{G_{adv}} + \alpha_2 L_{PASE} + \alpha_3 L_{power} + \alpha_4 L_{MFCC}. \quad (4.9)$$

We tune the coefficients (α_1 , α_2 , α_3 , and α_4) by sequentially training multiple models on GRID (4 speakers, seen speaker split) and incrementally finding the coefficients that yield the best WER on the validation set. Through our search, we find that $\alpha_1 = 1$, $\alpha_2 = 140$, $\alpha_3 = 50$,

¹<https://github.com/skaws2003/pytorch-mfcc>

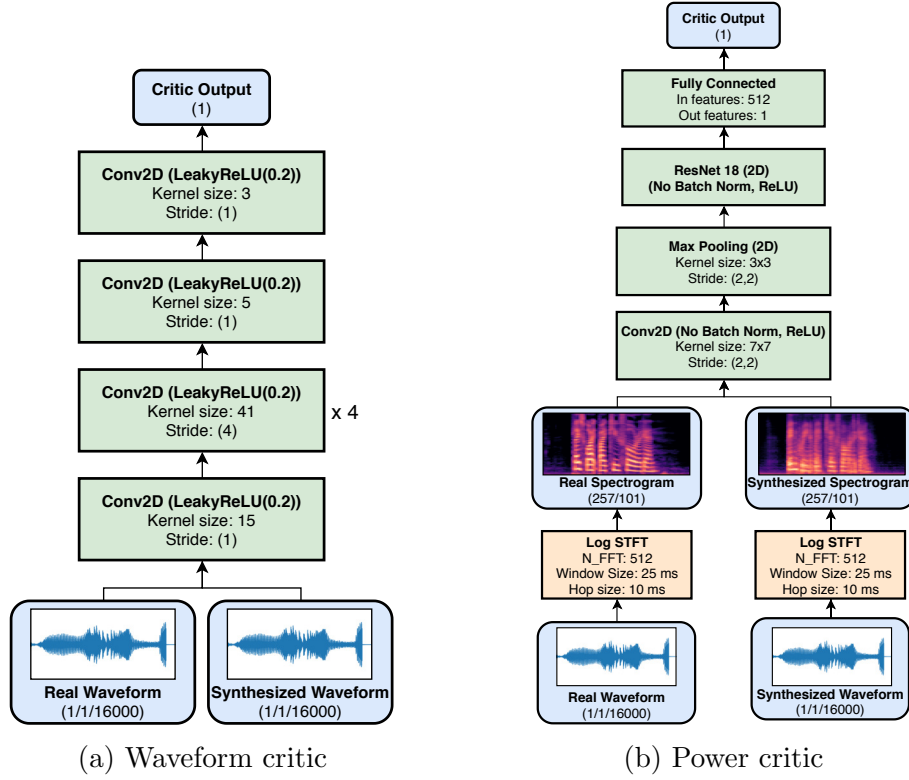


Figure 4.4: Description of the layers in the waveform critic (a) and power critic (b) used to train our model.

$\alpha_4 = 0.4$ yield the best results.

4.3.4 Training details

We use the Adam optimizer with a learning rate of 0.0001 and $\beta_1 = 0.5$, $\beta_2 = 0.99$ to train our generator and critics end-to-end. Given that the critics should be trained to completion before every generator training step, we perform 6 training steps on the critics before every training step of the generator. It should also be noted that we feed a one-second clip randomly sampled from the real and synthesized audio to each of the critics, rather than the entire utterance. The other losses are computed using the entire real and synthesized utterances. Additionally, we employ random cropping and horizontal flipping during training, as in Chapter 3.

Training our model for each of the experiments generally takes approximately one week on an Nvidia RTX 2080 Ti GPU. We train our model to completion, i.e., until the validation metrics plateau, using a batch size of 10 for all experiments. Synthesizing a 3-second audio clip

Table 4.1: Number of speech clips and total number of hours of speech for each dataset used in our study.

Corpus	Training set (clips / hours)	Validation set (clips / hours)	Test set (clips / hours)
GRID (4 speakers, seen speakers)	3,576 / 2.98	210 / 0.18	210 / 0.18
GRID (33 speakers, seen speakers)	29,584 / 24.65	1,642 / 1.37	1,641 / 1.37
GRID (33 speakers, unseen speakers)	15,888 / 13.24	7,000 / 5.83	9,982 / 8.32
TCD-TIMIT (3 lip speakers)	1014 / 1.64	57 / 0.09	60 / 0.09
LRW (full)	488,763 / 157.49	25,000 / 8.06	25,000 / 8.06
FLRW 500 Words	112,811 / 36.35	5878 / 1.89	5,987 / 1.93
FLRW 100 Words	22,055 / 7.11	1151 / 0.37	1,144 / 0.37
FLRW 20 Words	4,347 / 1.40	266 / 0.09	248 / 0.08

sampled at 16 kHz from 75 frames of video takes approximately 32 ms on the same high-end GPU, excluding pre-processing. Our generator has a total of 45.8 million parameters, whereas the waveform and power critics have 5.6 million and 11.2 million parameters respectively.

4.4 Datasets

For the purposes of this chapter, we use three separate audio-visual datasets to train and evaluate our model: GRID, TCD-TIMIT, and LRW. The content of these datasets is explained in detail in Chapter 2. To compare with previous works (Prajwal *et al.*, 2020a) on TCD-TIMIT, we only use the audio-visual data uttered by the three lip speakers, rather than the entire corpus. This results in a total of 1,131 clips. For LRW, in addition to the full corpus, we use a subset of this corpus that keeps only the approximately frontal videos, i.e., videos with yaw, pitch, and roll below 10 degrees. This leads to a corpus containing 124,676 samples in total and will be referred to as $F(\text{rontal})LRW$. We also randomly select 20/100 words from this subset to experiment with different ranges of vocabulary during training/testing. These

smaller sets will be referred to as *FLRW20* and *FLRW100*, respectively. Further statistics for each dataset are presented in Table 4.1. Rather than using the full face as input to our network, as is standard in other speech reconstruction works (Ephrat *et al.*, 2017; Akbari *et al.*, 2018; Prajwal *et al.*, 2020a), we crop the mouth of the speaker using dlib’s 68 landmark model (King, 2009), as explained in Chapter 2. The mouth ROI is of size 128x74 for GRID and 96x96 for TCD-TIMIT and LRW.

4.5 Evaluation Metrics

Although many metrics have been proposed for evaluating the quality of speech (Loizou, 2011), it is widely acknowledged that none of the existing metrics are highly correlated with human perception. For this reason, we evaluate our speech reconstruction model using 4 objective metrics which capture different properties of the audio: PESQ, STOI, MCD, and WER. Each of these metrics is explained in detail in Chapter 2

4.6 Results on Seen Speakers

In this section, we present our experiments for seen speakers. For direct comparison with other works we use 4 speakers from GRID (1, 2, 4 and 29), as proposed by Vougioukas *et al.* (2019a), Akbari *et al.* (2018), Prajwal *et al.* (2020a), and Michelsanti *et al.* (2020a), and the 3 lip speakers from TCD-TIMIT, as proposed by Prajwal *et al.* (2020a). In order to investigate the impact of the number of speakers and the amount of training data, we also present results for all 33 speakers from the GRID dataset. We split the utterances in each of these datasets using a 90–5–5 % ratio for training, validation, and testing respectively similarly to Vougioukas *et al.* (2019a), Akbari *et al.* (2018), Prajwal *et al.* (2020a), and Michelsanti *et al.* (2020a), such that the speakers in the validation and test sets are identical to the speakers seen in the training set (but the utterances are different). To measure the Word Error Rate (WER) for our GRID samples, we use a pre-trained ASR model (based on the work of Ma *et al.* (2019)) which was

trained and tested on the full GRID dataset (using the split mentioned in Section 4.7), achieving a baseline of 4.23 % WER on the test set. During training, we save the checkpoint with the lowest WER on the validation set at the end of each epoch, to obtain the best possible model. For TCD-TIMIT, we were unable to obtain an accurate publicly available speech recognition model and thus did not use WER as a metric. Therefore, we use STOI instead of WER as our criteria to select the best checkpoint for our TCD-TIMIT experiments. Audio samples, as well as spectrogram and waveform figures, are presented on our website² for the experiments presented in Sections 4.6, 4.7 and 4.8. Additionally, we present a publicly available repository³ which can be used to reproduce each of the evaluation metrics presented in this chapter. We are also available to provide generated test samples for researchers hoping to reproduce or compare with our work.

4.6.1 Ablation Study

Results for the ablation study are shown in Table 4.2. For this study, we use the GRID (4 speakers, seen speakers) dataset. Firstly, we observe that each of the three comparative losses L_{PASE} , L_{power} and L_{MFCC} yield considerable improvements in the verbal accuracy of samples (as shown by the WER), even when only one is removed. We can also observe that L_{MFCC} and L_{power} are particularly impactful on the MCD of the reported samples, which is unsurprising since this is an MFCC-based metric. On the other hand, it is clear that L_{PASE} is essential for achieving high intelligibility, given its particular impact on STOI. Finally, all three losses also seem to positively impact the PESQ score, indicating an increase in overall audio clarity.

We can see that the simultaneous removal of L_{PASE} and L_{power} greatly decreases PESQ and STOI, indicating that these losses are particularly important towards the clarity of generated samples. We also show that the absence of L_{MFCC} and L_{power} sharply increases MCD, indicating that these two losses greatly increase the similarity between real and synthesized audio. On the other hand, this model maintains a WER below 10 %, which means that L_{PASE} alone (together with the adversarial losses) can achieve intelligible audio. Finally, the removal of all three L1

²<https://sites.google.com/view/video-to-speech/home>

³<https://github.com/miraodasilva/evalaudio>

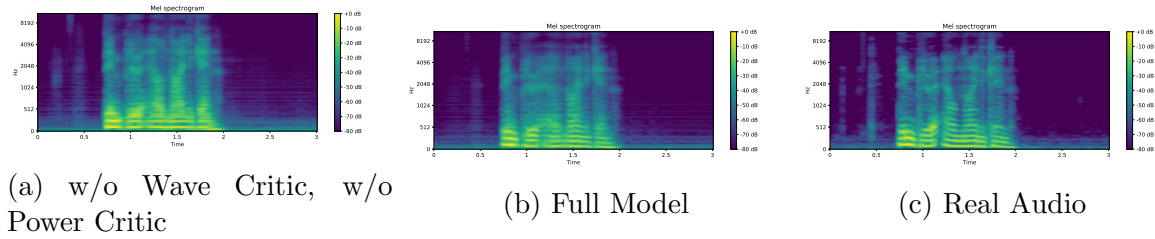


Figure 4.5: Mel-frequency spectrograms taken from the audio reconstructed with our seen speaker ablation models. The clip we present is from GRID, speaker 1, utterance 'Bin blue at L 9 again'.

losses results in realistic yet unintelligible audio. This is because the adversarial losses are the only objective used for training, and therefore there is no incentive for the network to learn the exact words corresponding to the input video.

We observe that the use of the waveform critic yields noticeable improvements through our metrics, particularly in WER and STOI, suggesting that its inclusion substantially increases intelligibility. Additionally, the power critic also yields moderate improvements in PESQ, STOI, and WER. Finally, we observe that the removal of both critics results in substantially lower MCD and WER, but maintains PESQ and STOI at a similar value. This again indicates that our model can generate intelligible and accurate words without adversarial losses. However, these synthesized samples lack realism, which drastically improves when the critics are used. To demonstrate this effect, readers are encouraged to listen to examples on our website².

We also experiment with using the full face as input, as this is commonly used in previous studies. Through this ablation, we show that using a cropped mouth region instead of the full face improves our results substantially in terms of WER, effectively improving intelligibility. We also prove that the use of overlap improves all metrics slightly, suggesting that its purpose of minimizing the issue of frame transitions is beneficial toward output quality. A qualitative comparison with other works can be seen in Figure 4.5. Compared to the real audio, our spectrogram is similar overall, but is slightly blurrier and fails to model some of the fine details in the frequency bins, especially in the higher frequencies. The model trained without adversarial critics features a much blurrier spectrogram than the full model, failing to reproduce even the lower frequency bands during voiced speech, highlighting the importance of adversarial training.

Table 4.2: Ablation study for our end-to-end video-to-speech model performed on GRID (4 speakers, seen speakers).

Model	PESQ	STOI	MCD	WER
w/o L_{PASE}	2.06	0.597	26.44	8.97 %
w/o L_{power}	2.05	0.575	28.64	9.54 %
w/o L_{MFCC}	2.08	0.591	28.09	9.09 %
w/o L_{PASE} , w/o L_{power}	1.86	0.545	27.47	13.44 %
w/o L_{PASE} , w/o L_{MFCC}	2.02	0.589	28.82	13.33 %
w/o L_{MFCC} , w/o L_{power}	2.00	0.569	31.43	9.71 %
w/o L_{PASE} , w/o L_{power} , w/o L_{MFCC}	1.14	0.311	53.63	89.12 %
w/o waveform critic	2.07	0.583	26.66	8.47 %
w/o power critic	2.08	0.594	26.73	7.30 %
w/o waveform critic, w/o power critic	2.07	0.584	27.45	9.01 %
w/o overlap	2.06	0.590	26.73	7.40 %
w/ full face	2.07	0.596	26.46	9.94 %
full model	2.10	0.595	26.78	7.03 %

4.6.2 Comparison with Other Works

We compare our proposed model with previous works on GRID (4 speakers, seen speakers) as shown in Table 4.3. We note that the metrics reported on Lip2Wav (Prajwal *et al.*, 2020a) are taken directly from their paper due to test samples not being publicly available and that their WER was calculated using the Google Speech-to-Text (STT) API rather than our ASR model.

Regarding PESQ, it is clear that our model is superior to the previous approaches by a sizeable margin. This suggests that the quality of our synthesized speech is somewhat higher than past models. Our model also outperforms previous works on STOI, excluding Lip2Wav. This shows that our samples are more intelligible than most other approaches, but are outperformed by the robustness and consistency of the speech produced by Lip2Wav. In particular, Lip2Wav’s very impressive performance on STOI compared to other metrics indicates that the generated speech is easily intelligible, but is lacking in perceptual quality and realism, which is confirmed by an empirical analysis of their samples. Furthermore, our generated samples achieve a better MCD than previous works, indicating that our reconstructed audio is more accurate than previous approaches on the frequency domain. Finally, our work achieves the best WER out of all methods, which shows that our model is more accurate than any of the previous approaches by a large factor, outperforming our previous model by more than 10 %.

A qualitative comparison is shown in Figure 4.6, which displays waveforms, mel-frequency spectrogram, and mel-frequency spectrogram differences, i.e., the element-wise absolute difference between the real and synthesized spectrograms. This difference is calculated as:

$$\|MelSpec(x) - MelSpec(\tilde{x})\|_1, \quad (4.10)$$

where x is the real waveform and $\tilde{x}$ is the synthesized waveform. Through the spectrograms, it is clear that Lip2Audspec is the least accurate in the frequency domain, failing to model many frequencies, particularly in the higher bands. The other three approaches are clearly more accurate, but all feature some inaccuracies during voiced speech and also noise in unvoiced segments. While the samples generated by Vougioukas *et al.* (2019a) and Michelsanti *et al.*

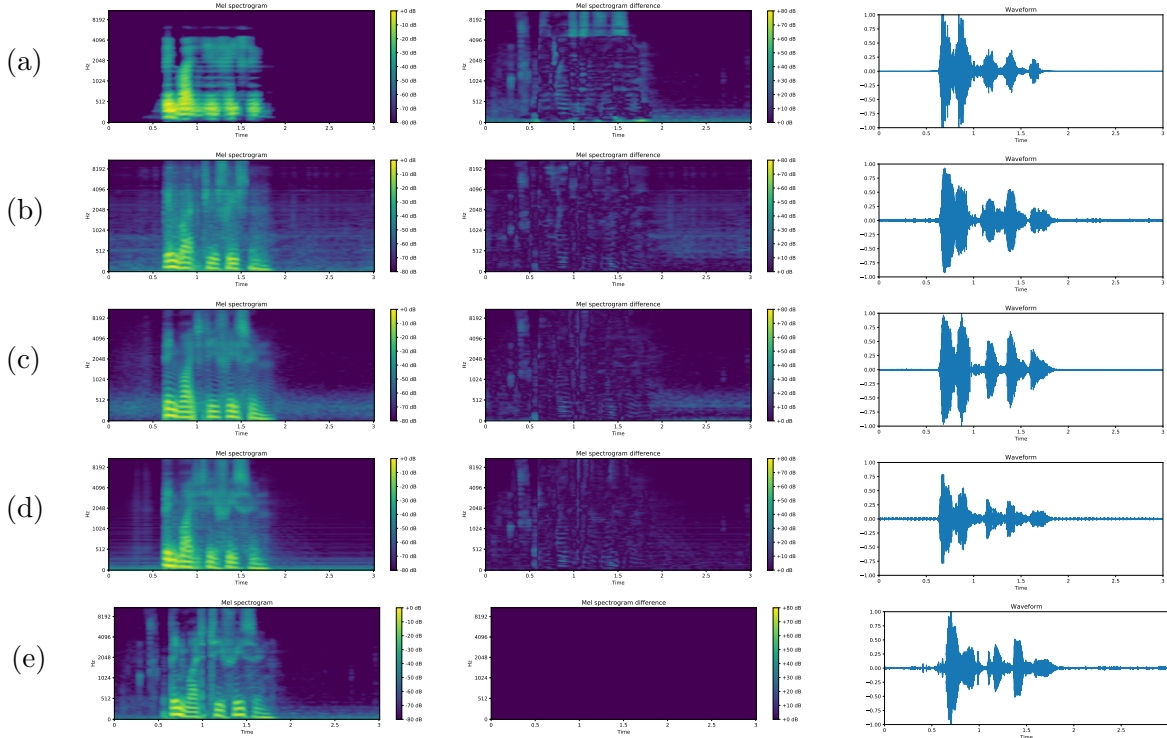


Figure 4.6: Mel-frequency spectrograms (left), mel-frequency spectrogram differences (middle) and waveforms (right) taken from the audio reconstructed with Lip2AudSpec (Akbari *et al.*, 2018) (a), our previous work (Vougioukas *et al.*, 2019a) (b), a previous vocoder-based model (Michelsanti *et al.*, 2020a) (c) and our model (d), as well as the real audio (e) – GRID, Speaker 1, utterance ‘Bin white at T 3 soon’. All models were trained on the same split of GRID (4 speakers, seen speakers), as presented in our comparison.

(2020a) feature an excessive amount of low-frequency noise, our model seems to accurately emulate the low amount of noise in the real audio and therefore achieves the least substantial spectrogram difference.

We also compare our model to Lip2Wav on TCD-TIMIT (3 lip speakers) in Table 4.4. Once more, it is clear that our model outperforms Lip2Wav (Prajwal *et al.*, 2020a) on PESQ, but achieves lower performance on STOI, which indicates that our model produces clearer, yet somewhat less intelligible audio. Additionally, our samples achieve a reasonably low MCD, indicating moderate similarity in the frequency domain.

Table 4.3: Comparison between our model and previous works, using GRID (4 speakers, seen speakers).

Method	PESQ	STOI	MCD	WER
Lip2Audspec (Akbari <i>et al.</i> , 2018)	1.81	0.425	63.88	46.36 %
GAN-based (Vougioukas <i>et al.</i> , 2019a)	1.70	0.539	45.37	21.11 %
Vocoder-based (Michelsanti <i>et al.</i> , 2020a)	1.90	0.553	46.64	22.14 %
Lip2Wav (Prajwal <i>et al.</i> , 2020a)	1.77	0.731	-	14.08 ^a %
Ours	2.10	0.595	26.78	7.03 %

^aReported using Google STT API.

Table 4.4: Comparison between our model and Lip2Wav, using TCD-Timit (3 lip speakers) with a seen speaker split.

Method	PESQ	STOI	MCD
Lip2Wav (Prajwal <i>et al.</i> , 2020a)	1.35	0.558	-
Ours	1.61	0.295	32.12

4.6.3 Performance as a Function of Training Set Size

For the purposes of this study, we use GRID (33 speakers, seen speakers) and we report results as we vary the size of the training set from 20 % to 100 % in steps of 20 %. Results are shown in Table 4.5. When compared to the results reported for GRID (4 speakers, seen speakers), we observe comparable performance for 33 speakers when using the full training set. This shows that our network adapts well to larger datasets and is able to model a large number of speakers with no substantial drop in performance.

Regarding the models which are trained using a smaller subset of the training set, it is clear that the performance drops as the amount of training data is gradually reduced. However, it is worth highlighting that the overall performance remains moderately consistent, even when we use only 20 % of the training data. This shows that our model adapts well to smaller datasets. We note that all 5 models were trained for the same amount of total training steps to avoid any bias in our comparative results.

Table 4.5: Study on the performance of our model using varying training set sizes, using GRID (33 speakers, seen speakers)

% of Training Set	PESQ	STOI	MCD	WER
20 %	1.96	0.583	29.22	11.78 %
40 %	2.00	0.594	28.49	10.10 %
60 %	2.02	0.595	27.94	9.06 %
80 %	2.02	0.596	27.68	8.36 %
100 %	2.02	0.601	27.78	8.03 %

4.7 Results on Unseen Speakers

In this section, we investigate the performance of the proposed approach on unseen speakers. For this study, we use all speakers from the GRID dataset, using a 50–20–30 % split ratio similar to Vougioukas *et al.* (2019a) and Michelsanti *et al.* (2020a), such that there is no overlap between the speakers featured in the training, validation, and test sets. To measure WER, we use the model trained on GRID mentioned in the previous section. We also select the checkpoint with the lowest WER on the validation set.

4.7.1 Ablation Study

In this study, we use GRID (33 speakers, unseen speakers). The results of the ablation study are shown in Table 4.6. For this task, we find that L_{power} provides the greatest impact on the quality of results, providing a substantial improvement in all metrics. On the other hand, L_{PASE} and L_{MFCC} show noticeable improvements in PESQ and STOI, indicating that these losses contribute to the clarity and intelligibility of the generated samples. Furthermore, we once more find that L_{MFCC} and L_{power} are particularly important towards achieving a low MCD, meaning that these losses are essential for achieving accurate MFCCs in our synthesized samples.

Regarding the adversarial loss, we can see that, as reported in the seen speaker ablation, PESQ, STOI, and MCD improve with the addition of the waveform and power critics. This suggests

Table 4.6: Ablation study for our model performed on GRID (33 speakers, unseen speakers).

Model	PESQ	STOI	MCD	WER
w/o L_{PASE}	1.44	0.520	38.19	22.66 %
w/o L_{power}	1.37	0.503	39.59	24.32 %
w/o L_{MFCC}	1.44	0.518	39.03	21.70 %
w/o Waveform Critic, w/o Power Critic	1.43	0.516	38.48	22.82 %
Full Model	1.47	0.523	37.91	23.13 %

that these critics have a positive effect on the clarity and intelligibility of samples and that the accuracy on the frequency domain is improved as well. However, we observe that the WER remains at a similar value with the removal of both critics, indicating that the network is generally capable of reproducing the correct words from the corresponding video samples while relying only on the three proposed L1 losses.

4.7.2 Comparison with Other Works

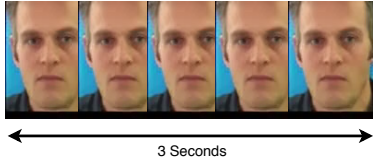
We present our comparison with other works (Vougioukas *et al.*, 2019a; Michelsanti *et al.*, 2020a) on GRID (33 speakers, unseen speakers) in Table 4.7. It is clear that our model outperforms previous works in all performance measures. Although the improvement in PESQ and STOI compared to these works is not as emphatic as the gains reported for seen speakers, WER sees a very substantial reduction. This improvement in WER can easily be observed in our synthesized speech and clearly shows that our model is far more consistent for this task than previous approaches. Furthermore, the observed MCD is substantially lower in our work, indicating that our synthesized speech yields more accurate spectrograms, which suggests a greater similarity between the content of real and synthesized samples.

4.7.3 Additional Experiments

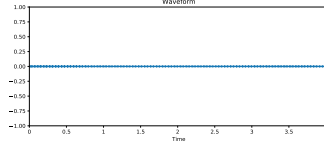
Additionally, we present a study on silent speakers. For this experiment, we artificially produce a video of a speaker from the GRID corpus being silent for five seconds by feeding Brownian

Table 4.7: Comparison between our current and previous model, using GRID (33 speakers, unseen speakers)

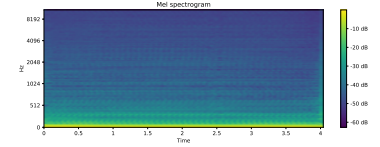
Method	PESQ	STOI	MCD	WER
GAN-based (Vougioukas <i>et al.</i> , 2019a)	1.24	0.470	51.28	37.10 %
Vocoder-based (Michelsanti <i>et al.</i> , 2020a)	1.23	0.477	55.02	55.23 %
Ours	1.47	0.523	37.91	23.13 %



(a) Silent Video



(b) Waveform



(c) Mel-frequency spectrogram

Figure 4.7: The spectrogram and waveform for the audio produced by our model for a video of a silent speaker (Speaker 2 from GRID) are portrayed in (a). As displayed in the waveform (b), the audio is almost completely silent, disregarding some low-frequency noise which is highlighted in the spectrogram (c). This shows that our model is robust to the scenario of silent speakers and produces minimal baseline noise under these circumstances. This audio sample is also available on our website².

noise into the facial animation model proposed by Vougioukas *et al.* (2020). We then use this video as input for our model trained on GRID (33 speakers, unseen speakers). This aims to measure two distinct factors: firstly, our model’s ability to recognize a silent speaker and not produce any voiced speech; and secondly, the baseline noise that is present in the audio we synthesize with our network, which is clear to observe when the speaker is silent. As discussed in Figure 4.7, our model performs well in this scenario and produces minimal noise for this silent example.

4.8 Results in the Wild

In this section, we investigate the performance of the proposed approach on utterances recorded ‘in the wild’. For this purpose, we use the full LRW dataset, and its subsets FLRW 500 Words, FLRW 100 Words, and FLRW 20 Words, which are introduced in Section 4.4. We split the utterances using the default split for LRW (90–5–5% ratio), such that there is no overlap between the utterances in the training, validation, and test sets. To measure the Word Error

Table 4.8: Comparison between our model and Lip2Wav, using the full LRW dataset.

Method	PESQ	STOI	MCD	WER
Lip2Wav (Prajwal <i>et al.</i> , 2020a)	1.20	0.543	-	34.20^a %
Ours	1.45	0.556	39.32	42.51 %

^aReported using Google STT API.

Rate (WER) for our samples, we use a pre-trained model (based on the work of Petridis *et al.* (2018a)) which was trained and tested on full LRW using the same split and achieve a baseline WER of 1.68% on the test set. As in previous experiments, we select the checkpoint with the lowest WER on the validation set.

Our comparison with Lip2Wav (Prajwal *et al.*, 2020a) on LRW (500 Words) is presented in Table 4.8. We compare our model to Lip2Wav on LRW (500 Words), in order to compare our model’s performance “in the wild” to this recent work. Our work shows a great improvement in PESQ compared to Lip2Wav, which suggests that our samples are able to achieve superior clarity in this regard. On the other hand, our STOI is very similar to the one reported in Lip2Wav, achieving a slight edge which could indicate a minor improvement in intelligibility.

In order to demonstrate our model’s ability to reconstruct speech in less constrained conditions, we experiment with the LRW dataset, as well as some of its subsets. These subsets present increasing degrees of challenge, culminating with the full LRW dataset which presents the greatest challenge given its large vocabulary and large variance in video perspective.

