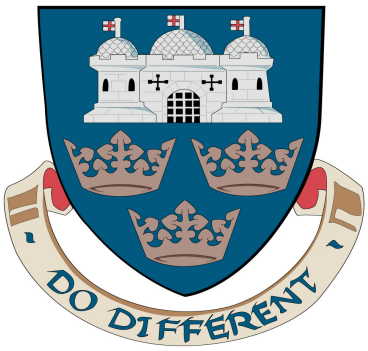
Audio speech enhancement using masks derived from visual speech

Danny Roy Websdale

A thesis submitted for the Degree of
Doctor of Philosophy

University of East Anglia
School of Computing Sciences



September 2018

Abstract

The aim of the work in this thesis is to explore how visual speech can be used within monaural masking based speech enhancement to remove interfering noise, with a focus on improving intelligibility. Visual speech has the advantage of not being corrupted by interfering noise and can therefore provide additional information within a speech enhancement framework. More specifically, this work considers audio-only, visual-only and audio-visual methods of mask estimation within deep learning architectures with application to both seen and unseen noise types.

To estimate masks from audio and visual speech information, models are developed using deep neural networks, specifically feed-forward (DNN) and recurrent (RNN) neural networks for temporal modelling and convolutional neural networks (CNN) for visual feature extraction. It was found that the proposed layer normalised bi-directional feed-forward hybrid network using gated recurrent units (LNBiGRU-DNN) provided best performance across all objective measures for temporal modelling. Also, extracting visual features using both pre-trained and end-to-end trained CNNs outperform traditional active appearance model (AAM) feature extraction across all noise types and SNRs tested. End-to-end CNNs trained on images focused on mouth-only regions-of-interest provided best performance for both audio-visual and visual-only models.

The best performing audio-visual masking method outperformed both audio-only and visual-only masking methods in both matched and unseen noise type and SNR dependent conditions. For example, in unseen cafeteria babble noise at $-10\,\mathrm{dB}$, audio-visual masking had an ESTOI of 46.8, while audio-only and visual-only mask-

ing scored 15.0 and 42.4, and the unprocessed audio scored 9.3. Formal tests show that visual information is critical for improving intelligibility at low SNRs and for generalisation to unseen noise conditions. Experiments in large unconstrained vocabulary speech confirm that the model architectures and approaches developed can generalise to unconstrained speech across noise independent conditions and can be considered for monaural speaker dependent real-world applications.

Acknowledgements

First and foremost I would like to express my sincere gratitude to my supervisor, Dr Ben Milner, for his continuous support, encouragement and patience throughout my studies. Furthermore, I extend my thanks to my secondary supervisors, Prof. Stephen Cox, Prof. Graham Finlayson and Dr Philip Glasson, for their additional help, to my examiners Dr Wenwu Wang and Prof. Amir Hussain for their insights and comments, and also to Prof. Richard Harvey for officiating my Viva.

My heartfelt gratitude goes to my family, my parents, Roy and Lorraine, my sister, Debbie, my brother, Harry, and my grandmother, Christine, for their continued, unrivalled support throughout my life and in all my endeavours. Thanks also go to all my friends, both old and new, for all the laughter, advice and support.

Finally, I would like to thank the UEA and the UK Home Office – Centre for Applied Science and Technology, for supporting this work.

Contents

List of PhD publications

Websdale, D., Le Cornu, T., and Milner, B. (2015). Objective measures for predicting the intelligibility of spectrally smoothed speech with artificial excitation. *Proceedings of Interspeech*.

Websdale, D. and Milner, B. (2015). Analysing the importance of different visual feature coefficients. *Proceedings of FAAVSP*.

Websdale, D. and Milner, B. (2017a). A comparison of perceptually motivated loss functions for binary mask estimation in speech separation. *Proceedings of Interspeech*.

Websdale, D. and Milner, B. (2017b). Using visual speech information and perceptually motivated loss functions for binary mask estimation. *Proceedings of AVSP*.

List of Abbreviations

A	Audio-Only
AAM	Active Appearance Model
ANOVA	Analysis Of Variance
AMS	Amplitude Modulation Spectrum
ARPMG	AMS RASTA-PLP MFCC GFB
ASA	Auditory Scene Analysis
AV	Audio-Visual
ASR	Automatic Speech Recognition
BiGRU	Bi-Directional Recurrent Neural Network With GRU Units
BiGRU-DNN	Bi-Directional Recurrent Feed-Forward Hybrid Neural Network With LSTM Cells
BiLSTM	Bi-Directional Recurrent Neural Network With LSTM Cells
BSS	Blind Source Separation
CASA	Computational Auditory Scene Analysis
CNN	Convolutional Neural Network
CE	Binary Cross-Entropy Loss Function
CEHF	Binary Cross-Entropy HIT-FA Hybrid Loss Function
DNN	Deep Feed-Forward Neural Network
ERB	Equivalent Rectangular Bandwidth
ESTOI	Extended Short-Time Objective Intelligibility
FA	False Alarm
FN	False Negative
FP	False Positive
GFB	Gammatone Filterbank
GMM	Gaussian Mixture Model
GRU	Gated Recurrent Unit
HF	HIT-FA Loss Function

HIT-FA	Hit Minus False Alarm
IBM	Ideal Binary Mask
ILSVRC	ImageNet Large Scale Visual Recognition Challenge
IRM	Ideal Ratio Mask
LC	Local Criterion
LNBiGRU	Layer Normalised Bi-Directional Recurrent Neural Network With GRU Units
LNBiGRU-DNN	Layer Normalised Bi-Directional Recurrent Feed-Forward Hybrid Neural Network With LSTM Cells
LSTM	Long Short-Term Memory
MFCC	Mel-Frequency Cepstral Coefficients
MLP	Multilayer Perceptrons
MMSE	Minimum Mean Square Error
MSE	Mean Square Error
MRCG	Multi-Resolution Cochleagram
PBM	Predicted Binary Mask
PCA	Principal Component Analysis
PESQ	Perceptual Evaluation Of Speech Quality
PRM	Predicted Ratio Mask
RASTA-PLP	Relative Spectral Transformed Perceptual Linear Prediction
ReLU	Rectified Linear Units
RGB	Red-Green-Blue
ROI	Regions-Of-Interest
RNN	Recurrent Neural Network
RNN-DNN	Recurrent Feed-forward Hybrid Neural Network
SNR	Signal-To-Noise Ratio
SVD	Singularity Value Decomposition
SVM	Support Vector Machine
T-F	Time-Frequency
TN	True Negative
TP	True Positive
TTS	Text-To-Speech
V	Visual-Only

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation and problem statement

In real-world scenarios, an audio speech signal can be contaminated or distorted by many artefacts. These include acoustic noise, channel distortion, echo, packet loss and codec distortion. These effects will affect both the intelligibility and quality of the resulting speech signal. To remove or reduce these effects and thereby improve the quality of the signal, speech enhancement is necessary. This can remove or reduce these through noise removal or suppression (Boll [1979]; Lim and Oppenheim [1978]; Ephraim and Malah [1984]; Kim et al. [2009]), channel equalisation (Chen et al. [1993]), echo cancellation (Benesty et al. [1998]), packet loss mitigation (Rappaport et al. [1996]; Perkins et al. [1998]) and codec distortion reduction (Ikeda and Sugiyama [1999]). However, this thesis focuses on speech enhancement for removing the effect of interfering noise on the perception of speech.

Noise has two main effects on the perception of speech. Firstly, the perceived quality of the speech signal is reduced by introducing artefacts and distortions onto the speech signal. This results in a signal that is unpleasant and increases fatigue for the listener, particularly in situations where the listener is exposed to high levels of noise for long periods of time. Secondly, the perceived intelligibility of the speech

signal is also effected. With the introduction of high levels of noise, the recognition of words uttered from the speech signal is reduced for human listeners.

The need to enhance speech signals arises in many situations where the speech signal is corrupted by noise from the source environment. Many applications desire speech enhancement for improving either the perceived quality or intelligibility. Voice communication over cellular telephone systems typically suffer from background noise in the environment, for example, whether in a car or restaurant. Similarly, video conferencing suffer from background noise caused by competing speakers. Speech enhancement algorithms can be used to remove or suppress the background noise from restaurants or competing speakers, to improve the quality of the received speech signal. In an air-ground communication systems between a pilot and ground control, the speech signal is corrupted in high levels of background environment noise from the cockpit. However, although improving quality is still important, improving intelligibility is more important. The information from the pilot is likely to be more important and desired by ground control over a better quality signal. Also for military communication applications, the improvement of intelligibility is more desired than quality. For hearing-aid and cochlear implant applications, improving both the quality and intelligibility of the enhanced signal is important for improving the quality of life of people suffering from hearing impairment. Within each example, acoustic information is captured through microphones, however few also capture visual information. Visual information within video conferencing could be used to track the current speaker, cameras could be placed within the cockpit to track the pilots speech, and for hearing impairment based applications, video could be captured through body cameras and smart glasses.

From the examples shown, the desired result from speech enhancement depends on the application, whether to improve quality, intelligibility or both. Ideally, speech enhancement would improve both quality and intelligibility. Generally, most speech enhancement algorithms which reduce background noise focus on the improvement of quality, at the expense of introducing speech distortions, which can reduce the re-

sulting intelligibility. The challenge, and focus of this thesis, is to remove or suppress background noise without introducing perceptual distortions, targeting the improvement of intelligibility. The remainder of this introduction first states the aims of the work, before providing some background on speech enhancement algorithms, applications using visual speech and an overview of deep learning.

1.2 Aims

The overall goal of this work is improving the perceived intelligibility of speech corrupted with interfering noise from the environment, through noise removal speech enhancement. This is explored with the following aims:

1.) **Visual speech for speech enhancement** Visual speech is the visual modality of the speaker, capturing information about mouth, lips, and other visual articulators of speech production. The visual modality is not affected or degraded by interfering noise, and as such offers a robust *clean* source of information. This work explores using visual speech and acoustic speech within the speech enhancement system for counteracting the affect of noise through audio-only, visual-only and audio-visual models.

2.) **Deep learning architecture for modelling** The supervised learning framework selected is key to the success and accuracy of the speech enhancement system. The emergence of deep learning through neural networks has provided large gains in performance across the majority of supervised learning tasks. This work explores and proposes using deep feed-forward (DNN), recurrent (RNN) and convolutional (CNN) neural networks for modelling within the supervised learning speech enhancement framework.

3.) **Feature extraction from input data** The features extracted from input data are key to allow modelling to learn strong relationships between inputs and target outputs. This work uses acoustic and visual information as input,

and as such explores both traditional feature extraction methods and feature extraction via deep learning through CNNs for extracting acoustic and visual information.

4.) **Generalisation to unseen noise conditions** In real-world applications, the environmental noise condition is likely to change in both noise type and noise level (SNR) across time, and as such the speech enhancement model must be able to perform across varying conditions. This work explores building noise type and SNR independent models to allow generalisation to new unseen noise conditions.

1.3 Speech enhancement for noise removal

The solution to the general problem of speech enhancement for noise removal depends on the desired application environment, types of noise present and the number of available microphones. The noise type can be noise-like, such as fans spinning or the engine and road surface when in a vehicle, or can be speech-like such as a restaurant or cafe with competing speakers. The number of available microphones has a significant impact on the performance of the speech enhancement algorithm. Generally, the larger number of microphones available, the easier the task becomes (single-channel or monaural verses multi-channel).

Given a simple example of two microphones, one microphone can be placed next to the speaker, and the second can be placed next to the noise source, providing two strong information streams for the speech enhancement algorithm. When a single microphone is used, only a single source containing both the speech and noise information is provided to the speech enhancement algorithm. Adaptive cancellation can be used when at least one microphone is placed near the noise source. When considering applications using audio and video, one source is provided from the acoustic stream and a second source is provided from the visual stream. This can be treated as a source of the speaker (video) and a source of the noise plus speaker

(audio). This provides an extension to single-microphone applications, allowing a source to contain only information of the speaker and not noise, albeit only visual information, but is not as effective as multi-microphone setups which can provide a source containing only noise information.

This work focuses on speech enhancement in additive noise with only a single microphone available (monaural). Figure 1.1 shows a simplified pipeline of an audio-only speech enhancement system. The speech enhancement algorithm takes as input the noisy (speech plus noise) signal, and outputs the enhanced signal.

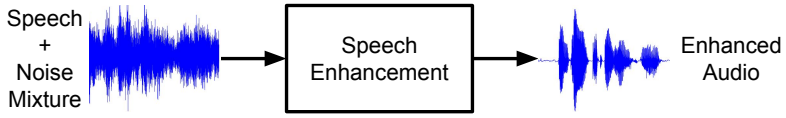


Figure 1.1: Overview of a simplified audio-only speech enhancement pipeline.

The speech enhancement algorithm can take the form of various approaches, including those based on filtering, such as spectral subtraction, statistical-model-based, subspace and masking algorithms, and those based on reconstructing or synthesising a clean audio signal. Spectral subtractive algorithms are the simplest form of speech enhancement algorithm to implement, and can only be used for additive noise environments. Such algorithms (Weiss et al. [1975]; Boll [1979]) make an estimate of the noise when no speech is present in the signal (i.e only noise is present in the signal), and subtract this estimate from the combined noisy speech to leave only the speech signal. Statistical-model-based algorithms form the problem in a statistical estimation framework. Given measurements of the noisy speech, a linear or non-linear transform is calculated to estimate parameters of the clean speech. Common algorithms using this approach are Weiner filtering (Lim and Oppenheim [1978, 1979]) and minimum mean square error (MMSE) (Ephraim and Malah [1984]). Subspace algorithms are based on linear algebra on the principle that the clean speech signal could be confined to a subset of the noisy speech. Decomposition of the noisy signal into subspaces primarily confined of speech signal and subspaces primarily confined to the noise signal, the enhanced signal can be produced by nulling

or removing the noise signal subspaces. The decomposition of the signal into subspaces can be achieved using well-known orthogonal matrix factorisation techniques from linear algebra, namely singularity value decomposition (SVD) (Dendrinos et al. [1991]) and eigenvector-eigenvalue factorisation (Ephraim and Van Trees [1995]).

Masking algorithms originate from the field of auditory scene analysis (ASA) (Brown and Cooke [1994]; Weintraub [1985]), forming computational auditory scene analysis (CASA) (Wang and Brown [2006]), and produce a time-frequency (T-F) mask derived from ideal separate clean and noise sources (i.e sources containing only clean or noise information) and train a model to predict such masks using the combined noisy speech as input (Kim et al. [2009]; Chen et al. [2014]; Zhao et al. [2016]; Healy et al. [2017]; Chen and Wang [2018]). The mask represents time-frequency (T-F) units that are either speech dominant or noise dominant, an ideal mask is calculated given a specific criterion using the separate sources, the criterion is not unique to each source, but produces masks which are specific for each speech plus noise combination. The criterion determines which T-F units are speech or noise dominant, and produces a mask such that the speech dominant units are kept (retained) and the noise dominant units are removed (suppressed). Criterion functions can either produce binary masks, where T-F unit values are fixed to either 0 (noise dominant) or 1 (speech dominant), or ratio masks, where T-F unit values are between 0 and 1 representing the proportion of speech present in each T-F unit. The mask can then be applied to the noisy speech signal, suppressing noise dominant T-F units yet retaining speech dominant T-F units, producing the enhanced audio signal. The algorithm then uses a model trained with known noisy speech and ideal mask pairings to predict and output the ideal mask. The model can then be used in *live* noisy conditions, when separate clean and noise sources are not available, to predict an estimation of an ideal mask. This estimated mask can then be applied to the noisy speech signal to produce the enhanced speech signal.

Speech reconstruction or synthesis based speech enhancement aims to reconstruct clean speech from speech parameters instead of retrieving clean speech from filtering

noisy speech. For speech reconstruction applications (Harding and Milner [2012, 2015]; Kato and Milner [2016]), parameters are estimated from the noisy speech to drive a speech production model. The model requires acoustic parameters of spectral envelope, fundamental frequency, phase and voicing (voiced, unvoiced on non-speech). The main benefit of using a speech reconstruction model as opposed to filtering is the constraints applied to the reconstructed signal. Reconstruction models are designed to only reconstruct components of the signal relating to speech, and therefore artefacts, that result from inaccurate estimation of noise contribution for filtering based algorithms, are not reconstructed. Speech reconstruction can also be used as a post-filter to filtering based speech enhancement algorithms to reduce these artefacts in the enhanced signal.

Spectral subtraction, statistical-model-based and subspace algorithms have been shown to improve perceived quality and reduce listener fatigue, but do not improve intelligibility, however masking algorithms have been shown to improve intelligibility in monaural conditions (Kim et al. [2009]). Speech reconstruction has been shown to be effective at removing noise from noisy speech, however errors in estimating the spectral envelope and voicing classification can lead to artefacts in the reconstructed speech reducing quality compared to filtering based algorithms (Harding and Milner [2015]).

The main focus of this work is on improving speech intelligibility, therefore masking algorithms are selected as the speech enhancement framework. The key part of the masking algorithms is the model used to predict ideal masks. The model has to learn a relationship from the noisy input signal and the target ideal masks. This allows speech enhancement to be treated as a mask estimation problem that uses supervised learning to map features extracted from noisy speech to an ideal mask. Therefore, the effectiveness of the model is determined by the criterion function, quality of the features extracted from the input data, and the framework or architecture of the model used in supervised learning. This provides three main areas of focus: selection of criterion function (binary or ratio masking), model input through

input data and feature extraction, and model architecture through supervised learning framework.

1.4 Visual speech processing

Using audio as the only source of information within a monaural speech enhancement system may be limited when applied to speech contaminated in high levels of noise. In high levels of noise most of the speech signal is lost and overpowered by the interfering noise. For speech enhancement using masking algorithms this introduces difficulties in determining which time-frequency (T-F) units within the mask are speech dominant or noise dominant causing over-suppression and under-suppression of noise within the enhanced speech. Over-suppressions are caused by incorrectly labelling speech dominant T-F units as noise dominant T-F units. This incorrectly suppresses speech dominant T-F units, causing the speech enhancement algorithm to suppress both noise and speech, resulting in the enhanced speech signal to lose speech content. Under-suppressions are caused by incorrectly labelling noise dominant T-F units as speech dominant T-F units. This incorrectly retains noise dominant T-F units, causing the speech enhancement algorithm to not fully suppress the noise, resulting in interfering noise to be retained in the enhanced speech signal. Effects introduced from over-suppressing or under-suppressing will reduce the resulting intelligibility (Loizou and Kim [2011]).

One way to avoid this problem is to exploit visual speech information. This has had success in a range of audio-only speech processing applications, such as automatic speech recognition (ASR), voice activity detection, speaker separation and speaker verification, have used information extracted from the visual modality, captured from video of the speaker, to counteract the affect from interfering noise on performance. The visual modality captures information about the mouth, lips, and other visual articulators of the speaker. Unlike the acoustic signal, visual information is not affected or degraded by interfering noise, and as such offers a robust *clean*

source of information for these applications.

Audio-visual ASR uses visual information to complement audio information for improving recognition accuracy in noise (Heckmann et al. [2002]; Liang et al. [2002]; Potamianos et al. [2003]; Thangthai et al. [2015]). However, using visual information for ASR does not only provide benefits in noisy conditions, but can also help the ASR system in clean conditions. This is because certain acoustic sounds are easier to recognise in the visual modality. For instance, /b/ (a bilabial) and /d/ (an alveolar), or nasal sounds like /m/ (a bilabial) and /n/ (an alveolar) are visually distinct but often confused in the audio domain (Potamianos et al. [2004]). Due to the benefits of the visual-stream, ASR applications have explored removing the acoustic stream and applied within a visual-only setting, called lip-reading (Matthews et al. [2002]; Lan et al. [2009, 2010]; Chung and Zisserman [2016]; Assael et al. [2016]; Thangthai et al. [2018]). Such applications, such as surveillance, can be applied where the acoustic information is not present. However, using visual-only information can add confusion into the ASR system, as certain acoustic sounds are produced with similar visual articulation. For instance, acoustic sounds such as /b/, /p/, and /m/ (all bilabials) are impossible to separate in their visual appearance.

Given the robustness of the visual modality to acoustic noise, voice activity detection has benefited from using either visual-only information (Sodoyer et al. [2006]; Liu et al. [2014]) or both audio and visual information (Almajai and Milner [2008]), where contributions between the audio and visual streams can be weighted depending on the SNR of the interfering noise. Furthermore, monaural speaker separation systems, which attempt to extract speech of a target speaker from signals containing speech of two or more speakers, have also shown improvements over audio-only systems by integrating visual information (Wang et al. [2005]; Khan and Milner [2013, 2015]; Khan et al. [2018]). The visual modality tends to be more speaker dependent, and harder to spoof compared to the audio modality, and as such has been applied for speaker verification (Dean and Sridharan [2010]; Fox and Reilly [2003]).

Due to the benefits of combining audio and visual modalities within many ap-

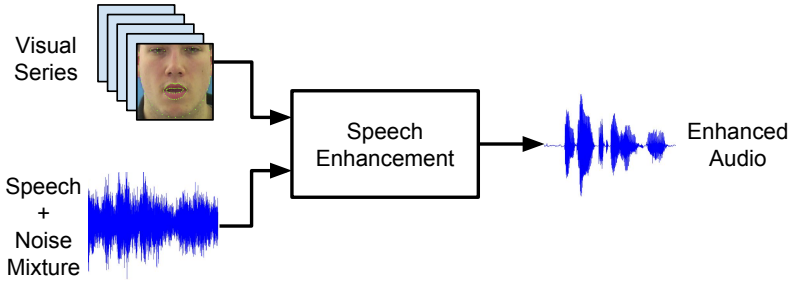


Figure 1.2: Overview of a simplified audio-visual speech enhancement pipeline.

plications, and the robustness of visual information to noise, audio-visual speech enhancement has also been considered for improving the perceived quality of the enhanced signal. Not only have the approaches taken increased in complexity, but the method of visual feature extraction has also increased in complexity over time. Initial work in (Rivet et al. [2007]) found success by combining audio and simple visual information (height and width of inner lip) within blind source separation (BSS) for solving indeterminacy problems, albeit within a highly constrained and limited dataset. In (Almajai and Milner [2009]), visual information (2D-DCT features) was used to estimate both clean speech and noise filterbanks used within Weiner filtering, showing further improvements over audio-only systems and approaches which previously only estimated the clean speech from audio-visual information (Girin et al. [2001]; Berthommier [2004]). The approach in (Liu et al. [2013]) used audio-visual dictionary learning, using Active Appearance Models (AAM) for visual feature extraction, for time-frequency masking. This method produced separate audio-only and visual-only masks, before combining them into a single audio-visual fusion mask used to enhance speech. Findings showed how combining both audio-only and visual-only generated masks provided improved quality over the audio-only mask and higher resolution over the visual-only mask. An overview of key methodologies within audio-visual speech enhancement is provided in (Rivet et al. [2014]). Figure 1.2 extends the previously defined audio-only speech enhancement system to an audio-visual speech enhancement system. This work also considers the performance of visual-only speech enhancement with the acoustic stream removed.

1.5 Deep learning

The supervised learning framework selected is key to the success and accuracy of the speech enhancement system. For speech enhancement using masking algorithms, the model used is required to learn a mapping from input features to target masks. This is achieved within model training, where examples of known noisy speech and ideal mask pairings are used to train a model capable of learning target masks. The model can then be used in *live* noisy conditions, when separate clean and noise sources are not available and as such an ideal mask cannot be produced, to predict an estimation of an ideal mask. This estimated mask can then be applied to the noisy speech signal to produce the enhanced speech signal.

Previous audio-only speech enhancement systems have shown a clear progression on the supervised learning framework used to train such models. The models used have progressed from simple multilayer perceptrons (MLPs) (Hu and Wang [2004]; Jin and Wang [2009]; Chen et al. [2014]), Gaussian mixture models (GMMs) (Kim et al. [2009]), support vector machines (SVMs) (Han and Wang [2012]; Wang and Wang [2013]) and finally deep neural networks (DNNs) (Xu et al. [2015b]; Yu et al. [2016]; Zhao et al. [2016]; Healy et al. [2017]; Chen and Wang [2018]; Gogate et al. [2018]).

The emergence of deep learning through neural networks has also provided large gains in accuracy across the majority of supervised learning tasks. Such applications include speech recognition (Graves et al. [2013a]; Graves and Jaitly [2014]; Chung and Zisserman [2016]; Assael et al. [2016]; Thangthai et al. [2018]), text-to-speech (TTS) synthesis (Fan et al. [2014]), image classification (Krizhevsky et al. [2012]; Simonyan and Zisserman [2014]; Szegedy et al. [2015]; He et al. [2016a]) and object detection (Krizhevsky et al. [2012]), among many other applications. Therefore, this work explores and proposes using deep feed-forward (DNN), recurrent (RNN) and convolutional (CNN) neural networks for modelling within the supervised learning speech enhancement framework.

1.6 Thesis structure

The remainder of this thesis is organised as follows, where further literature reviews are provided in each chapter. An initial exploration into speech enhancement using binary masking is performed in Chapter 2, providing a baseline architecture for the thesis. Binary masking is explored within a deep feed-forward neural network (DNN) framework for audio-only, visual-only and audio-visual models and is considered as a classification problem. Traditional acoustic and visual feature extraction methods, model architecture and loss functions used for network training are optimised. Novel loss functions used within DNN training are proposed to maximise intelligibility with comparisons made to traditional classification loss functions.

Chapter 3 explores the use of ratio masking and compares against the results found for binary masking. Ratio masking is explored within a DNN framework, and is considered as a regression problem instead of classification found for binary masking. Traditional acoustic and visual feature extraction methods and model architecture are optimised for ratio masking. Comparisons are made between ratio masking and binary masking to evaluate the best performing masking algorithm.

The focus of Chapter 4 is improving the neural network architecture used within supervised learning. Previously, only feed-forward neural network architectures have been considered. In this chapter the use of recurrent neural networks (RNN) are explored, with a proposed recurrent feed-forward hybrid architecture introduced. Recurrent neural networks are designed and optimised for data with temporal structure and are therefore well suited for speech processing applications. Evaluations are provided using the best performing feature extraction methods and speech enhancement algorithm (binary or ratio masking) found in Chapter 3 for audio-only, visual-only and audio-visual models.