Regarding the experiments with frontal LRW, we observe that our model maintains a similar overall quality of outputs for larger vocabularies, as demonstrated by the consistency in PESQ, STOI, and MCD. However, it is clear that the more difficult task presented by larger vocabularies yields a decrease in the average accuracy of samples, shown by the increasing WER. This implies that our model scales well with larger datasets, but has difficulties in adapting to larger vocabularies in very unconstrained and inconsistent environments. Even still, the word error rate reported for FLRW 20 Words is noticeably low, implying that our model can realistically reconstruct speech for hundreds of different speakers, even under such ‘wild’ conditions. Fi-

Table 4.9: Study on the performance of our speech reconstruction model for the three subsets of LRW mentioned in Section 4.4, as well as the full LRW dataset.

Corpus	PESQ	STOI	MCD	WER
FLRW 20 Words	1.43	0.523	43.87	25.00 %
FLRW 100 Words	1.40	0.528	41.56	36.54 %
FLRW 500 Words	1.44	0.555	39.72	44.28 %
LRW 500 Words	1.45	0.556	39.32	42.51 %

nally, we found that the full LRW dataset yields better performance than our full frontal subset (FLRW 500 Words). Although we expected the frontal data to provide an easier task for the network during training and testing, this result shows that the network benefits strongly from a larger training set, even if the visual data is less consistent.

4.9 Conclusion

In this chapter, we have presented our end-to-end video-to-waveform synthesis model using a generative adversarial network with two critics on waveform and spectrogram. First, we showed through an ablation study on GRID that the use of our losses, adversarial critics, and other choices in training methodology provides a positive impact on the quality of our results for both seen and unseen speaker video-to-speech. Furthermore, we demonstrated through our experiments on LRW that our model is able to generate intelligible speech for videos recorded entirely in the wild by hundreds of different speakers. Finally, we compared our model to previous video-to-speech models and found that it produces the best results on most metrics for GRID and LRW and achieves state-of-the-art performance on PESQ for TCD-TIMIT.

We observed that the choice of good critics as well as adequate comparative losses are fundamental to obtaining realistic results. Therefore, we believe that the pursuit of alternative loss functions (including different adversarial losses) is a promising option for future work. Additionally, we believe that there would be substantial benefit in experimenting with a speaker embedding as input to the generator, in addition to the video, in order to generalize to unseen

speakers with a more accurate voice profile, as proposed by Shen *et al.* (2018) and Prajwal *et al.* (2020a). We will explore this methodological addition in Chapter 5. Finally, extending our model towards other practical applications such as speech inpainting i. e., reconstructing missing audio segments in an audio-visual stream, would be a promising research pursuit in order to show the empirical value of video-to-speech synthesis.

CHAPTER 5

SYNTHESIZING SPEECH FROM SILENT VIDEO USING NEURAL VOCODERS

Contents

5.1	Introduction	99
5.2	Methodology	100
5.2.1	Video-to-spectrogram model	100
5.2.2	Spectrogram-to-waveform	102
5.3	Experimental setup	103
5.3.1	Datasets	103
5.3.2	Data pre-processing and augmentation	104
5.3.3	Training details	104
5.3.4	Evaluation metrics	105
5.4	Results	105
5.4.1	Experiments	105

5.4.2 Ablations	106
---------------------------	-----

5.5 Conclusion	108
---------------------------------	------------

This chapter is based on *SVTS: Scalable Video-to-Speech Synthesis*, a journal paper published in *Interspeech 2022*. I was the first author of this paper, designed the framework, conducted the experimental procedure, and wrote the manuscript. Alexandros Haliassos contributed to our implementation substantially to enable efficient training procedures, helped conduct many of the experiments, and provided useful feedback on the manuscript. Stavros Petridis, Björn Schuller, and Maja Pantic supervised the project throughout its entirety, providing helpful suggestions, feedback, and guidance regarding the design of the framework and the writing of the manuscript.

As in previous chapters, we would like to note that the related work and comparison with other works described in this chapter only include papers that were released before the time of submission (March 2022). Since then, several new video-to-speech approaches have been proposed, some of which outperform SVTS on LRS3 (Kim *et al.*, 2023; Hsu *et al.*, 2022), for instance.

5.1 Introduction

As explained in the previous chapter, video-to-speech synthesis is a promising task with many exciting applications such as audio recovery from existing videos, live speech reconstruction in videoconferencing, and even helping people suffering from aphonia. Furthermore, when designed without the need for intermediate text transcriptions, video-to-speech models such as the one presented in the previous chapter can be trained on unlabelled audio-visual speech, meaning that, in theory, they could be trained directly on large-scale datasets such as AVspeech Ephrat *et al.*, 2018. Remarkably, however, most recent works focus on corpora with small pools of speakers, constrained vocabularies, and video recorded in studio conditions (e.g., 4-Speaker GRID and 3-Lipspeaker TCD-TIMIT (Harte and Gillen, 2015)) (Akbari *et al.*, 2018; Vougioukas *et al.*, 2019b; Yadav *et al.*, 2021; Michelsanti *et al.*, 2020b; Oneata *et al.*, 2021; Um *et*

al., 2021; Hong *et al.*, 2021), achieving improvements in performance via the use of intricate loss ensembles (Um *et al.*, 2021; Kim *et al.*, 2021) and complex architectures (Yadav *et al.*, 2021; Um *et al.*, 2021; Hong *et al.*, 2021; Prajwal *et al.*, 2020b) (e.g., Chapter 4). While these developments are meaningful within ideal conditions, they fail to leverage the massive amount of audio-visual data available publicly and propose training procedures that do not easily scale to very large datasets (Kim *et al.*, 2021).

In this chapter, we aim to address these issues by proposing a simple video-to-speech system that efficiently scales with more data. It consists of a video-to-spectrogram predictor followed by a spectrogram-to-waveform synthesizer. The former is a ResNet18+conformer network (Gulati *et al.*, 2020; He *et al.*, 2016), which becomes deeper and wider for larger datasets and is trained using a combination of two established comparative losses. The latter is a pre-trained neural vocoder, which accurately synthesizes the corresponding audio waveform with low computational overhead. Our contributions are as follows: **(1)** We present a simple and effective video-to-speech approach that can easily scale to large and complex datasets. **(2)** We conduct a detailed ablation study demonstrating the differences between commonly-used spectrogram inversion methods, as well as validating our choice of loss functions. **(3)** We outperform previous approaches on most metrics on the small but popular GRID dataset and achieve state-of-the-art performance on the larger LRW dataset. **(4)** To the best of our knowledge, we are the first to present intelligible results on the challenging LRS3 (Afouras *et al.*, 2018b) dataset and show that scaling our model even further with a combination of LRS3 and VoxCeleb2 (Chung *et al.*, 2018a) (containing more than 1,500 hours of data) yields significant improvements.

5.2 Methodology

5.2.1 Video-to-spectrogram model

Our spectrogram predictor comprises two main components: (1) a visual encoder composed of a 3D convolutional stem followed by a standard 2D ResNet-18 (He *et al.*, 2016), as in the

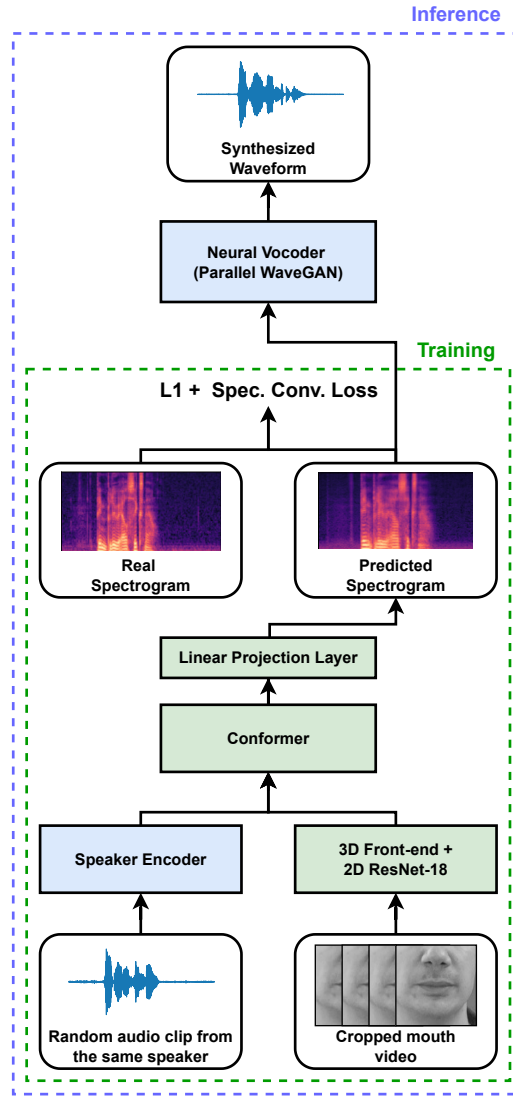


Figure 5.1: Summary of our video-to-speech synthesis approach during training and inference. In this figure, the components pictured in blue are pre-trained and kept frozen, while the components pictured in green are trained from scratch.

previous chapters, and (2) a conformer (Gulati *et al.*, 2020), as in Chapter 3, which receives the features from the visual encoder and aims to model the temporal correlations between them. The latter contains an initial linear layer, followed by a set of conformer blocks that vary in depth and width based on the model version, as shown in Table 5.1. Finally, each feature vector, corresponding to a video frame, is projected into a hidden size of 320 using a linear projection layer and reshaped into 4×80 spectrogram frames, similarly to our projector in Chapter 3. The input video is sampled at 20 fps and the extracted spectrogram contains 80 frames per second. We train our predictor using a combination of the L_1 loss and the spectral convergence loss (Yamamoto *et al.*, 2020).

Table 5.1: Summary of our proposed SVTS architectures. *refers to the total number of parameters in the model (ResNet + conformer + projection layer)

Model	SVTS-S	SVTS-M	SVTS-L
Num. parameters* (M)	27.3	43.1	87.6
Conformer blocks	6	12	12
Attention dim.	256	256	512
Attention heads	4	4	8
Conv. kernel size	31	31	31
Feedforward dim.	2048	2048	2048

As in multi-speaker text-to-speech systems (Jia *et al.*, 2018), our video-to-speech model requires information about the speaker’s voice characteristics, which cannot be derived accurately from silent video only. To this end, we use a pre-trained speaker encoder¹ originally trained for speaker verification on a combination of VoxCeleb (Nagrani *et al.*, 2017), VoxCeleb2 (Chung *et al.*, 2018a), and Librispeech (Panayotov *et al.*, 2015b). For each video clip, an embedding is extracted from a randomly selected audio clip from the same speaker and concatenated with the visual features extracted by the visual encoder, which are then fed into the conformer. Note that the speaker encoder is kept frozen during training.

5.2.2 Spectrogram-to-waveform

To generate waveform speech from the spectrograms, we opt for the use of a neural vocoder, Parallel WaveGAN (Yamamoto *et al.*, 2020). This WaveNet-based (Oord *et al.*, 2016) model is trained using a combination of comparative and adversarial losses. We employ a version pre-trained on LibriTTS (Zen *et al.*, 2019) for 1 million iterations. Note that it is used only at inference time, allowing for a substantially simpler training procedure than related video-to-speech works, which train their own vocoder from scratch (Um *et al.*, 2021; Hong *et al.*, 2021). This is, however, a trade-off between simplicity and performance, since a vocoder trained on the same corpus (rather than a separate set of datasets) will typically yield better results, as we will explore in Chapter 6. An overview of our approach is illustrated in Figure 5.1.

¹<https://github.com/CorentinJ/Real-Time-Voice-Cloning>.

5.3 Experimental setup

5.3.1 Datasets

The first corpus we experiment with is GRID, which has become an established benchmark in video-to-speech literature due to its small vocabulary, predictable structure, and clean recording conditions. We experiment with two versions of the dataset: (1) a seen speaker version, proposed in Chapter 4, where the 33 speakers are present in the training, validation, and testing sets, and (2) an unseen speaker version, introduced in (Vougioukas *et al.*, 2019b), where there is no overlap in the speakers between the sets.

The second corpus is the full LRW dataset, as in Chapter 4. Due to its lack of speaker labels, it is not possible to select a random audio clip from the same speaker to produce the corresponding speaker embedding. Therefore, for this corpus, we generate the speaker embeddings using the audio clip from the corresponding video, which is consistent with previous multi-speaker video-to-speech approaches on LRW (Prajwal *et al.*, 2020b).

To demonstrate our method’s scalability to even larger and less constrained datasets, we run experiments on the 312-hour-long LRS3 dataset, which was not featured in the experiments for our previous model (Chapter 4). As in GRID, we use two different versions of LRS3: seen speaker, where all speakers’ utterances are split into training, validation, and testing sets using an 80–10–10 % ratio, and unseen speaker, following the original split proposed by Afouras *et al.* (2018b). Finally, we experiment with combining the LRS3 training dataset with an English-only version (Shi *et al.*, 2022a) of VoxCeleb2 (while keeping the same LRS3 validation and test sets to ease comparison), amounting to around 1,550 hours of footage. An extended version of this combined training set is also used for the pre-training stage in Chapter 7. For both corpora, utterances exceeding 24 seconds are excluded from training due to hardware limitations. The contents of these datasets are explored in detail in Chapter 2

5.3.2 Data pre-processing and augmentation

To produce the cropped mouth video, we first extract 68-point landmarks using RetinaFace² (Deng *et al.*, 2020) and a pre-trained 2D-FAN³ (Bulat and Tzimiropoulos, 2017b). We average the landmarks across 12 frames through a sliding window to reduce motion jitter and align each frame to the mean face. We then crop a 96×96 region centered around the mouth and convert the frames to grayscale. The audio is sampled at 24 kHz, and the log-mel spectrograms are extracted using 80 mel bands, frequency bins of size 2048, a hop size of 12.5 ms, a window length of 50 ms, and a Hann window.

During training, we apply random cropping and horizontal flipping as in previous Chapters and also add random erasing with a probability of 0.5. The erased area is randomly sampled between 2 and 33 % of the full frame, with an aspect ratio ranging from 0.3 to 3.3. For our LRS3 experiments, we apply time-masking by randomly replacing each frame with the average pixel value in the video, since we find it aids generalization when training on long sentences. We apply one contiguous time mask for each second of the utterance, and each mask’s length is uniformly sampled from 0 to 0.4 seconds.

5.3.3 Training details

For our GRID and LRW experiments, we train our models using AdamW (Loshchilov and Hutter, 2017a) with a learning rate of 1×10^{-3} , $\beta_1 = 0.9$, $\beta_2 = 0.98$, and a weight decay of 1×10^{-2} . We warm up the learning rate for the first 10 % of iterations, and then decay it with a cosine schedule (Loshchilov and Hutter, 2016). For LRS3 seen speakers, we use a maximum learning rate of 7×10^{-3} , while for unseen speakers (including the combination with VoxCeleb2) we use 1×10^{-3} . We train for a total of 200, 150, 500, and 150 epochs for GRID, LRW, LRS3 seen speakers, and LRS3 unseen speakers, respectively. We save a checkpoint at the end of each epoch and at the end of training select the one with the lowest validation loss.

²https://github.com/biubug6/Pytorch_Retinaface

³<https://github.com/1adrianb/face-alignment>

5.3.4 Evaluation metrics

We use a set of four objective metrics to evaluate the quality of the generated speech in our experiments. Compared to Chapter 4, PESQ, STOI, and WER are kept, but MCD is replaced by ESTOI, as we found that measuring intelligibility was particularly important in this work. Although these metrics are commonly referenced in video-to-speech works and are therefore useful for comparison, it is widely known that no objective speech metric correlates perfectly with human perception of quality and intelligibility (Vougioukas *et al.*, 2019b), as we claim in Chapter 4. Therefore, we highly encourage readers to listen to the generated samples available on our project website⁴ rather than rely solely on the reported metrics.

5.4 Results

5.4.1 Experiments

Our results are presented in Table 5.2. We begin by discussing our findings on the small-scale GRID dataset. For the seen speaker split, our SVTS-S model clearly outperforms our previous approach presented in Chapter 4, as well as the recent model by Kim *et al.* (2021), on STOI and ESTOI. It also achieves a significant improvement in WER. These metrics indicate that our samples are more intelligible than previous works. On unseen speakers, our model achieves a better PESQ, STOI, and ESTOI but is outperformed by our previous GAN-based work (presented in Chapter 4) in WER. By perceptually evaluating the generated samples, we find that our seen speaker reconstruction is highly realistic and could be mistaken for real audio. On the other hand, our unseen speaker samples sound considerably less noisy than previous works and capture the unseen speaker’s voice with remarkable accuracy, thanks to our speaker embedding strategy.

On the more challenging and diverse LRW dataset, SVTS-M is superior to previous approaches on all metrics by a wide margin. We achieve a low WER of 13.4 %, indicating that our samples

⁴<https://sites.google.com/view/scalable-vts>

are consistently intelligible. Perceptually, we find that our samples sound substantially more realistic and accurate than previous approaches, including our GAN-based approach presented in Chapter 4. This strong performance is a consequence of our SVTS architecture, which allows us to efficiently scale to this larger dataset.

Finally, we experiment with LRS3, which is undoubtedly the most challenging corpus we approach, as discussed in Section 5.3.1. On the seen speaker setting, we find that our model achieves reasonable PESQ, STOI, and ESTOI performance, comparable to what had been reported by previous works on LRW. The unseen speaker protocol is naturally more challenging and therefore does not achieve the same level of quality. Interestingly, we find that results are greatly improved with the addition of the VoxCeleb2 data, as shown by the significant boost in all metrics. This empirically demonstrates our model’s ability to improve its reconstructions by leveraging additional training data, even if its distribution is different from the testing set (which only contains samples from LRS3). It also suggests that we may have not yet reached a saturation point: There are likely still gains to be made in the future with even more data.

Perceptually, we find that the most intelligible samples are produced by our seen speaker model, closely followed by our model trained on LRS3+VoxCeleb2. Although there is room for improvement, we find that most syllables in the reconstructed speech are discernible, and each speaker’s voice profile is reproduced with considerable accuracy, which is particularly impressive in the unseen speaker scenario.

5.4.2 Ablations

To motivate our use of Parallel WaveGAN (PWG) as our waveform synthesis model, we compare it in Table 5.3 with other recently proposed neural vocoders as well as the commonly used Griffin-Lim algorithm. All models, including our version of PWG, are pre-trained on LibriTTS and are publicly available⁵. The Griffin-Lim synthesis is performed using the fast version of the algorithm⁶ (Perraudin *et al.*, 2013) and runs for 30 iterations. It can be observed that

⁵<https://github.com/kan-bayashi/ParallelWaveGAN>

⁶<https://librosa.org/doc/main/generated/librosa.griffinlim.html>

Table 5.2: Summary of SVTS’s results compared to previous works on GRID, LRW, and LRS3. Due to LRS3’s complex vocabulary and long sentence structure, we are unable to find a speech recognition model that works accurately on our generated samples (e.g., the word “teacher” is often mistaken for “teachers”), and therefore do not report WER for this dataset. *reported using Google speech-to-text API.

Method	Corpus	Speaker split (seen/unseen)	Training data (hours)	PESQ	STOI	ESTOI	WER (%)
End-to-end GAN (Mira <i>et al.</i> , 2022a) (Chapter 4)	GRID	seen	24	1.70	0.667	0.466	4.60
VCA-GAN + Griffin-Lim (Kim <i>et al.</i> , 2021)	GRID	seen	20	1.97	0.695	0.505	5.13
SVTS-S	GRID	seen	24	1.97	0.705	0.523	2.36
End-to-end GAN (Vougioukas <i>et al.</i> , 2019b)	GRID	unseen	13	1.26	0.494	0.198	32.79
Conv.+GRU+WORLD vocoder (Michelsanti <i>et al.</i> , 2020b)	GRID	unseen	13	1.26	0.541	0.227	38.15
End-to-end GAN (Mira <i>et al.</i> , 2022a) (Chapter 4)	GRID	unseen	13	1.37	0.568	0.289	16.12
VCA-GAN+Griffin-Lim (Kim <i>et al.</i> , 2021)	GRID	unseen	13	1.39	0.570	0.282	24.57
Conv.+LSTM+WaveNet (Hong <i>et al.</i> , 2021)	GRID	unseen	13	1.33	0.531	0.271	26.17
SVTS-S	GRID	unseen	13	1.40	0.588	0.318	17.85
Conv.+LSTM+Griffin-Lim (Prajwal <i>et al.</i> , 2020b)	LRW	unseen	157	1.20	0.543	0.344	34.20*
End-to-end GAN (Mira <i>et al.</i> , 2022a) (Chapter 4)	LRW	unseen	157	1.33	0.552	0.330	42.60
VCA-GAN+Griffin-Lim (Kim <i>et al.</i> , 2021)	LRW	unseen	157	1.34	0.565	0.364	37.07
SVTS-M	LRW	unseen	157	1.49	0.649	0.483	13.40
SVTS-L	LRS3	seen	256	1.30	0.553	0.331	-
SVTS-L	LRS3	unseen	296	1.25	0.507	0.271	-
SVTS-L	LRS3 + VoxCeleb2	unseen	1556	1.26	0.530	0.313	-

Table 5.3: Vocoder ablation for SVTS on GRID (seen speakers). Speed is measured on an Nvidia RTX 2080 Ti. *computed on CPU

Metric	PESQ	STOI	ESTOI	WER (%)	Speed (clips/sec.)
Griffin-Lim* (Griffin and Jae Lim, 1984)	2.00	0.696	0.513	2.41	1.2
Multiband MelGAN (Yang <i>et al.</i> , 2021)	1.86	0.683	0.487	2.50	184.9
Style MelGAN (Mustafa <i>et al.</i> , 2021)	1.93	0.702	0.520	2.38	83.7
Parallel WaveGAN (Yamamoto <i>et al.</i> , 2020)	1.97	0.705	0.523	2.36	54.7

Table 5.4: Loss ablation for SVTS on GRID (seen speakers).

Metric	PESQ	STOI	ESTOI	WER (%)
w/o Spec. Conv.	1.97	0.705	0.523	2.90
w/o L_1	1.91	0.700	0.514	2.74
L_1 +Spec. Conv.	1.97	0.705	0.523	2.36

Parallel WaveGAN outperforms its peers Multiband Melgan (Yang *et al.*, 2021) and Style Melgan (Mustafa *et al.*, 2021) on all four evaluation metrics. Furthermore, through perceptual evaluation, we find that PWG produces substantially more realistic audio. Regarding Griffin-Lim, although it achieves a slightly higher PESQ score, we find that it consistently produces noisy speech with frequent artifacts. This highlights the limitations of PESQ as a metric, as it is often not sensitive to artifacts that are immediately noticeable to human listeners. Thanks to efficient GPU implementations, the vocoders are roughly $50\times$ faster than Griffin-Lim, with the fastest vocoder, Multiband Melgan, being able to process almost 200 GRID clips per second.

In Table 5.4, we experiment with each of our loss functions separately and compare them with the combined loss (baseline). We find that the baseline’s performance is roughly similar to the individual losses on PESQ, STOI, and ESTOI, but is clearly superior on WER. Interestingly, we find that our model achieves comparable performance with only an L_1 loss, which contrasts greatly with previous approaches’ (including Chapter 4) reliance on elaborate loss combinations (Um *et al.*, 2021).

5.5 Conclusion

In this chapter, we propose SVTS, a scalable approach for video-to-speech synthesis. We present three architectures of varying sizes, which allow us to efficiently adapt our training procedure to datasets ranging from GRID (27 hours) to LRS3+VoxCeleb2 ($> 1,500$ hours). We show that our method outperforms previous approaches (including our end-to-end approach presented in the previous Chapter) on most metrics for two popular versions of GRID, and establishes a new state-of-the-art for LRW. Finally, we experiment with the large and unconstrained LRS3 corpus, achieving intelligible results, and combine it with VoxCeleb2 to further improve our performance, demonstrating our method’s scalability. We hope our work will encourage a shift towards larger corpora, as this aligns with the current ubiquity of unlabelled audio-visual data.

CHAPTER 6

AUDIO-VISUAL SPEECH ENHANCEMENT

Contents

6.1	Introduction	110
6.2	Methodology	113
6.2.1	Audio-visual spectral enhancement	113
6.2.2	Waveform synthesis	114
6.3	Experimental Setup	115
6.3.1	Datasets	115
6.3.2	Pre-processing, augmentation, and evaluation	116
6.3.3	Comparison models	117
6.3.4	Training details	117
6.4	Results	118
6.4.1	Comparison with other works	118
6.4.2	Noise and interference study	119
6.4.3	Spectrogram inversion comparison	121

6.5 Conclusion 122

This chapter is based on *LA-VocE: Low-SNR Audio-visual Speech Enhancement using Neural Vocoders*, a conference paper published in *ICASSP 2023*. I was the first author of this paper, designed the framework, conducted the experimental procedure, and wrote the manuscript. Buye Xu made substantial contributions to the design of the framework, the implementation of our code base, and our experimental procedure, and provided helpful feedback on the manuscript. Jacob Donley helped establish our data processing pipeline, made substantial contributions to our evaluation procedure (namely through the proposal of new speech quality/intelligibility metrics), and provided extensive feedback and guidance during the writing of the manuscript. Anurag Kumar actively contributed to discussions surrounding data pre-processing, model design, the use of baselines for comparison, and the use of evaluation metrics in our results. Stavros Petridis, Vamsi Krishna Ithapu, and Maja Pantic supervised the project throughout its entirety, providing helpful suggestions, feedback, and guidance regarding the design of the framework and the writing of the manuscript.

As in previous chapters, the related work and comparison with other works presented below only include papers released before the time of submission (September 2022). Since then, ReVISE (Hsu *et al.*, 2022), for instance, has notably achieved impressive audio-visual speech enhancement results, although it does not compare directly with our work and is trained/evaluated on different datasets, under different conditions.