Only traditional feature extraction methods have been evaluated thus far, and instead Chapter 5 uses neural networks to perform feature extraction within the network architecture. The focus is on improving visual features which were found

to be important for speech enhancement at low signal-to-noise ratios (SNRs). Convolutional neural networks (CNN) are used to process raw images extracted from the video sequence instead of using traditional feature extraction methods. Images extracted from the video sequence are cropped to either a region-of-interest of the mouth-only or full-face and are upsampled via repetition or interpolation prior to input to the CNN. The use of pre-trained CNNs and end-to-end trained CNNs are compared within visual-only and audio-visual models.

All evaluations provided in Chapters 2 to 5 have optimised noise type and SNR dependent models within a small constrained vocabulary dataset. Chapter 6 evaluates the best performing audio-only, visual-only and audio-visual models for real-world conditions, by training noise type and SNR independent models with focus being generalisation to unseen noise conditions. Theses models are compared against the previous dependent models, before being applied to a larger unconstrained vocabulary dataset. It is found that visual information is critical for generalisation to unseen noise types and SNR conditions, and expanding to a large scale dataset has minimal performance degradation compared to the small dataset.

Finally, in Chapter 7 the work and results presented in this thesis on monaural speech enhancement using masks derived from acoustic and visual speech are summarised. Additionally, the limitations of this work are discussed with a number of possible avenues of future work outlined, with focus on extending to speaker independent applications and the adjustments required for use in real-time applications such as hearing aids or cochlear implants.

Chapter 2

Binary masking

2.1 Introduction

Chapter 1 presented a review of current speech enhancement algorithms, most algorithms are developed to improve perceived quality of the enhanced signal, whereas this work focuses on improving intelligibility. From this review, it was found that masking based algorithms are able to improve intelligibility, and are therefore selected as the speech enhancement framework in this work. Masking algorithms originate from the field of auditory scene analysis (ASA) (Brown and Cooke [1994]; Weintraub [1985]), forming computational auditory scene analysis (CASA) (Wang and Brown [2006]), and produce a channel or frequency mask derived from ideal separate clean and noise sources (i.e sources containing only clean or noise information) and train a model to predict such masks using the combined noisy speech as input (Kim et al. [2009]; Healy et al. [2013]; Chen et al. [2014]; Zhao et al. [2016]; Healy et al. [2017]; Chen and Wang [2018]).

The mask represents time-frequency (T-F) units that are either speech dominant or noise dominant, an ideal mask is calculated given a specific criterion using the separate sources. The criterion is not unique to each source, but produces masks which are specific for each speech plus noise combination. The criterion determines

which T-F units are speech or noise dominant, and produces a mask such that the speech dominant units are kept (retained) and the noise dominant units are removed (suppressed). Criterion functions can either produce ideal binary masks (IBM) or ideal ratio masks (IRM). In this chapter the focus is on binary masking based speech enhancement to develop a baseline system, and as such a binary criterion function is selected. Binary masks are constructed from values which are fixed to either 0 (noise dominant) or 1 (speech dominant), determined by whether the binary criterion function for each T-F unit is below (resulting in a 0) or above (resulting in a 1) the local criterion (LC), also known as the speech dominance threshold. The mask can then be applied to the noisy speech signal, suppressing noise dominant T-F units yet retaining speech dominant T-F units, producing the enhanced audio signal.

Several studies have reported subjective test results where ideal binary masking (IBM) improved intelligibility for speech in noise for both normal-hearing and hearing-impaired listeners (Ahmadi et al. [2013]; Brungart et al. [2006]; Li and Loizou [2008]; Wang et al. [2009]; Healy et al. [2013]). In practice an IBM is not available and instead the binary mask must be estimated from the noisy signal. This allows speech enhancement to be treated as a mask estimation problem that uses supervised learning to map features extracted from noisy speech to an ideal binary mask (Kim et al. [2009]; Han and Wang [2012]; Wang and Wang [2013]).

The algorithm then uses a model trained with known noisy speech and ideal mask pairings to predict and output the ideal mask. The model can then be used in *live* noisy conditions, when separate clean and noise sources are not available, to predict an estimation of an ideal mask (predicted mask). This predicted binary mask (PBM) can then be applied to the noisy speech signal to produce the enhanced speech signal. This work uses deep feed-forward neural networks (DNN) for modelling the relationship between input noisy speech and target binary masks, which have previously been shown to perform well for binary mask estimation (Healy et al. [2013]; Chen et al. [2014]).

This work considers two extensions to binary mask estimation. Firstly, this work

proposes the development of perceptually motivated loss functions within a DNN framework (Websdale and Milner [2017a,b]; Liu et al. [2017]). Most existing methods of binary mask estimation using DNNs maximise the classification accuracy of predicted masks. This is achieved by optimising the binary cross-entropy (CE) (Rubinstein and Kroese [2013]) loss function during training (Kim et al. [2009]; Healy et al. [2013]; Chen et al. [2014]). However, several studies have shown that the hit minus false alarm (HIT-FA) rate, where a hit is a correctly labelled speech dominant T-F unit (i.e correctly retaining speech) and a false alarm is an incorrectly labeled noise dominant T-F unit (i.e incorrectly retaining noise), of the predicted mask correlates more closely to speech intelligibility than classification accuracy (Healy et al. [2013]; Kim et al. [2009]; Chen et al. [2014]; Healy et al. [2015]; Chen et al. [2016]; Websdale and Milner [2017a,b]). Further details of the HIT-FA rate and classification accuracy are provided in Section 2.5.1. Therefore, we propose using perceptually motivated loss functions that are based on maximising the HIT-FA rate with the aim of increasing the intelligibility of the resulting masked speech (detailed in Section 2.5). Some attention has been focussed on improving the classifier to reduce perceptual error by changing the loss function for text-to-speech applications (Valentini-Botinhao et al. [2015]), and introducing signal approximation loss functions (Weninger et al. [2014]; Erdogan et al. [2015]) as a replacement for mask approximation within speech separation applications. Signal approximation loss functions apply the output of the network to the noisy spectrum within the loss function, and minimise this with respect to the target. Signal approximation works well when the network target is the power spectrum, outperforming mask approximation, however is not applicable to a cochleagram framework, due to the cochleagram is constructed from overlapping gammatone filterbanks.

This second extension to mask estimation investigates the use of visual speech information as a supplement to the acoustic information used in DNN training. The use of visual speech information in traditionally audio-only speech processing applications has given significant gains in performance in noisy conditions. For example, in automatic speech recognition (ASR), supplementing the audio with

visual features has reduced error rates in low SNR conditions (Thangthai et al. [2015]; Potamianos et al. [2003]; Heckmann et al. [2002]). A benefit of using visual speech information within mask estimation is that visual features are not degraded by acoustic noise, although in themselves they may not have the discriminative ability that audio features possess in terms of mask estimation. To investigate this we explore mask estimation, and subsequently speech intelligibility, by comparing audio-only, visual-only and audio-visual speech enhancement models. Figure 2.1 shows the training pipeline of the proposed audio-visual speech enhancement system using feed-forward neural networks. Visual features are extracted from video and combined with acoustic features extracted from noisy speech, before input into the feed-forward neural network (DNN) for temporal modelling to estimate the binary mask. For testing purposes, estimated masks are applied to a cochleagram of the noisy speech which suppresses noise-dominated T-F units and the remaining signals are overlapped and summed to produce the enhanced signal, shown in Figure 2.2. The same pipeline is used for all speech enhancement configurations, except the visual stream is removed for audio-only models, and the audio stream is removed for visual-only models.

The remainder of this chapter is organised as follows. Section 2.2 details how ideal binary masks are produced, and subsequently used for enhancement of noisy speech. Section 2.3 provides an overview of acoustic and visual feature extraction methods. The DNN architecture and training is introduced in Section 2.4, showing the importance of loss function selection. Section 2.5 first reviews the classification accuracy and HIT-FA rate objective measures (Section 2.5.1) before introducing the loss functions used in DNN training, specifically the standard binary cross-entropy (CE) (Section 2.5.2), and proposed HIT-FA (HF) (Section 2.5.3) and binary cross-entropy HIT-FA hybrid (CEHF) (Section 2.5.4) loss functions. Performance evaluations are made in Section 2.6 which first compare the effectiveness of the feature extraction methods outlined in Section 2.3 for audio-only, visual-only and audio-visual models using the CE loss function. Section 2.6.2 optimises the CEHF loss function for maximising HIT-FA rate. Section 2.6.3 compares the performance

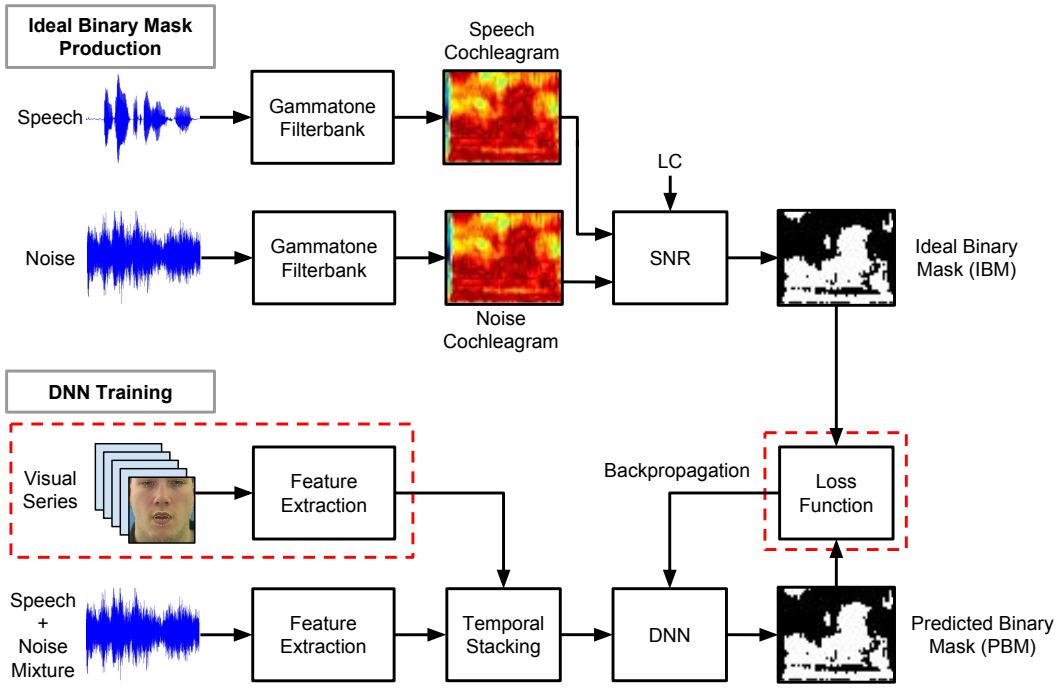


Figure 2.1: Overview of training the DNN binary masking speech enhancement system.

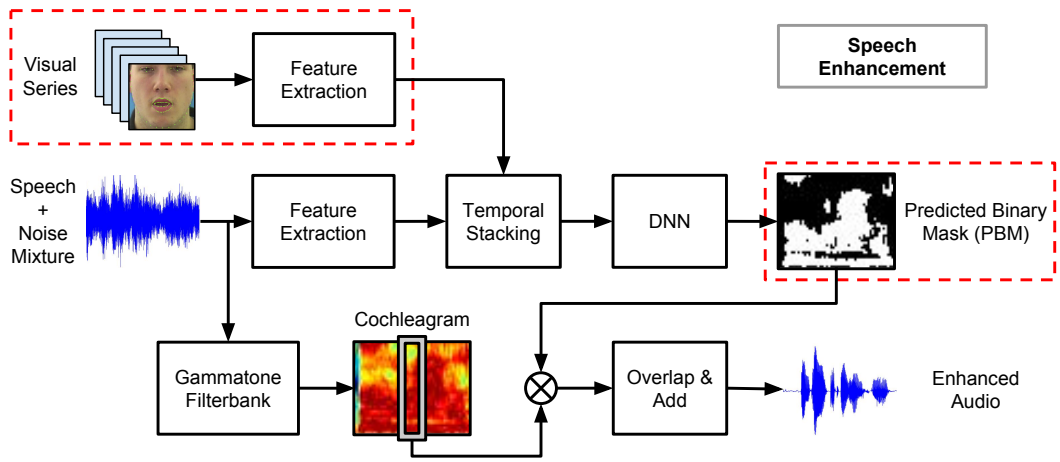


Figure 2.2: Overview of applying the DNN predicted binary mask to noisy speech for speech enhancement testing.

of all loss functions across varying noise type and SNR conditions and used the best performing feature extraction methods from Section 2.6.1. Finally, this chapter is concluded in Section 2.7.

2.2 Ideal binary masking

In CASA, enhanced speech is extracted by applying a mask to a time-frequency (T-F) representation of noisy speech. In ideal conditions separate clean and noise sources (i.e sources containing only clean or noise information) can be used to calculate an ideal binary mask (IBM). Figure 2.3 shows the pipeline for producing IBMs, where cochleagrams are produced from separate speech and noise sources for producing T-F units, the SNR between speech and noise cochleagrams is calculated before producing the IBM. Section 2.2.1 details the production of cochleagram based T-F units, before calculating the IBM in Section 2.2.2.

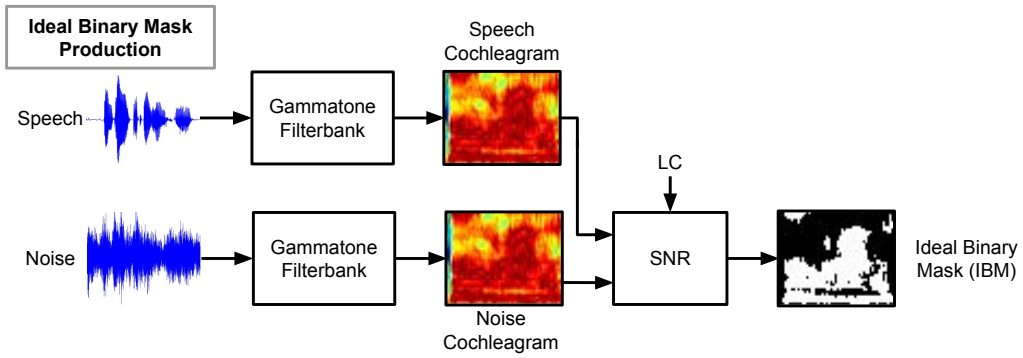


Figure 2.3: Overview of producing ideal binary masks (IBM).

2.2.1 Cochleagram production

In order to calculate ideal binary masks (IBM), the clean speech and noise source are decomposed from the time domain into time-frequency (T-F) units. The IBM criterion calculates which T-F units are speech dominant or noise dominant. The combined noisy speech time domain signal is decomposed into T-F units, the calculated mask is applied which subsequently suppresses noise dominant T-F units and retains speech dominant T-F units, before retuning back into the time domain producing the enhanced signal. From this, the decomposition into T-F units is key. Previous studies (Healy et al. [2013, 2015, 2017]; Wang and Brown [2006]; Chen et al. [2014]), have shown that using a cochleagram for producing T-F units is successful

for both binary and ratio masking, and as such is selected in this work.

The same cochleagram implementation in (Wang and Brown [2006]), is used in this work. Time domain signals (sampled at 16 kHz) are decomposed into T-F units by first applying a 64-channel gammatone filterbank (Patterson et al. [1987]), calculated as:

$$g_{f_c} = t^{N-1} \exp[-2\pi t b(f_c)] \cos(2\pi f_c t) u(t) \tag{2.1}$$

where f_c is the centre frequency, N is the filter order, $u(t)$ the step function and b is the bandwidth of the filter. The centre frequencies of the channels are uniformly spaced along the equivalent rectangular bandwidth (ERB) scale ranging from 50 Hz to 8 kHz, this is used to imitate the human auditory filters, providing the relation between $b(f_c)$ and f_c as:

$$b(f_c) = 1.019 \times \text{ERB}(f_c) = 1.019 \times 24.7 \times (4.37 \times f_c/1000 + 1) \tag{2.2}$$

The response signals from the filterbanks are then framed into 20 ms frames with 10 ms frame shift, hamming windowed and summed to provide the total energy of each windowed frame, producing the final cochleagram. Each 20 ms of the signal in the time domain is decomposed into 64 T-F units, each representing the energy contained within the gammatone filterbanks.

2.2.2 Ideal binary mask production

Ideal binary masks (IBM) determine whether T-F units within the noisy signal are speech dominant or noise dominant. An IBM is calculated from the premixed speech and noise sources using a criterion function. To calculate and IBM, first the SNR between speech and noise T-F units is calculated as:

$$\text{SNR}(t, f) = 10 \log_{10}(X(t, f)^2 / D(t, f)^2) \tag{2.3}$$

where X is the clean speech cochleagram, D the noise segment cochleagram and LC is the local criterion used as the threshold between speech and noise dominance. The production of cochleagrams is shown in Section 2.2.1). An IBM is then calculated as:

$$\text{IBM}(t,f) = \begin{cases} 1, & \text{if SNR}(t,f) \geq \text{LC} \\ 0, & \text{otherwise} \end{cases} \tag{2.4}$$

where t and f represent time frame and frequency bin respectively and LC is a local criterion. Speech dominant T-F units are assumed to have SNRs greater than or equal to LC, and are represented by 1 and retained. Noise dominant T-F units are assumed to have SNRs less than LC, and are represented by 0 and suppressed.

The choice of LC determines how many T-F units are deemed to be speech dominant or noise dominant. Ideally, and LC of 0 dB should be used, as any SNRs above 0 dB are speech dominant, any below are noise dominant. However, if the mixed noisy speech has an overall low SNR (below 0 dB), the majority of T-F units would be removed. This introduces artefacts into the enhanced signal, which will degrade resulting intelligibility. Previous studies have compared different values for LC (Loizou and Kim [2011]; Healy et al. [2015]; Chen et al. [2016]), which have shown an LC set 5 dB lower than the overall SNR provides best performance, and is subsequently used in this work for all SNRs.

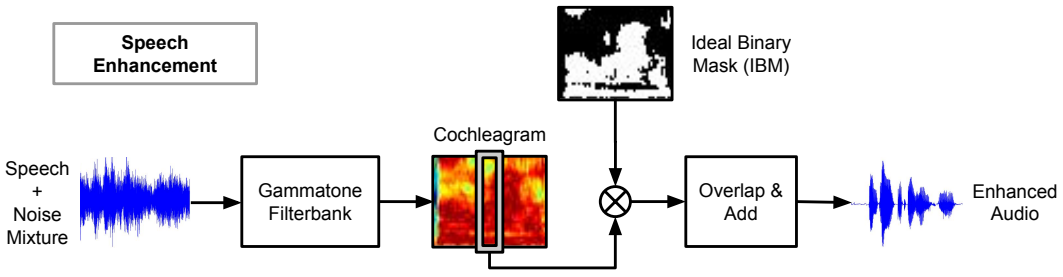


Figure 2.4: Overview of enhancing noisy speech through binary masking.

Calculated masks can then be applied to noisy speech producing the enhanced signal. Figure 2.4 shows the pipeline for generating enhanced signals. The combined

noisy speech time domain signal is decomposed into T-F units, following the same implementation to calculate a cochleagram, however the hamming windowed frames are not summed. The IBM is then multiplied with the hamming windowed frames, before returning back into a time domain signal through overlap and adding, and summing across gammatone filterbank responses. The same enhancement procedure is applied when testing different models, except the IBM is replaced with the predicted binary mask (PBM) produced from the model output.

2.3 Feature extraction for mask estimation

Feature extraction aims to identify suitably discriminative information in the noisy input speech and video that enables a model to determine whether T-F units are speech dominant (1) or noise domoninant (0). We investigate two different acoustic features and one visual feature extraction method.

2.3.1 Acoustic feature extraction

Acoustic feature extraction is applied to noisy speech (sampled at 16 kHz) for extracting suitably discriminant information from the noisy speech signal, used as input into the supervised learning model. Two acoustic feature extraction methods are considered, namely multi-resolution cochleagram feature (MRCG), specifically designed for cochleagram based speech enhancement, and an ensemble of complimentary features (ARPMG) traditionally used other speech processing applications.

2.3.1.1 Multi-resolution cochleagram feature (MRCG)

The multi-resolution cochleagram feature (MRCG) feature was designed specifically for masking based speech enhancement estimation and combines four cochleagrams at different resolutions (Chen et al. [2014]). The first captures high resolution localised detail while the remaining cochleagrams capture lower resolution spectrotem-

poral content. Cochleagrams are calculated using the same implementation outlined in Section 2.2.1.

Cochleagrams are computed by passing the input signal through a 64-channel gammatone filterbank. The response from the gammatone filterbanks are framed into 20 ms frames with 10 ms frame shift, hamming windowed and summed producing the energy of each frame. Unlike the approach in Section 2.2.1, a further log operation is applied, which gives the first cochleagram, CG_1. Similarly, CG_2 is calculated folowing the same method as CG_1, except frames are 200 ms in length with 10 ms frame shift. High resolution is captured by CG_1, and low resolution is captured by CG_2. Finally, CG_3 and CG_4 are derived by applying an 11×11 and 23×23 mean filter kernel to CG_1 respectively, capturing information across both time and frequency. The final MRCG feature, $\mathbf{x}^{\text{MRCG}}$, is produced by stacking all four CGs, an example is shown in Figure 2.5 for a clean utterance (clean utterance selected for clarity of diagram) of *"bin blue at e one now"*.

2.3.1.2 Complementary feature set (ARPMG)

The complementary feature set (ARPMG) is an ensemble of commonly used acoustic features (Chen et al. [2014]; Healy et al. [2015]; Chen et al. [2016]). This combines amplitude modulation spectrum (AMS) (Kollmeier and Koch [1994]; Tchorz and Kollmeier [2003]; Kim et al. [2009]), relative spectral transformed perceptual linear prediction (RASTA-PLP) (Hermansky and Morgan [1994]) and mel-frequency cepstral coefficients (MFCCs) (ETSI [2002]) with a gammatone filterbank (GFB) (Chen et al. [2014]).

The specific implementation used is taken from (Healy et al. [2015]) where AMS features are computed from 32 ms frames with 10 ms frame shift to give a 15-D vector. RASTA-PLP features are also computed from 32 ms frames with 10 ms frame shift and result in a 13-D vector. MFCCs are computed from 20 ms frames and 10 ms frame shift producing a 31-D vector. The GFB feature is computed from a 64-channel gammatone filterbank, decimating to 100 Hz to give a 10 ms frame shift

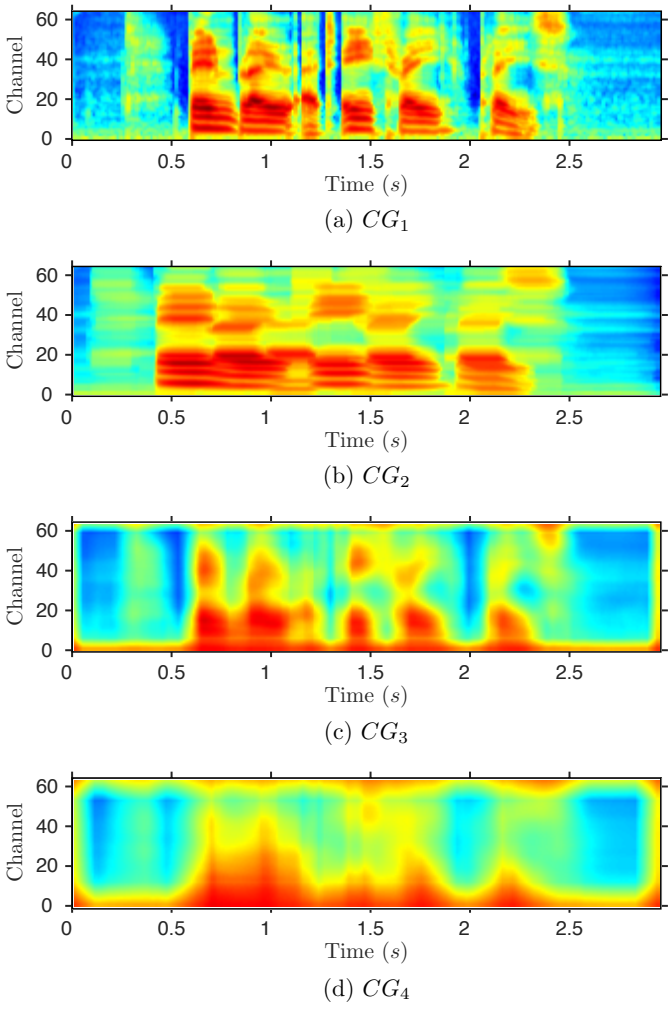


Figure 2.5: Multi-resolution cochleagram feature for clean utterance *"bin blue at e one now"*.

and results in a 64-D vector. Combining these gives the 123-D ARPMG feature, $\mathbf{x}^{\text{ARPMG}}$, which is produced at a 100 Hz frame rate.

2.3.2 Visual feature extraction

The visual feature selected is the active appearance model (AAM) which has proven to be an effective feature within visual-only and audio-visual ASR (Thangthai et al. [2015]; Lan et al. [2009]; Websdale and Milner [2015]) and is a model-based combination of shape and appearance. AAMs require labelled data with landmarks to generate features and use a model to perform this task automatically. The model

requires hand labelled training images to learn the variation in mouth shapes and in this work 43 training images were selected with 101 landmarks tracked. An example labelled image is shown in Figure 2.6, where 40 and 26 landmarks represent the outer and inner lip respectively, with extra landmarks for the eyes and jaw line, which assist the model in locating the face and fitting landmarks. A new model is produced by selecting only the mouth landmarks, and is used to produce AAM features, $\mathbf{x}^{\text{AAM}} = [\mathbf{s}_t\ \mathbf{a}_t]$, that comprise shape, $\mathbf{s}_t$, and appearance, $\mathbf{a}_t$, components for time t.

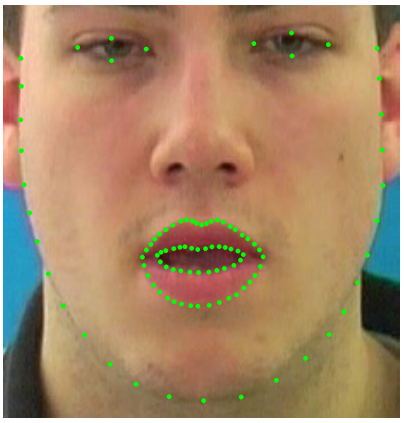


Figure 2.6: Hand labelled landmarks for AAM tracking and feature extraction.

The shape feature, $\mathbf{s}$, is obtained by concatenating n, x and y coordinates that form a two-dimensional mesh of the mouth, $\mathbf{s} = (x_1 y_1, ..., x_n y_n)^T$. A model that allows linear variation in shape is produced using PCA,

$$\mathbf{s} = \mathbf{s}_0 + \sum_{i=1}^{m} p_i \mathbf{s}_i \tag{2.5}$$

where $\mathbf{s}_0$ is the base shape, $\mathbf{s}_i$ are the shapes corresponding to the m largest eigenvectors and p_i are shape parameters. Coefficients comprising 90 % of the variation are selected, resulting in a vector size of 8 shape coefficients, $\mathbf{s}_t$.

The appearance feature, $\mathbf{a}$, is obtained from the pixels that lie inside the base mesh, $\mathbf{s}_0$ (Cootes et al. [2001]). As with the shape model, an appearance model, $\mathbf{a}$,

can also be expressed with linear variation,

$$\mathbf{a} = \mathbf{a}_0 + \sum_{i=1}^{m} q_i \mathbf{a}_i \tag{2.6}$$

where $\mathbf{a}_0$ is the base appearance, $\mathbf{a}_i$ are the appearances that correspond to the m largest eigenvectors and q_i are appearance parameters. Coefficients comprising 95 % of the variation are selected, giving a vector size of 15 appearance coefficients, $\mathbf{a}_t$. Combining the shape and appearance features gives an AAM vector, $\mathbf{x}^{\text{AAM}}$, with 23 dimensions which is extracted from the video at a rate of 25fps. Due to the difference in frame rates between acoustic and AAM features, visual features are upsampled through interpolation to that of the acoustic features.

2.4 Feed-forward neural network (DNN) architecture and training

The supervised learning framework selected is key to the success and accuracy of the speech enhancement system. For speech enhancement using masking algorithms, the model used is required to learn a mapping from input features to target ideal binary masks (IBM). This is achieved within model training, where examples of known noisy speech plus video and ideal mask pairings are used to train a model capable of learning target masks. The model can then be used in *live* noisy conditions, when separate clean and noise sources are not available and as such an ideal mask cannot be produced, to predict an estimation of an ideal mask. This estimated mask can then be applied to the noisy speech signal to produce the enhanced speech signal.

Previous speech enhancement systems have shown a clear progression on the supervised learning framework used to train such models. The models used have progressed from simple multilayer perceptrons (MLPs) (Hu and Wang [2004]; Jin and Wang [2009]; Chen et al. [2014]), Gaussian mixture models (GMMs) (Kim et al. [2009]), support vector machines (SVMs) (Han and Wang [2012]; Wang and Wang

[2013]) and finally deep neural networks (DNNs) (Zhao et al. [2016]; Chen and Wang [2018]).

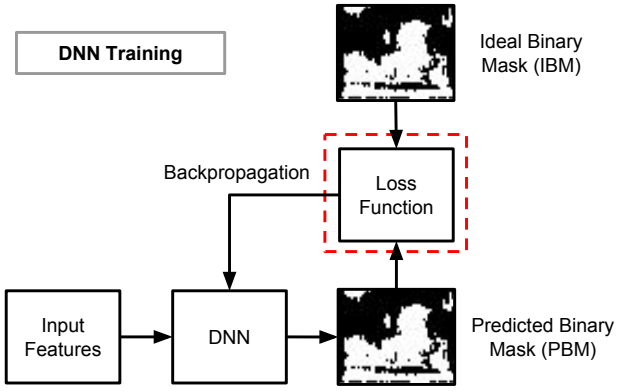


Figure 2.7: Overview of training DNNs for binary mask estimation.

This work uses DNNs as the model for learning the mapping between input features and target masks. Figure 2.7 shows an overview of how DNNs are trained. The DNN takes features as input and outputs a predicted binary mask (PBM). The error between the PBM and IBM is calculated through a loss function, and passed back through the DNN through backpropagation. The DNN updates internal weighting, and the procedure of calculating error is repeated, with error minimisation being the overall goal.

Feed-forward neural networks are constructed from a series of fully-connected layers. Layers consist of a number of nodes, each take as input all outputs from the previous layer for calculating a single output per node, where a non-linear activation function is then applied to the output of certain layers. A simple single-layer DNN is shown in Figure 2.8, which learns a mapping from input sequence, $\mathbf{x} = [\mathbf{x}_1; \mathbf{x}_2; \mathbf{x}_3]$, to target $\mathbf{y}$. The DNN consists of a single hidden layer, $\mathbf{h}$, and outputs an estimation of the target output, $\hat{\mathbf{y}}$. The hidden layers perform feature extraction by learning non-linear combinations of the inputs, where individually the features may not be particularly descriptive (Murphy [2012]).

The output of the hidden layer, $\mathbf{h}$, is calculated as:

$$\mathbf{h} = \sigma(\mathbf{W}^\intercal \mathbf{x} + b) \tag{2.7}$$

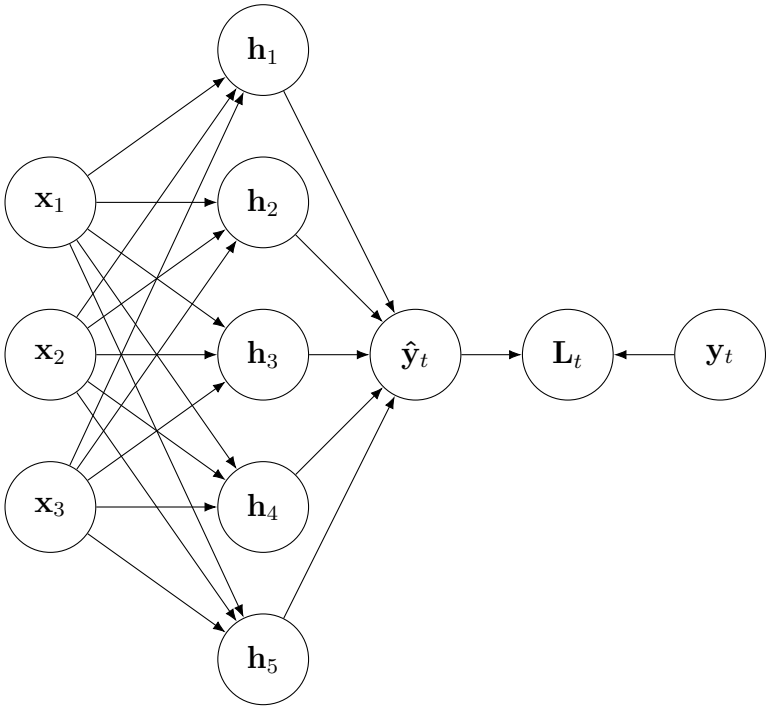


Figure 2.8: Computation of a typical feed-forward neural network.

which is a function of the input parameters, $\mathbf{x}$, and the weighted connections between the two layers, $\mathbf{W}$. A bias term, b, is included to provide each neuron (or node) in the input layer with a constant output, performing a similar role to the intercept in standard linear regression. In practice, however, the bias terms are usually incorporated into the weight parameter matrix. In order to learn non-linear mappings of inputs, the output from $\mathbf{h}$ is subject to a non-linear and differentiable activation function, σ, such as the sigmoid (logistic), tanh function, or the rectified linear unit (ReLU) (Maas et al. [2013]). The ReLU function is a non-saturating activation function, and is calculated as:

$$\sigma(x) = \max(0, x) \tag{2.8}$$

Conversely, the tanh and sigmoid functions both saturate given large input values. A benefit of using ReLU activations is that training is completed several times faster over sigmoid functions (Krizhevsky et al. [2012]). It is important that a non-linearity

activation is applied otherwise the DNN will be linear combinations of the inputs, and the activation function is required to be differentiable for training weight (and bias) parameters when the gradient descent method is used in training.

To derive the required weight parameters for each of the layer connections, the backpropagation of errors algorithm, used in conjunction with gradient descent optimisation, is applied to minimise the error between the estimated output, $\hat{\mathbf{y}}$, and ideal target, $\mathbf{y}$. The error is calculated through the loss function, $\mathbf{L}$. Loss functions determine the overall performance of the DNN, and effectively decide which features are learnt within the DNN. The choice of loss function is key for the success of the model. Loss functions are therefore application dependent, whether classification based (for binary masking) or regression based (for ratio masking). This work explores and proposes different loss functions within classification, for binary mask estimation to improve the resulting intelligibility of the enhanced signal. Details of the loss functions used in this work are provided in Section 2.5.

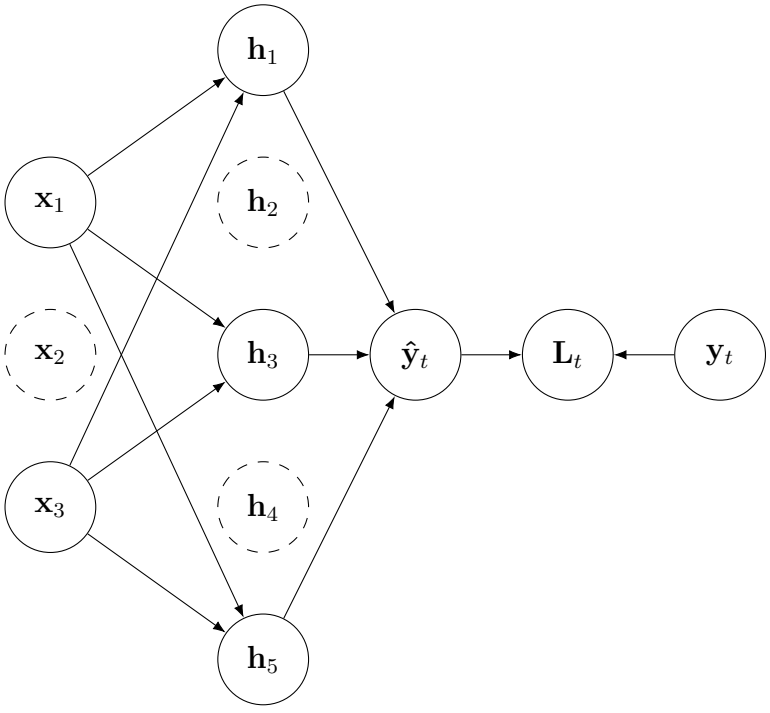


Figure 2.9: Computation of a typical feed-forward neural network with dropout.

Care must be taken when training neural networks as they are prone to over-

fitting on the training set if there is a lack of variation within the training material. To prevent over-fitting the dropout technique (Srivastava et al. [2014]), is used within the DNN architecture. During training, neurons (or nodes) are selected at random and dropped, meaning the neuron and its connections are temporarily removed from the network for that particular instance or set of training examples (mini-batch).

Figure 2.9 shows an example of dropout applied to the example network in Figure 2.8, where neurons $\mathbf{x}_2$, $\mathbf{h}_2$ and $\mathbf{h}_4$ were dropped, drastically reducing the number of connections. A probability of $p = 0.5$ is typically used for dropout applied to fully-connected hidden layers, and a probability close to or equal to 0, for dropping input units. The effect of applying dropout during training reduce the large model, into smaller *thinned* models. For estimation, the output is the average of all thinned models. This achieves a similar effect to training a large ensemble of models and averaging the predictions of all models (Goodfellow et al. [2013, 2016]).

Training of the networks is performed using the resilient backpropagation algorithm (Adam) (Kingma and Ba [2014]), as training concludes considerably faster than the standard backpropagation algorithm. The training data (training set and validation set) are grouped into mini-batches of 256 examples, with z-score normalisation applied to the input features. Normalising standardises each input using its mean and standard deviation, and helps the network to converge. Normalisation through z-score calculates the mean and standard deviation for each coefficient in the input feature vector, across all samples in the training set:

$$z\text{-score}(\mathbf{x}) = \frac{(\mathbf{x} - \boldsymbol{\mu})}{\boldsymbol{\sigma}} \tag{2.9}$$

where $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ are the mean and standard deviation at for all feature coefficients within the input $\mathbf{x}$, calculated across all examples within the training set. The calculated $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ are used to z-score normalise data within the validation set and test set.

The DNN weight values are initialised with uniformly distributed random variables in the range -0.01 to 0.01, and the learning rate is fixed at 0.0001 for DNNs.

To further prevent over-fitting early stopping of training (Prechelt [1998]) was used when the validation score (error score calculated on the validation set) did not improve after 5 further epochs, where an epoch is a full pass of the training set and validation set.

The final DNN architecture selected for this task is shown in Figure 4.3 and comprises 4 dense layers containing 1024 rectified linear units (ReLU) and a final sigmoid output layer. The number of fully-connected or dense layers, and number of units in each layer were optimised within a parameter grid search, the total number of layers was varied between 2 and 5, and the number of units was selected from $[256, 512, 1024, 2048]$. The model takes as input a window of stacked input features $\mathbf{X}_t = [\mathbf{x}_{t-K}; ...; \mathbf{x}_t; ...; \mathbf{x}_{t+K}]$, and outputs a vector corresponding to the central frame from the input window at time t, $\hat{\mathbf{Y}}_t = [\mathbf{y}_t]$. Including temporal information along with static features has shown to improve performance across many applications (Furui [1986]; Hanson and Applebaum [1990]; Healy et al. [2013, 2015]; Zhao et al. [2016]; Chen and Wang [2018]).

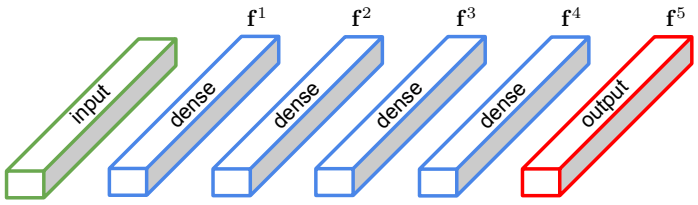


Figure 2.10: Feed-forward (DNN) speech enhancement architecture.

2.5 Perceptually motivated loss functions

Feed-forward neural networks (DNN) are used to learn a mapping from input features to target binary masks. Section 2.4 detailed how DNNs are trained, revealing that the choice of loss function plays an important role in the performance of the DNN. Binary mask estimation is treated as a classification problem. Classification applications usually maximise the classification accuracy through the binary cross-entropy (CE) loss function in training, whereas this work proposes two new

perceptually motivated loss functions as alternatives to the CE loss function, based on the HIT-FA rate, specifically designed for speech enhancement applications (Websdale and Milner [2017a,b]). A review of classification accuracy and HIT-FA rate objective measures is provided, before introducing the CE loss function and two new perceptually motivated loss functions inspired by the HIT-FA rate.

2.5.1 Objective measures of predicted binary mask quality

The quality of the predicted binary masks (PBM) is key to the success of the speech enhancement system. Without high quality masks, the enhanced speech signal will contain unwanted artefacts, caused by errors from over-suppressing speech dominant T-F units and under-suppressing noise dominant T-F units. Mask quality is calculated by comparing the PBM with the reference ideal binary mask (IBM).

Table 2.1: Relationship between the IBM and PBM.

PBM(t,f) \ IBM(t,f)	1	0
1	True Positive (TP)	False Positive (FP)
0	False Negative (FN)	True Negative (TN)

Table 2.1 shows the relationship between the PBM and the reference (or target) ideal binary mask (IBM). Given that the PBM(t, f) is 1 (i.e predicted to be speech dominant), the outcome is either a true positive (TP) if the IBM(t, f) is also 1 (i.e also speech dominant), or is a false positive (FP) if the IBM(t, f) is 0 (i.e noise dominant). Now given that the PBM(t, f) is 0 (i.e predicted to be noise dominant), the outcome is either a true negative (TN) if the IBM(t, f) is also 0 (i.e also noise dominant), or is a false negative (FN) if the IBM(t, f) is 1 (i.e speech dominant). Therefore, the overall outcome is either true or false whether the PBM(t, f) = IBM(t, f), and is either positive or negative whether the PBM$(t, f) = 1$ or 0. The sum of TPs, FPs, FNs and TNs across all T-F units are calculated from Equations 2.10 through 2.13, where for clarity $\mathbf{y}$ and $\hat{\mathbf{y}}$ are used to represent the IBM and

PBM respectively.

$$\text{TP} = \sum_{t=1}^{T}\sum_{f=1}^{F}\left[\mathbf{y}_{(t,f)}\hat{\mathbf{y}}_{(t,f)}\right] \tag{2.10}$$

$$\text{FP} = \sum_{t=1}^{T}\sum_{f=1}^{F}\left[(1-\mathbf{y}_{(t,f)})\hat{\mathbf{y}}_{(t,f)}\right] \tag{2.11}$$

$$\text{FN} = \sum_{t=1}^{T}\sum_{f=1}^{F}\left[\mathbf{y}_{(t,f)}(1-\hat{\mathbf{y}}_{(t,f)})\right] \tag{2.12}$$

$$\text{TN} = \sum_{t=1}^{T}\sum_{f=1}^{F}\left[(1-\mathbf{y}_{(t,f)})(1-\hat{\mathbf{y}}_{(t,f)})\right] \tag{2.13}$$

Objective measures then use these four summations (TP, FP, FN, TN) to calculate the quality of the predicted mask. A commonly used objective measure for binary mask quality is classification accuracy, calculated as:

$$\text{Accuracy} = \frac{\text{TP}+\text{TN}}{\text{TP}+\text{FP}+\text{FN}+\text{TN}} \tag{2.14}$$

$$= \frac{1}{TF}\sum_{t=1}^{T}\sum_{f=1}^{F}\left[\mathbf{y}_{(t,f)}\hat{\mathbf{y}}_{(t,f)} + (1-\mathbf{y}_{(t,f)})(1-\hat{\mathbf{y}}_{(t,f)})\right] \tag{2.15}$$

which provides a measure of how accurate the PBM is compared to the IBM, calculating the percentage of correctly labeled T-F units (TP+TN) from all T-F units ($T \times F$). An alternative measure of mask quality is the hit minus false alarm (HIT-FA) rate (Kim et al. [2009]), calculated as:

$$\text{HIT-FA} = \frac{\text{TP}}{\text{TP}+\text{FN}} - \frac{\text{FP}}{\text{FP}+\text{TN}} \tag{2.16}$$

$$= \frac{1}{R}\sum_{t=1}^{T}\sum_{f=1}^{F}\left[\mathbf{y}_{(t,f)}\hat{\mathbf{y}}_{(t,f)}\right] - \frac{1}{S}\sum_{t=1}^{T}\sum_{f=1}^{F}\left[(1-\mathbf{y}_{(t,f)})\hat{\mathbf{y}}_{(t,f)}\right] \tag{2.17}$$

where R is the number of T-F units within $\mathbf{y}$ that should be retained (1s) and S is

the number of T-F units within $\mathbf{y}$ that should be suppressed (0s),

$$R = \sum_{t=1}^{T} \sum_{f=1}^{F} \left[\mathbf{y}_{(t,f)} \right] \tag{2.18}$$

$$S = \sum_{t=1}^{T} \sum_{f=1}^{F} \left[(1 - \mathbf{y}_{(t,f)}) \right] \tag{2.19}$$

which instead provides a measure which compares accuracy of labelling speech dominant T-F units (HITs or TPs) minus errors in labelling noise dominant T-F units (FAs or FPs). The number of HITs determines how much speech is retained, whereas the number of FAs determines how much noise is incorrectly retained (i.e noise which has not been sufficiently suppressed).

Both classification accuracy and HIT-FA are maximised by achieving high TPs and high TNs (or specifically low FPs within HIT-FA), however the HIT-FA rate has been shown to correlate more closely with speech intelligibility compared with classification accuracy (Kim et al. [2009]). This is due to a potential bias found within classification accuracy, which is discussed further in Section 2.5.4.

2.5.2 Binary cross-entropy (CE) loss function

Binary cross-entropy (CE) is a standard loss function used within DNN training for classification tasks (Rubinstein and Kroese [2013]) and forms the baseline loss function. The aim of CE is to maximise the accuracy of the estimated mask where accuracy is defined as the proportion of correctly labeled T-F units. The CE loss, L^{CE}, is calculated as:

$$\mathrm{L}^{\mathrm{CE}} = -\frac{1}{TF} \sum_{t=1}^{T} \sum_{f=1}^{F} \left[\mathbf{y}_{(t,f)} \log(\hat{\mathbf{y}}_{(t,f)}) + (1 - \mathbf{y}_{(t,f)}) \log(1 - \hat{\mathbf{y}}_{(t,f)}) \right] \tag{2.20}$$

where $\mathbf{y}$ and $\hat{\mathbf{y}}$ are vectors that comprise concatenated frames of T-F units for each mini-batch in DNN training, from the IBM and PBM respectively. Each of these vectors comprises T time frames and F filterbanks.

2.5.3 HIT-FA (HF) loss function

The first perceptually motivated loss function (HF) is based on optimising the HIT-FA rate, which several studies have shown correlates more closely to intelligibility than mask accuracy (Kim et al. [2009]; Healy et al. [2013]; Chen et al. [2014]; Healy et al. [2015]; Chen et al. [2016]). In terms of the loss function, HITs refer to the proportion of correctly labeled target-dominant T-F units while FAs refer to the proportion of incorrectly labeled noise-dominant T-F units. Studies have shown that achieving high HITs (accurately labelling speech dominant T-F units) and low FAs (accurately labelling noise dominant T-F units) produces higher intelligibility (Kim et al. [2009]). Details of the HIT-FA rate are provided in Section 2.5.1.

The key difference between the CE and HF loss functions is that CE calculates accuracy over all T-F units together, whereas HF calculates the accuracy of target-dominant (1) and noise-dominant (0) T-F units separately. HIT-FA has a range between 1 and -1, with 1 being best performance. However within DNN training loss is minimised, therefore $-$(HIT-FA) = $-$HIT+FA is minimised, to give best performance at -1 and remove this discrepancy. The HIT-FA loss, L^{HF}, is calculated as:

$$\mathrm{L}^{\mathrm{HF}} = -\frac{1}{R}\sum_{t=1}^{T}\sum_{f=1}^{F}\left[\mathbf{y}_{(t,f)}\hat{\mathbf{y}}_{(t,f)}\right] + \frac{1}{S}\sum_{t=1}^{T}\sum_{f=1}^{F}\left[(1-\mathbf{y}_{(t,f)})\hat{\mathbf{y}}_{(t,f)}\right] \tag{2.21}$$

where R is the number of T-F units within $\mathbf{y}$ that should be retained (1s) and S is the number of T-F units within $\mathbf{y}$ that should be suppressed (0s). The first part concerning R calculates the number of HITs ($-$HIT), and the second concerning S calculates the number of FAs (+FA).

2.5.4 Binary cross-entropy HIT-FA hybrid (CEHF) loss function

Within an IBM the number of retained T-F units, R, and number of suppressed units, S, are generally not equal. In most cases there are more noise-dominant T-F units than target-dominant units, due mainly to non-speech areas. The HF loss function is calculated as proportions of R and S separately, and is therefore less affected by bias towards a difference between R and S (shown later for an example binary mask). Conversely, the CE loss function is calculated as an overall accuracy of R and S and is therefore biased towards the greater of the two (shown later for an example binary mask).

Inspiration from both the HF loss function and CE loss function produces a hybrid cross-entropy HIT-FA (CEHF) loss function by modifying the CE function to remove this bias. To do this the ratio between R and S is adjusted by applying a weighting term ω to the portion related to S. The cross-entropy HIT-FA loss function, $\mathrm{L}^{\mathrm{CEHF}}$, is calculated as:

$$\mathrm{L}^{\mathrm{CEHF}} = -\frac{1}{TF}\sum_{t=1}^{T}\sum_{f=1}^{F}\Big[\mathbf{y}_{(t,f)}\log(\hat{\mathbf{y}}_{(t,f)}) + \omega(1-\mathbf{y}_{(t,f)})\log(1-\hat{\mathbf{y}}_{(t,f)})\Big] \quad (2.22)$$

where the weighting term, ω, can adjust the balance between 1s and 0s, by reducing or increasing the influence of errors from suppressions (right side of equation 2.22). The weighting term could have been applied for adjusting the influence of errors from retained T-F units, but this can be achieved by using a value of $\omega > 1$.

To further show the differences between the loss functions, each loss function is represented as a set of scales given an example mask. Each set of scales represent a summation within the loss function. An example ideal binary mask is shown in Figure 2.11, where the black squares represent retain and white suppress. In our example there are 12 black squares and 24 white squares, giving a ratio of $1:2$ between R and S.

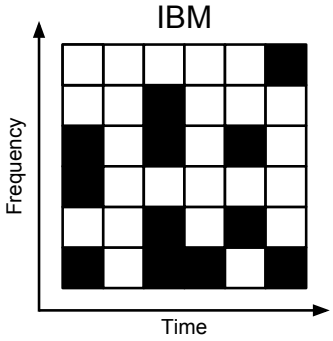


Figure 2.11: Ideal binary mask example.

The CE loss function can be represented as a single scale between the number of target-dominant (retain) and noise-dominant (suppress) T-F units within the summation, shown in Figure 2.12. In the example the number of suppressions outweighs the number of retains, which causes the CE loss function to favour its accuracy towards suppressing (0).

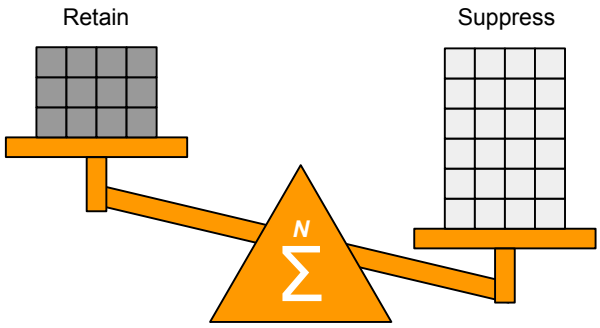


Figure 2.12: Binary cross-entropy loss function example.