6.1 Introduction

Speech enhancement, defined as the extraction of clean speech from a noisy signal, is a well-established signal processing task that has benefited greatly from the advent of deep learning (Wang and Chen, 2018). Recently-proposed models excel at denoising and dereverberation (Tan and Wang, 2020; Défossez *et al.*, 2020), but often struggle with very low signal-to-noise ratios (SNR) (Hao *et al.*, 2019). Furthermore, audio-only methods struggle to accurately remove background speech, as they are limited in the information they can use to distinguish it

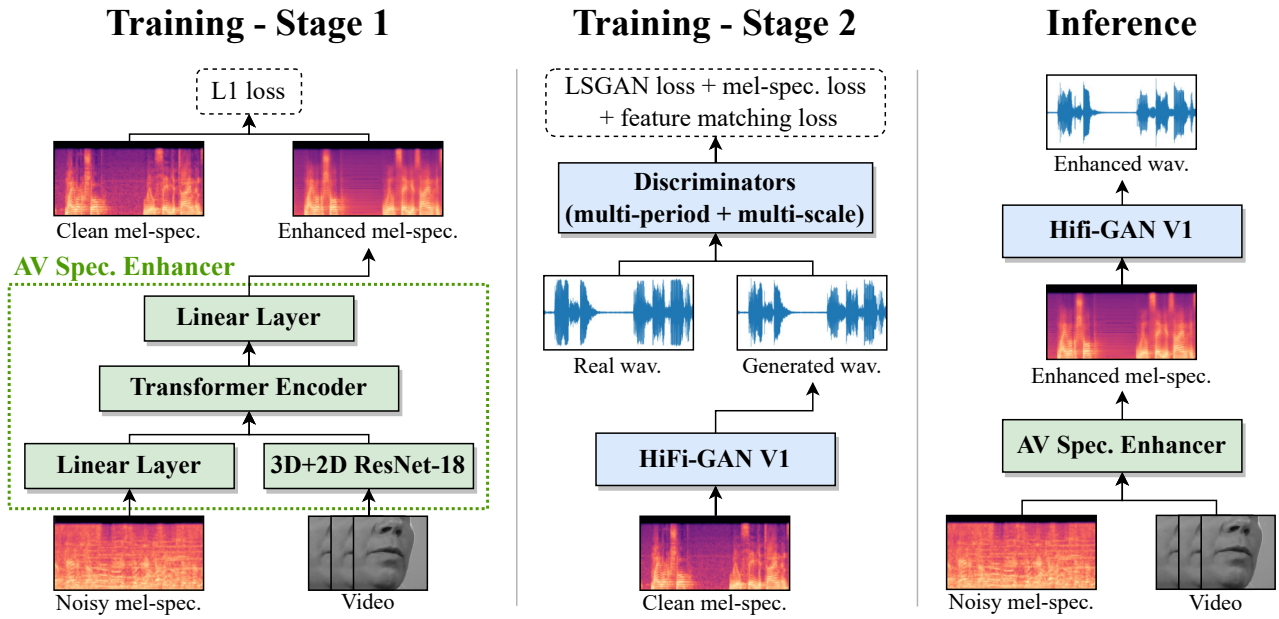


Figure 6.1: Summary of LA-VocE’s two-stage training approach and inference procedure.

from the target signal. These limitations have drawn researchers to leverage visual cues of the target speaker’s lip movements as additional supervision – an approach known as audio-visual speech enhancement (AVSE), which was discussed briefly in Chapter 1. This can be particularly valuable for applications such as video conferencing, streaming, recording, and hearing augmentation in a crowded and / or noisy environment, where the target speaker’s video stream can help the model enhance their speech. This method may also be leveraged to improve speech recognition in low-SNR conditions. Furthermore, the recent success of video-to-speech synthesis, as portrayed in Chapters 4 and 5, where the audio is reproduced using only silent video, highlights the importance of the visual modality and shows a promising direction for audio-visual speech enhancement in very low-SNR conditions.

Recent AVSE methods are often based on U-Nets (Gao and Grauman, 2021; Gabbay *et al.*, 2018; Pan *et al.*, 2021; Xu *et al.*, 2022), inspired by their audio-only counterparts (Tan and Wang, 2020; Défossez *et al.*, 2020; Luo and Mesgarani, 2019), or simple convolutional networks (Hou *et al.*, 2018), frequently combined with LSTMs (Yang *et al.*, 2022; Afouras *et al.*, 2018e). Existing speech enhancement models are typically combined with a video encoder which extracts visual features and concatenates them with acoustic features to perform audio-visual enhancement. These approaches draw heavily from speech enhancement literature, but do not

take advantage of state-of-the-art transformer-based audio-visual encoders (Ma *et al.*, 2021d; Shi *et al.*, 2022b). Most methods estimate (either directly or via a mask) the magnitude and phase of the clean spectrogram, which are converted into waveform using the inverse Short-Time Fourier Transform (iSTFT) (Afouras *et al.*, 2018e; Gao and Grauman, 2021; Gabbay *et al.*, 2018; Hou *et al.*, 2018), while others attempt to perform enhancement in the time domain directly (Pan *et al.*, 2021; Xu *et al.*, 2022). Both of these reconstruction techniques rely on very accurate predictions, which can be difficult to achieve, especially in low-SNR environments where audio supervision is unreliable. Yang *et al.* (2022) introduce neural vocoders as an alternative synthesis method but choose to focus on constrained studio-recorded scenarios.

To address these shortcomings, we propose a new two-stage approach for audio-visual speech enhancement entitled **Low-SNR Audio-visual Vocoder-based Speech Enhancement** (LA-VocE). First, we train an audio-visual spectrogram enhancer, which receives noisy speech and video of the cropped mouth and aims to predict a clean spectrogram. This model features a ResNet-18-based visual encoder (He *et al.*, 2016), as in previous chapters, and a large transformer encoder (Vaswani *et al.*, 2017) to model the temporal correlations in the audio-visual features and is trained using an L1 loss between the real and predicted mel-spectrogram magnitudes. We then train a neural vocoder (HiFi-GAN V1 (Kong *et al.*, 2020b)) to predict waveform audio from clean mel-spectrograms on the same corpus. This fully convolutional model is trained using a mixture of adversarial and comparative losses, with an ensemble of eight discriminators operating on multiple periods and scales. During inference, the enhancer and the vocoder are combined to perform end-to-end audio-visual speech enhancement.

Our contributions are as follows: **(1)** We present a new audio-visual speech enhancement approach based on a large transformer-based spectrogram enhancer, combined with our version of HiFi-GAN V1 trained on the same corpus. **(2)** We train our model to remove background noise and speech on the challenging AVSpeech dataset, which features thousands of speakers and multiple languages. **(3)** We compare our approach with previous state-of-the-art audio-visual and audio-only speech enhancement methods, and show that it significantly outperforms all methods across all metrics and noise conditions. **(4)** We study our model’s ability to generate clean audio for varying levels of noise and interference and find that it consistently achieves

improvements in speech intelligibility. **(5)** We measure our trained vocoder’s effectiveness against other spectrogram inversion approaches and observe that it significantly outperforms existing pre-trained models, as well as the Griffin-Lim algorithm (Griffin and Jae Lim, 1984).

6.2 Methodology

6.2.1 Audio-visual spectral enhancement

LA-VocE is summarized in Fig. 6.1. The first stage in our framework consists of training an audio-visual spectrogram enhancer, which is inspired by the video-to-spectrogram model in Chapter 5. This model extracts visual features using a 2D ResNet-18 (He *et al.*, 2016) with a 3D convolutional stem (as in Chapters 3, 4, and 5), and acoustic features using a single linear layer. The video features are then upsampled (via nearest neighbors interpolation) to match the audio features’ frame rate, and the features from the two modalities are concatenated along the channel dimension. The fused audio-visual features are fed into the transformer encoder (Vaswani *et al.*, 2017) - the largest component in the network. This module comprises an initial embedding layer, with a linear layer followed by relative positional encoding (Dai *et al.*, 2019), and 12 transformer encoder blocks, where the attention dimension, feedforward dimension, and the number of attention heads are 768, 2048 and 8, respectively. Finally, these features are decoded via a linear layer into the predicted mel-frequency spectrogram. We train the model via an L1 Loss:

$$\mathcal{L}_1 = ||s_{clean} - E(s_{noisy}, v)||_1, \quad (6.1)$$

where s_{clean} and s_{noisy} are the clean and noisy mel-spectrograms, respectively, v is the video of the speaker’s lip movements and E is our audio-visual spectrogram enhancer. We remove the spectral convergence loss from Chapter 5, as we find that it does not noticeably improve overall performance.

6.2.2 Waveform synthesis

The second stage in our method involves training a neural vocoder to convert the enhanced spectrograms into waveform audio, rather than using a pre-trained vocoder as we did in Chapter 5. We use HiFi-GAN (Kong *et al.*, 2020b), which upsamples the spectrogram gradually using a set of transposed convolutions. In particular, we opt for HiFi-GAN V1, the largest HiFi-GAN variant, which features 12 ResBlocks with hidden size 512, amounting to 13.92 million parameters. As proposed by Kong *et al.* (2020b), the model is trained via a multi-period discriminator (MPD), composed of five convolutional sub-discriminators which analyze the waveform along different periods (i. e., every 2, 3, 5, 7 and 11 samples), and a multi-scale discriminator (MSD), consisting of one sub-discriminator for the raw audio and two sub-discriminators that receive downsampled versions of the same waveform (via $2\times$ and $4\times$ average pooling). Our training objective (as in the original HiFi-GAN) combines the Least Squares Generative Adversarial Network (LSGAN) loss (Mao *et al.*, 2017) with an L1 loss on the mel-spectrogram magnitudes and a feature matching loss (Larsen *et al.*, 2016):

$$\mathcal{L}_G = \alpha_1 \mathcal{L}_{G_{adv}} + \alpha_2 \mathcal{L}_{spec} + \alpha_3 \mathcal{L}_{FM}, \quad (6.2)$$

$$\mathcal{L}_{G_{adv}} = \sum_{i=1}^{N_D} (D_i(G(s_{clean})) - 1)^2, \quad (6.3)$$

$$\mathcal{L}_{spec} = \|m(w_{clean}) - m(G(s_{clean}))\|_1, \quad (6.4)$$

$$\mathcal{L}_{FM} = \sum_{i=1}^{N_D} \sum_{l=1}^{N_{L_i}} \frac{\|D_i^l(w_{clean}) - D_i^l(G(s_{clean}))\|_1}{d_i^l}, \quad (6.5)$$

$$\mathcal{L}_D = \sum_{i=1}^{N_D} (D_i(w_{clean}) - 1)^2 + D_i(G(s_{clean}))^2, \quad (6.6)$$

where $\mathcal{L}_G$ is the generator loss, $\mathcal{L}_D$ is the discriminator loss, $\mathcal{L}_{G_{adv}}$ is the generator's adversarial loss, $\mathcal{L}_{spec}$ is the mel-spectrogram loss, $\mathcal{L}_{FM}$ is the feature matching loss, G is the generator

(HiFi-GAN V1), D_i is the i -th discriminator, N_D is the number of discriminators, w_{clean} is the clean waveform, m is the function that computes the mel-spectrogram, N_{L_i} is the number of layers in discriminator i , and D_i^l and d_i^l refer to the features extracted from layer l / discriminator i and their dimension, respectively. Loss coefficients α_1 , α_2 , and α_3 are set to 1, 45, and 2, respectively, as in the original HiFi-GAN (Kong *et al.*, 2020b). After both stages of training, the spectrogram enhancer and neural vocoder are combined during inference, as shown in Fig. 6.1.

6.3 Experimental Setup

6.3.1 Datasets

Our experiments focus on AVSpeech (Ephrat *et al.*, 2018), one of the largest publicly available audio-visual speech datasets. It contains roughly 4,700 hours of 3- to 10-second-long video clips obtained from 290,000 YouTube videos. These feature approximately 150,000 different subjects, speaking several languages with a virtually unconstrained vocabulary. The scale and heterogeneity of the data make for a substantially more challenging and realistic scenario than many commonly-used corpora such as GRID (Yang *et al.*, 2022; Cooke *et al.*, 2006b; Gabbay *et al.*, 2018) or Facestar (Yang *et al.*, 2022), which are recorded in studios.

For background noise, we sample clips from the Deep Noise Suppression challenge (Reddy *et al.*, 2020) noise dataset. It contains roughly 70,000 noise clips, amounting to around 150 classes, ranging from music to machine sounds. Both datasets are split into training, validation, and testing sets using an 80–10–10 % ratio. Due to the computational cost of computing the evaluation metrics, we randomly sample 1 % of the AVSpeech testing set, amounting to 1,552 samples, and use this as the evaluation set for our experiments. We add two types of corruption to the clean speech: background noise (denoted ‘noise’) and background speech (denoted ‘interference’). The corruption level is controlled by the Signal-to-Noise Ratio (SNR) and the Signal-to-Interference Ratio (SIR):

$$\text{SNR} = \frac{P_{\text{signal}}}{P_{\text{noise}}}, \quad \text{SIR} = \frac{P_{\text{signal}}}{P_{\text{interference}}}, \quad (6.7)$$

where P refers to the power of each waveform. The interfering speech is also obtained from AVSpeech. During training, the SNR and SIR are independently randomly sampled between 5 and -15 dB. The number of background noises and interfering speakers in each sample vary randomly from 1 to 5 and 1 to 3, respectively. During validation, we propose three different noise conditions to compare with other methods, ranging from least to most noisy. Noise conditions 1 (low), 2 (medium), and 3 (high) feature 1, 3, and 5 background noises at 0, -5, and -10 dB SNR, and 1, 2, and 3 background speakers at 0, -5, and -10 dB SIR, respectively.

6.3.2 Pre-processing, augmentation, and evaluation

The noisy and clean signals are normalized via peak normalization and are converted into log-scale mel-spectrograms using the following parameters: frequency bin size and Hann window size 1024, hop size 256, and 80 mel bands. The audio sampling rate is 16 kHz and the video frame rate is 25 frames per second (fps). To model the speaker’s lip movements, we extract the grayscale mouth Region Of Interest (ROI) from each video, following Chapter 5. To augment our data during training, we apply random cropping, random horizontal flipping, random erasing, and time-masking, as in Chapter 5. We evaluate our results using a set of five objective speech metrics. For more details on each metric, please refer to Chapter 2. Following other works (Pan *et al.*, 2021; Luo and Mesgarani, 2019), we denote improvements between noisy and enhanced speech metrics with the lowercase ‘i’, e.g., PESQ-WB i. In practice, this is computed as:

$$\text{Mi}(x, \tilde{x}, \hat{x}) = \text{M}(x, \hat{x}) - \text{M}(x, \tilde{x}) \quad (6.8)$$

where M can be any of the five metrics we propose, x is the clean speech (ground truth), $\tilde{x}$ is the noisy speech, before any enhancement, and $\hat{x}$ is the enhanced speech (the output of the

model).

6.3.3 Comparison models

We compare our results with two recent AVSE models: VisualVoice (Gao and Grauman, 2021), a complex spectral mapping approach originally proposed for speech separation that we adapt to perform enhancement, and Multi-modal Speaker Extraction (MuSE) (Pan *et al.*, 2021), a feature masking approach based on Conv-TasNet (Luo and Mesgarani, 2019). To provide a broader comparison with other reconstruction techniques, we also adapt two recent speech enhancement models for AVSE - Gated Convolutional Recurrent Network (GCRN) (Tan and Wang, 2020) and Demucs (Défossez *et al.*, 2020). We achieve this by adding a visual stream (3D front-end + ResNet-18, as in our model) which encodes the video into temporal features that are concatenated with the audio features from the original audio encoder (preceding the LSTM/GLSTM). We refer to these models as AV-GCRN and AV-Demucs. We also compare with the original audio-only GCRN to highlight the importance of the visual stream. All models are implemented based on official open-source code.

6.3.4 Training details

We train our spectrogram enhancer for 150 epochs using AdamW (Loshchilov and Hutter, 2019) with learning rate 7×10^{-4} , $\beta_1 = 0.9$, $\beta_2 = 0.98$ and weight decay 3×10^{-2} . We increase the learning rate for the first 15 epochs using linear warmup and then apply a cosine decay schedule (Loshchilov and Hutter, 2017b). Our comparison models are trained for 150 epochs using Adam (Kingma and Ba, 2015) with $\beta_1 = 0.9$, $\beta_2 = 0.98$ and the same learning rate schedule. The learning rate for GCRN, AV-GCRN, AV-Demucs, VisualVoice, and MuSE is set to 7×10^{-4} , 1×10^{-3} , 7×10^{-4} , 3×10^{-4} and 3×10^{-4} , respectively. To train MuSE, we replace the original SI-SDR objective (Pan *et al.*, 2021) with the loss from Demucs (L1 + multi-resolution STFT (Défossez *et al.*, 2020)), as we find this increases training stability and yields better results. We train HiFi-GAN for roughly 1 million iterations on AVSpeech using

AdamW with learning rate 2×10^{-4} , $\beta_1 = 0.8$, $\beta_2 = 0.99$ and weight decay 1×10^{-2} , decaying the learning rate by a factor of 0.999 every epoch.

6.4 Results

6.4.1 Comparison with other works

We compare with previous state-of-the-art methods in Table 6.1, and present a demo of these results on our project website¹. For noise condition 1, LA-VocE outperforms other approaches in quality and intelligibility, achieving significant improvements across all metrics. Indeed, even in this less noisy scenario, our vocoder-based approach can reproduce the target speech more accurately than mask-based methods such as MuSE (Pan *et al.*, 2021) and VisualVoice (Gao and Grauman, 2021), which are designed for separation with one to two background speakers. Previous AVSE methods yield decreased improvements for noise condition 2, particularly for speech quality metrics such as PESQ and ViSQOL, while LA-VocE yields significant gains in quality and especially intelligibility, as indicated by ESTOI i. This shows that despite identical training conditions, previous methods adapt poorly to lower SNR / SIR conditions compared to our new model.

Finally, on the noisiest scenario (noise condition 3), it is clear that other audio-visual methods, including mapping-based approaches (AV-GCRN (Tan and Wang, 2020) and AV-Demucs (Défossez *et al.*, 2020)), are unable to increase speech quality, achieving effectively no improvement on PESQ-WB and small increases on other metrics. LA-VocE, on the other hand, can still achieve significant gains in all metrics, indicating that our approach is substantially more robust to extremely low-SNR scenarios. Notably, both audio-only models (GCRN Tan and Wang, 2020 and LA-VocE) yield poor results in all scenarios - without visual information, the model is unable to accurately distinguish target speech from background speech.

¹<https://sites.google.com/view/la-voce-avse>

Table 6.1: Comparison between LA-VocE and other speech enhancement methods for different noise conditions. In the second column, “A” and “AV” stand for audio-only and audio-visual, respectively.

Method	Input	MCD $\downarrow$	PESQ-WB $\uparrow$	ViSQOL $\uparrow$	STOI $\uparrow$	ESTOI $\uparrow$
Noise condition 1 (1 background noise at 0 dB SNR + 1 interfering speaker at 0 dB SIR)						
GCRN (Tan and Wang, 2020)	A	0.410	0.044	0.093	-0.052	-0.038
AV-GCRN (Tan and Wang, 2020)	AV	-1.193	0.394	0.499	0.220	0.235
AV-Demucs (Défossez <i>et al.</i> , 2020)	AV	-5.581	0.738	0.688	0.270	0.298
MuSE (Pan <i>et al.</i> , 2021)	AV	-5.528	0.787	0.679	0.276	0.299
VisualVoice (Gao and Grauman, 2021)	AV	-3.781	0.606	0.645	0.249	0.270
LA-VocE (audio-only)	A	-3.189	0.248	0.135	0.055	0.047
LA-VocE	AV	-6.653	0.931	1.100	0.294	0.333
Noise condition 2 (3 background noises at -5 dB SNR + 2 interfering speakers at -5 dB SIR)						
GCRN (Tan and Wang, 2020)	A	-0.416	-0.010	0.163	-0.015	-0.015
AV-GCRN (Tan and Wang, 2020)	AV	-1.354	0.096	0.398	0.234	0.214
AV-Demucs (Défossez <i>et al.</i> , 2020)	AV	-5.548	0.274	0.426	0.308	0.300
MuSE (Pan <i>et al.</i> , 2021)	AV	-5.314	0.297	0.409	0.308	0.289
VisualVoice (Gao and Grauman, 2021)	AV	-3.388	0.164	0.367	0.253	0.237
LA-VocE (audio-only)	A	-2.817	0.056	0.087	0.066	0.043
LA-VocE	AV	-6.863	0.511	0.700	0.379	0.397
Noise condition 3 (5 background noises at -10 dB SNR + 3 interfering speakers at -10 dB SIR)						
GCRN (Tan and Wang, 2020)	A	-0.414	-0.015	0.210	-0.020	-0.005
AV-GCRN (Tan and Wang, 2020)	AV	-1.263	-0.043	0.217	0.171	0.139
AV-Demucs (Défossez <i>et al.</i> , 2020)	AV	-4.866	0.013	0.298	0.262	0.230
MuSE (Pan <i>et al.</i> , 2021)	AV	-4.185	0.011	0.242	0.231	0.182
VisualVoice (Gao and Grauman, 2021)	AV	-2.518	-0.045	0.248	0.181	0.160
LA-VocE (audio-only)	A	-1.982	-0.015	0.073	0.032	0.008
LA-VocE	AV	-6.170	0.159	0.447	0.371	0.358

6.4.2 Noise and interference study

We study our model’s performance in Table 6.2 by varying the SNR and SIR between 5 dB and -15 dB (as in training) while keeping the number of background noises and interfering speakers fixed at 3 and 2, respectively. On the left, we can see a very clear trend in PESQ-WB improvement, as it peaks for higher SNR/SIR conditions and deteriorates as the noise and interference increase. This suggests that the model excels at improving speech quality for higher SNR/SIR, even with the higher PESQ baseline, but struggles to achieve substantial gains when the environment becomes too noisy. On the other hand, ESTOI is substantially more consistent across all conditions and is in fact higher for -15 dB SNR/SIR than it is for 5 dB SNR/SIR. This indicates that, while LA-VocE struggles to produce high-quality audio for extremely noisy scenarios, it manages to achieve impressive improvements in intelligibility even for -15 dB SNR/SIR, where the target speech is entirely imperceptible for human listeners.

Table 6.2: LA-VocE’s performance for different SNR / SIR conditions with 3 background noises and 2 interfering speakers.

		PESQ-WB $\uparrow$					ESTOI $\uparrow$				
SNR (dB)		5	0	-5	-10	-15	5	0	-5	-10	-15
SIR (dB)	5	0.970	0.876	0.715	0.486	0.245	0.269	0.316	0.356	0.375	0.362
	0	0.904	0.795	0.630	0.411	0.210	0.327	0.354	0.375	0.378	0.355
	-5	0.789	0.679	0.511	0.319	0.136	0.386	0.394	0.397	0.383	0.349
	-10	0.617	0.523	0.405	0.248	0.092	0.429	0.426	0.414	0.388	0.344
	-15	0.438	0.383	0.289	0.195	0.081	0.443	0.433	0.414	0.381	0.330

Table 6.3: LA-VocE’s performance for different numbers of background noises and interfering speakers (-5 dB SNR / SIR).

		PESQ-WB $\uparrow$					ESTOI $\uparrow$				
# noises		1	2	3	4	5	1	2	3	4	5
# spk.	1	0.709	0.642	0.601	0.580	0.557	0.396	0.402	0.404	0.404	0.403
	2	0.602	0.553	0.511	0.497	0.482	0.396	0.398	0.397	0.395	0.393
#	3	0.539	0.490	0.462	0.455	0.431	0.390	0.390	0.388	0.387	0.384

This is consistent with our perceptual evaluation - LA-VocE consistently produces intelligible audio despite the noticeable artifacts for lower SNR / SIRs.

Notably, LA-VocE performs better for lower SIRs compared to lower SNRs, e. g., 5 dB SNR / -15 dB SIR substantially outperforms -15 dB SNR / 5 dB SIR on both metrics. This disparity is likely due to the nature of these two signals. Speech typically has a consistent frequency range and timbre and often contains gaps that the model can easily exploit, while noise is substantially more heterogeneous, ranging from impulses to continuous noises, presenting a greater denoising challenge. We also evaluate our model’s ability to perform enhancement under multiple noise sources and background speakers in Table 6.3, keeping the SNR / SIR at -5 dB. Unsurprisingly, we find that the strongest PESQ-WB improvements are achieved with 1 noise and 1 speaker, and become worse as they are increased, achieving a minimum of 0.431 for 5 noises and 3 speakers. While it is expected that increasing the number of sources will increase the complexity of the background noise, therefore making the enhancement task more difficult, we hypothesize that the sharper drop in performance when increasing the number of speakers is related to the temporal and spectral gaps in the interference. A single stream

Table 6.4: Comparison between different spectrogram inversion methods for LA-VocE (noise condition 2). In the second column, “Train. corp.” stands for training corpus.

Method	Train. corp.	MCD $\downarrow$	PESQ-WB $\uparrow$	ViSQOL $\uparrow$	STOI $\uparrow$	ESTOI $\uparrow$
Griffin-Lim (Griffin and Jae Lim, 1984)	-	-6.805	0.333	0.806	0.311	0.318
Noisy phase	-	-6.640	0.461	0.721	0.305	0.310
HiFi-GAN V1 (Kong <i>et al.</i> , 2020b)	VCTK	-6.570	0.384	0.655	0.374	0.388
HiFi-GAN V1 (Kong <i>et al.</i> , 2020b)	LJSpeech	-6.601	0.432	0.670	0.370	0.382
HiFi-GAN V1 (Kong <i>et al.</i> , 2020b)	AVSpeech	-6.863	0.511	0.700	0.379	0.397

of speech will contain pauses that will ease denoising, but these disappear as we increase the number of speakers, resembling continuous noise.

6.4.3 Spectrogram inversion comparison

Finally, we compare our trained HiFi-GAN with other spectrogram inversion methods in Table 6.4. We observe that our HiFi-GAN achieves better performance than existing pre-trained models² (trained on VCTK (Yamagishi *et al.*, 2012) and LJSpeech (Ito and Johnson, 2017), as presented in the original HiFi-GAN paper (Kong *et al.*, 2020b)) on all five metrics, highlighting the importance of training our own vocoder on AVSpeech, rather than applying a publicly available pre-trained model as in Chapter 5. As we demonstrate here, using a pre-trained vocoder makes for a simpler training procedure (since there is no need for a second stage), but yields worse results. We also compare with Griffin-Lim (Griffin and Jae Lim, 1984), a commonly-used spectrogram inversion algorithm, and experiment by applying iSTFT using the phase from the noisy input to reconstruct the waveform, as proposed by Gabbay *et al.* (2018) and Hou *et al.* (2018). In our experiments, both methods consistently produce artifacts that make the resulting waveforms sound noticeably more robotic than those produced by neural vocoders (this is particularly noticeable for Griffin-Lim). We show that these inversion methods yield significant drops in PESQ-WB $\uparrow$, STOI $\uparrow$, and ESTOI $\uparrow$, but surprisingly achieve competitive MCD $\downarrow$ performance, and substantially better ViSQOL $\uparrow$. This inconsistency likely implies that these three metrics are less sensitive to the specific artifacts introduced by these phase estimation strategies, and emphasizes the need for multiple evaluation metrics when evaluating synthesized speech.

²<https://github.com/jik876/hifi-gan>

6.5 Conclusion

In this chapter, we propose LA-VocE, a new framework for audio-visual speech enhancement under low-SNR conditions. Our method consists of two stages of training: audio-visual spectral enhancement via a transformer-based encoder, and waveform synthesis via HiFi-GAN. We train our model on thousands of hours of multilingual audio-visual speech and find that it significantly outperforms previous state-of-the-art AVSE approaches, particularly for higher noise conditions. We study LA-VocE’s performance under varying conditions of noise and interference, showing that even in the noisiest scenarios our vocoder-based approach can achieve large improvements in speech intelligibility. Finally, we compare our vocoder with existing spectrogram inversion methods, highlighting the importance of training our own HiFi-GAN. In the future, we believe it would be promising to adapt and optimize our architecture for real-time synthesis, which would enable speech enhancement in live video streams.

CHAPTER 7

LEARNING FROM BOTH MODALITIES SIMULTANEOUSLY

Contents

7.1	Introduction	125
7.2	RealForensics	126
7.2.1	Methodology	126
7.2.2	Experimental Setup	127
7.2.3	Results	129
7.3	RAVE_n	131
7.3.1	Methodology	132
7.3.2	Experimental Setup	134
7.3.3	Results	135
7.4	Conclusion	137

This chapter is based on two papers: *Leveraging Real Talking Faces via Self-Supervision for Robust Forgery Detection*, also known as *RealForensics*, a conference paper published in *CVPR 2022*, and *Jointly Learning Visual and Auditory Speech Representations from Raw Data*, also known as *RAVEN*, a conference paper published in *ICLR 2023*.

Alexandros Haliassos was the first author of *RealForensics*, designed both the pretext and downstream tasks, conducted the experimental procedure, and wrote the manuscript. I, the second author, made substantial contributions to discussions around the design of the self-supervised framework, contributed extensively to the codebase, conducted a wide variety of experiments (many of which were included in the final paper), and helped write the manuscript. Stavros Petridis and Maja Pantic supervised the project throughout its entirety, providing helpful suggestions, feedback, and guidance regarding the design of the framework and the writing of the manuscript.