The HF loss function can be represented as two scales, one for the summation for HITs calculated from the retain T-F units, and the second for FAs calculated from the suppress T-F units, shown in Figure 2.13. Due to the separate summations, the difference in number between retain and suppress has no impact on the loss function favouring one over the other.

Figures 2.12 and 2.13 show that the CE loss function is biased to the S due to the greater number of white squares, and that the HF loss function is not affected

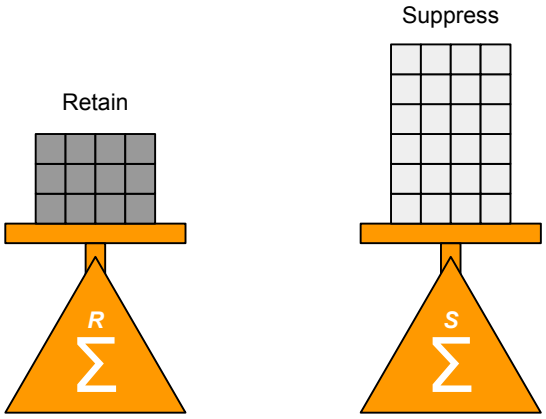


Figure 2.13: HIT-FA loss function example.

by this bias. Therefore setting the weighting term ω within CEHF to remove the bias could maximise HIT-FA rate. Therefore, in order for the ratio between R and S to be such that $R = S$, a weight of $\omega = {}^{R}/_{S}$ is applied. Figure 2.14 shows how the weighting applied to suppressions causes the overall scale to become balanced and unbiased just as the HF loss function, with $\omega = {}^{12}/_{24} = {}^{1}/_{2}$. The example has a bias towards S, therefore this normalisation will cause an increase of HITs at a cost of increasing FAs. The opposite would occur if the bias was towards R. A reduction to overall classification accuracy will occur in all cases where $R \neq S$ prior to normalisation.

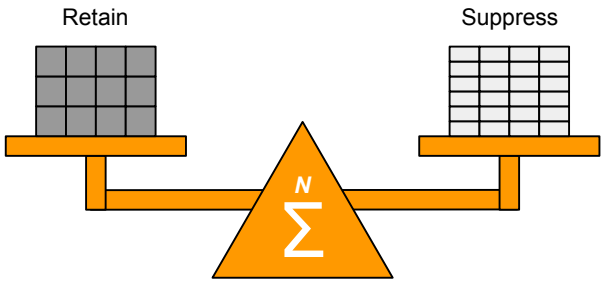


Figure 2.14: Cross-entropy HIT-FA hybrid loss function example.

2.6 Experimental results

The performance of feature extraction methods, loss functions, and the inclusion of visual information within feed-forward neural networks (DNN) is compared within binary mask estimation speech enhancement. Firstly, the feature extraction methods outlined in Section 2.3 are optimised across audio-only, visual-only and audio-visual models using the binary cross-entropy (CE) loss function (Section 2.5.2). Secondly, the binary cross-entropy HIT-FA hybrid (CEHF) loss function is optimised for HIT-FA rate. Finally, the best performing feature extraction methods from Section 2.6.1 is used to compare the performance of using the binary cross-entropy (CE) loss function, HIT-FA (HF) loss function and binary cross-entropy HIT-FA hybrid (CEHF) loss function for binary mask estimation across varying noise type and SNR conditions.

The first experiment compares feature extraction methods outlined in Section 2.3, namely multi-resolution cochleagrams (MRCG), complementary feature set (ARPMG) and active appearance models (AAM). Initially audio-only and visual-only models are compared to find the best performing feature extraction methods, before combining into audio-visual models. This experiment is conducted in babble noise at $-5\,\text{dB}$, for audio-only, visual-only and audio-visual models using the validation set, and optimises input window width. The best performing features are selected for further analysis.

The second experiment takes the best performing features, and maximises the proposed CEHF loss function for HIT-FA rate. The CEHF loss function contains a weighting term, ω, which is adjusted to achieve maximum HIT-FA rate, and can either be set to a fixed value, or dynamically set per minibatch during training (i.e $\omega = {}^{R}/_{S}$). This experiment is conducted in babble noise and factory noise at $-5\,\text{dB}$, for audio-visual models using the validation set. The best performing ω configuration within CEHF is selected for further analysis.

The final experiment compares the performance of the CE loss function, HF

loss function and CEHF loss function in varying noise types and SNRs conditions, specifically babble and factory noise at SNRs of $-10\,\text{dB}$, $-5\,\text{dB}$, $0\,\text{dB}$ and $5\,\text{dB}$, again in audio-only, visual-only and audio-visual models.

Across all experiments, objective measures are used to evaluate the performance of each model configuration. Various objective measures are available, such as frequency-weighted segmental SNR (Hu and Loizou [2008]), normalised covariance metric (Holube and Kollmeier [1996]) and the coherence speech intelligibility index (Kates and Arehart [2005]). This work uses classification accuracy (Rubinstein and Kroese [2013]) and HIT-FA rate (Kim et al. [2009]) to determine the accuracy of the predicted masks. This work also uses PESQ (Rix et al. [2001]) and ESTOI (Jensen and Taal [2016]) for measuring the quality and intelligibility of the enhanced signals, which have been shown to work well in many applications (Ma et al. [2009]; Websdale et al. [2015]).

All experiments are speaker dependent and use a single speaker (speaker 12) from the GRID dataset (details provided in Section A.1) which contains 1000 utterances, 800 are selected for neural network training of which 160 are removed for the validation set. The remaining 200 utterances are allocated to the test set. The same split of data is used across all experiments, i.e the training, validation and test set is unchanged between experiments. Noise files are taken from the NOIZEUS dataset (details provided in Section A.2).

The DNN models were implemented within the Lasagne framework (Dieleman et al. [2015]) with the Theano (Theano Development Team [2016]) back-end. Input data was z-score normalised and grouped into mini-batches of 256. To prevent overfitting, dropout of 0.2 was applied between all layers and early stopping (Prechelt [1998]) was used when the validation score did not improve after 5 further epochs. Training used backpropagation with the Adam optimiser (Kingma and Ba [2014]) and a learning rate of 0.0001, minimising the selected loss function.

2.6.1 Comparing feature extraction methods

An initial comparison is made between the feature extraction methods outlined in Section 2.3, namely multi-resolution cochleagrams (MRCG), complementary feature set (ARPMG) and active appearance models (AAM). Audio-only and visual-only experiments are first performed, with the best performing acoustic feature then selected for use in audio-visual experiments. The window width size of input feature context $\mathbf{X}_t = [\mathbf{x}_{t-K}; ...; \mathbf{x}_t; ...; \mathbf{x}_{t+K}]$ is also compared, with width K ranging from 1 to 17 across all experiments. For this investigation, experiments are conducted in babble noise at -5 dB only.

Figure 2.15 shows the intelligibility score produced from ESTOI for all features across the validation set. The validation set was selected for this initial comparison as it is effectively parameter searching for the optimal window width K and the feature set to be used in future experiments.

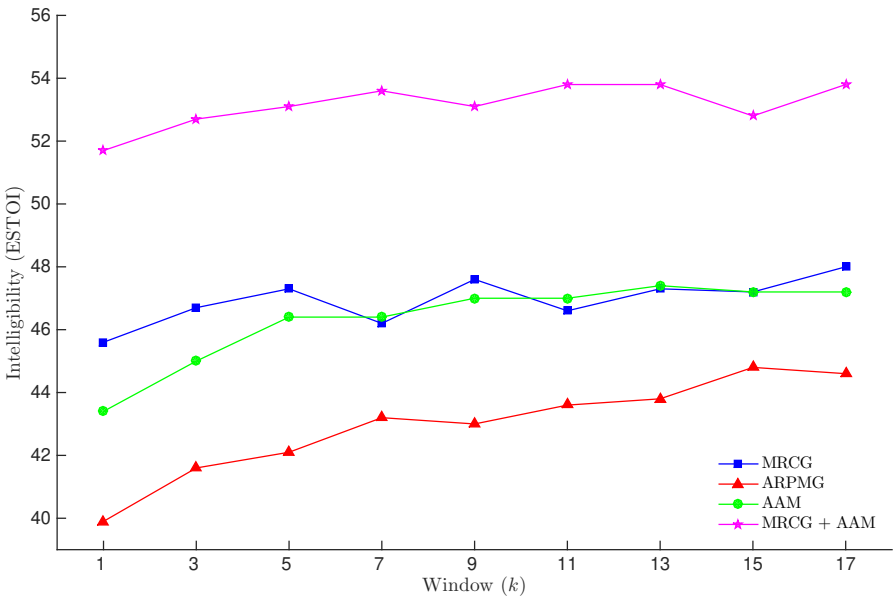


Figure 2.15: Effect of feature extraction methods and temporal window width on intelligibility (ESTOI) in babble noise at -5 dB.

Comparing initially the window width K across all features, we find performance increases up to a width of 11 before flattening. On average the best performing

window is of size 15, which is equivalent to a total window of 320 ms. Therefore a window of $k = 15$ is selected for future experiments.

When comparing the two acoustic features for audio-only mask estimation, MRCG and ARPMG, results show that the MRCG feature consistently outperforms ARPMG across all window widths with an average improvement of 4.0 in terms of ESTOI. This is attributed to using a cochleagram based framework. The MRCG is more closely related to the framework and target mask, where both are generated using cochleagrams. The complementary feature set does contain a gammatone feature, however, does not represent the energy of each frame as found within the cochleagrams used in MRCG. Although the features used within the complementary feature set are successful in other tasks, for this specific task and framework, the MRCG feature is more appropriate.

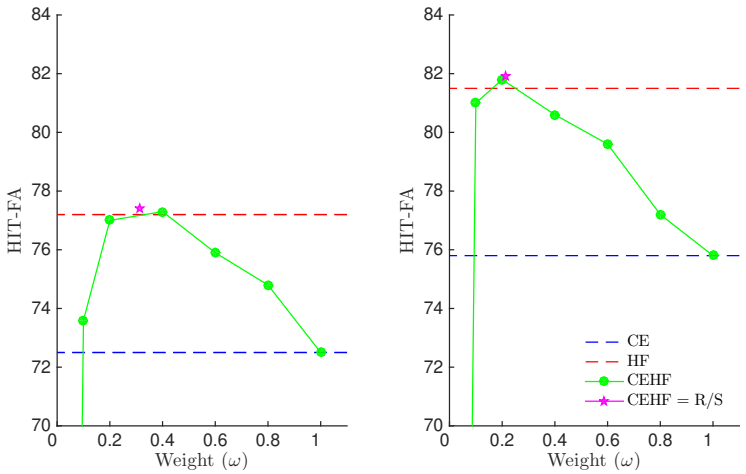
Now looking at the visual-only performance using AAMs, the performance is almost the same as that of MRCG with a window width above 7, and consistently better than ARPMG across all window widths, with an average of 3.4 in terms of ESTOI over ARPMG. This performance is surprising given that visual-only does not contain information about the noise type or level. The DNN is learning a mapping from purely mouth movements to an acoustic T-F mask. If the exact same mouth sequence is seen in different utterances with different noise segments, the DNN would still output the exact same mask, which therefore suggests the DNN is learning a mean output mask for each input sequence of mouth movements. It is worth recognising that these experiments are at low SNR (-5 dB) where the acoustic information is more corrupted, yet the visual information is unaffected.

For audio-visual masking, the acoustic MRCG feature and visual AAM feature are combined through stacking on input to the DNN. This combination provides large gains over both audio-only and visual-only models, with an average gain of 6.2 in terms of ESTOI over MRCG. Combining both modalities into a single feature provides a complementary feature for the DNN at this low SNR, where the visual information provides the mouth movement (narrowing down the potential output

mask options), and the acoustic information provides information about the noise (fine tuning the output mask per noise level).

2.6.2 Maximising HIT-FA rate with loss functions

In Section 2.5 two perceptual loss functions were proposed to maximise HIT-FA rate instead of classification accuracy. Within CEHF, a weighting term ω is introduced to counter the bias found in CE. From the example shown in Section 2.5.4, it was hypothesised that setting $\omega = {}^{R}/_{S}$ would maximise HIT-FA rate. This hypothesis is explored by varying ω between 0 and 1, and show the performance of the other loss functions. The experiments are performed within an audio-visual model (MRCG+AAM), with a window of $k = 15$ as this was found to have best performance in Section 2.6.1, in babble and factory noise at $-5\,\text{dB}$. Figure 2.16 shows the resulting HIT-FA rate for the validation set.



(a) babble $-5\,\text{dB}$ (b) factory $-5\,\text{dB}$

Figure 2.16: Effect on HIT-FA rate across loss functions in babble and factory noise at $-5\,\text{dB}$ for audio-visual.

When comparing HF against CE, results find that the HF provides large improvements over CE for HIT-FA rate across both noise conditions. This is unsurprising considering HF was designed to maximise the HIT-FA rate. Now considering the CEHF loss function, performance of CEHF steadily improves over CE as ω is re-

duced from 1 (where $\omega = 1$ is equivalent to using the CE loss function), reducing the bias towards suppression and increasing the importance of retaining T-F units. When setting $\omega = {}^{R}/_{S}$, shown as a magenta star, peak HIT-FA rate is achieved across both noise conditions confirming our hypothesis, giving performance equivalent to that of HF. When using the CEHF loss function in future experiments, ω is set to ${}^{R}/_{S}$.

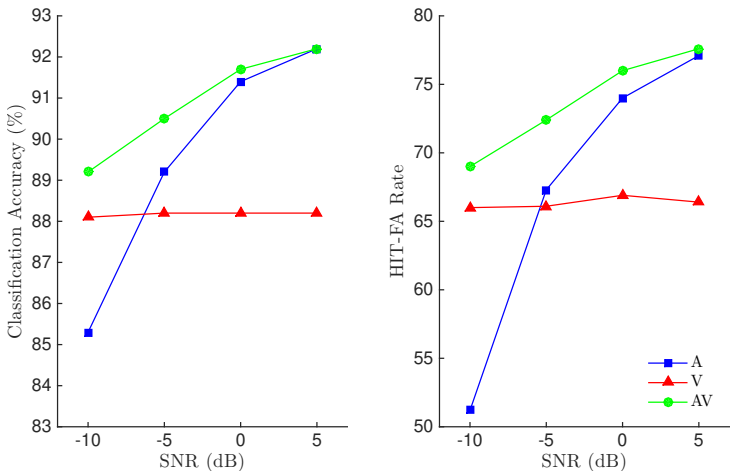
It is worth noting that setting $\omega = 0$ causes the HIT-FA rate to become 0. This is due to the error produced from suppressions is calculated as 0 within the loss function (see Equation 2.22 within Section 2.5.4), causing the DNN to only minimise error calculated from the T-F units concerned with retaining (speech dominant). This results in an output mask containing only 1s, i.e all speech dominant (to reduce error) and subsequently noise dominant T-F units are set to 1, producing a HIT-FA rate of 0. When this mask containing only 1s is applied to the noisy speech, no masking occurs and all T-F units are retained, producing the same unprocessed noisy signal before enhancement, i.e no enhancement would occur.

2.6.3 Analysis across noise type and SNR

In Section 2.6.1 feature extraction methods were tested in babble noise at $-5\,\text{dB}$ only, revealing extracting acoustic MRCG features, and visual AAM features to perform best. This experiment uses the best performing features, and expands on those tests to consider SNRs of $-10\,\text{dB}$, $-5\,\text{dB}$, $0\,\text{dB}$ and $5\,\text{dB}$ in both babble and factory noise. Previously, the focus was on solely on intelligibility through ESTOI, however now classification accuracy, HIT-FA rate and PESQ objective measures are considered. Initially the performance of audio-only, visual-only and audio-visual systems are compared using the CE loss function before evaluating the proposed perceptually motivated loss function.

2.6.3.1 Evaluating the binary cross-entropy loss function

This experiment explores the robustness of audio-only, visual-only and audio-visual systems across varying SNR and noise conditions using the CE loss function using the acoustic MRCG feature and visual AAM feature. Tables 2.2, 2.3 and 2.4 show the full set of objective measures for the test across all noise type and SNR conditions tests, for audio-only, visual-only and audio-visual models respectively. Figures 2.17 and 2.18 provide detailed breakdowns from Tables 2.2, 2.3 and 2.4 for babble noise at −10 dB, −5 dB, 0 dB and 5 dB.



(a) Classification Accuracy (b) HIT-FA

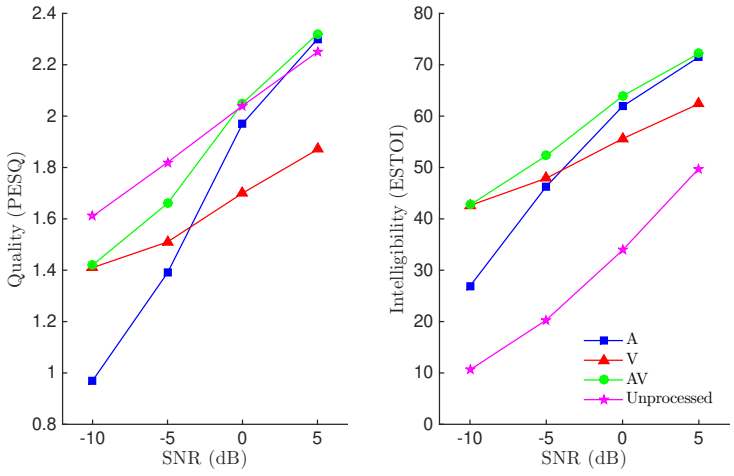
Figure 2.17: Effect on mask classification accuracy and HIT-FA rate across SNR for audio-only, visual-only and audio-visual in babble noise for binary mask estimation using the binary cross-entropy (CE) loss function.

Focusing first on classification accuracy and HIT-FA rate, results show similar trends for both measures across noise types and SNRs, with regards to model performance, shown in Figure 2.17 for babble noise at SNRs from −10 dB to 5 dB. Audio-visual models provide best performance across all noise types and SNRs for both classification accuracy and HIT-FA rate. Audio-only performs well at high SNRs, outperforming visual-only and reaching equivalent performance to audio-visual. However, at low SNRs, audio-only performs particularly poor, falling below visual-only. This is due to the high levels of noise at low SNRs, corrupting the acoustic signal. The DNN is unable to extract useful information from the heavily

degraded signal.

For visual-only systems, classification accuracy and HIT-FA rate provides a consistent score across all SNRs for each noise type. This is due to the visual feature being unaffected by noise type or SNR corrupting the audio stream, and the performance is provided by how well the DNN can map the input visual features to the target mask. The only difference between noise type and SNR configurations are the configuration dependant target masks.

When comparing the performance of audio-visual against audio-only and visual-only, an average improvement across both babble and factory noise at $-10\,\text{dB}$ for classification accuracy of 3.2 and 1.2, and for HIT-FA rate of 15.6 and 2.9 can be found over audio-only and visual-only models. Now comparing the performance of audio-visual against visual-only (as audio-only performs equivalently to audio-visual) across both babble and factory noise at $5\,\text{dB}$, an average improvement of 3.9 and 11.4, is found for classification accuracy and HIT-FA rate respectively.



(a) PESQ (b) ESTOI

Figure 2.18: Effect on quality through PESQ and intelligibility through ESTOI across SNR for audio-only, visual-only and audio-visual in babble noise for binary mask estimation using the binary cross-entropy (CE) loss function.

Looking now at quality scores through PESQ and intelligibility with ESTOI, similar trends as with classification accuracy and HIT-FA rate are found, where audio-visual outperforms audio-only and visual-only. Audio-only performs poorly

at low SNRs and well at high SNRs reaching equivalent performance to audio-visual. Visual-only again performs well at low SNRs, this time reaching equivalent performance to audio-visual, but poorly at high SNRs, shown in Figure 2.18 for babble noise at SNRs from −10 dB to 5 dB. The performance between all systems for PESQ is below that of unprocessed audio. However, for intelligibility all systems provide large gains over unprocessed audio. This is a known side-effect from binary masking based speech enhancement. Binary masking is focused on improving intelligibility over quality, and our intelligibility results through ESTOI confirm this.

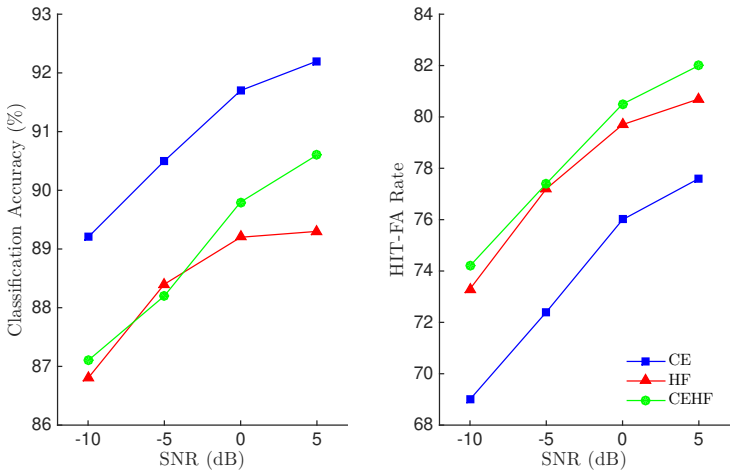
When comparing the performance of audio-visual against audio-only (as visual-only performs equivalently to audio-visual) across both babble and factory noise at −10 dB, an average improvement of 0.41 and 14.4, is found for PESQ and ESTOI respectively. Now comparing the performance of audio-visual against visual-only (as audio-only performs similarly to audio-visual) across both babble and factory noise at 5 dB, an average improvement of 0.45 and 12.5, is found for PESQ and ESTOI respectively. Audio-visual provides large gains over unprocessed audio across all noise types and SNRs, specifically, an average improvement of 27.3 in ESTOI is found for audio-visual models over unprocessed audio across babble and factory noise across all SNRs.

Comparing the overall performance of audio-only, visual-only and audio-visual systems across all noise types and SNRs, audio-visual models were found to consistently provide best performance across all objective measures over audio-only and visual-only. At high SNRs, the benefit gained from combining audio and visual information over audio-only is reduced as the audio features are less degraded by noise which allows the DNN to more effectively map to the target masks in these less challenging conditions. At very low SNRs (−10 dB), the benefit gained with audio-visual over visual-only is only found in classification accuracy and HIT-FA, with both PESQ and ESTOI giving equivalent scores in performance.

2.6.3.2 Evaluating the perceptually motivated loss functions

Section 2.6.3.1 explored the performance of audio-only, visual-only and audio-visual systems across SNR and noise type with the CE loss function. This experiment compares the performance of the proposed perceptually motivated loss functions against the CE loss function under the same previously used noise conditions.

Tables 2.2, 2.3 and 2.4 show the full set of objective measures for the test across all noise type and SNR conditions tests, for audio-only, visual-only and audio-visual models respectively. Figures 2.19 and 2.20 provide detailed breakdowns from Tables 2.2, 2.3 and 2.4 for babble noise at -10 dB, -5 dB, 0 dB and 5 dB. The audio-visual model was shown to consistently outperform (or match) both audio-only and visual-only for all objective measures across all noise types and SNRs and is selected for detailed analysis. The same performance trends between loss functions found for audio-visual are also found for audio-only and visual-only.



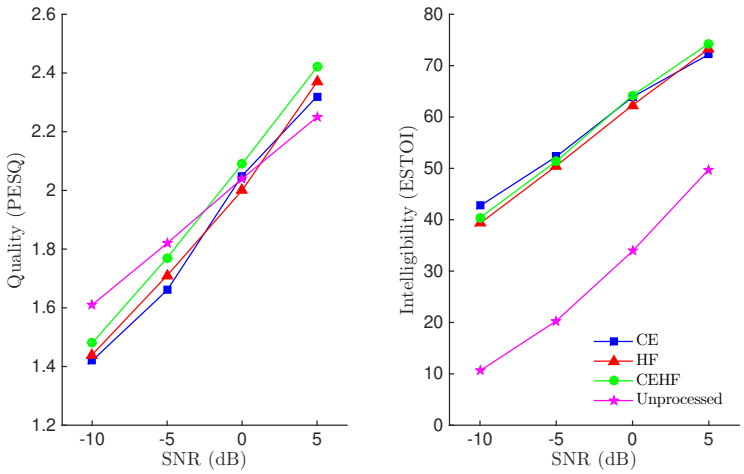
(a) Classification Accuracy (b) HIT-FA

Figure 2.19: Effect on mask classification accuracy and HIT-FA rate across SNR for CE, HF and CEHF loss functions in babble noise for audio-visual binary mask estimation.

Focusing first on classification accuracy and HIT-FA rate, results show clear differences between the CE and proposed HF and CEHF loss functions, shown in Figure 2.19 for audio-visual models in babble noise at SNRs from -10 dB to 5 dB. In classification accuracy, the CE loss function provides large gains over both HF

and CEHF loss functions across all noise type and SNRs. This is expected as the CE loss function is targeted to maximise accuracy. The hybrid CEHF loss function has accuracy higher than HF at high SNRs but is equivalent at low. However, when comparing HIT-FA rate, both the proposed HF and CEHF loss functions outperform CE across all noise types and SNRs, with the CEHF loss function providing peak performance over the HF loss function.

In terms of HITs, the CEHF and HF loss functions perform similarly, but the main difference between them is that the CEHF loss function generates fewer FAs compared to the HF loss function (shown in Tables 2.2, 2.3 and 2.4). Lowest HITs and lowest FAs are found with the CE loss function due to it favouring 0s over 1s in the mask, which is caused by the bias towards the larger of S and R. The CEHF loss function is able to remove this bias and provides a balance between increasing HITs without increasing as many FAs.



(a) PESQ (b) ESTOI

Figure 2.20: Effect on quality through PESQ and intelligibility through ESTOI across SNR for CE, HF and CEHF loss functions in babble noise for audio-visual binary mask estimation.

Looking now at quality scores through PESQ and intelligibility with ESTOI, all loss functions perform similarly across all noise types and SNRs, shown in Figure 2.20 for audio-visual models in babble noise at SNRs from -10 dB to 5 dB. Focusing on quality through PESQ, both the proposed loss functions (HF and CEHF) provide

improvements over the CE loss function, but again do not provide improvements over unprocessed audio below 0 dB, with any performance gain being marginal.

Comparing now intelligibility as measured by ESTOI, the CE loss function outperforms the HF loss function at lower SNRs while the HF loss function outperforms CE at the higher 5 dB SNR for all noise types. Even though the HF loss function outperforms CE with regards to the HIT-FA rate across all configurations, the large number of FAs introduced by HF reduces the intelligibility to be lower than CE at low SNRs. This shows that even a large increase in HITs does not compensate for a large increase in FAs, which are more detrimental to intelligibility at low SNR than at high SNR. When comparing the hybrid CEHF loss function the performance is shown to outperform both CE at SNRs above -5 dB, and is slightly worse than CE at -5 dB, but is shown to outperform the HF loss function across all noise types and SNRs. The CEHF loss function had higher HIT-FA rate over CE across all SNR for all systems, confirming that increasing the HIT-FA rate does increase intelligibility, but the number of FAs introduced affects the resulting intelligibility. Reducing FAs at low SNRs is critical whereas a higher HIT rate is more important at high SNRs.

Overall, with intelligibility being the main focus, all systems provide large gains in ESTOI over unprocessed audio, with the bimodal audio-visual system outperforming both audio-only and visual-only across all configurations. With regards to loss functions, if the SNR is very low, CE is the loss function of choice, however at all other SNRs, CEHF is the best performing loss function. CEHF provides the best balance between both classification accuracy and the HIT-FA rate regarding all loss functions, but favours HIT-FA rate over classification accuracy.

Table 2.2: (AUDIO-ONLY) Classification accuracy (in %), HIT-FA (in %) PESQ and ESTOI scores for the GRID dataset for audio-only binary mask estimation with different loss functions in babble and factory noise at −10 dB, −5 dB, 0 dB and 5 dB.

Noise (dB)		Loss	Acc	HIT-FA (FA)	PESQ	ESTOI
babble	-10	CE	**85.3**	51.2 (6.0)	0.97	**26.9**
		HF	82.0	58.7 (15.7)	1.00	26.1
		CEHF	81.3	**59.4** (17.3)	**1.04**	25.2
		unprocessed audio			1.61	10.6
	-5	CE	**89.2**	67.3 (5.8)	1.39	**46.3**
		HF	87.0	71.7 (12.0)	1.46	45.0
		CEHF	86.9	**72.2** (12.5)	**1.50**	44.5
		unprocessed audio			1.82	20.3
	0	CE	**91.4**	74.0 (4.7)	1.97	61.9
		HF	89.7	78.8 (10.1)	1.98	**62.2**
		CEHF	89.6	**79.0** (10.4)	**2.02**	**62.2**
		unprocessed audio			2.04	33.9
	+5	CE	**92.2**	77.1 (4.6)	2.30	71.5
		HF	90.2	81.5 (10.2)	2.33	73.2
		CEHF	90.3	**81.8** (10.2)	**2.39**	**73.8**
		unprocessed audio			2.25	49.8
factory	-10	CE	**89.9**	58.1 (4.2)	**0.91**	**28.1**
		HF	85.0	**64.0** (13.5)	0.80	24.2
		CEHF	82.2	63.6 (17.6)	0.86	22.5
		unprocessed audio			1.46	10.5
	-5	CE	**92.6**	70.5 (3.5)	**1.44**	**44.4**
		HF	89.3	75.9 (10.0)	1.31	41.0
		CEHF	89.3	**76.7** (10.3)	1.39	42.9
		unprocessed audio			1.66	20.1
	0	CE	**94.1**	77.5 (3.0)	**1.97**	58.7
		HF	92.0	83.1 (7.8)	1.90	59.0
		CEHF	91.7	**83.3** (8.3)	1.93	**59.3**
		unprocessed audio			1.87	33.5
	+5	CE	**94.9**	80.6 (2.8)	2.24	67.3
		HF	92.1	85.7 (8.3)	2.24	69.9
		CEHF	92.7	**86.4** (7.5)	**2.35**	**71.0**
		unprocessed audio			2.09	49.9

Table 2.3: (VISUAL-ONLY) Classification accuracy (in %), HIT-FA (in %) PESQ and ESTOI scores for the GRID dataset for visual-only binary mask estimation with different loss functions in babble and factory noise at −10 dB, −5 dB, 0 dB and 5 dB.

Noise (dB)		Loss	Acc	HIT-FA (FA)	PESQ	ESTOI
babble	-10	CE	**88.1**	66.0 (7.0)	1.41	**42.6**
		HF	84.5	72.3 (16.9)	1.47	38.1
		CEHF	84.9	**72.4** (16.2)	**1.50**	38.6
		unprocessed audio			1.61	10.6
	-5	CE	**88.2**	66.1 (7.1)	1.51	**47.9**
		HF	84.4	**72.5** (17.2)	1.62	45.7
		CEHF	84.7	**72.5** (16.6)	**1.65**	45.8
		unprocessed audio			1.82	20.3
	0	CE	**88.2**	66.9 (7.4)	1.70	55.6
		HF	85.2	**72.5** (15.7)	1.81	**55.7**
		CEHF	84.4	72.2 (17.1)	**1.84**	55.1
		unprocessed audio			2.04	33.9
	+5	CE	**88.2**	66.4 (7.2)	1.87	62.4
		HF	84.8	72.3 (16.3)	2.00	65.1
		CEHF	85.0	**72.6** (16.1)	**2.05**	**65.4**
		unprocessed audio			2.25	49.8
factory	-10	CE	**91.0**	68.7 (5.4)	1.22	**40.7**
		HF	87.1	**76.6** (13.5)	1.26	37.7
		CEHF	87.3	76.3 (13.1)	**1.31**	38.0
		unprocessed audio			1.46	10.5
	-5	CE	**90.9**	69.1 (5.5)	1.42	**46.0**
		HF	87.1	**76.6** (13.5)	1.42	44.3
		CEHF	87.2	76.4 (13.2)	**1.47**	44.9
		unprocessed audio			1.66	20.1
	0	CE	**90.9**	68.3 (5.3)	1.66	52.0
		HF	87.4	76.5 (13.0)	1.69	**54.1**
		CEHF	86.8	**76.6** (13.9)	**1.73**	53.6
		unprocessed audio			1.87	33.5
	+5	CE	**91.1**	69.3 (5.4)	1.84	58.0
		HF	86.8	76.9 (14.1)	1.94	62.7
		CEHF	87.4	**77.0** (13.1)	**2.00**	**63.6**
		unprocessed audio			2.09	49.9

Table 2.4: (AUDIO-VSIUAL) Classification accuracy (in %), HIT-FA (in %) PESQ and ESTOI scores for the GRID dataset for audio-visual binary mask estimation with different loss functions in babble and factory noise at −10 dB, −5 dB, 0 dB and 5 dB.

Noise (dB)		Loss	Acc	HIT-FA (FA)	PESQ	ESTOI
babble	-10	CE	**89.2**	69.0 (6.5)	1.42	**42.7**
		HF	86.8	73.3 (13.1)	1.44	39.3
		CEHF	87.1	**74.2** (12.9)	**1.48**	40.4
		unprocessed audio			1.61	10.6
	-5	CE	**90.5**	72.4 (5.7)	1.66	**52.3**
		HF	88.4	77.2 (11.8)	1.71	50.5
		CEHF	88.2	**77.4** (12.2)	**1.77**	51.4
		unprocessed audio			1.82	20.3
	0	CE	**91.7**	76.0 (5.0)	2.05	63.9
		HF	89.2	79.7 (11.5)	2.00	62.3
		CEHF	89.8	**80.5** (10.6)	**2.09**	**64.2**
		unprocessed audio			2.04	33.9
	+5	CE	**92.2**	77.6 (4.7)	2.32	77.6
		HF	89.3	80.7 (11.6)	2.37	73.3
		CEHF	90.6	**82.0** (9.7)	**2.42**	**74.3**
		unprocessed audio			2.25	49.8
factory	-10	CE	**92.3**	71.5 (4.2)	1.28	**41.0**
		HF	89.4	78.3 (10.5)	1.26	38.3
		CEHF	89.2	**78.5** (10.9)	**1.29**	38.3
		unprocessed audio			1.46	10.5
	-5	CE	**93.6**	75.8 (3.4)	**1.65**	**51.6**
		HF	90.9	81.5 (9.0)	1.54	48.9
		CEHF	90.7	**81.9** (9.5)	1.61	49.4
		unprocessed audio			1.66	20.1
	0	CE	**94.3**	78.9 (3.1)	**2.01**	60.6
		HF	91.8	84.1 (8.4)	1.93	60.7
		CEHF	92.0	**84.5** (8.2)	**2.01**	**61.5**
		unprocessed audio			1.87	33.5
	+5	CE	**94.9**	80.9 (2.8)	2.28	67.7
		HF	92.3	86.0 (8.1)	2.29	70.4
		CEHF	92.8	**86.6** (7.5)	**2.38**	**71.5**
		unprocessed audio			2.09	49.9

2.7 Conclusions

This work has examined the effect on intelligibility (ESTOI), quality (PESQ) and mask accuracy (classification accuracy and HIT-FA) of including visual information in binary mask estimation for speech enhancement. It was found that all systems provide large gains in intelligibility over unprocessed, with largest gains found at lower SNRs. Combining both audio and visual modalities into a single bimodal audio-visual system provides largest gains across all noise types and SNRs, confirming that combining audio and visual features provides a robust complimentary feature set.

This work also proposed two new perceptually motivated loss functions for binary mask estimation based speech enhancement, inspired by the HIT-FA rate which is known to correlate closely to speech intelligibility. A hybrid binary cross-entropy HIT-FA loss function (CEHF) was proposed to reduce the bias found within binary cross-entropy (CE) by adjusting the ratio between 1s and 0s inspired by HIT-FA. Evaluations using classification accuracy, HIT-FA rate, PESQ and ESTOI reveal that the proposed loss functions provide performance gains in HIT-FA and PESQ across all noise types and SNRs tested over the standard binary cross-entropy (CE) loss function.

Even though both HF and CEHF loss functions outperform the CE loss function for HIT-FA rate across all noise types and SNRs, the large number of FAs introduced reduces the intelligibility to be lower than CE at low SNRs. This shows that even a large increase in HITs does not compensate for a large increase in FAs, which are more detrimental to intelligibility at low SNR than at high SNR. Reducing FAs at low SNRs is critical whereas a higher HIT rate is more important at high SNRs.

Chapter 3

Ratio masking

3.1 Introduction

Previous work in Chapter 2 explored speech enhancement using binary masks. It was found that large gains in intelligibility could be found over unprocessed audio for audio-only, visual-only and audio-visual models. However, binary masking failed to provide any benefits in quality. This work instead explored speech enhancement using ratio masking, which is known to provide improvements in quality and intelligibility in ideal conditions (Wang et al. [2014]; Healy et al. [2017]). Unlike in binary masking where each time-frequency (T-F) unit is either fully retained or fully suppressed, ratio masking aims to retain only a proportion which is associated with the level of speech present in the T-F unit. The ideal ratio mask calculation is shown in Section 3.2, which produces a mask containing values in the range 0 to 1, compared to an IBM which produces masks containing fixed values of 0 and 1. This change from fixed values to varying values reduces the effect of transitioning between speech dominant and noise dominant frames, and effectively smooths the resulting signal reducing distortions. This reduction in introduced distortions is how the IRM is able to improve both quality and intelligibility compared to an IBM. Overall speech intelligibility and quality is found to be increased for the IRM over the IBM.

Just as with binary masking (Chapter 2), in practice an IRM is not available and instead the ratio mask must be estimated from the noisy signal. This allows supervised learning to be used to map features extracted from noisy speech to a ratio mask. The algorithm then uses a model trained with known noisy speech and ideal mask pairings to predict and output the ideal mask. The model can then be used in *live* noisy conditions, when separate clean and noise sources are not available, to predict an estimation of an ideal mask (predicted mask). This predicted ratio mask (PRM) can then be applied to the noisy speech signal to produce the enhanced speech signal. This work uses deep feed-forward neural networks (DNN) for modelling the relationship between input noisy speech and target ratio masks, which have previously been shown to perform well for DNN based ratio mask estimation (Narayanan and Wang [2013]; Healy et al. [2015, 2017]; Chen and Wang [2018]).

This work also considers supplementing acoustic features with visual speech to improve the mapping for mask estimation, which was found to be provide peak performance in binary masking. The use of visual speech information in traditionally audio-only speech processing applications has given significant gains in performance in noisy conditions. For example, in automatic speech recognition (ASR), supplementing the audio with visual features has reduced error rates in low SNR conditions (Thangthai et al. [2015]; Potamianos et al. [2003]; Heckmann et al. [2002]). A benefit of using visual speech information within mask estimation is that visual features are not degraded by acoustic noise, although in themselves they may not have the discriminative ability that audio features possess in terms of mask estimation. To investigate this we explore mask estimation, and subsequently speech intelligibility, by comparing audio-only, visual-only and audio-visual speech enhancement models. Figure 3.1 shows the training pipeline of the proposed audio-visual speech enhancement system using feed-forward neural networks, and follows the same pipeline as previously used for binary masking (Chapter 2), except now ratio masks are estimated. Visual features are extracted from video and combined with acoustic features extracted from noisy speech, before input into the feed-forward neural network

(DNN) for temporal modelling to estimate the ratio mask. For testing purposes, estimated masks are applied to a cochleagram of the noisy speech which suppresses noise-dominated T-F units and the remaining signals are overlapped and summed to produce the enhanced signal, shown in Figure 3.2. The same pipeline is used for all speech enhancement configurations, except the visual stream is removed for audio-only models, and the audio stream is removed for visual-only models.

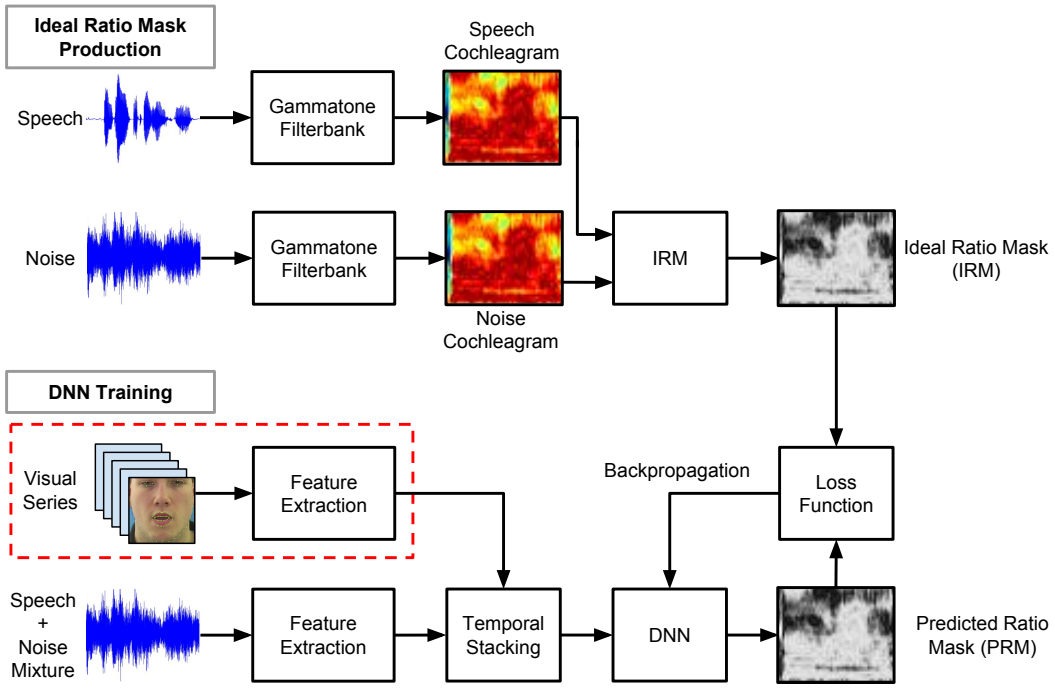


Figure 3.1: Overview of training the ratio masking speech enhancement system.

The remainder of this chapter is organised as follows. Section 3.2 details how ideal ratio masks are produced, and subsequently used for enhancement of noisy speech. Section 3.3 provides an overview of acoustic and visual feature extraction methods. The DNN architecture and training is introduced in Section 3.4, showing the difference between classification based and regression based DNNs. Performance evaluations are made in Section 3.5 which first compare the effectiveness of the feature extraction methods outlined in Section 2.3 for audio-only, visual-only and audio-visual models (Section 3.5.1). Section 3.5.2 compares the performance of ratio masking across varying noise type and SNR conditions using the best performing

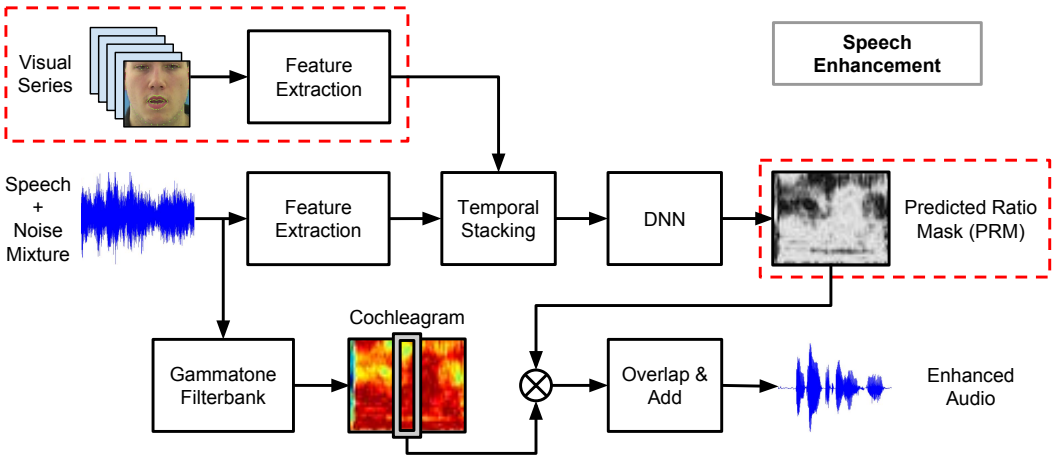


Figure 3.2: Overview of applying the predicted ratio mask for speech enhancement testing.

feature extraction methods from Section 3.5.1. A comparison is made between estimating binary masks and ratio masks in Section 3.5.3. Finally, this chapter is concluded in Section 3.6.

3.2 Ideal ratio masking

In CASA, enhanced speech is extracted by applying a mask to a time-frequency (T-F) representation of noisy speech. In ideal conditions separate clean and noise sources (i.e sources containing only clean or noise information) can be used to calculate an ideal ratio mask (IRM). Figure 3.3 shows the pipeline for producing IRMs, where cochleagrams are produced from separate speech and noise sources for producing T-F units, the IRM is then calculated between speech and noise cochleagrams. Details of cochleagram production are provided in Section 2.2.1.

Similar to ideal binary masking (IBM), ideal ratio masking (IRM) retains speech dominant T-F units and suppresses noise dominant T-F units. However, where as an IBM is restricted to values of 1 and 0, an IRM can take any value in between, and is instead defined as the ratio between speech and noisy speech with only the proportion associated with speech for each T-F unit retained. The IRM is calculated

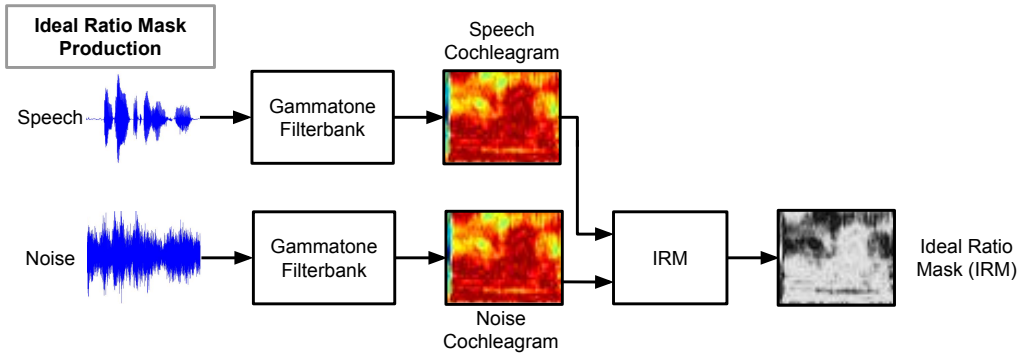


Figure 3.3: Overview of producing ideal ratio masks (IRM).

as:

$$\mathrm{IRM}(t,f) = \left(\frac{X(t,f)^2}{X(t,f)^2 + D(t,f)^2}\right)^{\beta} \tag{3.1}$$

where X is the clean speech cochleagram, D the noise-only cochleagram, t and f represent time frame and frequency bin respectively and β is a tuneable parameter usually set to 0.5 as standard (Narayanan and Wang [2013]; Healy et al. [2015]). Just as seen within binary masking, speech dominant T-F units produce values closer to 1 and noise dominant T-F units produce values closer to 0.

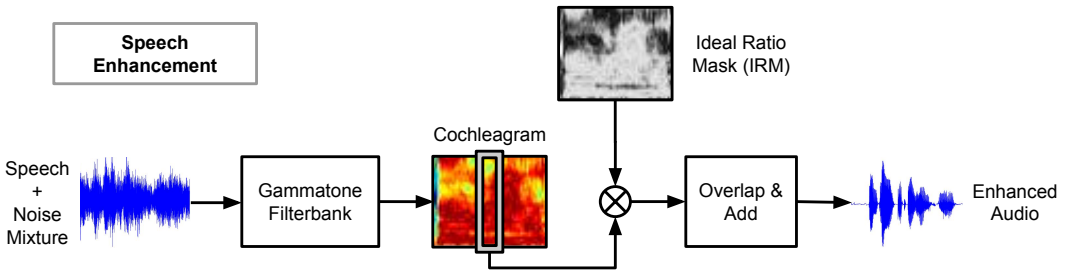


Figure 3.4: Overview of enhancing noisy speech through ratio masking.

Calculated masks can then be applied to noisy speech producing the enhanced signal. Figure 3.4 shows the pipeline for generating enhanced signals. The combined noisy speech time domain signal is decomposed into T-F units, following the same implementation to calculate a cochleagram, however the hamming windowed frames are not summed. The IRM is then multiplied with the hamming windowed frames,

before returning back into a time domain signal through overlap and adding, and summing across gammatone filterbank responses. The same enhancement procedure is applied when testing different models, except the IRM is replaced with the predicted ratio mask (PRM) produced from the model output.

For the calculation of classification accuracy and HIT-FA rate objective measures, a binary mask is required instead of a ratio mask. Therefore the ratio mask is converted in to a binary mask, $\widehat{\text{IBM}}$, and calculated as:

$$\widehat{\text{IRM}}(t,f) = \begin{cases} 1, & \text{if SNR}(t,f) \geq \text{LC} \\ 0, & \text{otherwise} \end{cases} \tag{3.2}$$

with

$$\text{SNR}(t,f) = 10\log_{10}\left(\frac{\text{IRM}(t,f)^{1/\beta}}{1-\text{IRM}(t,f)^{1/\beta}}\right) \tag{3.3}$$

where the local criterion (LC) is the threshold between speech dominant (1), and noise dominant (0) T-F units. Previous studies have compared different values for LC (Loizou and Kim [2011]; Healy et al. [2015]; Chen et al. [2016]), which have shown an LC set 5 dB lower than the overall SNR provides best performance, and is subsequently used in this work for all SNRs, and is equivalent to that used for binary masking (Section 2.2.2).

3.3 Feature extraction for mask estimation

Feature extraction aims to identify suitably discriminative information in the noisy input speech and video that enables a model to determine the speech dominant proportion of each T-F units which will be retained. Previous work in Chapter 2, explored two methods of acoustic feature extraction (MRCG and ARPMG) and visual feature extraction (AAM) within a binary mask estimation framework. The same features are selected here for evaluation in ratio mask estimation.

The multi-resolution cochleagram (MRCG) feature combines 4 different cochleagrams, of both high and low resolution, into a single feature, and was specifically designed for mask estimation based within a cochleagram framework (Chen et al. [2014]). The complementary feature set (ARPMG) is an ensemble of commonly used acoustic features, comprised of amplitude modulation spectrum (AMS), relative spectral transformed perceptual linear prediction (RASTA-PLP), mel-frequency cepstral coefficients (MFCCs) and a gammatone filterbank (GFB). The specific implementation used is taken from (Healy et al. [2015]). The active appearance model (AAM) is a model-based combination of shape and appearance, producing a compact feature representation of a mesh fitted to the speaker lips. Details of the implementations of MRCG, ARPMG and AAM feature extraction methods are discussed in Sections 2.3.1.1, 2.3.1.2 and 2.3.2 respectively.

3.4 Feed-forward neural network (DNN) for ratio mask estimation

This work uses the same pipeline and training procedure as previously used within binary mask estimation (Section 2.4), except is trained to estimate an ideal ratio mask (IRM). A DNN is used as the model for learning the mapping between input features and target masks. Figure 3.5 shows an overview of how DNNs are trained. The DNN takes features as input and outputs a predicted ratio mask (PRM). The error between the PRM and IRM is calculated through a loss function, and passed back through the DNN through backpropagation. The DNN updates internal weighting, and the procedure of calculating error is repeated, with error minimisation being the overall goal.