Alexandros Haliassos was also the first author of *RAVEN*, designed both the pretext and downstream tasks, conducted the experimental procedure, and wrote the manuscript. Pingchuan Ma contributed to the design of the downstream frameworks (including substantial contributions to the codebase), conducted many downstream experiments which were featured in the paper, and helped write the manuscript. I, the third author, helped design the self-supervised framework and the underlying experimental procedure, conducted several experiments, and helped write the manuscript. Stavros Petridis and Maja Pantic supervised the project throughout its entirety, providing helpful suggestions, feedback, and guidance regarding the design of the framework and the writing of the manuscript.

As in previous Chapters, the related work and comparison with other works presented below only include papers released before the time of submission (November 2021 and September 2022, respectively). Since then, many works have competed with *RealForensics*, with some of them outperforming our model under specific conditions (Bai *et al.*, 2023; Feng *et al.*, 2023). Similarly, some of *RAVEN*'s results on LRS3 have been outperformed by the new AV-data2vec (Lian *et al.*, 2023).

7.1 Introduction

As discussed in Chapter 3, self-supervised learning has become a popular way to learn representations from unlabelled data by leveraging the data itself as a training objective. While many approaches have focused on image (Chen *et al.*, 2020; Grill *et al.*, 2020), audio (Baevski *et al.*, 2020; Hsu *et al.*, 2021), and video (Tong *et al.*, 2022; Fernando *et al.*, 2017) individually, not many works have considered video and audio together for self-supervised learning. This is a particularly promising topic since audio-visual speech data has become ubiquitous in modern online platforms, and can be easily leveraged to train new deep learning models.

So far, we have presented three projects that demonstrate how visual speech encoders can learn from acoustic speech to perform video-to-speech synthesis, or to train better lip reading models. These approaches are motivated by the intuitive idea that acoustic speech may contain better or more salient information about specific aspects of the speech (e. g., verbal information) which the visual encoder can leverage to produce better features. However, it may also be the case that video contains better information about other aspects of the speech, e. g., emotional cues. Therefore, one could train a self-supervised acoustic speech encoder to predict visual features in order to produce better features for emotion recognition, for example.

Inspired by this idea, Shukla *et al.* (2021) propose a self-supervised approach to improve emotion recognition from speech by leveraging video as an objective. Specifically, the model learns by reconstructing the corresponding video based on a previous speech-driven facial animation work (Vougioukas *et al.*, 2018), while simultaneously applying a self-supervised audio loss based on the work of Fernando *et al.*, 2017. The results show that this self-supervised framework consistently outperforms traditional supervised learning on emotion recognition and that visual supervision plays a fundamental role in self-supervision. This method also outperforms previous audio-only self-supervised approaches.

These two research directions lead us to the idea of designing a framework that can perform both tasks simultaneously, i. e., learn visual representations from audio and acoustic representations from video. Inspired by this idea, we propose two audio-visual self-supervised frameworks:

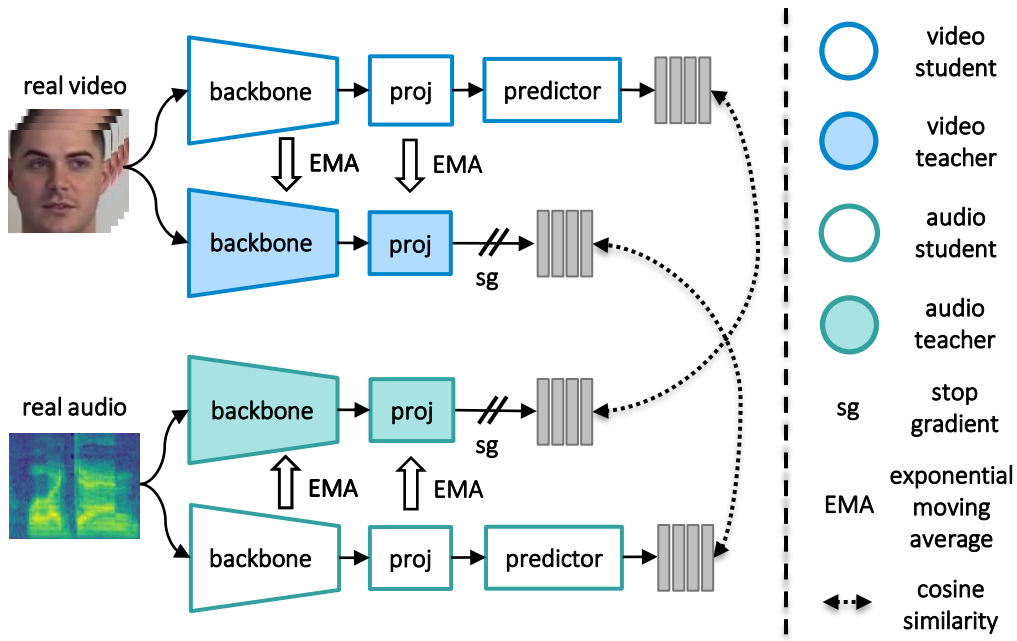


Figure 7.1: Summary of the RealForensics self-supervised training procedure.

RealForensics and RAVeN.

7.2 RealForensics

RealForensics is a self-supervised model inspired by Bootstrap Your Own Latent (BYOL) (Grill *et al.*, 2020), a recent self-supervised framework that achieves state-of-the-art performance in image representation learning. The method works by employing a student-teacher pair for each modality (audio and video), where the student in one modality predicts the other modality’s teacher’s features. We train it on an unlabelled version of LRW and apply the pre-trained visual encoder for DeepFake detection. We show that this method achieves state-of-the-art on multiple forgery detection tasks.

7.2.1 Methodology

Our self-supervised pre-training method is summarized in Figure 7.1. Inspired by BYOL (Grill *et al.*, 2020), we propose a student-teacher pair for each modality, where the encoder architectures of the students and teachers are identical, and the teachers’ weights are updated as an

exponential moving average (with momentum set to 0.999) of the students' weights. The video encoder is composed of a Channel-Separated convolutional Network (CSN) (Tran *et al.*, 2019) with the temporal strides set to 1, which serves as our visual backbone, followed by a linear projector, which consists of a single convolutional layer with kernel size 1, and a batch normalization layer. This model receives raw RGB video of the full face as input sampled at 25 fps and outputs a 256-dimension visual feature per input frame. The audio encoder, on the other hand, receives log-scaled mel-spectrograms sampled at 100 fps, which are downsampled to 25fps via a 2D ResNet-18 backbone (He *et al.*, 2016) followed by a linear projector (identical to the one in the video encoder).

Each student's features are fed into single-block transformer encoders, which serve as our predictors. The predicted features are first L2-normalised and then compared with the other modality's teacher's features via an L2 Loss, which we aim to minimize. We note that, as shown in Figure 7.1, we apply a stop-gradient in each teacher to prevent training collapse, as proposed by Grill *et al.* (2020). We optimize our model using AdamP (Heo *et al.*, 2021), with the learning rate set to 7×10^{-4} , $\beta_1 = 0.9$, $\beta_2 = 0.999$ and weight decay set to 0.01. We train for 150 epochs, where we linearly increase the backbones' and projectors' learning rate until epoch 20 using warmup, and then apply a cosine decay learning rate schedule (Loshchilov and Hutter, 2017b), as in Chapters 5 and 6. In contrast, the predictors' learning rate is kept fixed throughout training, inspired by Chen and He (2021).

7.2.2 Experimental Setup

We train our self-supervised model on LRW (Chung and Zisserman, 2016a) (without labels) using the original training/validation/testing file splits. We find that this dataset is large enough to provide a good pre-training corpus for this specific project, and allows us to have reasonable computational costs per experiment. During pre-training, we apply random cropping to size 140×140 , and then resize the frame to size 112×112 , and also apply random horizontal flipping and grayscale transformation, both with a probability of 50%, similarly to previous chapters. In addition, we apply random masking on the inputs of the student networks, which forces them

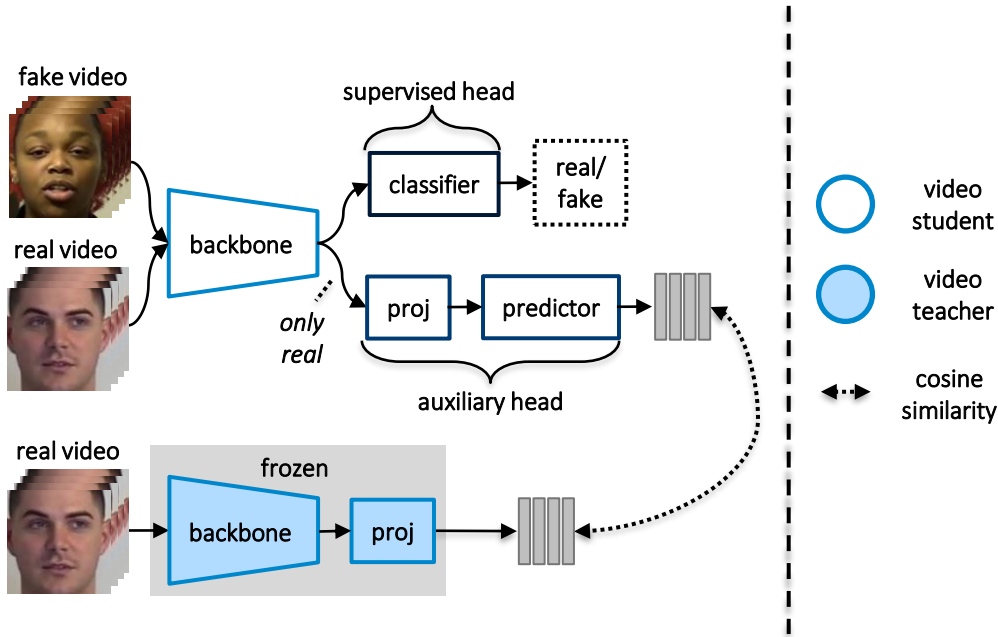


Figure 7.2: Summary of the RealForensics downstream training procedure, after self-supervised pre-training.

to predict the missing information. We randomly erase a set of frames from each video, as in Chapter 5, and similarly erase temporal frames and frequency bins from each spectrogram, as proposed by Park *et al.* (2019).

We evaluate the performance of our pre-trained visual encoder by applying it for deepfake detection (also known as forgery detection). The CSN backbone is transferred for this new downstream task and is fine-tuned to perform two tasks simultaneously. On one side of the network, the visual features are fed into a cosine classifier (Wang *et al.*, 2017a) which classifies whether the video sample is real or fake. At the same time, if the input is a real sample, its visual features are fed into a projector and predictor (identical to the ones seen previously, but trained from scratch) which are trained to predict the frozen teacher’s features from the pre-training stage. Our downstream training procedure is summarized in Figure 7.2. This second auxiliary learning task encourages the network to classify the samples based on high-level semantic features, such as mouth movements and emotion, due to the teacher likely having learned to capture such information in the self-supervised stage. We train the downstream model on a FaceForensics++ (FF++) (Rössler *et al.*, 2019), a large and well-established deepfake detection corpus made using two face swapping methods and two face reenactment methods (4000 fake

Table 7.1: Cross-dataset forgery detection performance (Area Under Curve (AUC) in %) of RealForensics compared to previous works. Our model is pre-trained on LRW via our self-supervised framework, and fine-tuned for forgery detection on FaceForensics++, as described in Section 7.2.2. The acronyms for each dataset presented on the first row are explained in the last paragraph of Section 7.2.1

Method	CDF	DFDC	FSh	DFo	Avg
Xception (Rössler <i>et al.</i> , 2019)	73.7	70.9	72.0	84.5	75.3
CNN-aug (Wang <i>et al.</i> , 2020a)	75.6	72.1	65.7	74.4	72.0
Patch-based (Chai <i>et al.</i> , 2020)	69.6	65.6	57.8	81.8	68.7
Face X-ray (Li <i>et al.</i> , 2020b)	79.5	65.5	92.8	86.8	81.2
CNN-GRU (Sabir <i>et al.</i> , 2019)	69.8	68.9	80.8	74.1	73.4
Multi-task (Nguyen <i>et al.</i> , 2019)	75.7	68.1	66.0	77.7	71.9
DSP-FWA (Li and Lyu, 2019)	69.5	67.3	65.5	50.2	63.1
Two-branch (Masi <i>et al.</i> , 2020)	76.7	—	—	—	—
LipForensics (Haliassos <i>et al.</i> , 2021)	82.4	73.5	97.1	97.6	87.7
FTCN (Zheng <i>et al.</i> , 2021)	86.9	74.0	98.8	98.8	89.6
CSN (supervised training only)	69.4	68.1	87.9	89.3	78.7
RealForensics (our self-supervised method)	86.9	75.9	99.7	99.3	90.5

videos and 1000 real videos in total), and measure its cross-dataset performance by evaluating it on separate forgery detection datasets: CelebDF-v2 (CDF) (Li *et al.*, 2020c), which contains 5,639 synthetic videos generated using their improved face swapping algorithm; a subset of DeepFake Detection Challenge (DFDC) (Dolhansky *et al.*, 2019; Haliassos *et al.*, 2021), amounting to 3,215 videos filmed under extreme conditions (e.g., low lighting); FaceShifter (FSh) (Li *et al.*, 2020a) and DeeperForensics (DFo) (Jiang *et al.*, 2020), each of which applies their own face swapping methods on the real videos from FF++ to generate new deepfakes. We apply random time masking (as in pre-training) and random erasing on the input video during downstream training.

7.2.3 Results

We begin by comparing our method with previous forgery detection approaches on all downstream datasets in Table 7.1, where we pre-train on LRW (via self-supervised learning) and train on FaceForensics++. It is clear that RealForensics outperforms all previous approaches, spanning a wide variety of techniques, on all presented datasets. Our method yields partic-

Table 7.2: Ablation study for RealForensics on the downstream deepfake detection task. The results shown refer to the forgery detection accuracy (%). All models are pre-trained on LRW and fine-tuned on FaceForensics++.

Method	FSh	DFo
RealForensics	97.1	97.1
CSN (supervised training only)	82.1	83.1
self-supervised pre-training + fine-tuning (w/o auxiliary task)	95.0	95.2
w/o time masking	96.1	95.9
w/o random erasing	96.3	96.3

Table 7.3: Comparison of RealForensics with other self-supervised approaches. The results shown refer to the forgery detection accuracy (%). All models are pre-trained on LRW and fine-tuned on FaceForensics++.

Method	FSh	DFo
Perfect Match (Chung <i>et al.</i> , 2019)	91.4	87.9
Symmetrised Perfect Match	91.8	90.2
Video-only RealForensics	92.9	89.7
RealForensics	97.1	97.1

ularly impressive improvements on DFDC and FaceShifter (FSh), while on average achieving 0.9 % higher AUC (in absolute terms) than the previous state-of-the-art result. Notably, the model considerably outperforms its supervised counterpart by a wide margin on all datasets, indicating the importance of our audio-visual self-supervised pre-training. Indeed, if we apply the same video encoder (CSN) for naïve supervised downstream training, the results are well below the previous state-of-the-art.

We move on to perform an ablation on the main aspects of RealForensics in Table 7.2. We experiment by training a CSN for deepfake detection from scratch and find that this configuration yields a decrease of 15 and 14 % for FaceShifter (FSh) and DeeperForensics (DFo) compared to our baseline method. Then, we transfer the pre-trained model from the self-supervised stage (trained on LRW) and fine-tune it for forgery classification without the auxiliary teacher loss and find that this substantially increases downstream accuracy, but is still noticeably worse than the baseline. This leads us to two fundamental conclusions: **1)** The audio-visual pre-training stage plays a very important role in RealForensics’ performance, and removing it yields a very

large decrease in downstream performance, and **2)** the proposed regularization effect of the auxiliary task during downstream training plays a less important role, but still yields tangible improvements in classification accuracy. We also experiment by removing time masking and random erasing on the video input of the downstream task and find that this yields worse results, highlighting the importance of these data augmentations.

Finally, we compare RealForensics with other self-supervised methods in Table 7.3. First, we compare with a recent contrastive method - Perfect Match (Chung *et al.*, 2019). This approach works by feeding a short video clip into our visual encoder, and feeding multiple audio clips (from the same video sample) into our audio encoder, one of which is correctly aligned with the video. Through a simple contrastive loss, the encoders learn to produce features that are similar for the aligned audio/video pair, and distant for misaligned pairs. We propose to symmetrize this architecture by also performing the inverse task of having multiple misaligned videos for one audio input, and find that this achieves better results. Overall, however, RealForensics significantly outperforms both of these approaches. Finally, we investigate the importance of paired audio-visual data, by pre-training on video only (from LRW) using the BYOL framework (Grill *et al.*, 2020). We find that this pre-training approach is clearly inferior to our audio-visual baseline (RealForensics) in downstream performance. This clearly highlights the significance of applying a pre-training scheme that leverages the duality between video and audio.

7.3 RAVEn

Raw Audio-Visual Encoder (RAVEn) is a self-supervised audio-visual model which extends and reformulates the ideas of RealForensics. Inspired by He *et al.* (2022), we introduce masking mechanisms before the predictor networks, which provide direct improvements in downstream performance. We also entirely forego the use of handcrafted features by encoding raw waveform directly, adopt transformer encoders as the main backbone for both modalities, remove feature projectors since we find they no longer improve performance and introduce intra-modality self-

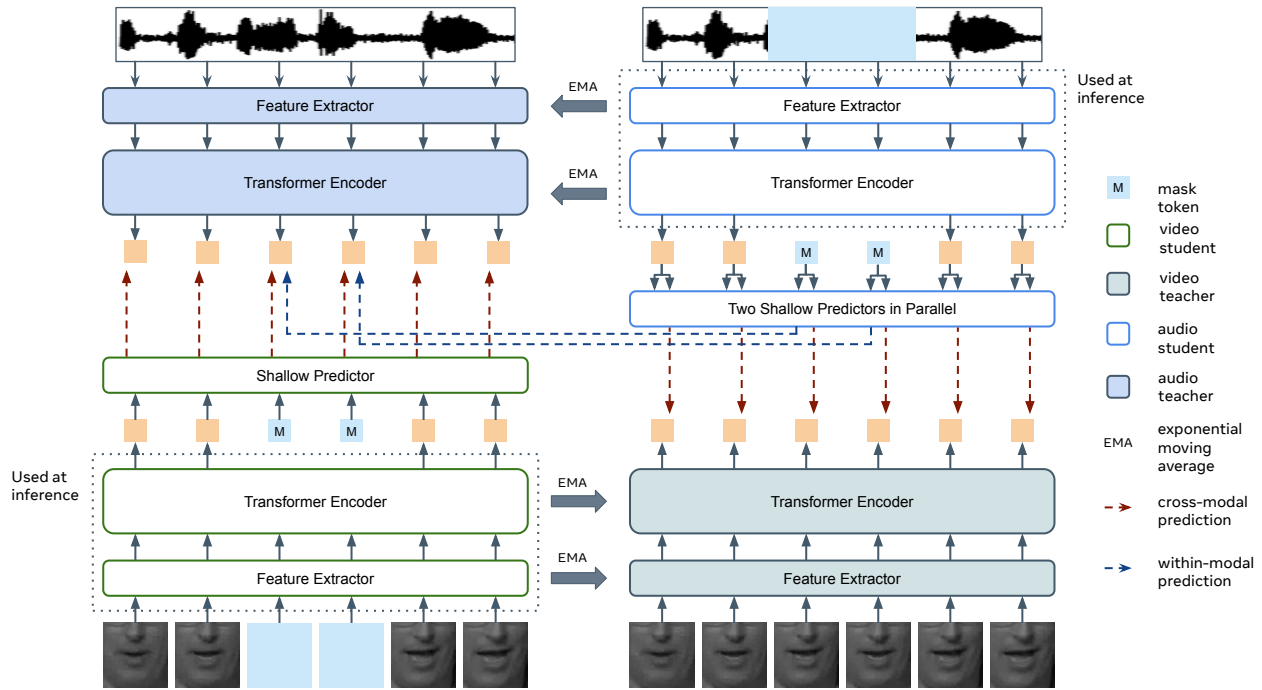


Figure 7.3: Summary of RAVEn’s self-supervised training procedure.

supervised learning in addition to the cross-modal learning. In order to make full use of both our pre-trained encoders (trained on a combination of LRS3 and VoxCeleb2), we apply the audio and video branches for speech recognition and lip reading, respectively. We show that RAVEn achieves state-of-the-art lip reading results on LRS3, and outperforms all previous self-supervised approaches for speech recognition and lip reading on a low-resource scenario where only 30 hours of labeled data (from LRS3) are available.

7.3.1 Methodology

As in RealForensics, we propose a student-teacher pair for each modality, inspired by Grill *et al.* (2020), as shown in Figure 7.3. In RAVEn, however, both backbones (video and audio) are based on deep transformer encoders with 12/24 blocks for the Base/Large variants of our model (RAVEn-Base and RAVEn-Large). On the audio side, we encode raw waveform using a 1D ResNet-18 which is adjusted to produce features that match the video’s temporal dimension (25 fps). On the video side, we encode raw grayscale cropped mouth video by applying a 3D+2D ResNet-18, as in previous chapters. Unlike RealForensics, we apply no projection layers after

Table 7.4: Comparison with other supervised and self-supervised works on LRS3 using only 30 hours of labeled training data (Lab. hours) and 433 (full LRS3 training set) / 1,759 (LRS3+Vox2-en) hours of unlabelled data (Unlab. hours). “LM” refers to whether a Language model was used in the downstream task. *Includes non-publicly available data, some of which is not hand-labeled. †Result obtained from (Hsu *et al.*, 2021). ‡Result obtained with the same pre-training strategy for VSR and ASR.

Method	Encoder	LM	Unlab. hours	Lab. hours	WER (%)	
					VSR	ASR
Supervised						
Afouras <i>et al.</i> (2018c)	Transformer	✓	-	1,519*	58.9	8.3
Xu <i>et al.</i> (2020)	RNN	✗	-	590	57.8	7.2
Shillingford <i>et al.</i> (2019)	RNN	✓	-	3,886*	55.1	-
Ma <i>et al.</i> (2022a)	Conformer	✓	-	813	34.7	-
Makino <i>et al.</i> (2019)	RNN	✗	-	31,000*	33.6	4.8
Prajwal <i>et al.</i> (2022)	Transformer	✓	-	2,676*	30.7	-
Serdyuk <i>et al.</i> (2021)	Transformer	✗	-	90,000*	25.9	2.3
Serdyuk <i>et al.</i> (2022)	Conformer	✗	-	90,000*	19.3	1.6
Trained from scratch (supervised)						
Base model	Transformer	✗	-	30	93.4	18.5
Large model	Transformer	✗	-	30	95.5	9.9
Self-supervised						
Base models, less pre-training data						
Ma <i>et al.</i> (2021c) (Chapter 3)	Transformer	✗	433	30	71.9 [†]	-
Zhang <i>et al.</i> (2022)	Transformer	✗	433	30	67.8	10.9
Hsu <i>et al.</i> (2021)	Transformer	✗	433	30	-	5.4
Shi <i>et al.</i> (2022a)	Transformer	✗	433	30	51.8	4.9
RAVEN	Transformer	✗	433	30	47.0	4.7
Base models, more pre-training data						
Zhang <i>et al.</i> (2022)	Transformer	✗	1,759	30	-	5.0
Hsu <i>et al.</i> (2021)	Transformer	✗	1,759	30	46.1	4.6 [‡] / 3.8
RAVEN	Transformer	✗	1,759	30	40.2	3.8
Large models, more pre-training data						
Zhang <i>et al.</i> (2022)	Transformer	✗	1,759	30	-	3.2
Hsu <i>et al.</i> (2021)	Transformer	✗	1,759	30	32.5	2.9
Hsu <i>et al.</i> (2021) w/ self-train.	Transformer	✗	1,759	30	28.6	-
RAVEN	Transformer	✗	1,759	30	33.1	2.6
RAVEN w/ self-train.	Transformer	✗	1,759	30	26.2	2.2
RAVEN w/ self-train.	Transformer	✓	1,759	30	24.4	1.9

the feature encoders. Instead, we apply token-level masking before the predictors, a choice motivated by the findings of He *et al.* (2022). As in RealForensics, we use shallow transformer encoders as predictors but increase the number of blocks to 2, and the hidden size to 512.

We keep the cross-modal loss from RealForensics, which is an L2 loss between the real and predicted features. Furthermore, we add an intra-modal loss for the audio branch. The audio features are fed into a separate predictor which predicts the teacher’s features for the masked input frames (this is unlike the cross-modal prediction task, where all features from all input frames, masked and unmasked, must be predicted). We motivate this asymmetrical design through an ablation study presented later in Section 7.3.3. We optimize our self-supervised framework using AdamW with the learning rate set to $3/2 \times 10^{-3}$ (for RAVeN-Base/Large), $\beta_{a_1} = 0.9$, $\beta_{a_2} = 0.999$ and weight decay set to 0.04. We train for 150 epochs, where we use learning warmup for the first 30/40 epochs (for LRS3/LRS3+Vox2-en) and a cosine decay schedule for the remaining epochs, as in previous chapters.

7.3.2 Experimental Setup

For self-supervised training, we use two datasets. The first is LRS3, where we use the same training/validation/testing split as in Chapter 3, but remove the transcriptions. The second is a slightly extended version of the combination of the training sets from LRS3 and English-only VoxCeleb2 presented in Chapter 5, curated and originally presented by Shi *et al.* (2022a). For the remainder of this chapter, we refer to this corpus as LRS3+Vox2-en. We apply random horizontal flipping ($p = 0.5$) and random cropping (to size 88×88) for video, and random time masking for both video and audio. We crop the mouth in each video using RetinaFace and FAN, as in Chapter 5.

Instead of discarding our audio encoder, as we did in RealForensics, we use both modalities’ encoders for separate tasks: speech recognition (audio) and lip reading (video). In both cases, we transfer the full encoders (ResNet+Transformer encoder) from each student and discard the predictors. Instead, we add (for both tasks) a decoder, composed of a single linear layer followed by a 6/9-block transformer decoder (for the Base and Large variants), which predicts the

transcriptions. We fine-tune these models for speech recognition/lip reading using a combined CTC+attention loss. We also experiment with self-training by first fine-tuning the audio model on the available labeled data, then generating pseudo-labels for the remaining unlabelled data, and finally training both models (audio and video) on the new (larger) set of labeled data. To further boost performance, we also add a transformer-based pre-trained language model (as presented by Irie *et al.* (2019)), which is trained on a combination of many large language datasets totaling 166 million characters.

7.3.3 Results

We compare with other works in Table 7.4 using only 30 hours of labeled audio-visual speech from LRS3. We experiment with three configurations: “Base models, less pre-training data”, where we use the base variant of RAVEn (and other works) and only the unlabelled training data from LRS3; “Base models, more pre-training data”, where we increase the amount of unlabelled data by roughly $4\times$ by adding English-only data from VoxCeleb2; and “Large models, more pre-training data” where we increase the model size by using the large variant of RAVEn (and other works), while still using LRS3+Vox2-en for pre-training. RAVEn achieves state-of-the-art results in all three categories, yielding particularly impressive improvements for VSR. On the third task, we find that our self-training strategy boosts results by a significant margin, particularly for lip reading, indicating that the combination of self-supervised and semi-supervised learning yields the best results. Furthermore, we observe that the addition of the language model (seen in the last row) noticeably improves performance on both VSR and ASR, justifying its inclusion in our downstream methodology. Finally, on a more general note, we would like to highlight RAVEn’s remarkable scalability: by scaling up the model and leveraging more unlabelled data (and applying the additional techniques presented above), RAVEn roughly halves its WER on VSR (from 47.0% to 24.4%) and achieves an even larger relative improvement on ASR (4.7% to 1.9%) while training on the same amount of labeled data.