The same DNN architecture as used for binary masking is again used here for ratio masking. The DNN architecture is shown in Figure 3.6 and comprises 4 dense layers containing 1024 rectified linear units (ReLU) and a final output layer. To enable the estimation of ratio masks, the output layer is changed from a sigmoid

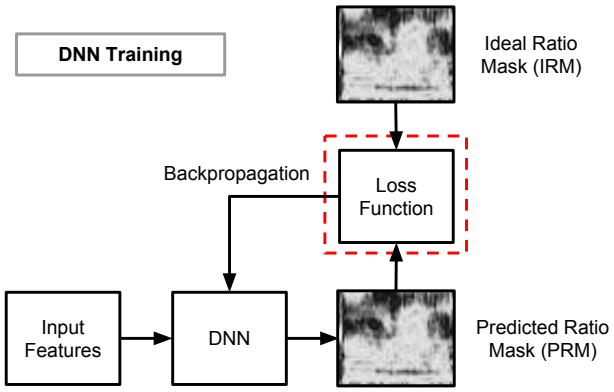


Figure 3.5: Overview of training DNNs for ratio mask estimation.

layer used in binary masking, to a linear layer for ratio masking, as the target output is no longer constrained to values of 0 or 1. The model takes as input a window of stacked input features $\hat{\mathbf{X}}_t = [\mathbf{x}_{t-K}; ...; \mathbf{x}_t; ...; \mathbf{x}_{t+K}]$, and outputs a vector corresponding to the central frame from the input window at time t, $\hat{\mathbf{Y}}_t = [\mathbf{y}_t]$.

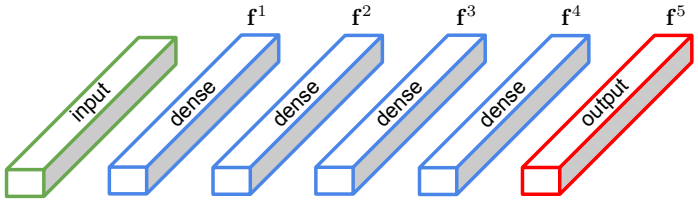


Figure 3.6: Feed-forward (DNN) speech enhancement architecture.

This task is no longer a classification problem but regression, and therefore requires a different loss function for the model to optimise. The standard loss function used for regression tasks is mean squared error (MSE), and is consequently selected for this task. The aim of MSE is to minimise the squared error between the ideal ratio target mask (IRM) and predicted ratio mask (PRM), with $\mathrm{L}^{\mathrm{MSE}}$ calculated as:

$$\mathrm{L}^{\mathrm{MSE}} = \frac{1}{TF} \sum_{t=1}^{T} \sum_{f=1}^{F} \left[(\mathbf{y}_{(t,f)} - \hat{\mathbf{y}}_{(t,f)})^2 \right] \tag{3.4}$$

where $\mathbf{y}$ and $\hat{\mathbf{y}}$ are vectors that comprise concatenated frames of T-F units for each mini-batch in DNN training, from the IRM and PRM respectively. Each of these

vectors comprises T time frames and F filterbanks.

3.5 Experimental results

The performance of feature extraction methods and the inclusion of visual information within feed-forward neural networks (DNN) is compared within ratio mask estimation speech enhancement. Firstly, the feature extraction methods outlined in Section 3.3 are optimised across audio-only, visual-only and audio-visual models. The best performing feature extraction methods is then used to compare performance cross varying noise type and SNR conditions.

The first experiment compares feature extraction methods outlined in Section 3.3, namely multi-resolution cochleagrams (MRCG), complementary feature set (ARPMG) and active appearance models (AAM). Initially audio-only and visual-only models are compared to find the best performing feature extraction methods, before combining into audio-visual models. This experiment is conducted in babble noise at $-5\,\text{dB}$, for audio-only, visual-only and audio-visual models using the validation set, and optimises input window width. The best performing features are selected for further analysis.

The second experiment expands on the previous experiment by introducing additional noise types and varying SNRs, specifically babble and factory noise at SNRs of $-10\,\text{dB}$, $-5\,\text{dB}$, $0\,\text{dB}$ and $5\,\text{dB}$, again in audio-only, visual-only and audio-visual models.

The DNN models were implemented within the Lasagne framework (Dieleman et al. [2015]) with the Theano (Theano Development Team [2016]) back-end. Input data was z-score normalised and grouped into mini-batches of 256. To prevent overfitting, dropout of 0.2 was applied between all layers and early stopping (Prechelt [1998]) was used when the validation score did not improve after 5 further epochs. Training used backpropagation with the Adam optimiser (Kingma and Ba [2014]) and a learning rate of 0.0001, minimising the MSE loss function. All experiments

use a single speaker (speaker 12) from the GRID dataset (details provided in Section A.1), containing 1000 utterances which are allocated into 640, 160 and 200 for the training, validation and test sets respectively.

3.5.1 Comparing feature extraction methods

An initial comparison is made between the feature extraction methods outlined in Section 2.3, namely multi-resolution cochleagrams (MRCG), complementary feature set (ARPMG) and active appearance models (AAM). Audio-only and visual-only experiments are first performed, with the best performing acoustic feature then selected for use in audio-visual experiments. The window width size of input feature context $\mathbf{X}_t = [\mathbf{x}_{t-K}; ...; \mathbf{x}_t; ...; \mathbf{x}_{t+K}]$ is also compared, with width K ranging from 1 to 17 across all experiments. For this investigation, experiments are conducted in babble noise at -5 dB only.

Figure 3.7 shows the intelligibility score produced from ESTOI for all features across the validation set. The validation set was selected for this initial comparison as it is effectively parameter searching for the optimal window width K and the feature set to be used in future experiments.

Comparing initially the window width k across all features, we find similar trends to that found within binary masking (Section 2.6.1), with performance increasing up to a width of 13 before flattening. On average the best performing window is of size 15, which is equivalent to a total window of 320 ms. Therefore a window of $k = 15$ is selected for future experiments.

When comparing the two acoustic features for audio-only mask estimation, MRCG and ARPMG, results show that the MRCG feature consistently outperforms ARPMG across all window widths, just as that found in binary masking. For ratio masking the difference between MRCG and ARPMG is increased with an average improvement of 6.3 in ESTOI compared with 4.0 for binary masking. This increase in performance for MRCG over ARPMG is attributed to using a cochleagram based

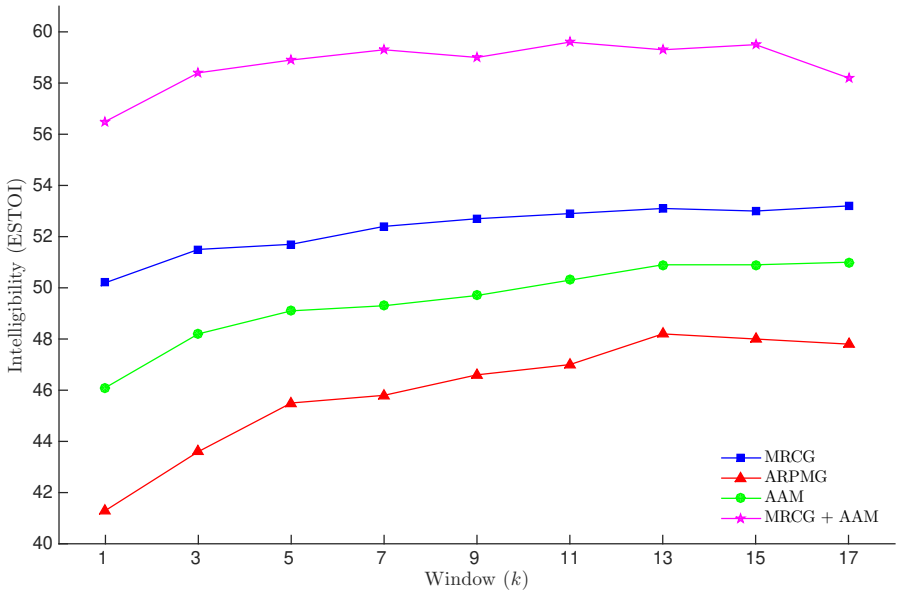


Figure 3.7: Effect of feature extraction methods and temporal window width on intelligibility through ESTOI in babble noise at -5 dB.

framework. The MRCG is more closely related to the framework and target mask, where both are generated using cochleagrams. The complementary feature set does contain a gammatone feature, however, does not represent the energy of each frame as found within the cochleagrams used in MRCG. Although the features used within the complementary feature set are successful in other tasks, for this specific task and framework, the MRCG feature is more appropriate.

Now looking at the visual-only performance using AAMs, unlike in binary masking where performance was almost the same as that of MRCG with a window width above 7, for ratio masking the performance is consistently worse than MRCG, sitting halfway between MRCG and ARPMG. The difference between AAM and ARPMG in ESTOI is an average improvement of 3.5 for ratio masking compared with 3.4 for binary masking. For ratio masking the performance of AAM is increased with an average improvement of 3.2 in ESTOI over that found for binary masking. This further confirms that the MRCG feature is the strongest and is more suited for ratio masking compared to the complementary feature. Again just as with the binary

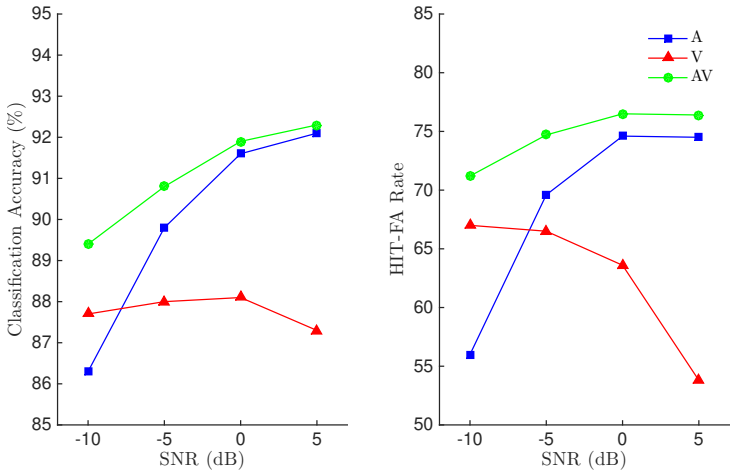
masking experiment (Section 2.6.1), it is worth recognising that these experiments are at low SNR (−5 dB) where the acoustic information is more corrupted, yet the visual information is unaffected.

For audio-visual masking the acoustic MRCG feature and visual AAM feature are combined through stacking on input to the DNN. This combination provides large gains over both audio-only and visual-only models. An average gain in ESTOI of 6.4 for ratio masking compared to 6.2 found in binary masking shows that even though the MRCG feature outperformed the AAM feature individually, when combined the visual AAM still compliments the acoustic MRCG providing further performance gains. Combining both modalities into a single feature provides a complementary feature for the DNN at this low SNR, where the visual information provides the mouth movement (narrowing down the potential output mask options), and the acoustic information provides information about the noise (fine tuning the output mask per noise level).

3.5.2 Analysis across noise type and SNR

In Section 3.5.1 feature extraction methods were tested in babble noise at −5 dB only, revealing extracting acoustic MRCG features, and visual AAM features to perform best. This experiment uses the best performing features, and expands on those tests to consider SNRs of −10 dB, −5 dB, 0 dB and 5 dB in both babble and factory noise. Previously, the focus was on solely on intelligibility through ESTOI, however now classification accuracy, HIT-FA rate and PESQ objective measures are considered.

Table 3.1 shows the full set of objective measures for the test set across all noise type and SNR conditions tested, for audio-only, visual-only and audio-visual models respectively. Objective measures selected are classification accuracy, HIT-FA rate, PESQ and ESTOI. Figures 3.8 and 3.9 provide detailed breakdowns from Table 3.1 for babble noise at −10 dB, −5 dB, 0 dB and 5 dB.



(a) Classification Accuracy (b) HIT-FA

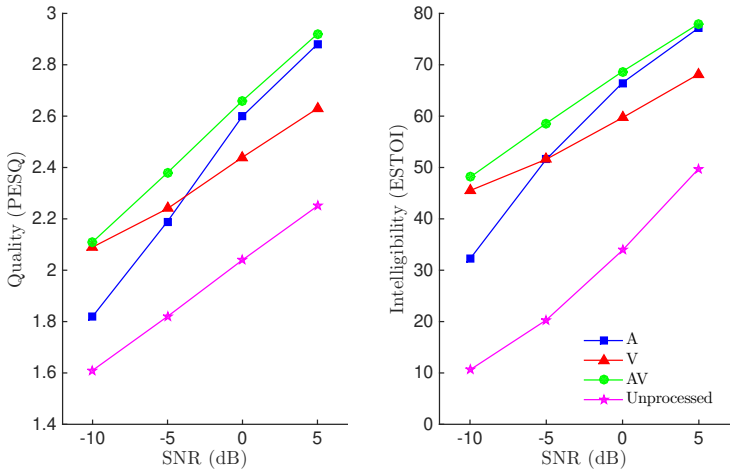
Figure 3.8: Effect on mask classification accuracy and HIT-FA rate across SNR for audio-only, visual-only and audio-visual models in babble noise for ratio mask estimation.

Focusing first on classification accuracy and HIT-FA rate, results show similar trends for both measures across noise types and SNRs, with regards to model performance, shown in Figure 3.8 for babble noise at SNRs from $-10\,\text{dB}$ to $5\,\text{dB}$. Audio-visual models provide best performance across all noise types and SNRs for both classification accuracy and HIT-FA rate. Audio-only performs well at high SNRs, outperforming visual-only and reaching equivalent performance to audio-visual for classification accuracy. However, at low SNRs, audio-only performs particularly poor, falling below visual-only. This is due to the high levels of noise at low SNRs, corrupting the acoustic signal. The DNN is unable to extract useful information from the heavily degraded signal.

For visual-only systems, classification accuracy provides a consistent score across all SNRs for each noise type. This is due to the visual feature being unaffected by noise type or SNR corrupting the audio stream, and the performance is provided by how well the DNN can map the input visual features to the target mask. However, for HIT-FA rate, the performance is best at low SNRs and drops at higher SNRs. The number of HITs is reduced at higher SNRs, and the number of FAs is also reduced at higher SNRs, suggesting the DNN is favouring towards the suppressions. This

is due from the increased difference between noise dominant and speech dominant IRM values found at higher SNRs. At low SNRs, the speech dominant T-F units produce IRM values closer to that of noise dominant, and as such when minimising the MSE loss function, the mean between noise dominant and speech dominant is closer, providing better performance in terms of HIT-FA rate. This reduction in both HITs and FAs is equal in terms of the number of T-F units due to the classification accuracy staying constant.

When comparing the performance of audio-visual against audio-only and visual-only, an average improvement across both babble and factory noise at $-10\,\text{dB}$ for classification accuracy of 2.7 and 1.7, and for HIT-FA rate of 13.3 and 4.4 can be found over audio-only and visual-only models. Now comparing the performance of audio-visual against visual-only (as audio-only performs similarly to audio-visual) across both babble and factory noise at $5\,\text{dB}$, an average improvement of 4.4 and 21.1, is found for classification accuracy and HIT-FA rate respectively.



(a) PESQ (b) ESTOI

Figure 3.9: Effect on quality through PESQ and intelligibility through ESTOI across SNR for audio-only, visual-only and audio-visual in babble noise for ratio mask estimation.

Looking now at quality scores through PESQ and intelligibility with ESTOI, similar trends as with classification accuracy and HIT-FA rate are found, where audio-visual outperforms audio-only and visual-only. Audio-only performs poorly at low SNRs and well at high SNRs reaching equivalent performance to audio-

visual. Visual-only again performs well at low SNRs, this time reaching equivalent performance to audio-visual, but poorly at high SNRs, shown in Figure 3.9 for babble noise at SNRs from -10 dB to 5 dB.

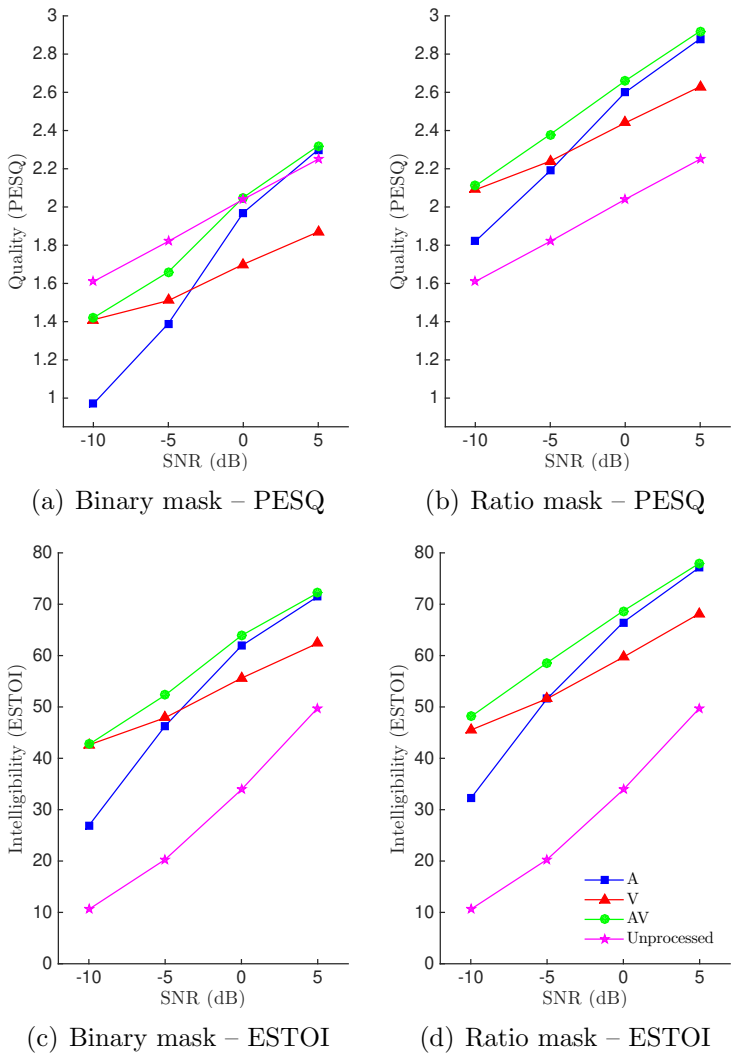
The performance of all systems in PESQ provide large gains over unprocessed audio. Average PESQ gains of 0.57, 0.53, and 0.69 are found for audio-only, visual-only and audio-visual across all noise types and SNRs respectively. Also, the performance of all systems in ESTOI provide large gains over unprocessed audio. Average ESTOI gains of 28.6, 28.3, and 35.1 are found for audio-only, visual-only and audio-visual across all noise types and SNRs respectively.

At low SNRs, when the audio is more corrupted, the visual-only system performs as well as the audio-visual system, suggesting all the important information is held within the visual stream. At high SNRs, the audio-only system performs as well as the audio-visual system, suggesting all the important information is held within the acoustic stream. This finding is similar to how stream weighting is controlled within audio-visual ASR applications (Thangthai et al. [2015]). This confirms why the audio-visual system performs best across all SNRs.

Comparing the overall performance of audio-only, visual-only and audio-visual systems across all noise types and SNRs, audio-visual models were found to consistently provide best performance across all objective measures over audio-only and visual-only. At high SNRs, the benefit gained from combining audio and visual information over audio-only is reduced as the audio features are less degraded by noise which allows the DNN to more effectively map to the target masks in these less challenging conditions. At low SNRs, the benefit gained over visual-only is only found in classification accuracy and HIT-FA, with both PESQ and ESTOI providing equivalent performance, showing visual information provides effectively all information needed.

3.5.3 Comparing binary mask and ratio mask estimation

In Section 3.5.2 the best performing acoustic and visual feature extraction methods were evaluated across varying noise type and SNR conditions for ratio mask estimation. Now, the difference in performance between estimating binary masks (Section 2.6.3) and estimating ratio masks is compared. Figure 3.10 shows a comparison between quality using PESQ and intelligibility using ESTOI for binary mask and ratio mask trained models in babble noise at -10 dB, -5 dB, 0 dB and 5 dB. Similar results and trends are also found across factory noise.



(a) Binary mask – PESQ (b) Ratio mask – PESQ

(c) Binary mask – ESTOI (d) Ratio mask – ESTOI

Figure 3.10: Effect on quality through PESQ and intelligibility through ESTOI across SNR for audio-only, visual-only and audio-visual in babble noise for binary mask estimation and ratio mask estimation.

Focusing first at quality through PESQ, the overall performance of ratio masking is considerably larger than binary masking for all models, across all noise types and SNRs, with large increases found over unprocessed audio across all ratio masking models. Binary masking is shown to provide little (if any) performance improvement over unprocessed audio, yet large gains are found for ratio masking. An average performance difference of 0.72, 0.73 and 0.66 can be found for audio-only, visual-only and audio-visual ratio masking models compared to binary masking models across all SNRs for babble noise. This reveals a key difference between binary masking and ratio masking and shows a clear indication of the benefit of ratio masking over binary masking. The ability to adjust the amount of each T-F unit is retained or suppressed instead of retaining/suppressing the entire unit smooths the transitions between non-speech units, removing distortions caused by enhancement via binary masking.

Looking now at intelligibility through ESTOI, a smaller difference between binary masking and ratio masking models is found than that found in PESQ, with ratio masking models still outperforming binary masking models across all SNRs. An average performance difference of 5.2, 4.1 and 5.6 can be found for audio-only, visual-only and audio-visual ratio masking models compared to binary masking models across all SNRs for babble noise.

Overall, audio-only, visual-only and audio-visual models have large gains in quality and gains in intelligibility for ratio masking models over binary masking models. This is attributed to the varying values within the IRM, compared to the fixed valued IBM. The performance increase in PESQ over unprocessed audio is particularly important, considering binary masking was unable to provide little/no benefit over unprocessed audio.

Table 3.1: Classification accuracy (in %), HIT-FA (in %) PESQ and ESTOI scores for the GRID dataset for ratio mask estimation in babble and factory noise at −10 dB, −5 dB, 0 dB and 5 dB.

Noise (dB)		Model	Acc	HIT-FA (FA)	PESQ	ESTOI
babble	-10	A	86.3	56.0 (6.3)	1.82	32.2
		V	87.7	67.0 (8.4)	2.09	45.5
		AV	**89.4**	**71.2** (7.0)	**2.11**	**48.1**
		unprocessed audio			1.61	10.6
	-5	A	89.8	69.6 (5.8)	2.19	51.6
		V	88.0	66.5 (7.6)	2.24	51.6
		AV	**90.8**	**74.7** (6.0)	**2.38**	**58.6**
		unprocessed audio			1.82	20.3
	0	A	91.6	74.6 (4.6)	2.60	66.5
		V	88.1	63.6 (6.0)	2.44	59.7
		AV	**91.9**	**76.5** (4.8)	**2.66**	**68.7**
		unprocessed audio			2.04	33.9
	+5	A	92.1	74.5 (3.5)	2.88	77.2
		V	87.3	53.8 (3.2)	2.63	68.2
		AV	**92.3**	**76.4** (4.0)	**2.92**	**78.0**
		unprocessed audio			2.25	49.8
factory	-10	A	90.3	59.7 (4.1)	1.95	32.8
		V	90.8	67.5 (5.3)	2.17	47.5
		AV	**92.5**	**72.0** (4.0)	**2.18**	**48.4**
		unprocessed audio			1.46	10.5
	-5	A	93.0	71.9 (3.2)	2.31	52.0
		V	91.0	66.6 (4.8)	2.32	53.0
		AV	**93.6**	**75.8** (3.3)	**2.43**	**58.8**
		unprocessed audio			1.66	20.1
	0	A	94.3	77.2 (2.7)	2.64	67.2
		V	91.0	64.0 (4.1)	2.49	60.2
		AV	**94.4**	**79.0** (2.9)	**2.71**	**69.9**
		unprocessed audio			1.87	33.5
	+5	A	94.7	77.8 (2.2)	2.93	78.2
		V	91.0	59.9 (3.1)	2.65	69.3
		AV	**94.8**	**78.5** (2.3)	**2.96**	**79.0**
		unprocessed audio			2.09	49.9

3.6 Conclusions

This work has examined the effect on intelligibility (ESTOI), quality (PESQ) and mask accuracy (classification accuracy and HIT-FA) of including visual information in ratio mask estimation for speech enhancement. It was found that all systems provide large gains in intelligibility over the unprocessed audio, with largest gains found at lower SNRs. Combining both audio and visual modalities into a single bimodal audio-visual system provides largest gains across all noise types and SNRs, confirming that combining audio and visual features provides a robust complimentary feature set. At high SNRs, the benefit gained from combining audio and visual information over audio-only is reduced as the audio features are less degraded by noise which allows the DNN to more effectively map to the target masks in these less challenging conditions. At low SNRs, the benefit gained over visual-only is only found in classification accuracy and HIT-FA, with both PESQ and ESTOI providing equivalent performance, showing visual information provides effectively all information needed.

This work also compared the previous results found within binary masking based speech enhancement (Section 2.6.3) with ratio masking based speech enhancement, revealing that across all measures, ratio masking outperforms binary masking for audio-only and audio-visual systems, for visual-only ratio masking outperferms binary masking across all measures except for HIT-FA rate. Largest gains are found for quality (PESQ) and intelligibility (ESTOI). The impact on quality provides a key difference between the two masking methods, where binary masking provides little/no improvement over unprocessed audio, yet ratio masking yields large improvements. This is attributed to the constrained nature of the binary mask, a binary mask is a representation of a ratio mask that has been quantised into two classes, speech dominant and noise dominant. This quantisation removes and reduces the resolution and detail found within the ratio mask, which produces the degradation in performance.

Chapter 4

Ratio masking using recurrent neural networks

4.1 Introduction

Previous work in Chapter 3 compared binary masking with ratio masking, finding ratio masking to outperform binary masking across most objective measures. This previous work was conducted using feed-forward neural networks (DNNs) for mask estimation. It was found for both binary and ratio masking that the amount of temporal context supplied to the DNN affected the performance of the model. Temporal modelling is key for learning the relationship between input features and target masks within supervised learning. In this chapter the focus is on improving temporal modelling by using recurrent neural networks (RNNs) instead of DNNs within the speech enhancement framework for ratio mask estimation. Specifically, comparisons are made between three methods of temporal modelling, using feed-forward neural networks, using standard bi-directional recurrent neural networks, and using the proposed bi-directional recurrent feed-forward hybrid neural network (RNN-DNN). Two implementations of recurrent cells are also compared, namely the traditional long short-term memory cell (LSTM) and gated recurrent unit (GRU).