We proceed by conducting an ablation on the prediction tasks presented in RAVEn, as illustrated in Figure 7.4 and described in Table 7.5. We begin by running a version of RAVEn where

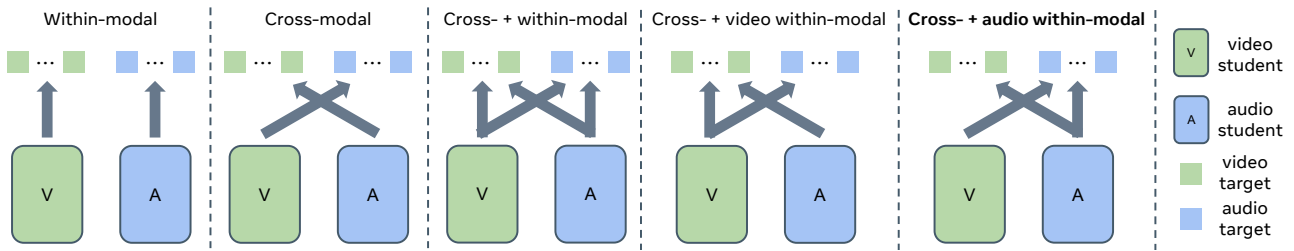


Figure 7.4: Summarized overview of RAVEn’s experimental configurations featured in our ablation study.

Table 7.5: Ablation study on the prediction tasks used in RAVEn. The models are pre-trained on LRS3 (full training set) and fine-tuned on 30 hours of LRS3 labeled data using RAVEn-Base, as in the “Base models, more pre-training data” configuration in Table 7.4. In the upper row, **V** and **A** stand for video and audio, respectively.

Setting	Prediction tasks				WER (%)	
	V → V	A → A	V → A	A → V	VSR	ASR
Within-modal	✓	✓	✗	✗	92.7	15.5
Cross-modal	✗	✗	✓	✓	40.8	14.0
Cross- + within-modal	✓	✓	✓	✓	49.0	14.0
Cross- + video within-modal	✓	✗	✓	✓	55.3	16.4
Cross- + audio within-modal	✗	✓	✓	✓	32.9	12.2

only within-modal losses (on both video and audio) are used in pre-training, effectively consisting of two separate uni-modal self-supervised models. Remarkably, this setup achieves very high WER, since it does not leverage the duality of video and audio, which plays a fundamental role in RAVEn’s success. Then, we apply only the cross-modal losses, as in RealForensics, yielding much better results. We attempt to combine the two approaches, having both cross-modal and within-modal losses on both sides of the network, and find that the addition of the within-modal losses, in fact, deteriorates performance. Finally, we combine each within-modal loss with both cross-modal losses in two separate experiments, to investigate the role of within-modal learning in each modality. The results clearly show that the video within-modal loss has a strong negative effect on performance, whereas the audio within-modal loss (on its own) improves downstream performance substantially. We hypothesize that this large discrepancy in within-modal performance can be attributed to the nature of the two modalities: in most scenarios, it is expected that the audio will contain better and more salient verbal information than the video, particularly due to the existence of homophemes in visual speech (Chung *et al.*, 2017).

7.4 Conclusion

In this chapter, we present RealForensics and RAVeN, two self-supervised frameworks that can leverage large datasets of unlabelled audio-visual speech to improve performance on practical downstream tasks. In RealForensics, we show that our self-supervised approach yields a video encoder that consistently improves forgery detection performance in multiple cross-dataset scenarios, achieving state-of-the-art performance on four different deepfake datasets. We also show that our predictive cross-modal methodology outperforms existing audio-visual synchronization-based contrastive methods, as well as video-only self-supervised pre-training based on state-of-the-art research. In RAVeN, we extend and adapt our methodology through multiple changes and additions, including a within-modal loss, and fine-tune both pre-trained encoders (audio and video) to outperform previous works on speech recognition and lip reading. Furthermore, we conduct an ablation on RAVeN’s prediction tasks, where we find that the cross-modal losses play a fundamental role in RAVeN’s downstream performance, and the addition of the audio-only within-modal loss yields a lower WER on both tasks, justifying its inclusion in the framework. In summary, we design two frameworks where both modalities can learn from each other simultaneously while directly improving each other’s representations, and find that these can be applied to real-world tasks. In the future, we believe it would be promising to apply our frameworks for neighboring speech-related tasks, such as emotion recognition. In addition, it would potentially be promising to combine the three most prevalent methods in recent lip reading works - self-supervised learning, semi-supervised learning, and distillation - into one unified framework.

CHAPTER 8

CONCLUSION

Contents

8.1	Summary	138
8.2	Applications	140
8.3	Ethical Considerations	141
8.4	Future Work	143

8.1 Summary

This thesis aims to leverage unlabelled audio-visual speech via deep learning to improve real-world performance on audio/video-related tasks. It begins by using speech-related features as a self-supervised objective to improve performance on visual speech recognition. Then, we take this idea to its furthest extent by trying to leverage audio not only as an objective to learn useful features, but as a synthesis target, and propose two separate video-to-speech synthesis frameworks that achieve state-of-the-art results. After this, we apply our developments in

video-to-speech to improve audio-visual speech enhancement under noisy conditions via the use of powerful audio-visual encoders and neural vocoders. Finally, we propose to leverage the full potential of audio-visual speech by using both modalities as inputs and objectives in a single unified framework. To this end, we propose two methods where acoustic and visual speech learn from each other simultaneously, progressively improving each other’s features, and apply the trained encoders for deepfake detection, speech recognition, and lip reading.

In Chapter 3, we leverage an existing pre-trained speech encoder (PASE+ (Ravanelli *et al.*, 2020a)) to generate an acoustic objective that can be used to train our ResNet+Conformer-based video encoder. We pre-train on an unlabelled version of LRS3 using this novel objective and then fine-tune the pre-trained visual model for lip reading. We show that the pre-training outperforms previous self-supervised methods on LRW (word-level lip reading) and achieves state-of-the-art performance on LRS2 (sentence-level lip reading). Furthermore, we compare the pre-trained model with our baseline (training from scratch) for different portions of labeled data and find that the pre-trained weights consistently yield an improvement in downstream performance, but are particularly important when the labeled dataset is very small.

In Chapter 4, we propose a new end-to-end video-to-speech synthesis framework that generates raw waveform from raw video of the speaker’s lip movements. We propose a convolutional encoder-decoder structure for the waveform generator (with a GRU-based temporal encoder), which is trained via two discriminators that operate on the waveform and spectrogram domains, and three comparative losses. We find that this model achieves state-of-the-art performance according to most metrics on three versions of GRID, TCD-TIMIT, and LRW. Through an extensive ablation study, we find that each of the proposed losses plays an important role in the quality of the reconstructed speech.

In Chapter 5, we design a new video-to-speech model that prioritizes simplicity and scalability. This is based on a ResNet+Conformer generator that predicts log-scaled mel-spectrograms from raw video, which can be scaled for different datasets. The predicted spectrograms are converted into raw waveform using a pre-trained neural vocoder (Parallel WaveGAN). We demonstrate that this approach outperforms all previous methods on GRID and LRW, and experiment for

the first time with LRS3, which is larger than any previously mentioned dataset in video-to-speech literature.

In Chapter 6, we borrow from our experience in video-to-speech synthesis by proposing a new audio-visual speech enhancement (AVSE) model for Low-SNR conditions. We propose a ResNet+Transformer encoder that receives raw mouth crop video and noisy audio and predicts log-mel spectrograms. Rather than using a pre-trained vocoder, as we did in Chapter 5, we train our own version of HiFi-GAN on the same audio-visual corpus (AVSpeech). We find that this approach considerably outperforms previous AVSE methods for three different noise conditions with multiple background speakers and noise sources. Furthermore, we conduct a study where we evaluate our model’s ability to perform different levels of noise and interference, and find that it consistently outputs intelligible results.

In Chapter 7, we propose two audio-visual self-supervised frameworks where acoustic and visual speech can learn from each other to produce better representations. These approaches are inspired by recent advances in image-based methods (Grill *et al.*, 2020; He *et al.*, 2022), and work by training the visual and acoustic speech encoders to predict each other’s latent features. We apply our first framework, RealForensics, by fine-tuning the pre-trained video encoder for deepfake detection, and find that it outperforms all previous works on four cross-dataset benchmarks. In our second project, RAVEn, we apply the audio and video encoders for ASR and VSR respectively and find that they both outperform previous self-supervised methods when few labeled examples are available (30 hours of transcribed data from LRS3).

8.2 Applications

This thesis focuses on a diverse array of downstream tasks. Firstly, lip reading and speech recognition, which are proposed in Chapter 3 and 7 can, and indeed already have, a variety of real-world applications in commercial speech-to-text systems^{1,2}. On the medical front, speech recognition can help hearing impaired patients by transcribing speech in real time, while lip

¹<https://cloud.google.com/speech-to-text>

²<https://support.apple.com/en-gb/guide/iphone/iph2c0651d2/ios>

reading can help people with aphonia, i.e., who cannot produce voiced speech, by transcribing their speech based on their facial movements. Video-to-speech synthesis, as proposed in Chapters 4 and 5, can similarly be applied to help people with aphonia by producing audible speech directly from their silent lip movements alone. On the other hand, audio-visual speech enhancement systems, such as the one proposed in Chapter 6, can instead be applied for video conferencing, where we can remove acoustic noise and background speech based by leveraging the speaker’s video stream as additional supervision.

Finally, deepfake detection (presented in Chapter 7) can be applied to automatically detect manipulated visual speech without the need for human intervention. This can have a direct positive impact on modern media-sharing platforms by alleviating the potential social harm of misusing recent advances in facial animation. Furthermore, the self-supervised visual and acoustic models in Chapters 3 and 7 can be applied to a variety of new tasks, other than the ones proposed in their respective Chapters. These include emotion recognition, speaker recognition/identification, speech re-synthesis from self-supervised audio embeddings (Polyak *et al.*, 2021) and facial animation, among others.

8.3 Ethical Considerations

Many of the tasks discussed in this thesis, including ASR, VSR, video-to-speech synthesis, and audio-visual speech enhancement can have a very positive effect on broader society by improving communication systems, particularly by helping those who suffer from hearing impairments or aphonia. However, these can also be easily applied for surveillance, which is particularly problematic in VSR and video-to-speech since they can be applied for long-distance speech recognition, e.g., transcribing the speech of someone being recorded on CCTV. Another aggravating point is the fact that these methods are designed to be autonomous, meaning that they can be deployed and operated with little to no human supervision. This newfound automation, combined with recent developments that substantially improve the accuracy of state-of-the-art models, can enable automatic mass surveillance on an unprecedented scale.

As I mentioned in a recent interview with VICE³, there are two ways to prevent and limit the misuse of these technologies. Firstly, state-of-the-art research should be published in open-access platforms where researchers can be held accountable for their claims and applications. This allows us the public to discuss the impact of potential applications before these can be developed and leveraged by bad actors. Secondly, to prevent the misuse of these new technologies by powerful institutions such as governments or large corporations, it is important to consider them as political matters. Inevitably, powerful organizations will eventually have access to new models and the resources to deploy them. Therefore, our democracies must be robust enough to prevent them from undermining the general public's safety or privacy, e.g., by passing laws that limit the use of facial recognition in public spaces ⁴.

Another important ethical consideration should be made regarding the data that is used to train the models presented in this thesis, i.e., audio-visual speech. The datasets we use to train our models are either recorded in studios (GRID, TCD-TIMIT) or taken from existing audio-visual media, e.g., television broadcasts (LRW, LRS2), Ted Talks (LRS3) or YouTube (VoxCeleb2, AVSpeech). In all of these datasets, the aim is to collect as much audio-visual speech as possible, which means that there are limited mechanisms in place to, for example, ensure that there is a representative percentage of each ethnicity. In fact, none of these datasets present a detailed study of the ethnicity of its speakers. Therefore, it is highly likely that some ethnicities may be underrepresented, which will lead to models that are less accurate for these specific individuals. This kind of implicit discrimination caused by biased datasets should be prevented by designing inclusive datasets that are designed with these factors in mind. A good example is Casual Conversations (Hazirbas *et al.*, 2022), a recent dataset that is explicitly made to take into account age, gender, and apparent skin tone.

³<https://www.vice.com/en/article/bvzvbw/tech-companies-are-training-ai-to-read-your-lips>

⁴<https://www.politico.eu/article/europe-edges-closer-to-a-ban-on-facial-recognition/>

8.4 Future Work

Although many research directions can be pursued by developing the research questions posed in this thesis, two future research topics should be highlighted. The first is diffusion-based speech synthesis. As we demonstrate in Chapters 5 and 6, neural vocoders play a fundamental role in developing state-of-the-art video-to-speech and audio-visual speech enhancement models. Indeed, modern vocoders such as HiFi-GAN (Kong *et al.*, 2020a) considerably outperform Griffin-Lim (Griffin and Lim, 1983), are relatively lightweight and are substantially faster than real-time during inference. The vocoders used in this thesis (Parallel WaveGAN (Yamamoto *et al.*, 2019) and HiFi-GAN (Kong *et al.*, 2020a)), as well as the waveform decoder in Chapter 4, are based on GANs, which have dominated speech synthesis literature and have shown very impressive results. However, with the emergence of diffusion models in recent years (Ho *et al.*, 2020), the image synthesis state-of-the-art has shifted away from GANs and achieved novel, more impressive results by leveraging diffusion-based architectures (Dhariwal and Nichol, 2021). Although there has been some initial research into diffusion-based neural vocoders (Kong *et al.*, 2021; Koizumi *et al.*, 2022), we would like to encourage a more substantial shift towards this research direction, as this could lead to better vocoders that can improve the state-of-the-art for all synthesis-based tasks, including video-to-speech and audio-visual speech enhancement.

We also believe it would be promising to investigate new learning objectives for audio-visual self-supervised learning. In Chapter 3, we propose a self-supervised objective that can successfully leverage audio as an objective for visual speech encoders by using a pre-trained speech encoder. In Chapter 7, on the other hand, we propose two different frameworks with different (yet similar) objectives that allow acoustic and visual speech to learn from each other simultaneously. We believe these objectives were a fundamental step towards our state-of-the-art results on downstream results. However, as discussed in these chapters, there is a wide variety of self-supervised objectives proposed for image recognition, for example, that we did not experiment with (Zbontar *et al.*, 2021; He *et al.*, 2022). We believe exploring this topic more thoroughly within the audio-visual domain could lead to frameworks that are faster, more stable, require less tuning, and, most importantly, produce better downstream results. Therefore, we would

like to encourage new research that either applies existing self-supervised objectives to learn from audio-visual speech or designs new objectives that are tailored to these modalities.

BIBLIOGRAPHY

- [1] A. Polyak *et al.*, “Speech resynthesis from discrete disentangled self-supervised representations,” in *Interspeech 2021, 22nd Annual Conference of the International Speech Communication Association, Brno, Czechia, 30 August - 3 September 2021*, H. Hermansky, H. Cernocký, L. Burget, L. Lamel, O. Scharenborg, and P. Motlíček, Eds., ISCA, 2021, pp. 3615–3619. DOI: 10.21437/Interspeech.2021-475. [Online]. Available: <https://doi.org/10.21437/Interspeech.2021-475>.
- [2] K. Yang, D. Markovic, S. Krenn, V. Agrawal, and A. Richard, “Audio-visual speech codecs: Rethinking audio-visual speech enhancement by re-synthesis,” in *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, IEEE, 2022, pp. 8217–8227. DOI: 10.1109/CVPR52688.2022.00805. [Online]. Available: <https://doi.org/10.1109/CVPR52688.2022.00805>.
- [3] C. Ittichaichareon, S. Suksri, and T. Yingthawornsuk, “Speech recognition using mfcc,” in *International conference on computer graphics, simulation and modeling*, vol. 9, 2012.
- [4] S. M. Kamruzzaman, A. N. M. R. Karim, M. S. Islam, and M. E. Haque, “Speaker identification using mfcc-domain support vector machine,” *CoRR*, vol. abs/1009.4972, 2010. arXiv: 1009.4972. [Online]. Available: <http://arxiv.org/abs/1009.4972>.
- [5] M. Gurban and J. Thiran, “Information theoretic feature extraction for audio-visual speech recognition,” *IEEE Trans. Signal Process.*, vol. 57, no. 12, pp. 4765–4776, 2009.

- DOI: 10.1109/TSP.2009.2026513. [Online]. Available: <https://doi.org/10.1109/TSP.2009.2026513>.
- [6] G. Strang, “The discrete cosine transform,” *SIAM Rev.*, vol. 41, no. 1, pp. 135–147, 1999. DOI: 10.1137/S0036144598336745. [Online]. Available: <https://doi.org/10.1137/S0036144598336745>.
- [7] A. Tharwat, T. Gaber, A. Ibrahim, and A. E. Hassanien, “Linear discriminant analysis: A detailed tutorial,” *AI Commun.*, vol. 30, no. 2, pp. 169–190, 2017. DOI: 10.3233/AIC-170729. [Online]. Available: <https://doi.org/10.3233/AIC-170729>.
- [8] J. Shlens, “A tutorial on principal component analysis,” *CoRR*, vol. abs/1404.1100, 2014. arXiv: 1404.1100. [Online]. Available: <http://arxiv.org/abs/1404.1100>.
- [9] K. Kumar, T. Chen, and R. M. Stern, “Profile view lip reading,” in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2007, Honolulu, Hawaii, USA, April 15-20, 2007*, IEEE, 2007, pp. 429–432. DOI: 10.1109/ICASSP.2007.366941. [Online]. Available: <https://doi.org/10.1109/ICASSP.2007.366941>.
- [10] S. Pascual, M. Ravanelli, J. Serrà, A. Bonafonte, and Y. Bengio, “Learning problem-agnostic speech representations from multiple self-supervised tasks,” in *Interspeech 2019, 20th Annual Conference of the International Speech Communication Association, Graz, Austria, 15-19 September 2019*, G. Kubin and Z. Kacic, Eds., ISCA, 2019, pp. 161–165. DOI: 10.21437/Interspeech.2019-2605. [Online]. Available: <https://doi.org/10.21437/Interspeech.2019-2605>.
- [11] M. Ravanelli *et al.*, “Multi-task self-supervised learning for robust speech recognition,” in *2020 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2020, Barcelona, Spain, May 4-8, 2020*, IEEE, 2020, pp. 6989–6993. DOI: 10.1109/ICASSP40776.2020.9053569. [Online]. Available: <https://doi.org/10.1109/ICASSP40776.2020.9053569>.
- [12] A. Gulati *et al.*, “Conformer: Convolution-augmented transformer for speech recognition,” in *Interspeech 2020, 21st Annual Conference of the International Speech Commu-*

- nication Association, *Virtual Event, Shanghai, China, 25-29 October 2020*, H. Meng, B. Xu, and T. F. Zheng, Eds., ISCA, 2020, pp. 5036–5040. DOI: 10.21437/Interspeech.2020-3015. [Online]. Available: <https://doi.org/10.21437/Interspeech.2020-3015>.
- [13] J. Shen *et al.*, “Natural TTS synthesis by conditioning wavenet on MEL spectrogram predictions,” in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2018, Calgary, AB, Canada, April 15-20, 2018*, IEEE, 2018, pp. 4779–4783. DOI: 10.1109/ICASSP.2018.8461368. [Online]. Available: <https://doi.org/10.1109/ICASSP.2018.8461368>.
- [14] N. Zeghidour, N. Usunier, G. Synnaeve, R. Collobert, and E. Dupoux, “End-to-end speech recognition from the raw waveform,” in *Interspeech 2018, 19th Annual Conference of the International Speech Communication Association, Hyderabad, India, 2-6 September 2018*, B. Yegnanarayana, Ed., ISCA, 2018, pp. 781–785. DOI: 10.21437/Interspeech.2018-2414. [Online]. Available: <https://doi.org/10.21437/Interspeech.2018-2414>.
- [15] J. Donahue, S. Dieleman, M. Binkowski, E. Elsen, and K. Simonyan, “End-to-end adversarial text-to-speech,” in *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*, OpenReview.net, 2021. [Online]. Available: <https://openreview.net/forum?id=rsf1z-JSj87>.
- [16] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, IEEE Computer Society, 2016, pp. 770–778. DOI: 10.1109/CVPR.2016.90. [Online]. Available: <https://doi.org/10.1109/CVPR.2016.90>.
- [17] M. Tan and Q. V. Le, “Efficientnet: Rethinking model scaling for convolutional neural networks,” in *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, K. Chaudhuri and R. Salakhutdinov, Eds., ser. Proceedings of Machine Learning Research, vol. 97, PMLR,

- 2019, pp. 6105–6114. [Online]. Available: <http://proceedings.mlr.press/v97/tan19a.html>.
- [18] A. Dosovitskiy *et al.*, “An image is worth 16x16 words: Transformers for image recognition at scale,” in *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*, OpenReview.net, 2021. [Online]. Available: <https://openreview.net/forum?id=YicbFdNTTy>.
- [19] O. Russakovsky *et al.*, “Imagenet large scale visual recognition challenge,” *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, 2015. DOI: 10.1007/s11263-015-0816-y. [Online]. Available: <https://doi.org/10.1007/s11263-015-0816-y>.
- [20] T. Stafylakis and G. Tzimiropoulos, “Combining residual networks with lstms for lipreading,” in *Interspeech 2017, 18th Annual Conference of the International Speech Communication Association, Stockholm, Sweden, August 20-24, 2017*, F. Lacerda, Ed., ISCA, 2017, pp. 3652–3656. [Online]. Available: http://www.isca-speech.org/archive/Interspeech_2017/abstracts/0085.html.
- [21] T. Afouras, J. S. Chung, and A. Zisserman, “Deep lip reading: A comparison of models and an online application,” in *Interspeech 2018, 19th Annual Conference of the International Speech Communication Association, Hyderabad, India, 2-6 September 2018*, B. Yegnanarayana, Ed., ISCA, 2018, pp. 3514–3518. DOI: 10.21437/Interspeech.2018-1943. [Online]. Available: <https://doi.org/10.21437/Interspeech.2018-1943>.
- [22] T. Afouras, J. S. Chung, A. W. Senior, O. Vinyals, and A. Zisserman, “Deep audio-visual speech recognition,” *CoRR*, vol. abs/1809.02108, 2018. arXiv: 1809.02108. [Online]. Available: <http://arxiv.org/abs/1809.02108>.
- [23] T. Afouras, J. S. Chung, and A. Zisserman, “The conversation: Deep audio-visual speech enhancement,” in *Interspeech 2018, 19th Annual Conference of the International Speech Communication Association, Hyderabad, India, 2-6 September 2018*, B. Yegnanarayana, Ed., ISCA, 2018, pp. 3244–3248. DOI: 10.21437/Interspeech.2018-1400. [Online]. Available: <https://doi.org/10.21437/Interspeech.2018-1400>.

- [24] R. Mira, K. Vougioukas, P. Ma, S. Petridis, B. W. Schuller, and M. Pantic, “End-to-end video-to-speech synthesis using generative adversarial networks,” *IEEE Transactions on Cybernetics*, pp. 1–13, 2022. DOI: 10.1109/TCYB.2022.3162495.
- [25] J. Baker *et al.*, “Research developments and directions in speech recognition and understanding, part 1,” *IEEE Signal Processing Magazine*, vol. 26, no. 3, pp. 75–80, 2009. [Online]. Available: <https://www.microsoft.com/en-us/research/publication/research-developments-and-directions-in-speech-recognition-and-understanding-part-1/>.
- [26] A. Kumar, S. Verma, and H. Mangla, “A survey of deep learning techniques in speech recognition,” in *2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN)*, 2018, pp. 179–185. DOI: 10.1109/ICACCCN.2018.8748399.
- [27] L. Rabiner and B. Juang, “An introduction to hidden markov models,” *IEEE ASSP Magazine*, vol. 3, no. 1, pp. 4–16, 1986. DOI: 10.1109/MASSP.1986.1165342.
- [28] D. Reynolds, “Gaussian mixture models,” *Encyclopedia of Biometrics*, Jan. 2008. DOI: 10.1007/978-0-387-73003-5_196.
- [29] T. K. Moon, “The expectation-maximization algorithm,” *IEEE Signal processing magazine*, vol. 13, no. 6, pp. 47–60, 1996.
- [30] L. E. Baum, T. Petrie, G. Soules, and N. Weiss, “A maximization technique occurring in the statistical analysis of probabilistic functions of markov chains,” *The Annals of Mathematical Statistics*, vol. 41, no. 1, pp. 164–171, 1970, ISSN: 00034851. [Online]. Available: <http://www.jstor.org/stable/2239727> (visited on 10/25/2022).
- [31] S. Fenghour, D. Chen, K. Guo, B. Li, and P. Xiao, “Deep learning-based automated lip-reading: A survey,” *IEEE Access*, vol. 9, pp. 121 184–121 205, 2021. DOI: 10.1109/ACCESS.2021.3107946. [Online]. Available: <https://doi.org/10.1109/ACCESS.2021.3107946>.

- [32] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, “Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks,” in *Proceedings of the 23rd international conference on Machine learning*, 2006, pp. 369–376.
- [33] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, and D. S. Pallett, “Darpa timit acoustic-phonetic continuous speech corpus cd-rom. nist speech disc 1-1.1,” *NASA STI/Recon technical report n*, vol. 93, p. 27 403, 1993.
- [34] J. J. Godfrey, E. Holliman, and J. McDaniel, “SWITCHBOARD: telephone speech corpus for research and development,” in *1992 IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP '92, San Francisco, California, USA, March 23-26, 1992*, IEEE Computer Society, 1992, pp. 517–520. DOI: 10.1109/ICASSP.1992.225858. [Online]. Available: <https://doi.org/10.1109/ICASSP.1992.225858>.
- [35] Z. Tüske, G. Saon, and B. Kingsbury, “On the limit of english conversational speech recognition,” in *Interspeech 2021, 22nd Annual Conference of the International Speech Communication Association, Brno, Czechia, 30 August - 3 September 2021*, H. Hermansky, H. Cernocký, L. Burget, L. Lamel, O. Scharenborg, and P. Motlíček, Eds., ISCA, 2021, pp. 2062–2066. DOI: 10.21437/Interspeech.2021-211. [Online]. Available: <https://doi.org/10.21437/Interspeech.2021-211>.
- [36] M. B. Hoy, “Alexa, siri, cortana, and more: An introduction to voice assistants,” *Medical reference services quarterly*, vol. 37, no. 1, pp. 81–88, 2018.
- [37] Y. M. Assael, B. Shillingford, S. Whiteson, and N. de Freitas, “Lipnet: Sentence-level lipreading,” *CoRR*, vol. abs/1611.01599, 2016. arXiv: 1611.01599. [Online]. Available: <http://arxiv.org/abs/1611.01599>.
- [38] M. Cooke, J. Barker, S. Cunningham, and X. Shao, “An audio-visual corpus for speech perception and automatic speech recognition,” *The Journal of the Acoustical Society of America*, vol. 120, no. 5, pp. 2421–2424, 2006.
- [39] T. Afouras, J. S. Chung, A. Senior, O. Vinyals, and A. Zisserman, “Deep audio-visual speech recognition,” in *arXiv:1809.02108*, 2018.

- [40] T. Afouras, J. S. Chung, and A. Zisserman, “Lrs3-ted: A large-scale dataset for visual speech recognition,” in *arXiv preprint arXiv:1809.00496*, 2018.
- [41] P. Ma, R. Mira, S. Petridis, B. W. Schuller, and M. Pantic, “Lira: Learning visual speech representations from audio through self-supervision,” in *Interspeech 2021, 22nd Annual Conference of the International Speech Communication Association, Brno, Czechia, 30 August - 3 September 2021*, H. Hermansky, H. Cernocký, L. Burget, L. Lamel, O. Scharenborg, and P. Motlíček, Eds., ISCA, 2021, pp. 3011–3015. DOI: 10.21437/Interspeech.2021-1360. [Online]. Available: <https://doi.org/10.21437/Interspeech.2021-1360>.
- [42] Y. Tabet and M. Boughazi, “Speech synthesis techniques. a survey,” in *International Workshop on Systems, Signal Processing and their Applications, WOSSPA*, 2011, pp. 67–70. DOI: 10.1109/WOSSPA.2011.5931414.
- [43] X. Tan, T. Qin, F. K. Soong, and T. Liu, “A survey on neural speech synthesis,” *CoRR*, vol. abs/2106.15561, 2021. arXiv: 2106.15561. [Online]. Available: <https://arxiv.org/abs/2106.15561>.
- [44] A. Simons and S. Cox, “Generation of mouthshapes for a synthetic talking head,” *Proceedings of the Institute of Acoustics, Autumn Meeting*, Jan. 1990.
- [45] E. Yamamoto, S. Nakamura, and K. Shikano, “Lip movement synthesis from speech based on hidden markov models,” *Speech Commun.*, vol. 26, no. 1-2, pp. 105–115, 1998. DOI: 10.1016/S0167-6393(98)00054-5. [Online]. Available: [https://doi.org/10.1016/S0167-6393\(98\)00054-5](https://doi.org/10.1016/S0167-6393(98)00054-5).
- [46] A. van den Oord *et al.*, “Wavenet: A generative model for raw audio,” in *The 9th ISCA Speech Synthesis Workshop, Sunnyvale, CA, USA, 13-15 September 2016*, ISCA, 2016, p. 125. [Online]. Available: http://www.isca-speech.org/archive/SSW_2016/abstracts/ssw9_DS-4_van_den_Oord.html.
- [47] C. Donahue, J. J. McAuley, and M. S. Puckette, “Adversarial audio synthesis,” in *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA*,

- USA, May 6-9, 2019, OpenReview.net, 2019. [Online]. Available: <https://openreview.net/forum?id=ByMVTsR5KQ>.
- [48] R. Yamamoto, E. Song, and J. Kim, “Parallel wavegan: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram,” *CoRR*, vol. abs/1910.11480, pp. 6199–6203, 2019. DOI: 10.1109/ICASSP40776.2020.9053795. arXiv: 1910.11480. [Online]. Available: <http://arxiv.org/abs/1910.11480>.
- [49] J. Kong, J. Kim, and J. Bae, “Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis,” in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., 2020. [Online]. Available: <https://proceedings.neurips.cc/paper/2020/hash/c5d736809766d46260d816d8dbc9eb44-Abstract.html>.
- [50] K. Vougioukas, S. Petridis, and M. Pantic, “End-to-end speech-driven facial animation with temporal gans,” in *British Machine Vision Conference 2018, BMVC 2018, Newcastle, UK, September 3-6, 2018*, BMVA Press, 2018, p. 133. [Online]. Available: <http://bmvc2018.org/contents/papers/0539.pdf>.
- [51] D. Das, S. Biswas, S. Sinha, and B. Bhowmick, “Speech-driven facial animation using cascaded gans for learning of motion and texture,” in *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XXX*, A. Vedaldi, H. Bischof, T. Brox, and J. Frahm, Eds., ser. Lecture Notes in Computer Science, vol. 12375, Springer, 2020, pp. 408–424. DOI: 10.1007/978-3-030-58577-8_25. [Online]. Available: https://doi.org/10.1007/978-3-030-58577-8_25.
- [52] K. Vougioukas, S. Petridis, and M. Pantic, “DINO: A conditional energy-based GAN for domain translation,” in *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*, OpenReview.net, 2021. [Online]. Available: <https://openreview.net/forum?id=WAlSmwsqDsb>.
- [53] I. J. Goodfellow *et al.*, “Generative adversarial networks,” *CoRR*, vol. abs/1406.2661, 2014. arXiv: 1406.2661. [Online]. Available: <http://arxiv.org/abs/1406.2661>.

- [54] S. Petridis, T. Stafylakis, P. Ma, F. Cai, G. Tzimiropoulos, and M. Pantic, “End-to-end audiovisual speech recognition,” in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2018, Calgary, AB, Canada, April 15-20, 2018*, IEEE, 2018, pp. 6548–6552. DOI: 10.1109/ICASSP.2018.8461326. [Online]. Available: <https://doi.org/10.1109/ICASSP.2018.8461326>.
- [55] P. Ma, S. Petridis, and M. Pantic, “End-to-end audio-visual speech recognition with conformers,” in *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2021, Toronto, ON, Canada, June 6-11, 2021*, IEEE, 2021, pp. 7613–7617. DOI: 10.1109/ICASSP39728.2021.9414567. [Online]. Available: <https://doi.org/10.1109/ICASSP39728.2021.9414567>.
- [56] J. Hou *et al.*, “Audio-visual speech enhancement using deep neural networks,” in *Asia-Pacific Signal and Information Processing Association Annual Summit and Conference, APSIPA 2016, Jeju, South Korea, December 13-16, 2016*, IEEE, 2016, pp. 1–6. DOI: 10.1109/APSIPA.2016.7820732. [Online]. Available: <https://doi.org/10.1109/APSIPA.2016.7820732>.
- [57] M. Gogate, K. Dashtipour, and A. Hussain, “Towards robust real-time audio-visual speech enhancement,” *CoRR*, vol. abs/2112.09060, 2021. arXiv: 2112.09060. [Online]. Available: <https://arxiv.org/abs/2112.09060>.
- [58] A. Ephrat *et al.*, “Looking to listen at the cocktail party: A speaker-independent audio-visual model for speech separation,” *ACM Trans. Graph.*, vol. 37, no. 4, p. 112, 2018. DOI: 10.1145/3197517.3201357. [Online]. Available: <https://doi.org/10.1145/3197517.3201357>.
- [59] R. Gao and K. Grauman, “Visualvoice: Audio-visual speech separation with cross-modal consistency,” in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, Computer Vision Foundation / IEEE, 2021, pp. 15 495–15 505. DOI: 10.1109/CVPR46437.2021.01524. [Online]. Available: https://openaccess.thecvf.com/content/CVPR2021/html/Gao_VisualVoice_Audio-

- Visual_Speech_Separation_With_Cross-Modal_Consistency_CVPR_2021_paper.html.
- [60] J. Devlin, M. Chang, K. Lee, and K. Toutanova, “BERT: pre-training of deep bidirectional transformers for language understanding,” in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, J. Burstein, C. Doran, and T. Solorio, Eds., Association for Computational Linguistics, 2019, pp. 4171–4186. DOI: 10.18653/v1/n19-1423. [Online]. Available: <https://doi.org/10.18653/v1/n19-1423>.
 - [61] T. B. Brown *et al.*, “Language models are few-shot learners,” in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., 2020. [Online]. Available: <https://proceedings.neurips.cc/paper/2020/hash/1457c0d6bfcb4967418bfb8ac142f64a-Abstract.html>.
 - [62] T. Chen, S. Kornblith, M. Norouzi, and G. E. Hinton, “A simple framework for contrastive learning of visual representations,” in *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, ser. Proceedings of Machine Learning Research, vol. 119, PMLR, 2020, pp. 1597–1607. [Online]. Available: <http://proceedings.mlr.press/v119/chen20j.html>.
 - [63] K. He, H. Fan, Y. Wu, S. Xie, and R. B. Girshick, “Momentum contrast for unsupervised visual representation learning,” in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, Computer Vision Foundation / IEEE, 2020, pp. 9726–9735. DOI: 10.1109/CVPR42600.2020.00975. [Online]. Available: <https://doi.org/10.1109/CVPR42600.2020.00975>.
 - [64] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, “Wav2vec 2.0: A framework for self-supervised learning of speech representations,” in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020*,

- NeurIPS 2020, December 6-12, 2020, virtual*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., 2020. [Online]. Available: <https://proceedings.neurips.cc/paper/2020/hash/92d1e1eb1cd6f9fba3227870bb6d7f07-Abstract.html>.
- [65] W. Hsu, B. Bolte, Y. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, “Hubert: Self-supervised speech representation learning by masked prediction of hidden units,” *IEEE ACM Trans. Audio Speech Lang. Process.*, vol. 29, pp. 3451–3460, 2021. DOI: 10.1109/TASLP.2021.3122291. [Online]. Available: <https://doi.org/10.1109/TASLP.2021.3122291>.
- [66] J. Grill *et al.*, “Bootstrap your own latent - A new approach to self-supervised learning,” in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., 2020. [Online]. Available: <https://proceedings.neurips.cc/paper/2020/hash/f3ada80d5c4ee70142b17b8192b2958-Abstract.html>.
- [67] J. Zbontar, L. Jing, I. Misra, Y. LeCun, and S. Deny, “Barlow twins: Self-supervised learning via redundancy reduction,” in *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, M. Meila and T. Zhang, Eds., ser. Proceedings of Machine Learning Research, vol. 139, PMLR, 2021, pp. 12 310–12 320. [Online]. Available: <http://proceedings.mlr.press/v139/zbontar21a.html>.
- [68] L. Schoneveld, A. Othmani, and H. Abdelkawy, “Leveraging recent advances in deep learning for audio-visual emotion recognition,” *Pattern Recognit. Lett.*, vol. 146, pp. 1–7, 2021. DOI: 10.1016/j.patrec.2021.03.007. [Online]. Available: <https://doi.org/10.1016/j.patrec.2021.03.007>.
- [69] J. S. Chung and A. Zisserman, “Lip reading in the wild,” in *Asian Conference on Computer Vision*, 2016.
- [70] R. S. C. de Mira, A. Haliassos, S. Petridis, B. W. Schuller, and M. Pantic, “SVTS: scalable video-to-speech synthesis,” in *Interspeech 2022, 23rd Annual Conference of*

- the International Speech Communication Association, Incheon, Korea, 18-22 September 2022*, H. Ko and J. H. L. Hansen, Eds., ISCA, 2022, pp. 1836–1840. DOI: 10.21437/Interspeech.2022-10770. [Online]. Available: <https://doi.org/10.21437/Interspeech.2022-10770>.
- [71] Y. Jia *et al.*, “Transfer learning from speaker verification to multispeaker text-to-speech synthesis,” in *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, S. Bengio, H. M. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds., 2018, pp. 4485–4495. [Online]. Available: <https://proceedings.neurips.cc/paper/2018/hash/6832a7b24bc06775d02b7406880b93fc-Abstract.html>.
- [72] R. Yamamoto, E. Song, and J. Kim, “Parallel wavegan: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram,” in *2020 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2020, Barcelona, Spain, May 4-8, 2020*, IEEE, 2020, pp. 6199–6203. DOI: 10.1109/ICASSP40776.2020.9053795. [Online]. Available: <https://doi.org/10.1109/ICASSP40776.2020.9053795>.
- [73] D. W. Griffin and J. S. Lim, “Signal estimation from modified short-time fourier transform,” in *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP ’83, Boston, Massachusetts, USA, April 14-16, 1983*, IEEE, 1983, pp. 804–807. DOI: 10.1109/ICASSP.1983.1172092. [Online]. Available: <https://doi.org/10.1109/ICASSP.1983.1172092>.
- [74] R. Mira *et al.*, “La-voce: Low-snr audio-visual speech enhancement using neural vocoders,” in *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023, pp. 1–5. DOI: 10.1109/ICASSP49357.2023.10096390.
- [75] A. Haliassos, R. Mira, S. Petridis, and M. Pantic, “Leveraging real talking faces via self-supervision for robust forgery detection,” in *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24,*

- 2022, IEEE, 2022, pp. 14 930–14 942. DOI: 10.1109/CVPR52688.2022.01453. [Online]. Available: <https://doi.org/10.1109/CVPR52688.2022.01453>.
- [76] A. Haliassos, P. Ma, R. Mira, S. Petridis, and M. Pantic, “Jointly learning visual and auditory speech representations from raw data,” in *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*, OpenReview.net, 2023. [Online]. Available: <https://openreview.net/pdf?id=BPwIgvf5iQ>.
- [77] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. B. Girshick, “Masked autoencoders are scalable vision learners,” in *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, IEEE, 2022, pp. 15 979–15 988. DOI: 10.1109/CVPR52688.2022.01553. [Online]. Available: <https://doi.org/10.1109/CVPR52688.2022.01553>.
- [78] N. Harte and E. Gillen, “TCD-TIMIT: an audio-visual corpus of continuous speech,” *IEEE Trans. Multim.*, vol. 17, no. 5, pp. 603–615, 2015. DOI: 10.1109/TMM.2015.2407694. [Online]. Available: <https://doi.org/10.1109/TMM.2015.2407694>.
- [79] J. S. Chung, A. Nagrani, and A. Zisserman, “Voxceleb2: Deep speaker recognition,” in *Interspeech 2018, 19th Annual Conference of the International Speech Communication Association, Hyderabad, India, 2-6 September 2018*, B. Yegnanarayana, Ed., ISCA, 2018, pp. 1086–1090. DOI: 10.21437/Interspeech.2018-1929. [Online]. Available: <https://doi.org/10.21437/Interspeech.2018-1929>.
- [80] B. Gold, N. Morgan, and D. Ellis, *Speech and Audio Signal Processing - Processing and Perception of Speech and Music, Second Edition*. Wiley, 2011, ISBN: 978-0-470-19536-9. [Online]. Available: <http://eu.wiley.com/WileyCDA/WileyTitle/productCd-0470195363.html>.
- [81] X. Huang, A. Acero, and H.-W. Hon, “Spoken language processing: A guide to theory, algorithm, and system development,” Jan. 2001.
- [82] A. Paszke *et al.*, “Automatic differentiation in pytorch,” 2017.
- [83] B. McFee *et al.*, *Librosa/librosa: 0.9.2*, version 0.9.2, Jun. 2022. DOI: 10.5281/zenodo.6759664. [Online]. Available: <https://doi.org/10.5281/zenodo.6759664>.

- [84] S. Davis and P. Mermelstein, “Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 28, no. 4, pp. 357–366, 1980. DOI: 10.1109/TASSP.1980.1163420.
- [85] K. K. Kishore and P. K. Satish, “Emotion recognition in speech using mfcc and wavelet features,” in *2013 3rd IEEE International Advance Computing Conference (IACC)*, IEEE, 2013, pp. 842–847.
- [86] D. E. King, “Dlib-ml: A machine learning toolkit,” *Journal of Machine Learning Research*, vol. 10, pp. 1755–1758, 2009.
- [87] J. Deng, J. Guo, E. Ververas, I. Kotsia, and S. Zafeiriou, “Retinaface: Single-shot multi-level face localisation in the wild,” in *CVPR*, 2020. DOI: 10.1109/CVPR42600.2020.00525. [Online]. Available: https://openaccess.thecvf.com/content/_CVPR/_2020/html/Deng_RetinaFace_Single-Shot_Multi-Level_Face_Localisation_in_the_Wild_CVPR_2020_paper.html.
- [88] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05)*, vol. 1, 2005, 886–893 vol. 1. DOI: 10.1109/CVPR.2005.177.
- [89] M. Hearst, S. Dumais, E. Osuna, J. Platt, and B. Scholkopf, “Support vector machines,” *IEEE Intelligent Systems and their Applications*, vol. 13, no. 4, pp. 18–28, 1998. DOI: 10.1109/5254.708428.
- [90] S. Yang, P. Luo, C. C. Loy, and X. Tang, “Wider face: A face detection benchmark,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [91] A. Bulat and G. Tzimiropoulos, “How far are we from solving the 2d & 3d face alignment problem? (and a dataset of 230, 000 3d facial landmarks),” in *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, IEEE Computer Society, 2017, pp. 1021–1030. DOI: 10.1109/ICCV.2017.116. [Online]. Available: <https://doi.org/10.1109/ICCV.2017.116>.

- [92] V. Kazemi and J. Sullivan, “One millisecond face alignment with an ensemble of regression trees,” in *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014*, IEEE Computer Society, 2014, pp. 1867–1874. DOI: 10.1109/CVPR.2014.241. [Online]. Available: <https://doi.org/10.1109/CVPR.2014.241>.
- [93] A. Newell, K. Yang, and J. Deng, “Stacked hourglass networks for human pose estimation,” in *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VIII*, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds., ser. Lecture Notes in Computer Science, vol. 9912, Springer, 2016, pp. 483–499. DOI: 10.1007/978-3-319-46484-8_29. [Online]. Available: https://doi.org/10.1007/978-3-319-46484-8_29.
- [94] X. Zhu, Z. Lei, X. Liu, H. Shi, and S. Z. Li, “Face alignment across large poses: A 3d solution,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, IEEE Computer Society, 2016, pp. 146–155. DOI: 10.1109/CVPR.2016.23. [Online]. Available: <https://doi.org/10.1109/CVPR.2016.23>.
- [95] J. Heaton, “Ian goodfellow, yoshua bengio, and aaron courville: Deep learning - the MIT press, 2016, 800 pp, ISBN: 0262035618,” *Genet. Program. Evolvable Mach.*, vol. 19, no. 1-2, pp. 305–307, 2018. DOI: 10.1007/s10710-017-9314-z. [Online]. Available: <https://doi.org/10.1007/s10710-017-9314-z>.
- [96] Y. LeCun *et al.*, “Backpropagation applied to handwritten zip code recognition,” *Neural Comput.*, vol. 1, no. 4, pp. 541–551, 1989. DOI: 10.1162/neco.1989.1.4.541. [Online]. Available: <https://doi.org/10.1162/neco.1989.1.4.541>.
- [97] V. Nair and G. E. Hinton, “Rectified linear units improve restricted boltzmann machines,” in *Proceedings of the 27th International Conference on Machine Learning (ICML-10), June 21-24, 2010, Haifa, Israel*, J. Fürnkranz and T. Joachims, Eds., Omnipress, 2010, pp. 807–814. [Online]. Available: <https://icml.cc/Conferences/2010/papers/432.pdf>.

- [98] P. Ramachandran, B. Zoph, and Q. V. Le, “Searching for activation functions,” in *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Workshop Track Proceedings*, OpenReview.net, 2018. [Online]. Available: <https://openreview.net/forum?id=Hkuq2EkPf>.
- [99] S. Ioffe and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” in *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, F. R. Bach and D. M. Blei, Eds., ser. JMLR Workshop and Conference Proceedings, vol. 37, JMLR.org, 2015, pp. 448–456. [Online]. Available: <http://proceedings.mlr.press/v37/ioffe15.html>.
- [100] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” *Commun. ACM*, vol. 60, no. 6, pp. 84–90, 2017. DOI: 10.1145/3065386. [Online]. Available: <http://doi.acm.org/10.1145/3065386>.
- [101] C. Szegedy *et al.*, “Going deeper with convolutions,” in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, IEEE Computer Society, 2015, pp. 1–9. DOI: 10.1109/CVPR.2015.7298594. [Online]. Available: <https://doi.org/10.1109/CVPR.2015.7298594>.
- [102] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015. [Online]. Available: <http://arxiv.org/abs/1409.1556>.
- [103] H. Li, Z. Xu, G. Taylor, C. Studer, and T. Goldstein, “Visualizing the loss landscape of neural nets,” in *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, S. Bengio, H. M. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds., 2018, pp. 6391–6401. [Online]. Available: <https://proceedings.neurips.cc/paper/2018/hash/a41b3bb3e6b050b6c9067c67f663b915-Abstract.html>.

- [104] A. Veit, M. J. Wilber, and S. J. Belongie, “Residual networks behave like ensembles of relatively shallow networks,” in *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, D. D. Lee, M. Sugiyama, U. von Luxburg, I. Guyon, and R. Garnett, Eds., 2016, pp. 550–558. [Online]. Available: <https://proceedings.neurips.cc/paper/2016/hash/37bc2f75bf1bcfe8450a1a41c200364c-Abstract.html>.
- [105] G. Saon *et al.*, “English conversational telephone speech recognition by humans and machines,” in *Interspeech 2017, 18th Annual Conference of the International Speech Communication Association, Stockholm, Sweden, August 20-24, 2017*, F. Lacerda, Ed., ISCA, 2017, pp. 132–136. [Online]. Available: http://www.isca-speech.org/archive/Interspeech_2017/abstracts/0405.html.
- [106] A. Kolesnikov *et al.*, “Big transfer (bit): General visual representation learning,” in *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part V*, A. Vedaldi, H. Bischof, T. Brox, and J. Frahm, Eds., ser. Lecture Notes in Computer Science, vol. 12350, Springer, 2020, pp. 491–507. DOI: 10.1007/978-3-030-58558-7_29. [Online]. Available: https://doi.org/10.1007/978-3-030-58558-7_29.
- [107] S. Xie, R. B. Girshick, P. Dollár, Z. Tu, and K. He, “Aggregated residual transformations for deep neural networks,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, IEEE Computer Society, 2017, pp. 5987–5995. DOI: 10.1109/CVPR.2017.634. [Online]. Available: <https://doi.org/10.1109/CVPR.2017.634>.
- [108] Z. Liu *et al.*, “Swin transformer V2: scaling up capacity and resolution,” in *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, IEEE, 2022, pp. 11 999–12 009. DOI: 10.1109/CVPR52688.2022.01170. [Online]. Available: <https://doi.org/10.1109/CVPR52688.2022.01170>.
- [109] P. Ma, B. Martínez, S. Petridis, and M. Pantic, “Towards practical lipreading with distilled and efficient models,” in *IEEE International Conference on Acoustics, Speech*

- and *Signal Processing, ICASSP 2021, Toronto, ON, Canada, June 6-11, 2021*, IEEE, 2021, pp. 7608–7612. DOI: 10.1109/ICASSP39728.2021.9415063. [Online]. Available: <https://doi.org/10.1109/ICASSP39728.2021.9415063>.
- [110] P. Ma, Y. Wang, S. Petridis, J. Shen, and M. Pantic, “Training strategies for improved lip-reading,” in *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2022, Virtual and Singapore, 23-27 May 2022*, IEEE, 2022, pp. 8472–8476. DOI: 10.1109/ICASSP43922.2022.9746706. [Online]. Available: <https://doi.org/10.1109/ICASSP43922.2022.9746706>.
- [111] J. J. Hopfield, “Neural networks and physical systems with emergent collective computational abilities,” *Proceedings of the National Academy of Sciences*, vol. 79, no. 8, pp. 2554–2558, 1982. DOI: 10.1073/pnas.79.8.2554. eprint: <https://www.pnas.org/doi/pdf/10.1073/pnas.79.8.2554>. [Online]. Available: <https://www.pnas.org/doi/abs/10.1073/pnas.79.8.2554>.
- [112] Z. C. Lipton, “A critical review of recurrent neural networks for sequence learning,” *CoRR*, vol. abs/1506.00019, 2015. arXiv: 1506.00019. [Online]. Available: <http://arxiv.org/abs/1506.00019>.
- [113] J. F. Kolen and S. C. Kremer, “Gradient flow in recurrent nets: The difficulty of learning longterm dependencies,” in *A Field Guide to Dynamical Recurrent Networks*. 2001, pp. 237–243. DOI: 10.1109/9780470544037.ch14.
- [114] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997. DOI: 10.1162/neco.1997.9.8.1735. [Online]. Available: <https://doi.org/10.1162/neco.1997.9.8.1735>.
- [115] F. A. Gers, J. Schmidhuber, and F. A. Cummins, “Learning to forget: Continual prediction with LSTM,” *Neural Comput.*, vol. 12, no. 10, pp. 2451–2471, 2000. DOI: 10.1162/089976600300015015. [Online]. Available: <https://doi.org/10.1162/089976600300015015>.
- [116] D. W. Otter, J. R. Medina, and J. K. Kalita, “A survey of the usages of deep learning for natural language processing,” *IEEE Trans. Neural Networks Learn. Syst.*, vol. 32,

- no. 2, pp. 604–624, 2021. DOI: 10.1109/TNNLS.2020.2979670. [Online]. Available: <https://doi.org/10.1109/TNNLS.2020.2979670>.
- [117] A. Graves, A. Mohamed, and G. E. Hinton, “Speech recognition with deep recurrent neural networks,” in *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2013, Vancouver, BC, Canada, May 26-31, 2013*, IEEE, 2013, pp. 6645–6649. DOI: 10.1109/ICASSP.2013.6638947. [Online]. Available: <https://doi.org/10.1109/ICASSP.2013.6638947>.
- [118] M. Schuster and K. K. Paliwal, “Bidirectional recurrent neural networks,” *IEEE Trans. Signal Process.*, vol. 45, no. 11, pp. 2673–2681, 1997. DOI: 10.1109/78.650093. [Online]. Available: <https://doi.org/10.1109/78.650093>.
- [119] K. Cho, B. van Merriënboer, D. Bahdanau, and Y. Bengio, “On the properties of neural machine translation: Encoder-decoder approaches,” in *Proceedings of SSST@EMNLP 2014, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation, Doha, Qatar, 25 October 2014*, D. Wu, M. Carpuat, X. Carreras, and E. M. Vecchi, Eds., Association for Computational Linguistics, 2014, pp. 103–111. DOI: 10.3115/v1/W14-4012. [Online]. Available: <https://aclanthology.org/W14-4012/>.
- [120] J. Chung, Ç. Gülçehre, K. Cho, and Y. Bengio, “Empirical evaluation of gated recurrent neural networks on sequence modeling,” *CoRR*, vol. abs/1412.3555, 2014. arXiv: 1412.3555. [Online]. Available: <http://arxiv.org/abs/1412.3555>.
- [121] P. Le and W. H. Zuidema, “Quantifying the vanishing gradient and long distance dependency problem in recursive neural networks and recursive lstms,” in *Proceedings of the 1st Workshop on Representation Learning for NLP, Rep4NLP@ACL 2016, Berlin, Germany, August 11, 2016*, P. Blunsom *et al.*, Eds., Association for Computational Linguistics, 2016, pp. 87–93. DOI: 10.18653/v1/W16-1610. [Online]. Available: <https://doi.org/10.18653/v1/W16-1610>.
- [122] J. Zhao *et al.*, “Do RNN and LSTM have long memory?” In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual*

- Event*, ser. Proceedings of Machine Learning Research, vol. 119, PMLR, 2020, pp. 11 365–11 375. [Online]. Available: <http://proceedings.mlr.press/v119/zhao20c.html>.
- [123] A. Vaswani *et al.*, “Attention is all you need,” in *NeurIPS*, 2017, pp. 5998–6008. [Online]. Available: <https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4Abstract.html>.
- [124] D. Bahdanau, K. Cho, and Y. Bengio, “Neural machine translation by jointly learning to align and translate,” in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015. [Online]. Available: <http://arxiv.org/abs/1409.0473>.
- [125] Z. Niu, G. Zhong, and H. Yu, “A review on the attention mechanism of deep learning,” *Neurocomputing*, vol. 452, pp. 48–62, 2021. DOI: 10.1016/j.neucom.2021.03.091. [Online]. Available: <https://doi.org/10.1016/j.neucom.2021.03.091>.
- [126] Z. Dai, Z. Yang, Y. Yang, J. G. Carbonell, Q. V. Le, and R. Salakhutdinov, “Transformer-xl: Attentive language models beyond a fixed-length context,” in *ACL*, Association for Computational Linguistics, 2019, pp. 2978–2988. DOI: 10.18653/v1/p19-1285. [Online]. Available: <https://doi.org/10.18653/v1/p19-1285>.
- [127] J. Yu, Z. Wang, V. Vasudevan, L. Yeung, M. Seyedhosseini, and Y. Wu, “Coca: Contrastive captioners are image-text foundation models,” *CoRR*, vol. abs/2205.01917, 2022. DOI: 10.48550/arXiv.2205.01917. arXiv: 2205.01917. [Online]. Available: <https://doi.org/10.48550/arXiv.2205.01917>.
- [128] X. Zhai, A. Kolesnikov, N. Houlsby, and L. Beyer, “Scaling vision transformers,” in *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, IEEE, 2022, pp. 1204–1213. DOI: 10.1109/CVPR52688.2022.01179. [Online]. Available: <https://doi.org/10.1109/CVPR52688.2022.01179>.
- [129] Z. Tong, Y. Song, J. Wang, and L. Wang, “Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training,” *CoRR*, vol. abs/2203.12602,

2022. DOI: 10.48550/arXiv.2203.12602. arXiv: 2203.12602. [Online]. Available: <https://doi.org/10.48550/arXiv.2203.12602>.
- [130] B. Shi, W. Hsu, K. Lakhotia, and A. Mohamed, “Learning audio-visual speech representation by masked multimodal cluster prediction,” *CoRR*, vol. abs/2201.02184, 2022. arXiv: 2201.02184. [Online]. Available: <https://arxiv.org/abs/2201.02184>.
- [131] Q. Fournier, G. M. Caron, and D. Aloise, “A practical survey on faster and lighter transformers,” *CoRR*, vol. abs/2103.14636, 2021. arXiv: 2103.14636. [Online]. Available: <https://arxiv.org/abs/2103.14636>.
- [132] N. Kitaev, L. Kaiser, and A. Levskaya, “Reformer: The efficient transformer,” in *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*, OpenReview.net, 2020. [Online]. Available: <https://openreview.net/forum?id=rkgNKkHtvB>.
- [133] S. Wang, B. Z. Li, M. Khabsa, H. Fang, and H. Ma, “Linformer: Self-attention with linear complexity,” *CoRR*, vol. abs/2006.04768, 2020. arXiv: 2006.04768. [Online]. Available: <https://arxiv.org/abs/2006.04768>.
- [134] K. M. Choromanski *et al.*, “Rethinking attention with performers,” in *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*, OpenReview.net, 2021. [Online]. Available: <https://openreview.net/forum?id=Ua6zuk0WRH>.
- [135] Y. Lu *et al.*, “Understanding and improving transformer from a multi-particle dynamic system point of view,” *CoRR*, vol. abs/1906.02762, 2019. arXiv: 1906.02762. [Online]. Available: <http://arxiv.org/abs/1906.02762>.
- [136] F. Chollet, “Xception: Deep learning with depthwise separable convolutions,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, IEEE Computer Society, 2017, pp. 1800–1807. DOI: 10.1109/CVPR.2017.195. [Online]. Available: <https://doi.org/10.1109/CVPR.2017.195>.