Recurrent neural networks have been applied successfully in many speech related tasks, such as speech recognition (Graves and Schmidhuber [2005]; Graves et al. [2013a,b]; Graves and Jaitly [2014]) and TTS synthesis (Fan et al. [2014]). Other fields of work where recurrent neural networks have been used are handwriting generation (Graves [2013]), image captioning (Kiros et al. [2014]; Vinyals et al. [2015b]; Xu et al. [2015a]) and parsing (Vinyals et al. [2015a]). Recurrent networks are specifically designed for processing a sequence of values, such as a context window of speech, and as such can process larger lengths of sequences than is feasibly possible with other network architectures, such as DNNs. This is because RNNs process each item in a sequence individually through time, and as such connections within the layer are only connected to a single frame on input, plus additional connections from within the recurrent cell. Whereas in DNNs the full sequence is concatenated into a single vector, and as such connections within the layer are connected to all frames of the sequence on input. Therefore, the full number of connections within an RNN layer is considerably smaller than that in a DNN for larger sequence lengths. Another property of RNNs traversing the sequence each frame sequentially allows the sequence to be of variable length, whereas DNNs require a fixed length of sequence due to the fully connected property of feed-forward dense layers.

Initially, recurrent networks were uni-directional and therefore processed the input sequence in a single direction, propagating information from the past to the current step within the sequence. In the context of speech processing this therefore accounts for carry-over coarticulation but does not account for anticipatory coarticulation. For the previously used DNN architecture, a symmetric context surrounding the speech at a particular time was used as input into the network, and as such the network had access to both carry-over and anticipatory coarticulations. To account for this within a recurrent network, the sequence can still contain both past and future context, but instead of traversing the sequence in a single forward direction, a second traversal is performed in the backward direction creating a bi-directional RNN. This allows the bi-directional RNN, at each step within the sequence, to have access to both past information (from the forward direction) and future information

(from the backward direction) providing information regarding both carry-over and anticipatory coarticulations. The choice of recurrent cell used within recurrent layers effects the ability to learn temporal information. Various studies have evaluated the performance of recurrent cells (Chung et al. [2014]; Jozefowicz et al. [2015]; Greff et al. [2017]), revealing no single implementation performs best for all applications, and as such this work compares two popular implementations, namely long short-term memory cell (LSTM) (Graves [2013]) and gated recurrent unit (GRU) (Cho et al. [2014]).

Due to the importance of modelling both carry-over and anticipatory coarticulation for speech enhancement we implement bi-directional recurrent neural networks only, and do not consider uni-directional RNNs. This work continues to consider audio-only, visual-only and audio-visual models. Figure 4.1 shows the training pipeline of the proposed audio-visual speech enhancement system using recurrent neural networks. Visual features are extracted from video and combined with acoustic features extracted from noisy speech, before input into the recurrent neural network (RNN) for temporal modelling to estimate the ratio mask. For testing purposes, estimated masks are applied to a cochleagram of the noisy speech which suppresses noise-dominated T-F units and the remaining signals are overlapped and summed to produce the enhanced signal, shown in Figure 4.2. The same pipeline is used for all speech enhancement configurations, except the visual stream is removed for audio-only models, and the audio stream is removed for visual-only models.

The remainder of this chapter is organised as follows. Section 4.2 provides an overview of the baseline DNN model and acoustic and visual feature extraction methods. Section 4.3 introduces recurrent neural networks and the proposed recurrent feed-forward hybrid neural network architecture, while Section 4.4 describes the implementation of recurrent cells used within recurrent neural networks, namely long short-term memory cells (LSTM) and gated recurrent units (GRU), additional layer normalisation extensions. Performance evaluations are made in Section 4.5 which

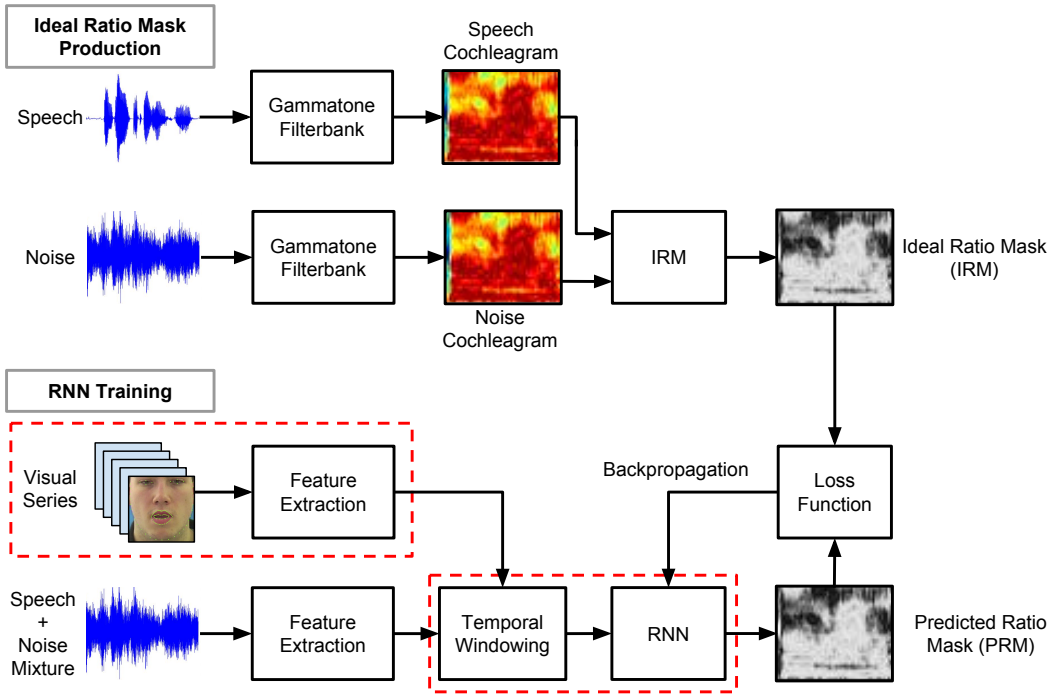


Figure 4.1: Overview of training the RNN ratio masking speech enhancement system.

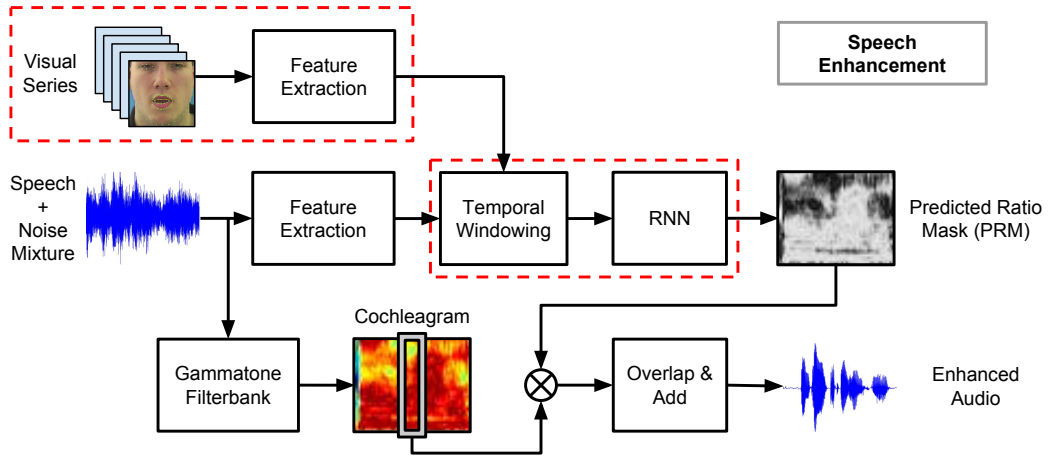


Figure 4.2: Overview of applying the RNN predicted ratio mask to noisy speech for speech enhancement testing.

first compare the effectiveness of the temporal architectures outlined in Sections 4.2 and 4.3 for audio-only, visual-only and audio-visual models (Section 4.5.1). Section 4.5.2 compares the performance of feed-forward neural networks, with standard recurrent neural networks and the proposed recurrent feed-forward hybrid neural

network for temporal modelling. Experiments are conducted across varying noise type and SNR conditions and used the best performing temporal architecture configurations from Section 4.5.1. Finally, this chapter is concluded in Section 4.6.

4.2 Baseline feed-forward neural network based temporal model

Previous work in Chapters 2 and 3 explored using feed-forward neural networks (DNNs) for temporal modelling, using standard feature extraction methods for input within binary masking and ratio masking speech enhancement. This found that using DNNs within ratio masking provided best performance across all objective measures for audio-only, visual-only and audio-visual models. This forms our baseline model for this work and is the chosen architecture for the temporal model. The DNN architecture is shown in Figure 4.3 and comprises 4 dense layers containing 1024 rectified linear units (ReLU) and a final linear output layer. The model takes as input a window of stacked input features $\hat{\mathbf{X}}_t = [\mathbf{x}_{t-K}; ...; \mathbf{x}_t; ...; \mathbf{x}_{t+K}]$, and outputs a vector corresponding to the central frame from the input window at time t, $\hat{\mathbf{Y}}_t = [\mathbf{y}_t]$. Detailed implementations of the DNN are in Section 3.4.

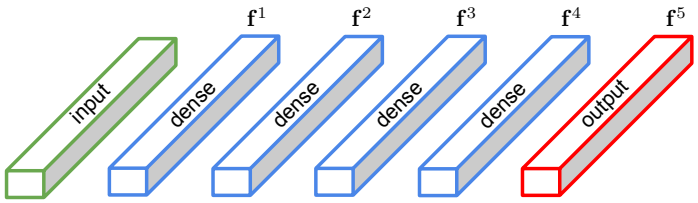


Figure 4.3: Feed-forward (DNN) speech enhancement architecture.

From our previous work in Chapters 2 and 3, the acoustic feature MRCG and visual feature AAM was found to perform best, and as such are selected for this work. The multi-resolution cochleagram (MRCG) feature combines 4 different cochleagrams, of both high and low resolution, into a single feature, and was specifically designed for mask estimation based within a cochleagram framework (Chen et al.

[2014]). The active appearance model (AAM) is a model-based combination of shape and appearance, producing a compact feature representation of a mesh fitted to the speaker lips. Details of the implementations of MRCG and AAM feature extraction methods are discussed in Sections 2.3.1.1 and 2.3.2 respectively. For audio-only experiments the input feature $\mathbf{x} = [\mathbf{x}^{\mathrm{MRCG}}]$, for visual-only experiments the input feature $\mathbf{x} = [\mathbf{x}^{\mathrm{AAM}}]$ while for audio-visual experiments the input feature $\mathbf{x} = [\mathbf{x}^{\mathrm{MRCG}}; \mathbf{x}^{\mathrm{AAM}}]$, where ; is a concatenation function.

4.3 Recurrent neural network based temporal models

Our previous work in Chapter 3 used feed-forward neural network regressors (DNNs) for temporal modelling to learn a mapping from input features $\mathbf{X}$ to target ideal ratio masks $\mathbf{Y}$. This is now expanded with bi-directional recurrent neural networks (RNNs) which model the temporal structure found within speech. This work also proposes a bi-directional recurrent feed-forward hybrid architecture (RNN-DNN) which combines the input and output of the recurrent network, allowing the recurrent layers to focus on temporal information with static information provided from the input. The aim of this section is to show the different recurrent architectures, while Section 4.4 provides detailed implementations of the recurrent cells.

4.3.1 Recurrent neural network (RNN)

We propose extending the DNN implementation (Section 4.2) by using bi-directional recurrent neural networks (RNN), which have shown improvements in other speech processing fields such as recognition (Graves and Schmidhuber [2005]; Graves et al. [2013a]) and TTS synthesis (Fan et al. [2014]), and model the temporal structure found within speech. Bi-directional RNNs process context windows in both forward and backward directions, enabling the model to learn both carry-over and anticipa-

tory coarticulations.

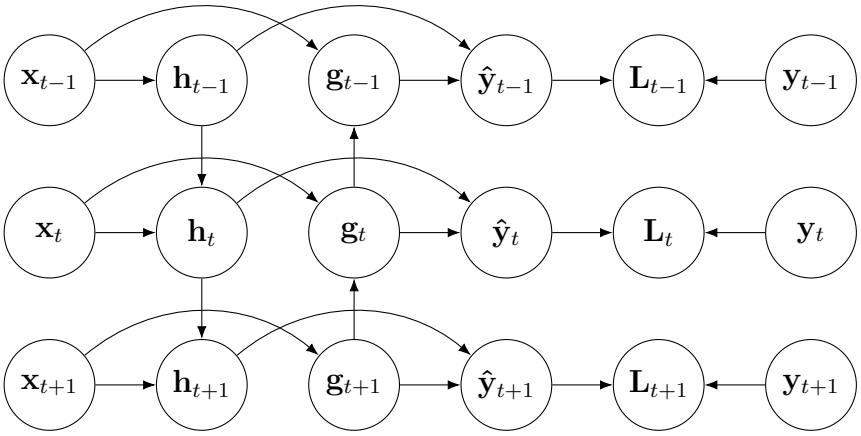


Figure 4.4: Computation of a typical bi-directional recurrent neural network.

A simple bi-directional RNN is shown in Figure 4.4 with $\mathbf{h}_t$ representing the state of the sub-RNN moving forward through time, and $\mathbf{g}_t$ representing the sub-RNN moving backward through time. This allows the output units $\hat{\mathbf{y}}_t$ to provide a prediction that depends on both past and future context. This mapping is then optimised with the target $\mathbf{y}_t$ via the loss function $\mathbf{L}_t$ for each time step t.

The RNN selected for this task comprises 2 pairs of recurrent forward and backward layers, each consisting of 512 (256 forward and 256 backward) recurrent cells (explained in Section 4.4), such that pair $\mathbf{f}^n = [\mathbf{h}^n; \mathbf{g}^n]$, followed by 2 further 1024 ReLU dense layers and a linear output layer, shown in Figure 4.5. The number of recurrent layer pairs, dense layers and number of units in each layer were optimised within a parameter grid search, the total number of layers was fixed at 4 to match the DNN architecture, i.e 1 to 3 recurrent layer pairs with 3 to 1 dense layers.

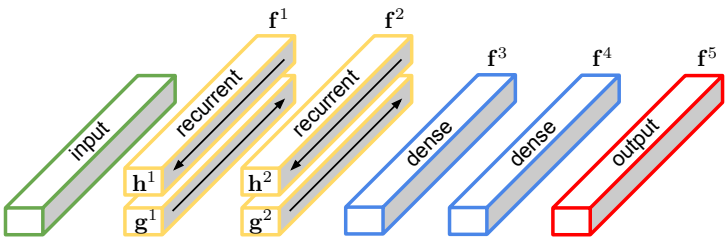


Figure 4.5: Recurrent (RNN) speech enhancement architecture.

The first recurrent layer pair $\mathbf{f}^1$ traverses the full input context window:

$$\mathbf{h}^1 = \overrightarrow{[\mathbf{x}_{t-K}, ..., \mathbf{x}_t, ..., \mathbf{x}_{t+K}]} \tag{4.1}$$

$$\mathbf{g}^1 = \overleftarrow{[\mathbf{x}_{t-K}, ..., \mathbf{x}_t, ..., \mathbf{x}_{t+K}]} \tag{4.2}$$

and the second recurrent layer pair $\mathbf{f}^2$ traverses the full context window from $\mathbf{f}^1$:

$$\mathbf{h}^2 = \overrightarrow{[\mathbf{h}^1_{t-K}, ..., \mathbf{h}^1_t, ..., \mathbf{h}^1_{t+K}]} \tag{4.3}$$

$$\mathbf{g}^2 = \overleftarrow{[\mathbf{h}^1_{t-K}, ..., \mathbf{h}^1_t, ..., \mathbf{h}^1_{t+K}]} \tag{4.4}$$

The output from $\mathbf{f}^2$ is then reshaped such that each time step from the context window is passed through the DNN layers ($\mathbf{f}^3$ and $\mathbf{f}^4$) separately, ie $\omega_x \times 512$, where ω_x is the width of the input window, producing a separate output from the network for each time step of size 64 ($\omega_x \times 64$), as our target IRM is a vector of size 64 for each time step, before reshaping back into context window form $\hat{\mathbf{Y}}_t = [64 \times \omega_x]$. This final reshaped predicted output is used within the loss function for training with target $\mathbf{Y} = [\mathbf{y}_{t-K}, ..., \mathbf{y}_t, ..., \mathbf{y}_{t+K}]$.

When applying this predicted mask to the noisy speech in the enhancement stage, only the central frame at time t is used, providing the same output as that from the DNN model. In preliminary tests, extracting the central frame outperformed taking an average across all frames within the context window. Although the network could have been trained to output only the central frame, it was found that the network learned a more accurate central frame at time t when also learning the target for all other time steps, $t - K$ to $t + K$.

4.3.2 Recurrent feed-forward hybrid neural network (RNN-DNN)

This work proposes an extension to the standard RNN architecture by introducing a skip connection between the input layer and the final output of the recurrent

layers. The aim is to allow the recurrent layers to focus on temporal modelling while the skip connection provides static information. This new network structure is defined as a recurrent feed-forward hybrid neural network (RNN-DNN). Inspiration comes from traditional speech processing tasks, where temporal derivatives are stacked with static features before passing into models, traditionally by concatenating velocity and acceleration features with static frames (Furui [1986]; Hanson and Applebaum [1990]). This approach is part of the ETSI standard (ETSI [2002]) for MFCC production used commonly in acoustic speech processing tasks.

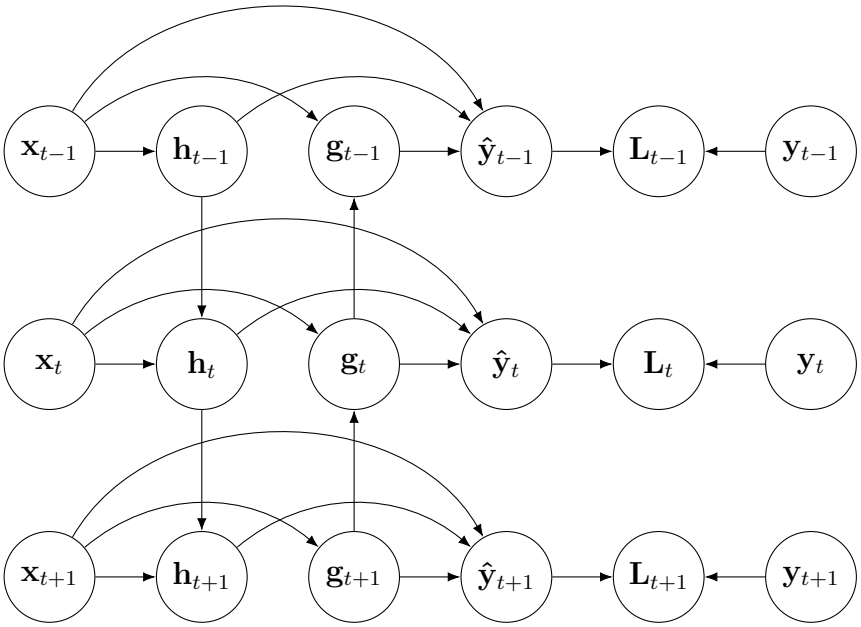


Figure 4.6: Computation of a typical bi-directional recurrent feed-forward hybrid neural network.

A simple bi-directional RNN-DNN is shown in Figure 4.6 with $\mathbf{h}_t$ representing the state of the sub-RNN moving forward through time, and $\mathbf{g}_t$ representing the sub-RNN moving backward through time. The extra connection from $\mathbf{x}_t$ to $\hat{\mathbf{y}}_t$ allows the output units $\hat{\mathbf{y}}_t$ to provide a prediction that depends on both past and future context and the current static input frame. This mapping is then optimised with the true target $\mathbf{y}_t$ via the loss function $\mathbf{L}_t$ for each time step t.

The RNN-DNN selected for this task follows the same construction as the RNN shown in Section 4.3.1 with the additional RNN-DNN skip connections, shown in

Figure 4.7.

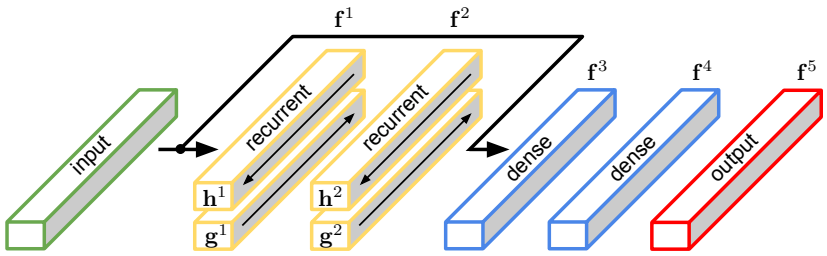


Figure 4.7: Recurrent feed-forward hybrid (RNN-DNN) speech enhancement architecture.

The skip connection between the input layer and the final output from the recurrent layers is achieved by concatenation such that the input into the first dense layer ($\mathbf{f}^3$) is:

$$\mathbf{f}^3 = [\mathbf{X}; \mathbf{f}^2], \quad \text{where } \mathbf{f}^2 = [\mathbf{h}^2; \mathbf{g}^2] \tag{4.5}$$

The remaining layers and reshaping steps are the same as that for the RNN described in Section 4.3.1, producing the same size output.

4.4 Recurrent neural network cells

The recurrent layers in the temporal models discussed in Section 4.3 are constructed from recurrent cells or units instead of the standard units used in dense layers. The recurrent units calculate their output based on not only the current time step, but previous time steps that they have seen. The two most popular recurrent units are the long short-term memory cell (LSTM) and gated recurrent unit (GRU), and are both selected for evaluation in our experiments. Various studies have evaluated which recurrent unit is best (Chung et al. [2014]; Jozefowicz et al. [2015]; Greff et al. [2017]), finding that with small modifications to either LSTM or GRU for the specific task, performance was generally equal using either, with no clear best unit. In this work we compare both units using their standard implementations.

4.4.1 Long short-term memory (LSTM)

The long short-term memory cell (LSTM) (Graves [2013]) has been shown to provide state-of-the-art performance in many speech processing fields such as recognition (Graves and Schmidhuber [2005]; Graves et al. [2013a,b]; Graves and Jaitly [2014]) and text-to-speech synthesis (Fan et al. [2014]), and is generally the recurrent unit of choice. The benefit of using LSTM cells comes from the ability to store information (within a cell), allowing long range information found within the input sequence can be exploited, unlike standard RNN units which do not contain cells and are only able to access short term information. Additionally, LSTM cells are able to overcome the problems of vanishing gradients typically found with standard RNN units.

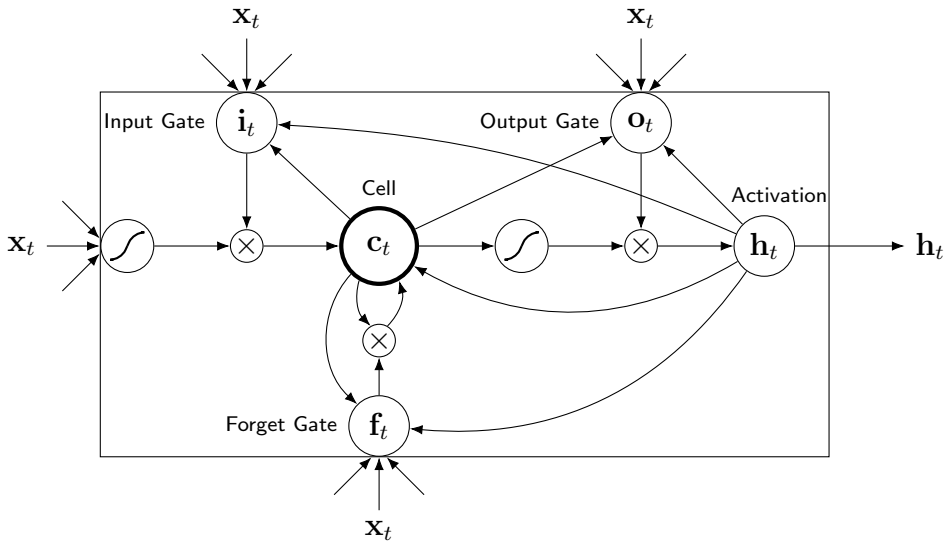


Figure 4.8: Long short-term memory cell.

The LSTM cell makes use of gates to control the flow of input and output information from both the cell itself and it's internal storage cell. Figure 4.8 shows a diagram of the connections between gates, storage cell, inputs and outputs found within the LSTM cell. The LSTM cell contains three gates: input, output, and forget. The forget gate, $\mathbf{f}_t$, decides which information currently within the internal

storage cell ($\mathbf{c}_t$) should be kept or forgotten, and is calculated as:

$$\mathbf{f}_t = \sigma(\mathbf{x}_t\mathbf{W}_{xf} + \mathbf{h}_{t-1}\mathbf{W}_{hf} + \mathbf{W}_{cf} \odot \mathbf{c}_{t-1} + b_f) \tag{4.6}$$

where $\mathbf{x}_t$ is the input data, $\mathbf{h}_{t-1}$ is the activation of the LSTM cell of the previous time step and $\mathbf{c}_{t-1}$ is the information stored within the internal storage cell of the previous time step. These connections to $\mathbf{c}_{t-1}$, providing information stored within the internal storage cell, for all gates are called peephole connections, which have been shown to provide increased performance (Gers and Schmidhuber [2000]). The contribution of input, previous activation and storage cell are determined by the weighting terms $\mathbf{W}_{xf}$, $\mathbf{W}_{hf}$, $\mathbf{W}_{cf}$ and bias b_f for the forget gate calculation. The output of the forget gate is controlled by the activation of the sigmoid function, σ. When the output is close to 0, the information in the storage cell is forgotten, whereas when the output is close to 1, the information in the storage cell is retained.