- [137] Q. Zhang *et al.*, “Transformer transducer: A streamable speech recognition model with transformer encoders and RNN-T loss,” in *2020 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2020, Barcelona, Spain, May 4-8, 2020*, IEEE, 2020, pp. 7829–7833. DOI: 10.1109/ICASSP40776.2020.9053896. [Online]. Available: <https://doi.org/10.1109/ICASSP40776.2020.9053896>.
- [138] Y. Zhang *et al.*, “Pushing the limits of semi-supervised learning for automatic speech recognition,” *CoRR*, vol. abs/2010.10504, 2020. arXiv: 2010.10504. [Online]. Available: <https://arxiv.org/abs/2010.10504>.
- [139] W. Chan, D. S. Park, C. A. Lee, Y. Zhang, Q. V. Le, and M. Norouzi, “Speechstew: Simply mix all available speech recognition data to train one large neural network,” *CoRR*, vol. abs/2104.02133, 2021. arXiv: 2104.02133. [Online]. Available: <https://arxiv.org/abs/2104.02133>.
- [140] P. Ma, S. Petridis, and M. Pantic, “Visual speech recognition for multiple languages in the wild,” *CoRR*, vol. abs/2202.13084, 2022. arXiv: 2202.13084. [Online]. Available: <https://arxiv.org/abs/2202.13084>.
- [141] ———, “End-to-end audio-visual speech recognition with conformers,” in *ICASSP*, 2021, pp. 7613–7617. DOI: 10.1109/ICASSP39728.2021.9414567.
- [142] D. P. Kingma and M. Welling, “Auto-encoding variational bayes,” in *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2014. [Online]. Available: <http://arxiv.org/abs/1312.6114>.
- [143] M. Mirza and S. Osindero, “Conditional generative adversarial nets,” *CoRR*, vol. abs/1411.1784, 2014. arXiv: 1411.1784. [Online]. Available: <http://arxiv.org/abs/1411.1784>.
- [144] S. E. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee, “Generative adversarial text to image synthesis,” in *Proceedings of the 33rd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, M. Balcan and K. Q. Weinberger, Eds., ser. JMLR Workshop and Conference Proceedings,

- vol. 48, JMLR.org, 2016, pp. 1060–1069. [Online]. Available: <http://proceedings.mlr.press/v48/reed16.html>.
- [145] P. Isola, J. Zhu, T. Zhou, and A. A. Efros, “Image-to-image translation with conditional adversarial networks,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, IEEE Computer Society, 2017, pp. 5967–5976. DOI: 10.1109/CVPR.2017.632. [Online]. Available: <https://doi.org/10.1109/CVPR.2017.632>.
- [146] A. Radford, L. Metz, and S. Chintala, “Unsupervised representation learning with deep convolutional generative adversarial networks,” in *ICLR*, 2016. [Online]. Available: <http://arxiv.org/abs/1511.06434>.
- [147] M. Arjovsky, S. Chintala, and L. Bottou, “Wasserstein GAN,” *CoRR*, vol. abs/1701.07875, 2017. arXiv: 1701.07875. [Online]. Available: <http://arxiv.org/abs/1701.07875>.
- [148] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, “Improved training of wasserstein gans,” in *Proc. of NeurIPS*, 2017, pp. 5767–5777. [Online]. Available: <http://papers.nips.cc/paper/7159-improved-training-of-wasserstein-gans>.
- [149] X. Mao, Q. Li, H. Xie, R. Y. K. Lau, Z. Wang, and S. P. Smolley, “Least squares generative adversarial networks,” in *ICCV*, IEEE, 2017, pp. 2813–2821. DOI: 10.1109/ICCV.2017.304. [Online]. Available: <https://doi.org/10.1109/ICCV.2017.304>.
- [150] X. Xu, R. Flynn, and M. Russell, “Speech intelligibility and quality: A comparative study of speech enhancement algorithms,” in *2017 28th Irish Signals and Systems Conference (ISSC)*, 2017, pp. 1–6. DOI: 10.1109/ISSC.2017.7983599.
- [151] 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 *Proc. of ICASSP*, 2001. DOI: 10.1109/ICASSP.2001.941023. [Online]. Available: <https://doi.org/10.1109/ICASSP.2001.941023>.

- [152] J. G. Beerends and J. A. Stemerdink, “A perceptual speech-quality measure based on a psychoacoustic sound representation,” *Journal of the Audio Engineering Society*, vol. 42, no. 3, pp. 115–123, 1994.
- [153] R. Kubichek, “Mel-cepstral distance measure for objective speech quality assessment,” in *Proceedings of IEEE Pacific Rim Conference on Communications Computers and Signal Processing*, vol. 1, 1993, 125–128 vol.1. DOI: 10.1109/PACRIM.1993.407206.
- [154] A. Hines, J. Skoglund, A. C. Kokaram, and N. Harte, “Visqol: An objective speech quality model,” *EURASIP J. Audio Speech Music. Process.*, vol. 2015, p. 13, 2015. DOI: 10.1186/s13636-015-0054-9. [Online]. Available: <https://doi.org/10.1186/s13636-015-0054-9>.
- [155] M. Chinen, F. S. C. Lim, J. Skoglund, N. Gureev, F. O’Gorman, and A. Hines, “ViSQOL v3: An open source production ready objective speech and audio metric,” in *QoMEX*, IEEE, 2020, pp. 1–6. DOI: 10.1109/QoMEX48832.2020.9123150. [Online]. Available: <https://doi.org/10.1109/QoMEX48832.2020.9123150>.
- [156] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen, “An algorithm for intelligibility prediction of time-frequency weighted noisy speech,” *IEEE Trans. Speech Audio Process.*, vol. 19, no. 7, pp. 2125–2136, 2011. DOI: 10.1109/TASL.2011.2114881. [Online]. Available: <https://doi.org/10.1109/TASL.2011.2114881>.
- [157] —, “An algorithm for intelligibility prediction of time–frequency weighted noisy speech,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 7, pp. 2125–2136, 2011. DOI: 10.1109/TASL.2011.2114881.
- [158] J. Jensen and C. H. Taal, “An algorithm for predicting the intelligibility of speech masked by modulated noise maskers,” *ACM Trans. Audio Speech Lang. Process.*, vol. 24, no. 11, pp. 2009–2022, 2016. DOI: 10.1109/TASLP.2016.2585878. [Online]. Available: <https://doi.org/10.1109/TASLP.2016.2585878>.
- [159] P. Ma, A. Haliassos, A. Fernandez-Lopez, H. Chen, S. Petridis, and M. Pantic, “Auto-avsr: Audio-visual speech recognition with automatic labels,” in *ICASSP 2023-2023*

- IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2023, pp. 1–5.
- [160] J MacQueen, “Classification and analysis of multivariate observations,” in *5th Berkeley Symp. Math. Statist. Probability*, 1967, pp. 281–297.
- [161] X. Chen, S. Xie, and K. He, “An empirical study of training self-supervised vision transformers,” in *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, IEEE, 2021, pp. 9620–9629. DOI: 10.1109/ICCV48922.2021.00950. [Online]. Available: <https://doi.org/10.1109/ICCV48922.2021.00950>.
- [162] M. Caron, P. Bojanowski, A. Joulin, and M. Douze, “Deep clustering for unsupervised learning of visual features,” in *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part XIV*, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds., ser. Lecture Notes in Computer Science, vol. 11218, Springer, 2018, pp. 139–156. DOI: 10.1007/978-3-030-01264-9_9. [Online]. Available: https://doi.org/10.1007/978-3-030-01264-9_9.
- [163] R. Zhang, P. Isola, and A. A. Efros, “Colorful image colorization,” in *ECCV*, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds., vol. 9907, Springer, 2016, pp. 649–666. DOI: 10.1007/978-3-319-46487-9_40. [Online]. Available: https://doi.org/10.1007/978-3-319-46487-9_40.
- [164] M. Noroozi and P. Favaro, “Unsupervised learning of visual representations by solving jigsaw puzzles,” in *ECCV*, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds., vol. 9910, Springer, 2016, pp. 69–84. DOI: 10.1007/978-3-319-46466-4_5. [Online]. Available: https://doi.org/10.1007/978-3-319-46466-4_5.
- [165] S. Pascual, M. Ravanelli, J. Serrà, A. Bonafonte, and Y. Bengio, “Learning problem-agnostic speech representations from multiple self-supervised tasks,” in *Proc. of Interspeech*, G. Kubin and Z. Kacic, Eds., ISCA, 2019, pp. 161–165. DOI: 10.21437/Interspeech.2019-2605. [Online]. Available: <https://doi.org/10.21437/Interspeech.2019-2605>.

- [166] M. Ravanelli *et al.*, “Multi-task self-supervised learning for robust speech recognition,” in *Proc. of ICASSP*, IEEE, 2020, pp. 6989–6993. DOI: 10.1109/ICASSP40776.2020.9053569. [Online]. Available: <https://doi.org/10.1109/ICASSP40776.2020.9053569>.
- [167] S. Chung, J. S. Chung, and H. Kang, “Perfect match: Self-supervised embeddings for cross-modal retrieval,” *J. Sel. Top. Signal Process.*, vol. 14, no. 3, pp. 568–576, 2020. DOI: 10.1109/JSTSP.2020.2987720. [Online]. Available: <https://doi.org/10.1109/JSTSP.2020.2987720>.
- [168] ———, “Perfect match: Improved cross-modal embeddings for audio-visual synchronisation,” in *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2019, Brighton, United Kingdom, May 12-17, 2019*, IEEE, 2019, pp. 3965–3969. DOI: 10.1109/ICASSP.2019.8682524. [Online]. Available: <https://doi.org/10.1109/ICASSP.2019.8682524>.
- [169] S. Chung, H. Kang, and J. S. Chung, “Seeing voices and hearing voices: Learning discriminative embeddings using cross-modal self-supervision,” in *Interspeech*, 2020, pp. 3486–3490. arXiv: 2004.14326. [Online]. Available: <https://arxiv.org/abs/2004.14326>.
- [170] M. K. Tellamekala, M. F. Valstar, M. Pound, and T. Giesbrecht, “Audio-visual predictive coding for self-supervised visual representation learning,” in *ICPR*, IEEE, 2020, pp. 9912–9919. DOI: 10.1109/ICPR48806.2021.9413295. [Online]. Available: <https://doi.org/10.1109/ICPR48806.2021.9413295>.
- [171] Y. Zhao, R. Xu, X. Wang, P. Hou, H. Tang, and M. Song, “Hearing lips: Improving lip reading by distilling speech recognizers,” in *AAAI*, AAAI Press, 2020, pp. 6917–6924. [Online]. Available: <https://aaai.org/ojs/index.php/AAAI/article/view/6174>.
- [172] T. Afouras, J. S. Chung, and A. Zisserman, “ASR is all you need: Cross-modal distillation for lip reading,” in *ICASSP*, IEEE, 2020, pp. 2143–2147. DOI: 10.1109/ICASSP40776.2020.9054253. [Online]. Available: <https://doi.org/10.1109/ICASSP40776.2020.9054253>.

- [173] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, “Librispeech: An ASR corpus based on public domain audio books,” in *ICASSP*, IEEE, 2015, pp. 5206–5210. DOI: 10.1109/ICASSP.2015.7178964. [Online]. Available: <https://doi.org/10.1109/ICASSP.2015.7178964>.
- [174] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in *ICLR*, 2015. [Online]. Available: <http://arxiv.org/abs/1412.6980>.
- [175] X. Weng and K. Kitani, “Learning spatio-temporal features with two-stream deep 3D CNNs for lipreading,” in *BMVC*, BMVA Press, 2019, p. 269. [Online]. Available: <https://bmvc2019.org/wp-content/uploads/papers/0016-paper.pdf>.
- [176] T. Stafylakis, M. H. Khan, and G. Tzimiropoulos, “Pushing the boundaries of audiovisual word recognition using residual networks and LSTMs,” *Computer Vision and Image Understanding*, vol. 176, pp. 22–32, 2018.
- [177] P. Ma, Y. Wang, J. Shen, S. Petridis, and M. Pantic, “Lip-reading with densely connected temporal convolutional networks,” in *WACV*, 2021, pp. 2857–2866.
- [178] P. Ma, B. Martinez, S. Petridis, and M. Pantic, “Towards practical lipreading with distilled and efficient models,” in *ICASSP*, 2021, pp. 7608–7612. DOI: 10.1109/ICASSP39728.2021.9415063.
- [179] S. Petridis, T. Stafylakis, P. Ma, G. Tzimiropoulos, and M. Pantic, “Audio-visual speech recognition with a hybrid ctc/attention architecture,” in *SLT*, IEEE, 2018, pp. 513–520. DOI: 10.1109/SLT.2018.8639643. [Online]. Available: <https://doi.org/10.1109/SLT.2018.8639643>.
- [180] X. Zhang, F. Cheng, and S. Wang, “Spatio-temporal fusion based convolutional sequence learning for lip reading,” in *ICCV*, IEEE, 2019, pp. 713–722. DOI: 10.1109/ICCV.2019.00080. [Online]. Available: <https://doi.org/10.1109/ICCV.2019.00080>.
- [181] J. Yu *et al.*, “Audio-visual recognition of overlapped speech for the LRS2 dataset,” in *ICASSP*, 2020, pp. 6984–6988. DOI: 10.1109/ICASSP40776.2020.9054127.

- [182] K. Vougioukas, P. Ma, S. Petridis, and M. Pantic, “Video-driven speech reconstruction using generative adversarial networks,” in *Proc. of Interspeech*, G. Kubin and Z. Kacic, Eds., ISCA, 2019, pp. 4125–4129. DOI: 10.21437/Interspeech.2019-1445. [Online]. Available: <https://doi.org/10.21437/Interspeech.2019-1445>.
- [183] M. Kim, J. Hong, and Y. M. Ro, “Lip-to-speech synthesis in the wild with multi-task learning,” *CoRR*, vol. abs/2302.08841, 2023. DOI: 10.48550/arXiv.2302.08841. arXiv: 2302.08841. [Online]. Available: <https://doi.org/10.48550/arXiv.2302.08841>.
- [184] W. Hsu, T. Remez, B. Shi, J. Donley, and Y. Adi, “Revise: Self-supervised speech resynthesis with visual input for universal and generalized speech enhancement,” *CoRR*, vol. abs/2212.11377, 2022. DOI: 10.48550/arXiv.2212.11377. arXiv: 2212.11377. [Online]. Available: <https://doi.org/10.48550/arXiv.2212.11377>.
- [185] H. Zhou, Z. Liu, X. Xu, P. Luo, and X. Wang, “Vision-infused deep audio inpainting,” in *Proc. of ICCV, CVF / IEEE*, 2019, pp. 283–292. DOI: 10.1109/ICCV.2019.00037. [Online]. Available: <https://doi.org/10.1109/ICCV.2019.00037>.
- [186] A. van den Oord *et al.*, “Parallel wavenet: Fast high-fidelity speech synthesis,” in *Proc. of ICML*, PMLR, 2018, pp. 3915–3923. [Online]. Available: <http://proceedings.mlr.press/v80/oord18a.html>.
- [187] T. L. Cornu and B. Milner, “Reconstructing intelligible audio speech from visual speech features,” in *Proc. of Interspeech*, ISCA, 2015, pp. 3355–3359. [Online]. Available: http://www.isca-speech.org/archive/interspeech_2015/i15_3355.html.
- [188] A. Ephrat, T. Halperin, and S. Peleg, “Improved speech reconstruction from silent video,” in *Proc. of ICCV*, IEEE, 2017, pp. 455–462. DOI: 10.1109/ICCVW.2017.61. [Online]. Available: <https://doi.org/10.1109/ICCVW.2017.61>.
- [189] H. Akbari, H. Arora, L. Cao, and N. Mesgarani, “Lip2audspec: Speech reconstruction from silent lip movements video,” in *Proc. of ICASSP*, IEEE, 2018, pp. 2516–2520. DOI: 10.1109/ICASSP.2018.8461856. [Online]. Available: <https://doi.org/10.1109/ICASSP.2018.8461856>.

- [190] K. R. Prajwal, R. Mukhopadhyay, V. P. Namboodiri, and C. V. Jawahar, “Learning individual speaking styles for accurate lip to speech synthesis,” in *Proc. of CVPR*, IEEE, 2020, pp. 13 793–13 802. DOI: 10 . 1109 / CVPR42600 . 2020 . 01381. [Online]. Available: <https://doi.org/10.1109/CVPR42600.2020.01381>.
- [191] D. Michelsanti, O. Slizovskaia, G. Haro, E. Gómez, Z. Tan, and J. Jensen, “Vocoder-based speech synthesis from silent videos,” in *Proc. of Interspeech*, H. Meng, B. Xu, and T. F. Zheng, Eds., ISCA, 2020, pp. 3530–3534. DOI: 10 . 21437 / Interspeech . 2020 - 1026. [Online]. Available: <https://doi.org/10.21437/Interspeech.2020-1026>.
- [192] M. Morise, F. Yokomori, and K. Ozawa, “WORLD: A vocoder-based high-quality speech synthesis system for real-time applications,” *IEICE Trans. Inf. Syst.*, vol. 99-D, no. 7, pp. 1877–1884, 2016. DOI: 10 . 1587 / transinf . 2015EDP7457. [Online]. Available: <https://doi.org/10.1587/transinf.2015EDP7457>.
- [193] K. Kumar *et al.*, “Melgan: Generative adversarial networks for conditional waveform synthesis,” in *Proc. of NeurIPS*, 2019, pp. 14 881–14 892. [Online]. Available: <http://papers.nips.cc/paper/9629-melgan-generative-adversarial-networks-for-conditional-waveform-synthesis>.
- [194] M. Cooke, J. Barker, S. Cunningham, and X. Shao, “An audio-visual corpus for speech perception and automatic speech recognition (1),” *The Journal of the Acoustical Society of America*, vol. 120, pp. 2421–4, 2006.
- [195] N. Harte and E. Gillen, “Tcd-timit: An audio-visual corpus of continuous speech,” *IEEE Transactions on Multimedia*, vol. 17, no. 5, pp. 603–615, 2015.
- [196] G. Zhao, M. Barnard, and M. Pietikäinen, “Lipreading with local spatiotemporal descriptors,” *IEEE Trans. Multimedia*, vol. 11, no. 7, pp. 1254–1265, 2009. DOI: 10 . 1109 / TMM . 2009 . 2030637. [Online]. Available: <https://doi.org/10.1109/TMM.2009.2030637>.
- [197] J. S. Chung and A. Zisserman, “Lip reading in the wild,” in *Proc. of ACCV*, ser. Lecture Notes in Computer Science, vol. 10112, 2016, pp. 87–103. DOI: 10 . 1007 / 978 - 3 - 319 - 54184 - 6 _ 6. [Online]. Available: https://doi.org/10.1007/978-3-319-54184-6_6.

- [198] S. Petridis, Z. Li, and M. Pantic, “End-to-end visual speech recognition with LSTMS,” in *Proc. of ICASSP*, IEEE, 2017, pp. 2592–2596. DOI: 10.1109/ICASSP.2017.7952625. [Online]. Available: <https://doi.org/10.1109/ICASSP.2017.7952625>.
- [199] S. Petridis, Y. Wang, P. Ma, Z. Li, and M. Pantic, “End-to-end visual speech recognition for small-scale datasets,” *Pattern Recognit. Lett.*, vol. 131, pp. 421–427, 2020. DOI: 10.1016/j.patrec.2020.01.022. [Online]. Available: <https://doi.org/10.1016/j.patrec.2020.01.022>.
- [200] B. Shillingford *et al.*, “Large-scale visual speech recognition,” in *Proc. of Interspeech*, ISCA, 2019, pp. 4135–4139. DOI: 10.21437/Interspeech.2019-1669. [Online]. Available: <https://doi.org/10.21437/Interspeech.2019-1669>.
- [201] S. Petridis, Y. Wang, Z. Li, and M. Pantic, “End-to-end multi-view lipreading,” in *BMVC*, BMVA, 2017. [Online]. Available: <https://www.dropbox.com/s/9me0ad6bbqf7h91/0737.pdf?dl=1>.
- [202] Y. Wang *et al.*, “Tacotron: Towards end-to-end speech synthesis,” in *Proc. of Interspeech*, ISCA, 2017, pp. 4006–4010. [Online]. Available: http://www.isca-speech.org/archive/Interspeech_2017/abstracts/1452.html.
- [203] X. Zhu, G. Beauregard, and L. L. Wyse, “Real-time signal estimation from modified short-time fourier transform magnitude spectra,” *IEEE Trans. Audio, Speech & Language Processing*, vol. 15, no. 5, pp. 1645–1653, 2007. DOI: 10.1109/TASL.2007.899236. [Online]. Available: <https://doi.org/10.1109/TASL.2007.899236>.
- [204] J. Donahue, S. Dieleman, M. Binkowski, E. Elsen, and K. Simonyan, “End-to-end adversarial text-to-speech,” *CoRR*, vol. abs/2006.03575, 2020. arXiv: 2006.03575. [Online]. Available: <https://arxiv.org/abs/2006.03575>.
- [205] T. L. Cornu and B. Milner, “Generating intelligible audio speech from visual speech,” *IEEE ACM Trans. Audio Speech Lang. Process.*, vol. 25, no. 9, pp. 1751–1761, 2017. DOI: 10.1109/TASLP.2017.2716178. [Online]. Available: <https://doi.org/10.1109/TASLP.2017.2716178>.

- [206] A. Ephrat and S. Peleg, “Vid2speech: Speech reconstruction from silent video,” in *Proc. of ICASSP*, IEEE, 2017, pp. 5095–5099. DOI: 10.1109/ICASSP.2017.7953127. [Online]. Available: <https://doi.org/10.1109/ICASSP.2017.7953127>.
- [207] Y. Kumar, M. Aggarwal, P. Nawal, S. Satoh, R. R. Shah, and R. Zimmermann, “Harnessing AI for speech reconstruction using multi-view silent video feed,” in *Proc. of MM*, ACM, 2018. DOI: 10.1145/3240508.3241911. [Online]. Available: <https://doi.org/10.1145/3240508.3241911>.
- [208] I. Anina, Z. Zhou, G. Zhao, and M. Pietikäinen, “Ouluvs2: A multi-view audiovisual database for non-rigid mouth motion analysis,” in *Proc. of FG*, IEEE, 2015, pp. 1–5. DOI: 10.1109/FG.2015.7163155. [Online]. Available: <https://doi.org/10.1109/FG.2015.7163155>.
- [209] Y. Kumar, R. Jain, K. M. Salik, R. R. Shah, R. Zimmermann, and Y. Yin, “Mylipper: A personalized system for speech reconstruction using multi-view visual feeds,” in *Proc. of ISM*, IEEE, 2018, pp. 159–166. DOI: 10.1109/ISM.2018.00–19. [Online]. Available: <https://doi.org/10.1109/ISM.2018.00–19>.
- [210] Y. Kumar, R. Jain, K. M. Salik, R. R. Shah, Y. Yin, and R. Zimmermann, “Lipper: Synthesizing thy speech using multi-view lipreading,” in *Proc. of AAAI*, 2019, pp. 2588–2595. DOI: 10.1609/aaai.v33i01.33012588. [Online]. Available: <https://doi.org/10.1609/aaai.v33i01.33012588>.
- [211] S. Uttam *et al.*, “Hush-hush speak: Speech reconstruction using silent videos,” in *Proc. of Interspeech*, ISCA, 2019, pp. 136–140. DOI: 10.21437/Interspeech.2019-3269. [Online]. Available: <https://doi.org/10.21437/Interspeech.2019-3269>.
- [212] A. Owens, P. Isola, J. H. McDermott, A. Torralba, E. H. Adelson, and W. T. Freeman, “Visually indicated sounds,” in *Proc. of CVPR*, IEEE, 2016, pp. 2405–2413. DOI: 10.1109/CVPR.2016.264. [Online]. Available: <https://doi.org/10.1109/CVPR.2016.264>.
- [213] Y. Zhou, Z. Wang, C. Fang, T. Bui, and T. L. Berg, “Visual to sound: Generating natural sound for videos in the wild,” in *Proc. of CVPR*, IEEE, 2018, pp. 3550–3558.

- DOI: 10.1109/CVPR.2018.00374. [Online]. Available: http://openaccess.thecvf.com/content/_cvpr/_2018/html/Zhou_Visual_to_Sound_CVPR_2018_paper.html.
- [214] L. Chen, S. Srivastava, Z. Duan, and C. Xu, “Deep cross-modal audio-visual generation,” in *Proc. of MM*, ACM, 2017, pp. 349–357. DOI: 10.1145/3126686.3126723. [Online]. Available: <https://doi.org/10.1145/3126686.3126723>.
- [215] W. Han, C. Chan, O. C. Choy, and K. Pun, “An efficient MFCC extraction method in speech recognition,” in *International Symposium on Circuits and Systems*, 2006. DOI: 10.1109/ISCAS.2006.1692543. [Online]. Available: <https://doi.org/10.1109/ISCAS.2006.1692543>.
- [216] K. V. Krishna Kishore and P. Krishna Satish, “Emotion recognition in speech using mfcc and wavelet features,” in *(Proc. of IACC)*, 2013, pp. 842–847.
- [217] P. C. Loizou, “Speech quality assessment,” in *Multimedia Analysis, Processing and Communications*, vol. 346, 2011, pp. 623–654. DOI: 10.1007/978-3-642-19551-8_23. [Online]. Available: https://doi.org/10.1007/978-3-642-19551-8_23.
- [218] P. Ma, S. Petridis, and M. Pantic, “Investigating the lombard effect influence on end-to-end audio-visual speech recognition,” in *INTERSPEECH*, 2019.
- [219] K. Vougioukas, S. Petridis, and M. Pantic, “Realistic speech-driven facial animation with gans,” *Int. J. Comput. Vis.*, vol. 128, no. 5, pp. 1398–1413, 2020. DOI: 10.1007/s11263-019-01251-8. [Online]. Available: <https://doi.org/10.1007/s11263-019-01251-8>.
- [220] K. Vougioukas, P. Ma, S. Petridis, and M. Pantic, “Video-driven speech reconstruction using generative adversarial networks,” in *Interspeech*, G. Kubin and Z. Kacic, Eds., ISCA, 2019, pp. 4125–4129. DOI: 10.21437/Interspeech.2019-1445. [Online]. Available: <https://doi.org/10.21437/Interspeech.2019-1445>.
- [221] R. Yadav, A. Sardana, V. P. Namboodiri, and R. M. Hegde, “Speech prediction in silent videos using variational autoencoders,” in *ICASSP*, IEEE, 2021, pp. 7048–7052. DOI: 10.1109/ICASSP39728.2021.9414040. [Online]. Available: <https://doi.org/10.1109/ICASSP39728.2021.9414040>.

- [222] D. Michelsanti, O. Slizovskaia, G. Haro, E. Gómez, Z. Tan, and J. Jensen, “Vocoder-based speech synthesis from silent videos,” in *Interspeech*, H. Meng, B. Xu, and T. F. Zheng, Eds., ISCA, 2020, pp. 3530–3534. DOI: 10.21437/Interspeech.2020-1026. [Online]. Available: <https://doi.org/10.21437/Interspeech.2020-1026>.
- [223] D. Oneata, A. Stan, and H. Cucu, “Speaker disentanglement in video-to-speech conversion,” in *EUSIPCO*, IEEE, 2021, pp. 46–50. DOI: 10.23919/EUSIPCO54536.2021.9616266. [Online]. Available: <https://doi.org/10.23919/EUSIPCO54536.2021.9616266>.
- [224] S. Um, J. Kim, J. Lee, S. Oh, K. Byun, and H. Kang, “Facetron: Multi-speaker face-to-speech model based on cross-modal latent representations,” *CoRR*, vol. abs/2107.12003, 2021. arXiv: 2107.12003. [Online]. Available: <https://arxiv.org/abs/2107.12003>.
- [225] J. Hong, M. Kim, S. J. Park, and Y. M. Ro, “Speech reconstruction with reminiscent sound via visual voice memory,” *IEEE ACM Trans. Audio Speech Lang. Process.*, vol. 29, pp. 3654–3667, 2021. DOI: 10.1109/TASLP.2021.3126925. [Online]. Available: <https://doi.org/10.1109/TASLP.2021.3126925>.
- [226] M. Kim, J. Hong, and Y. M. Ro, “Lip to speech synthesis with visual context attentional GAN,” in *NeurIPS*, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, Eds., 2021. [Online]. Available: <https://openreview.net/forum?id=x6z8J\17LP3>.
- [227] K. R. Prajwal, R. Mukhopadhyay, V. P. Namboodiri, and C. V. Jawahar, “Learning individual speaking styles for accurate lip to speech synthesis,” in *CVPR*, Computer Vision Foundation / IEEE, 2020, pp. 13 793–13 802. DOI: 10.1109/CVPR42600.2020.01381. [Online]. Available: https://openaccess.thecvf.com/content_CVPR_2020/html/Prajwal_Learning_Individual_Speaking_Styles_for_Accurate_Lip_to_Speech_Synthesis_CVPR_2020_paper.html.
- [228] J. S. Chung, A. Nagrani, and A. Zisserman, “Voxceleb2: Deep speaker recognition,” in *Interspeech*, 2018.
- [229] A. Nagrani, J. S. Chung, and A. Zisserman, “Voxceleb: A large-scale speaker identification dataset,” in *Interspeech*, F. Lacerda, Ed., ISCA, 2017, pp. 2616–2620. [Online].

Available: http://www.isca-speech.org/archive/Interspeech_2017/abstracts/0950.html.

- [230] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, “Librispeech: An ASR corpus based on public domain audio books,” in *Proc. of ICASSP*, IEEE, 2015, pp. 5206–5210. DOI: 10.1109/ICASSP.2015.7178964. [Online]. Available: <https://doi.org/10.1109/ICASSP.2015.7178964>.
- [231] H. Zen *et al.*, “Libritts: A corpus derived from librispeech for text-to-speech,” in *Interspeech*, G. Kubin and Z. Kacic, Eds., ISCA, 2019, pp. 1526–1530. DOI: 10.21437/Interspeech.2019-2441. [Online]. Available: <https://doi.org/10.21437/Interspeech.2019-2441>.
- [232] A. Bulat and G. Tzimiropoulos, “How far are we from solving the 2d & 3d face alignment problem? (and a dataset of 230,000 3d facial landmarks),” in *ICCV*, 2017.
- [233] I. Loshchilov and F. Hutter, “Decoupled weight decay regularization,” *arXiv preprint arXiv:1711.05101*, 2017.
- [234] ———, “Sgdr: Stochastic gradient descent with warm restarts,” *arXiv preprint arXiv:1608.03983*, 2016.
- [235] D. Griffin and Jae Lim, “Signal estimation from modified short-time fourier transform,” *IEEE Trans. on Acoustics, Speech, and Signal Processing*, vol. 32, no. 2, pp. 236–243, Apr. 1984, ISSN: 0096-3518. DOI: 10.1109/TASSP.1984.1164317.
- [236] G. Yang, S. Yang, K. Liu, P. Fang, W. Chen, and L. Xie, “Multi-band melgan: Faster waveform generation for high-quality text-to-speech,” in *SLT*, IEEE, 2021, pp. 492–498. DOI: 10.1109/SLT48900.2021.9383551. [Online]. Available: <https://doi.org/10.1109/SLT48900.2021.9383551>.
- [237] A. Mustafa, N. Pia, and G. Fuchs, “Stylemelgan: An efficient high-fidelity adversarial vocoder with temporal adaptive normalization,” in *ICASSP*, IEEE, 2021, pp. 6034–6038. DOI: 10.1109/ICASSP39728.2021.9413605. [Online]. Available: <https://doi.org/10.1109/ICASSP39728.2021.9413605>.

- [238] N. Perraudin, P. Balázs, and P. L. Søndergaard, “A fast griffin-lim algorithm,” in *WASPAA*, IEEE, 2013, pp. 1–4. DOI: 10.1109/WASPAA.2013.6701851. [Online]. Available: <https://doi.org/10.1109/WASPAA.2013.6701851>.
- [239] D. Wang and J. Chen, “Supervised speech separation based on deep learning: An overview,” *IEEE ACM Trans. Audio Speech Lang. Process.*, vol. 26, no. 10, pp. 1702–1726, 2018. DOI: 10.1109/TASLP.2018.2842159. [Online]. Available: <https://doi.org/10.1109/TASLP.2018.2842159>.
- [240] K. Tan and D. Wang, “Learning complex spectral mapping with gated convolutional recurrent networks for monaural speech enhancement,” *IEEE ACM Trans. Audio Speech Lang. Process.*, vol. 28, pp. 380–390, 2020. DOI: 10.1109/TASLP.2019.2955276. [Online]. Available: <https://doi.org/10.1109/TASLP.2019.2955276>.
- [241] A. Défossez, G. Synnaeve, and Y. Adi, “Real time speech enhancement in the waveform domain,” in *Interspeech*, ISCA, 2020, pp. 3291–3295. DOI: 10.21437/Interspeech.2020-2409. [Online]. Available: <https://doi.org/10.21437/Interspeech.2020-2409>.
- [242] X. Hao, X. Su, Z. Wang, H. Zhang, and Batushiren, “UNetGAN: A robust speech enhancement approach in time domain for extremely low signal-to-noise ratio condition,” in *Interspeech*, ISCA, 2019, pp. 1786–1790. DOI: 10.21437/Interspeech.2019-1567. [Online]. Available: <https://doi.org/10.21437/Interspeech.2019-1567>.
- [243] A. Gabbay, A. Shamir, and S. Peleg, “Visual speech enhancement,” in *Interspeech*, ISCA, 2018, pp. 1170–1174. DOI: 10.21437/Interspeech.2018-1955. [Online]. Available: <https://doi.org/10.21437/Interspeech.2018-1955>.
- [244] Z. Pan, R. Tao, C. Xu, and H. Li, “Muse: Multi-modal target speaker extraction with visual cues,” in *ICASSP*, IEEE, 2021, pp. 6678–6682. DOI: 10.1109/ICASSP39728.2021.9414023. [Online]. Available: <https://doi.org/10.1109/ICASSP39728.2021.9414023>.

- [245] X. Xu *et al.*, “VSEGAN: visual speech enhancement generative adversarial network,” in *ICASSP*, IEEE, 2022, pp. 7308–7311. DOI: 10.1109/ICASSP43922.2022.9747187. [Online]. Available: <https://doi.org/10.1109/ICASSP43922.2022.9747187>.
- [246] Y. Luo and N. Mesgarani, “Conv-TasNet: Surpassing ideal time-frequency magnitude masking for speech separation,” *ACM Trans. Audio Speech Lang. Process.*, vol. 27, no. 8, pp. 1256–1266, 2019. DOI: 10.1109/TASLP.2019.2915167. [Online]. Available: <https://doi.org/10.1109/TASLP.2019.2915167>.
- [247] J. Hou, S. Wang, Y. Lai, Y. Tsao, H. Chang, and H. Wang, “Audio-visual speech enhancement using multimodal deep convolutional neural networks,” *IEEE Trans. Emerg. Top. Comput. Intell.*, vol. 2, no. 2, pp. 117–128, 2018. DOI: 10.1109/TETCI.2017.2784878. [Online]. Available: <https://doi.org/10.1109/TETCI.2017.2784878>.
- [248] B. Shi, W. Hsu, K. Lakhotia, and A. Mohamed, “Learning audio-visual speech representation by masked multimodal cluster prediction,” in *ICLR*, OpenReview.net, 2022. [Online]. Available: <https://openreview.net/forum?id=Z1Q1m11u0M>.
- [249] J. Kong, J. Kim, and J. Bae, “HiFi-GAN: Generative adversarial networks for efficient and high fidelity speech synthesis,” in *NeurIPS*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., vol. 33, 2020, pp. 17 022–17 033. [Online]. Available: <https://proceedings.neurips.cc/paper/2020/file/c5d736809766d46260d816d8dbc9eb44-Paper.pdf>.
- [250] A. B. L. Larsen, S. K. Sønderby, H. Larochelle, and O. Winther, “Autoencoding beyond pixels using a learned similarity metric,” in *ICML*, ser. JMLR Workshop and Conference Proceedings, vol. 48, JMLR.org, 2016, pp. 1558–1566. [Online]. Available: <http://proceedings.mlr.press/v48/larsen16.html>.
- [251] C. K. A. Reddy *et al.*, “The INTERSPEECH 2020 deep noise suppression challenge: Datasets, subjective testing framework, and challenge results,” in *Interspeech*, ISCA, 2020, pp. 2492–2496. DOI: 10.21437/Interspeech.2020-3038. [Online]. Available: <https://doi.org/10.21437/Interspeech.2020-3038>.

- [252] I. Loshchilov and F. Hutter, “Decoupled weight decay regularization,” in *ICLR*, OpenReview.net, 2019. [Online]. Available: <https://openreview.net/forum?id=Bkg6RiCqY7>.
- [253] —, “SGDR: stochastic gradient descent with warm restarts,” in *ICLR*, OpenReview.net, 2017. [Online]. Available: <https://openreview.net/forum?id=Skq89Scxx>.
- [254] J. Yamagishi, C. Veaux, and K. MacDonald, “CSTR VCTK corpus: English multi-speaker corpus for CSTR voice cloning toolkit,” in *University of Edinburgh. The Centre for Speech Technology Research (CSTR)*, 2012.
- [255] K. Ito and L. Johnson, *The LJ speech dataset*, <https://keithito.com/LJ-Speech-Dataset/>, 2017.
- [256] W. Bai, Y. Liu, Z. Zhang, B. Li, and W. Hu, “Aunet: Learning relations between action units for face forgery detection,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 24 709–24 719.
- [257] C. Feng, Z. Chen, and A. Owens, “Self-supervised video forensics by audio-visual anomaly detection,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 10 491–10 503.
- [258] J. Lian, A. Baevski, W. Hsu, and M. Auli, “Av-data2vec: Self-supervised learning of audio-visual speech representations with contextualized target representations,” *CoRR*, vol. abs/2302.06419, 2023. DOI: 10.48550/arXiv.2302.06419. arXiv: 2302.06419. [Online]. Available: <https://doi.org/10.48550/arXiv.2302.06419>.
- [259] B. Fernando, H. Bilen, E. Gavves, and S. Gould, “Self-supervised video representation learning with odd-one-out networks,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, IEEE Computer Society, 2017, pp. 5729–5738. DOI: 10.1109/CVPR.2017.607. [Online]. Available: <https://doi.org/10.1109/CVPR.2017.607>.
- [260] A. Shukla, S. Petridis, and M. Pantic, “Does visual self-supervision improve learning of speech representations for emotion recognition,” *IEEE Transactions on Affective Computing*, pp. 1–1, 2021. DOI: 10.1109/TAFFC.2021.3062406.

- [261] D. Tran, H. Wang, M. Feiszli, and L. Torresani, “Video classification with channel-separated convolutional networks,” in *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, IEEE, 2019, pp. 5551–5560. DOI: 10.1109/ICCV.2019.00565. [Online]. Available: <https://doi.org/10.1109/ICCV.2019.00565>.
- [262] B. Heo *et al.*, “Adamp: Slowing down the slowdown for momentum optimizers on scale-invariant weights,” in *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*, OpenReview.net, 2021. [Online]. Available: <https://openreview.net/forum?id=Iz3zU3M316D>.
- [263] X. Chen and K. He, “Exploring simple siamese representation learning,” in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, Computer Vision Foundation / IEEE, 2021, pp. 15 750–15 758. DOI: 10.1109/CVPR46437.2021.01549. [Online]. Available: https://openaccess.thecvf.com/content/CVPR2021/html/Chen_Exploring_Simple_Siamese_Representation_Learning_CVPR_2021_paper.html.
- [264] D. S. Park *et al.*, “Specaugment: A simple data augmentation method for automatic speech recognition,” in *Interspeech 2019, 20th Annual Conference of the International Speech Communication Association, Graz, Austria, 15-19 September 2019*, G. Kubin and Z. Kacic, Eds., ISCA, 2019, pp. 2613–2617. DOI: 10.21437/Interspeech.2019-2680. [Online]. Available: <https://doi.org/10.21437/Interspeech.2019-2680>.
- [265] F. Wang, X. Xiang, J. Cheng, and A. L. Yuille, “Normface: L₂ hypersphere embedding for face verification,” in *Proceedings of the 2017 ACM on Multimedia Conference, MM 2017, Mountain View, CA, USA, October 23-27, 2017*, Q. Liu *et al.*, Eds., ACM, 2017, pp. 1041–1049. DOI: 10.1145/3123266.3123359. [Online]. Available: <https://doi.org/10.1145/3123266.3123359>.
- [266] A. Rössler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Nießner, “Faceforensics++: Learning to detect manipulated facial images,” in *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27*

- November 2, 2019, IEEE, 2019, pp. 1–11. DOI: 10.1109/ICCV.2019.00009. [Online]. Available: <https://doi.org/10.1109/ICCV.2019.00009>.
- [267] Y. Li, X. Yang, P. Sun, H. Qi, and S. Lyu, “Celeb-df: A large-scale challenging dataset for deepfake forensics,” in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, Computer Vision Foundation / IEEE, 2020, pp. 3204–3213. DOI: 10.1109/CVPR42600.2020.00327. [Online]. Available: https://openaccess.thecvf.com/content/_CVPR/_2020/html/Li_Celeb-DF_A_Large-Scale_Challenging_Dataset_for_DeepFake_Forensics_CVPR_2020_paper.html.
- [268] B. Dolhansky, R. Howes, B. Pflaum, N. Baram, and C. Canton-Ferrer, “The deepfake detection challenge (DFDC) preview dataset,” *CoRR*, vol. abs/1910.08854, 2019. arXiv: 1910.08854. [Online]. Available: <http://arxiv.org/abs/1910.08854>.
- [269] A. Haliassos, K. Vougioukas, S. Petridis, and M. Pantic, “Lips don’t lie: A generalisable and robust approach to face forgery detection,” in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, Computer Vision Foundation / IEEE, 2021, pp. 5039–5049. DOI: 10.1109/CVPR46437.2021.00500. [Online]. Available: https://openaccess.thecvf.com/content/CVPR2021/html/Haliassos_Lips_Dont_Lie_A_Generalisable_and_Robust_Approach_To_Face_CVPR_2021_paper.html.
- [270] L. Li, J. Bao, H. Yang, D. Chen, and F. Wen, “Advancing high fidelity identity swapping for forgery detection,” in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, Computer Vision Foundation / IEEE, 2020, pp. 5073–5082. DOI: 10.1109/CVPR42600.2020.00512. [Online]. Available: https://openaccess.thecvf.com/content/_CVPR/_2020/html/Li_Advancing_High_Fidelity_Identity_Swapping_for_Forgery_Detection_CVPR_2020_paper.html.
- [271] L. Jiang, R. Li, W. Wu, C. Qian, and C. C. Loy, “Deeperforensics-1.0: A large-scale dataset for real-world face forgery detection,” in *2020 IEEE/CVF Conference on Com-*

- puter Vision and Pattern Recognition, *CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, Computer Vision Foundation / IEEE, 2020, pp. 2886–2895. DOI: 10.1109/CVPR42600.2020.00296. [Online]. Available: https://openaccess.thecvf.com/content/_CVPR/_2020/html/Jiang/_DeeperForensics-1.0_A_Large-Scale_Dataset_for_Real-World_Face_Forgery_Detection_CVPR_2020_paper.html.
- [272] S. Wang, O. Wang, R. Zhang, A. Owens, and A. A. Efros, “Cnn-generated images are surprisingly easy to spot... for now,” in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, Computer Vision Foundation / IEEE, 2020, pp. 8692–8701. DOI: 10.1109/CVPR42600.2020.00872. [Online]. Available: https://openaccess.thecvf.com/content/_CVPR/_2020/html/Wang/_CNN-Generated_Images_Are_Surprisingly_Easy_to_Spot..._for_Now_CVPR_2020_paper.html.
- [273] L. Chai, D. Bau, S. Lim, and P. Isola, “What makes fake images detectable? understanding properties that generalize,” in *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XXVI*, A. Vedaldi, H. Bischof, T. Brox, and J. Frahm, Eds., ser. Lecture Notes in Computer Science, vol. 12371, Springer, 2020, pp. 103–120. DOI: 10.1007/978-3-030-58574-7_7. [Online]. Available: https://doi.org/10.1007/978-3-030-58574-7_7.
- [274] L. Li *et al.*, “Face x-ray for more general face forgery detection,” in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, Computer Vision Foundation / IEEE, 2020, pp. 5000–5009. DOI: 10.1109/CVPR42600.2020.00505. [Online]. Available: https://openaccess.thecvf.com/content/_CVPR/_2020/html/Li/_Face_X-Ray_for_More_General_Face_Forgery_Detection_CVPR_2020_paper.html.
- [275] E. Sabir, J. Cheng, A. Jaiswal, W. AbdAlmageed, I. Masi, and P. Natarajan, “Recurrent convolutional strategies for face manipulation detection in videos,” in *IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2019, Long Beach, CA, USA, June 16-20, 2019*, Computer Vision Foundation / IEEE, 2019, pp. 80–

87. [Online]. Available: http://openaccess.thecvf.com/content/_CVPRW/_2019/html/Media/_Forensics/Sabir/_Recurrent/_Convolutional/_Strategies/_for/_Face/_Manipulation/_Detection/_in/_Videos/_CVPRW/_2019/_paper.html.
- [276] H. H. Nguyen, F. Fang, J. Yamagishi, and I. Echizen, “Multi-task learning for detecting and segmenting manipulated facial images and videos,” in *10th IEEE International Conference on Biometrics Theory, Applications and Systems, BTAS 2019, Tampa, FL, USA, September 23-26, 2019*, IEEE, 2019, pp. 1–8. DOI: 10.1109/BTAS46853.2019.9185974. [Online]. Available: <https://doi.org/10.1109/BTAS46853.2019.9185974>.
- [277] Y. Li and S. Lyu, “Exposing deepfake videos by detecting face warping artifacts,” in *IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2019, Long Beach, CA, USA, June 16-20, 2019*, Computer Vision Foundation / IEEE, 2019, pp. 46–52. [Online]. Available: http://openaccess.thecvf.com/content/_CVPRW/_2019/html/Media/_Forensics/Li/_Exposing/_DeepFake/_Videos/_By/_Detecting/_Face/_Warping/_Artifacts/_CVPRW/_2019/_paper.html.
- [278] I. Masi, A. Killekar, R. M. Mascarenhas, S. P. Gurudatt, and W. AbdAlmageed, “Two-branch recurrent network for isolating deepfakes in videos,” in *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part VII*, A. Vedaldi, H. Bischof, T. Brox, and J. Frahm, Eds., ser. Lecture Notes in Computer Science, vol. 12352, Springer, 2020, pp. 667–684. DOI: 10.1007/978-3-030-58571-6_39. [Online]. Available: https://doi.org/10.1007/978-3-030-58571-6_39.
- [279] Y. Zheng, J. Bao, D. Chen, M. Zeng, and F. Wen, “Exploring temporal coherence for more general video face forgery detection,” in *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, IEEE, 2021, pp. 15 024–15 034. DOI: 10.1109/ICCV48922.2021.01477. [Online]. Available: <https://doi.org/10.1109/ICCV48922.2021.01477>.

- [280] B. Xu, C. Lu, Y. Guo, and J. Wang, “Discriminative multi-modality speech recognition,” in *CVPR*, IEEE, 2020, pp. 14 421–14 430. DOI: 10 . 1109 / CVPR42600 . 2020 . 01444. [Online]. Available: <https://doi.org/10.1109/CVPR42600.2020.01444>.
- [281] T. Makino *et al.*, “Recurrent neural network transducer for audio-visual speech recognition,” in *ASRU*, IEEE, 2019, pp. 905–912. DOI: 10.1109/ASRU46091.2019.9004036. [Online]. Available: <https://doi.org/10.1109/ASRU46091.2019.9004036>.
- [282] K. R. Prajwal, T. Afouras, and A. Zisserman, “Sub-word level lip reading with visual attention,” in *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, IEEE, 2022, pp. 5152–5162. DOI: 10 . 1109 / CVPR52688 . 2022 . 00510. [Online]. Available: <https://doi.org/10.1109/CVPR52688.2022.00510>.
- [283] D. Serdyuk, O. Braga, and O. Siohan, “Audio-visual speech recognition is worth 32\times 32\times 8\$ voxels,” in *IEEE Automatic Speech Recognition and Understanding Workshop, ASRU 2021, Cartagena, Colombia, December 13-17, 2021*, IEEE, 2021, pp. 796–802. DOI: 10.1109/ASRU51503.2021.9688191. [Online]. Available: <https://doi.org/10.1109/ASRU51503.2021.9688191>.
- [284] ———, “Transformer-based video front-ends for audio-visual speech recognition for single and multi-person video,” in *Interspeech 2022, 23rd Annual Conference of the International Speech Communication Association, Incheon, Korea, 18-22 September 2022*, H. Ko and J. H. L. Hansen, Eds., ISCA, 2022, pp. 2833–2837. DOI: 10.21437/Interspeech.2022-10920. [Online]. Available: <https://doi.org/10.21437/Interspeech.2022-10920>.
- [285] Z. Zhang, J. Zhang, J. Zhang, M. Wu, X. Fang, and L. Dai, “Learning contextually fused audio-visual representations for audio-visual speech recognition,” *CoRR*, vol. abs/2202.07428, 2022. arXiv: 2202.07428. [Online]. Available: <https://arxiv.org/abs/2202.07428>.
- [286] K. Irie, A. Zeyer, R. Schlüter, and H. Ney, “Language modeling with deep transformers,” in *Interspeech 2019, 20th Annual Conference of the International Speech Communication Association, Graz, Austria, 15-19 September 2019*, G. Kubin and Z. Kacic, Eds., ISCA,

- 2019, pp. 3905–3909. DOI: 10.21437/Interspeech.2019-2225. [Online]. Available: <https://doi.org/10.21437/Interspeech.2019-2225>.
- [287] J. S. Chung, A. W. Senior, O. Vinyals, and A. Zisserman, “Lip reading sentences in the wild,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, IEEE Computer Society, 2017, pp. 3444–3453. DOI: 10.1109/CVPR.2017.367. [Online]. Available: <https://doi.org/10.1109/CVPR.2017.367>.
- [288] C. Hazirbas, J. Bitton, B. Dolhansky, J. Pan, A. Gordo, and C. Canton-Ferrer, “Towards measuring fairness in AI: the casual conversations dataset,” *IEEE Trans. Biom. Behav. Identity Sci.*, vol. 4, no. 3, pp. 324–332, 2022. DOI: 10.1109/TBIOM.2021.3132237. [Online]. Available: <https://doi.org/10.1109/TBIOM.2021.3132237>.
- [289] J. Ho, A. Jain, and P. Abbeel, “Denoising diffusion probabilistic models,” in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., 2020. [Online]. Available: <https://proceedings.neurips.cc/paper/2020/hash/4c5bcfec8584af0d967f1ab10179ca4b-Abstract.html>.
- [290] P. Dhariwal and A. Q. Nichol, “Diffusion models beat gans on image synthesis,” in *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, M. Ranzato, A. Beygelzimer, Y. N. Dauphin, P. Liang, and J. W. Vaughan, Eds., 2021, pp. 8780–8794. [Online]. Available: <https://proceedings.neurips.cc/paper/2021/hash/49ad23d1ec9fa4bd8d77d02681df5cfa-Abstract.html>.
- [291] Z. Kong, W. Ping, J. Huang, K. Zhao, and B. Catanzaro, “Diffwave: A versatile diffusion model for audio synthesis,” in *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*, OpenReview.net, 2021. [Online]. Available: <https://openreview.net/forum?id=a-xFK8Ymz5J>.

- [292] Y. Koizumi, H. Zen, K. Yatabe, N. Chen, and M. Bacchiani, “Specgrad: Diffusion probabilistic model based neural vocoder with adaptive noise spectral shaping,” in *Interspeech 2022, 23rd Annual Conference of the International Speech Communication Association, Incheon, Korea, 18-22 September 2022*, H. Ko and J. H. L. Hansen, Eds., ISCA, 2022, pp. 803–807. DOI: 10.21437/Interspeech.2022-301. [Online]. Available: <https://doi.org/10.21437/Interspeech.2022-301>.