The forget gate decided how much of the previous data in the storage cell should be forgotten, now the input gate decides how much of the current input frame should also be stored within the storage cell. The input gate, $\mathbf{i}_t$, is similar to that of the forget gate and is calculated as:

$$\mathbf{i}_t = \sigma(\mathbf{x}_t\mathbf{W}_{xi} + \mathbf{h}_{t-1}\mathbf{W}_{hi} + \mathbf{W}_{ci} \odot \mathbf{c}_{t-1} + b_i) \tag{4.7}$$

where the contribution of input, previous activation and storage cell are determined by the weighting terms $\mathbf{W}_{xi}$, $\mathbf{W}_{hi}$, $\mathbf{W}_{ci}$ and bias b_i for the input gate calculation. Similar to the forget gate, the output of the input gate is controlled by the activation of the sigmoid function, σ. When the output is close to 0, the information in the input is not stored in the storage cell, whereas when the output is close to 1, the information in input is stored in the storage cell. The data stored in the internal storage cell, $\mathbf{c}_t$, is updated and calculated as:

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \tau(\mathbf{x}_t\mathbf{W}_{xc} + \mathbf{h}_{t-1}\mathbf{W}_{hc} + b_c) \tag{4.8}$$

where activations of the forget gate removes old data, and activations of the input gate store new data. Further contributions from the input and previous activation are provided by the element-wise multiplication, ($\odot$), with the tanh function, τ, of the weighted input and previous activation from terms $\mathbf{W}_{xc}$, $\mathbf{W}_{hc}$ and bias b_c. The τ operation is used instead of the sigmoid to avoid the vanishing gradient problem when training recurrent neural networks, as the second derivative can sustain for a long range before going to 0, unlike the sigmoid function.

The output gate, $\mathbf{o}_t$, determines what information currently stored in the internal storage cell ($\mathbf{c}_t$) should be output, and is calculated as:

$$\mathbf{o}_t = \sigma(\mathbf{x}_t\mathbf{W}_{xo} + \mathbf{h}_{t-1}\mathbf{W}_{ho} + \mathbf{W}_{co} \odot \mathbf{c}_t + b_o) \tag{4.9}$$

where the key difference between the calculation of the output gate and forget gate or input gate is only information stored in the storage cell and previous activation is used for output gate calculation. The contribution of input, previous activation and storage cell are determined by the weighting terms $\mathbf{W}_{xo}$, $\mathbf{W}_{ho}$, $\mathbf{W}_{co}$ and bias b_o for the output gate calculation. Similar to both the forget gate and input gate, the output of the output gate is controlled by the activation of the sigmoid function, σ. When the output is close to 0, the information in the storage cell is not output, whereas when the output is close to 1, the information in the storage cell is output. The final LSTM cell activation, $\mathbf{h}_t$, is calculated as:

$$\mathbf{h}_t = \mathbf{o}_t \odot \tau(\mathbf{c}_t) \tag{4.10}$$

where the final application of the tanh function, τ, ensures that the outputs from the cell are in the range of -1 to 1.

4.4.2 Gated recurrent unit (GRU)

An alternative recurrent unit to the LSTM cell is the gated recurrent unit (GRU) (Cho et al. [2014]) which instead of containing a specific internal storage cell to store

data, uses only the current input and the activation of previous time step. This means that to model long range information, the GRU must continue to output required information within it's activation, otherwise only short term information is available.

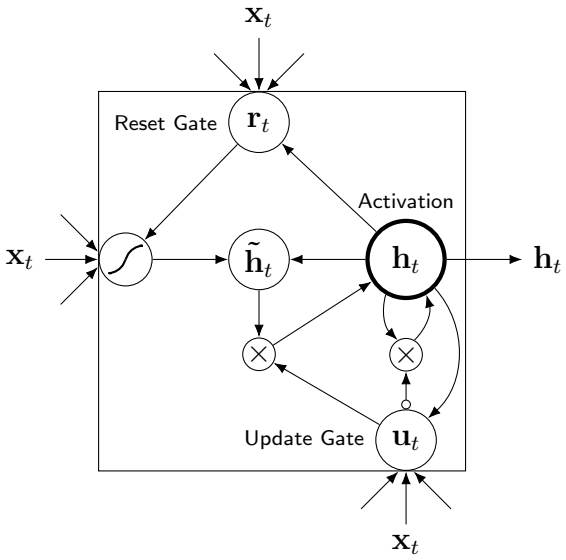


Figure 4.9: Gated recurrent unit.

Similar to the LSTM cell, the GRU makes use of gates to control the flow of input and output information. Figure 4.9 shows a diagram of the connections between gates, inputs and outputs found within the GRU. The GRU contains two gates: reset and update. The reset gate, $\mathbf{r}_t$, is similar to the forget gate, decides which information from the previous activation ($\mathbf{h}_{t-1}$) should be kept or reset, and is calculated as:

$$\mathbf{r}_t = \sigma(\mathbf{x}_t\mathbf{W}_{xr} + \mathbf{h}_{t-1}\mathbf{W}_{hr} + b_r) \tag{4.11}$$

where $\mathbf{x}_t$ is the input data and $\mathbf{h}_{t-1}$ is the activation of the GRU of the previous time step. The contribution of input and previous activation are determined by the weighting terms $\mathbf{W}_{xr}$, $\mathbf{W}_{hr}$ and bias b_r for the reset gate calculation. The output of the forget gate is controlled by the activation of the sigmoid function, σ. When the output is close to 0, the information in the previous activation is reset, whereas

when the output is close to 1, the information in the previous activation is retained.

The reset gate decided how much of the previous data in the storage cell should be forgotten, now the update gate decides how much of the current activation should be updated, similar to combining both the input gate and output gate within the LSTM cell. The update gate, $\mathbf{u}_t$, is similar to that of the reset gate and is calculated as:

$$\mathbf{u}_t = \sigma(\mathbf{x}_t\mathbf{W}_{xu} + \mathbf{h}_{t-1}\mathbf{W}_{hu} + b_u) \tag{4.12}$$

where the contribution of input and previous activation are determined by the weighting terms $\mathbf{W}_{xr}$, $\mathbf{W}_{hr}$ and bias b_r for the update gate calculation. Similar to the reset gate, the output of the update gate is controlled by the activation of the sigmoid function, σ. When the output is close to 0, the information from the previous activation is not updated, whereas when the output is close to 1, the information from the previous activation is updated.

Unlike the LSTM which has a dedicated cell state ($\mathbf{c}_t$), the GRU calculates a candidate state, $\tilde{\mathbf{h}}_t$, from the current input, reset gate and previous activation, calculated as:

$$\tilde{\mathbf{h}}_t = \tau(\mathbf{x}_t\mathbf{W}_{x\tilde{h}} + \mathbf{r}_t \odot (\mathbf{h}_{t-1}\mathbf{W}_{h\tilde{h}}) + b_{\tilde{h}}) \tag{4.13}$$

where activations of the reset gate removes the old activation, and activations of from the input add new data. Further contributions from the input and previous activation are provided by the element-wise multiplication, ($\odot$), of the weighted previous activation and reset gate before adding the new weighted input data, using terms $\mathbf{W}_{x\tilde{h}}$, $\mathbf{W}_{h\tilde{h}}$ and bias $b_{\tilde{h}}$. The tanh function, τ, is used instead of the sigmoid function to avoid the vanishing gradient problem when training recurrent neural networks, as the second derivative can sustain for a long range before going to 0, unlike the sigmoid function.

The final GRU activation, $\mathbf{h}_t$, is calculated as:

$$\mathbf{h}_t = (1 - \mathbf{u}_t) \odot \mathbf{h}_{t-1} + \mathbf{u}_t \odot \tilde{\mathbf{h}}_t \tag{4.14}$$

where the final activation is a linear sum of the previous activation, $\mathbf{h}_{t-1}$ and candidate state $\tilde{\mathbf{h}}_t$, controlled by the update gate $\mathbf{u}_t$.

4.4.3 Layer normalisation

Recurrent networks require longer processing time to train in comparison to feed-forward networks, due to the self-looping involved in unravelling the time steps. A recent approach to improve convergence in feed-forward and convolutional networks is to introduce batch normalisation (Ioffe and Szegedy [2015]), which adds normalisation steps within the network architecture. Normalisation standardises each input using its mean and standard deviation. This is similar to z-score normalisation, which is computed over the entire training set, is instead computed over batches. Batch normalisation works well when the input is a fixed length, which is required in feed-forward and convolutional networks. However, recurrent networks can take input sequences of varying length, which therefore requires an approach different to batch normalisation, called layer normalisation (Ba et al. [2016]). Layer normalisation adjusts the computation of recurrent units to include normalisation steps on gate inputs within each hidden layer. The normalisation function LN is defined as:

$$LN(\boldsymbol{\chi}; \boldsymbol{\alpha}, \boldsymbol{\beta}) = \frac{(\boldsymbol{\chi} - \boldsymbol{\mu})}{\boldsymbol{\sigma}} \odot \boldsymbol{\alpha} + \boldsymbol{\beta} \tag{4.15}$$

with

$$\mu_i = \frac{1}{N} \sum_{n=1}^{N} \chi_{i,n}, \quad \sigma_i = \sqrt{\frac{1}{N} \sum_{n=1}^{N} (\chi_{i,n} - \mu_i)^2} \tag{4.16}$$

where μ_i and σ_i are the mean and standard deviation at position i of the input vector $\boldsymbol{\chi}$ and N is the length of the input feature when training the network. Parameters $\boldsymbol{\alpha}$ (initialised as all 1s) and $\boldsymbol{\beta}$ (initialised as all 0s) are the gain and bias of the same size as $\boldsymbol{\chi}$. Both $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are not fixed but learnable parameters, as even though they are initialised to give zero mean and unit variance, this is unlikely be optimal for the network (particularly when considering non-linear activations), and as such are updated and learnt during training. Layer normalisation, is calculated over all elements across the feature vector. In our work we compare both LSTM cells and GRU units where layer normalisation can be applied to all connections using the input, $\mathbf{x}_t$, previous activation, $\mathbf{h}_{t-1}$, for both LSTM cells and GRUs, and the current internal cell state, $\mathbf{c}_t$, for LSTM cells. The layer normalised version of an LSTM cell is calculated as (where $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ parameters within LN are removed for simplicity):

$$\mathbf{i}_t = \sigma(LN(\mathbf{x}_t\mathbf{W}_{xi}) + LN(\mathbf{h}_{t-1}\mathbf{W}_{hi}) + \mathbf{W}_{ci} \odot \mathbf{c}_{t-1} + b_i) \tag{4.17}$$

$$\mathbf{f}_t = \sigma(LN(\mathbf{x}_t\mathbf{W}_{xf}) + LN(\mathbf{h}_{t-1}\mathbf{W}_{hf}) + \mathbf{W}_{cf} \odot \mathbf{c}_{t-1} + b_f) \tag{4.18}$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \tau(LN(\mathbf{x}_t\mathbf{W}_{xc}) + LN(\mathbf{h}_{t-1}\mathbf{W}_{hc}) + b_c) \tag{4.19}$$

$$\mathbf{o}_t = \sigma(LN(\mathbf{x}_t\mathbf{W}_{xo}) + LN(\mathbf{h}_{t-1}\mathbf{W}_{ho}) + \mathbf{W}_{co} \odot \mathbf{c}_t + b_o) \tag{4.20}$$

$$\mathbf{h}_t = \mathbf{o}_t \odot \tau(LN(\mathbf{c}_t)) \tag{4.21}$$

and to a GRUs as:

$$\mathbf{r}_t = \sigma(LN(\mathbf{x}_t\mathbf{W}_{xr}) + LN(\mathbf{h}_{t-1}\mathbf{W}_{hr}) + b_r) \tag{4.22}$$

$$\mathbf{u}_t = \sigma(LN(\mathbf{x}_t\mathbf{W}_{xu}) + LN(\mathbf{h}_{t-1}\mathbf{W}_{hu}) + b_u) \tag{4.23}$$

$$\tilde{\mathbf{h}}_t = \tau(LN(\mathbf{x}_t\mathbf{W}_{x\tilde{h}}) + \mathbf{r}_t \odot LN(\mathbf{h}_{t-1}\mathbf{W}_{h\tilde{h}}) + b_{\tilde{h}}) \tag{4.24}$$

$$\mathbf{h}_t = (1 - \mathbf{u}_t) \odot \mathbf{h}_{t-1} + \mathbf{u}_t \odot \tilde{\mathbf{h}}_t. \tag{4.25}$$

4.5 Experimental results

The performance of using feed-forward neural networks (DNN), traditional recurrent neural networks (RNN) and the proposed recurrent feed-forward hybrid neural networks (RNN-DNN) for temporal modelling is compared. Firstly, the temporal architectures outlined in Sections 4.2 and 4.3 are optimised across audio-only, visual-only and audio-visual models. The best performing temporal model is then used to compare the performance against DNNs in varying noise type and SNR conditions.

The first experiment compares temporal model architectures outlined in Sections 4.2 and 4.3, recurrent neural network cells and the layer normalisation extension outlined in Section 4.4. This experiment is conducted in babble noise at $-5\,\text{dB}$, for audio-only, visual-only and audio-visual models using the validation set for optimising input window width. The test set is then used with the best performing window width to select which temporal architectures will be used for further analysis.

The second experiment expands on the previous experiment by introducing additional noise types and varying SNRs, specifically babble and factory noise at SNRs of $-10\,\text{dB}$, $-5\,\text{dB}$, $0\,\text{dB}$ and $5\,\text{dB}$, again in audio-only, visual-only and audio-visual models.

The DNN and RNN models were implemented within the Lasagne framework (Dieleman et al. [2015]) with the Theano (Theano Development Team [2016]) backend. Input data was z-score normalised and grouped into mini-batches of 256. To prevent overfitting, dropout of 0.2 was applied between all layers and early stopping (Prechelt [1998]) was used when the validation score did not improve after 5 further epochs. Training used backpropagation with the Adam optimiser (Kingma and Ba [2014]) and learning rates of 0.0001 for DNN and 0.001 for RNN, minimising the MSE loss function. All experiments use a single speaker (speaker 12) from the GRID dataset (details provided in Section A.1), containing 1000 utterances which are allocated into 640, 160 and 200 for the training, validation and test sets respectively.

4.5.1 Comparing temporal model architectures

An initial comparison is made between the temporal model architectures outlined in Section 4.2 for DNN and Section 4.3 for RNNs, recurrent neural network cells and the layer normalisation extension outlined in Section 4.4. Experiments in Chapters 2 and 3 showed that the amount of temporal context supplied to the DNN affected the performance of the model. This experiment first optimises the amount of temporal context supplied to the recurrent temporal models using the validation set (Section 4.5.1.1). Then a comparison of the varying temporal architectures at the selected window width is performed using the test set, to select which architectures will be used in subsequent experiments (Section 4.5.1.2). All experiments are performed in audio-only (MRCG feature), visual-only (AAM feature) and audio-visual (MRCG + AAM features) models in babble noise at $-5\,\text{dB}$.

4.5.1.1 Optimising temporal window width

This experiment optimises the temporal window width of the different temporal architectures outlined in Table 4.1. Firstly, the DNN is compared with a baseline bi-directional RNN using LSTM cells (BiLSTM) and then against the GRU units (BiGRU). The best performing recurrent unit is then used in subsequent models which test our proposed RNN-DNN architecture and the effect of including layer normalisation (LN). Figure 4.10 shows the intelligibility scores produced from ESTOI for all models and architectures across the validation set. An optimal window width ω_k within the input context window, $\mathbf{X} = [\mathbf{x}_{t-K}, ..., \mathbf{x}_{t+K}]$, is explored by varying K between 1 and 17 producing context windows ranging from 40 ms to 360 ms.

Focusing first on comparing the performance of DNN against BiLSTM and BiGRU models, the DNN outperforms the BiLSTM architecture across nearly all window widths for audio-only, visual-only and audio-visual configurations. The DNN also performs better than the BiGRU architecture for short window widths in audio-

Table 4.1: Architecture configurations selected for analysis.

Model	Network architecture
DNN	Feed-forward neural network
BiLSTM	Bi-directional recurrent neural network with LSTM cells
BiGRU	Bi-directional recurrent neural network with GRU units
BiGRU-DNN	Bi-directional recurrent feed-forward hybrid neural network with LSTM cells
LNBiGRU	Layer normalised bi-directional recurrent neural network with GRU units
LNBiGRU-DNN	Layer normalised bi-directional recurrent feed-forward hybrid neural network with LSTM cells

only and audio-visual models, however at long window widths the BiGRU provides large gains over the DNN architecture. The DNN performance stays flat as the context window increases, showing it is unable to take advantage of the increased context available. This shows that the recurrent models are able to learn a better mapping of the longer temporal structure compared to the DNN, yet both perform well with only local context. For visual-only, the DNN performs best across all window widths. This is attributed to the lack of visual variation across time, the visual stream is upsampled to match the acoustic frame rate, and as such provides smoother yet smaller increments between frames. Using GRU units consistently outperformed LSTM cells across all conditions and is chosen for further analysis.

Comparing now the proposed BiGRU-DNN architecture against the standard BiGRU architecture, the BiGRU and BiGRU-DNN perform similarly across all window widths and models. For audio-only BiGRU performs slightly better, yet for visual-only and audio-visual models the BiGRU-DNN performed best. Introducing layer normalisation through LNBiGRU and LNBiGRU-DNN architectures increases performance for both architectures and across all models and window widths, offering best performance found on the validation set.

Overall, comparing the effect of varying window width K within the different temporal architectures across all model conditions, a similar trend is found with

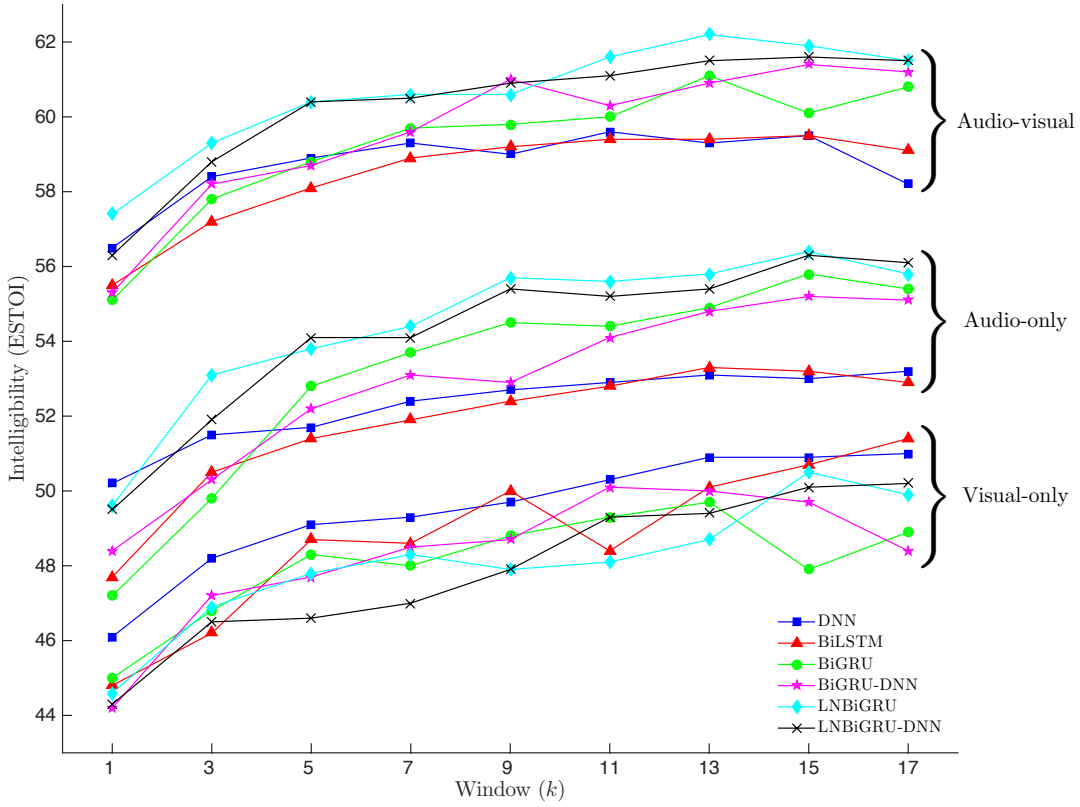


Figure 4.10: Effect of temporal network architecture and window width on intelligibility (ESTOI) in babble noise at −5 dB for audio-only, visual-only and audio-visual inputs.

the recurrent networks as with the feed-forward network. Intelligibility with ESTOI increases as the window width increases, on average the best performing window size is $K = 15$, giving an input context of 320 ms. A window width of $K = 15$ is selected for experiments on the test set (Section 4.5.1.2) for choosing which temporal architectures will be used in varying noise type and SNR conditions (Section 4.5.2).

4.5.1.2 Evaluating temporal model architecture performance

To compare the performance of the temporal architectures the test set is applied with a context window of $K = 15$, which was found optimal in Section 4.5.1.1, in babble noise at −5 dB to each model architecture outlined in Table 4.1. Table 4.2 shows the classification accuracy, HIT-FA rate, PESQ and ESTOI scores for all models using the test set.

Table 4.2: Classification accuracy (in %), HIT-FA (in %), PESQ and ESTOI scores for the GRID dataset in babble noise at −5 dB with different temporal network architectures, window = 31 ($K = 15$), using the test set.

Feat	Network	Acc	HIT-FA (FA)	PESQ	ESTOI
A	DNN	89.8	69.6 (5.8)	2.19	51.6
	BiLSTM	89.6	68.6 (5.6)	2.18	51.8
	BiGRU	90.3	70.5 (5.1)	2.25	54.0
	BiGRU-DNN	90.2	70.1 (5.1)	2.21	53.7
	LNBiGRU	90.4	**71.6** (5.5)	2.25	54.6
	LNBiGRU-DNN	**90.5**	71.0 (4.9)	**2.27**	**54.9**
V	DNN	**88.0**	66.5 (7.6)	2.24	**51.6**
	BiLSTM	87.9	66.2 (7.7)	**2.25**	51.0
	BiGRU	87.8	66.0 (7.7)	2.20	48.1
	BiGRU-DNN	87.8	**67.4** (8.5)	2.22	50.3
	LNBiGRU	87.7	67.1 (8.5)	2.23	51.5
	LNBiGRU-DNN	87.7	66.5 (8.2)	2.21	51.1
AV	DNN	90.8	74.7 (6.0)	2.38	58.6
	BiLSTM	90.7	74.2 (6.1)	2.38	58.7
	BiGRU	91.0	75.9 (6.3)	2.41	59.5
	BiGRU-DNN	91.3	74.9 (5.1)	2.41	60.5
	LNBiGRU	91.3	**76.0** (5.7)	**2.43**	60.8
	LNBiGRU-DNN	**91.5**	75.6 (5.2)	**2.43**	**61.0**
unprocessed audio				1.82	22.0

Focusing first on audio-only models, the BiLSTM model provides a small increase in ESTOI over the DNN, yet a small decrease with all other measures. When moving to the BiGRU system, all measures show gains over both the DNN and the BiLSTM system, particularly for ESTOI. The proposed BiGRU-DNN system performs slightly worse than the BiGRU system, yet when both have layer normalisation applied, the LNBiGRU-DNN becomes the best performing model. Gains are also found with the LNBiGRU system over the BiGRU system showing that layer normalisation does provide performance gains in both architectures. The best performing recurrent network (LNBiGRU-DNN) provides consistent gains over the DNN model for all objective measures, with largest gains found in intelligibility of

3.3 in ESTOI.

Looking now at the audio-visual models, similar trends are found as with audio-only. The BiGRU system outperforms the BiLSTM, and layer normalisation provides gains to both the BiGRU and BiGRU-DNN architectures. The best performing architecture across most measures is again the LNBiGRU-DNN, providing an intelligibility gain of 2.4 in ESTOI over the DNN.

Comparing now the visual-only models, we find that across most measures, no gains are achieved with any recurrent system. This is attributed to the lack of visual variation across time, the visual stream is upsampled to match the acoustic frame rate, and as such provides smoother yet smaller increments between frames. When comparing the recurrent architectures, the BiLSTM model outperforms the BiGRU and BiGRU-DNN models. Layer normalisation again provided gains over the non-normalised versions, allowing performance to beat the BiLSTM and almost match the DNN in terms of ESTOI. This confirms the importance of standardising the recurrent inputs within recurrent neural network cells.

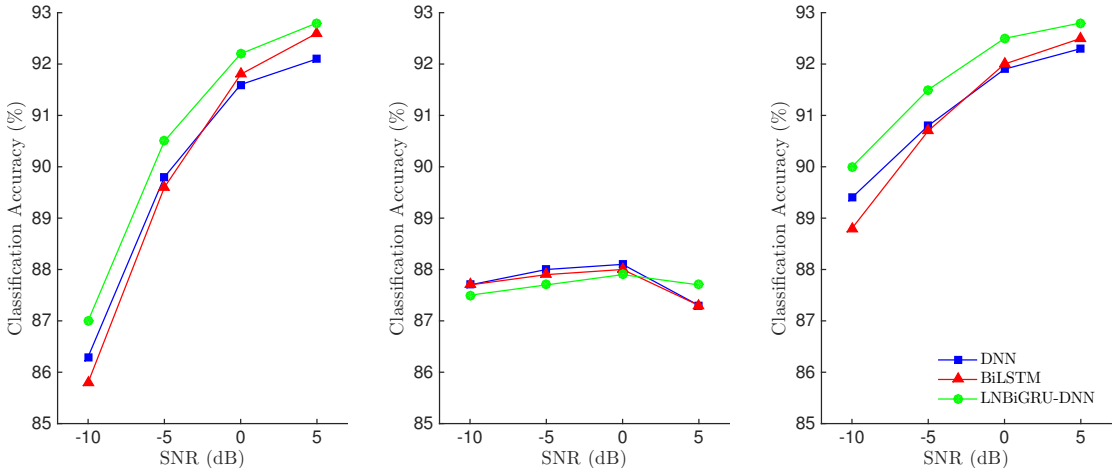
Overall, across both audio-only and audio-visual the best performing recurrent network is the LNBiGRU-DNN providing large gains over the DNN counterpart and the baseline BiLSTM network. We select the DNN, BiLSTM and LNBiGRU-DNN for further analysis in varying noise types and SNR.

4.5.2 Analysis across noise type and SNR

In Section 4.5.1 temporal architectures were compared and optimised, finding the layer normalised bi-directional recurrent feed-forward hybrid neural network using gated recurrent units (LNBiGRU-DNN) to perform best for audio-only and audio-visual models on the test set in babble noise at $-5\,\mathrm{dB}$. In this experiment a comparison between feed-forward neural networks (DNN), standard recurrent neural networks (BiLSTM) and the best performing LNBiGRU-DNN for temporal modelling in varying noise type and SNR conditions. The noise types and SNR conditions

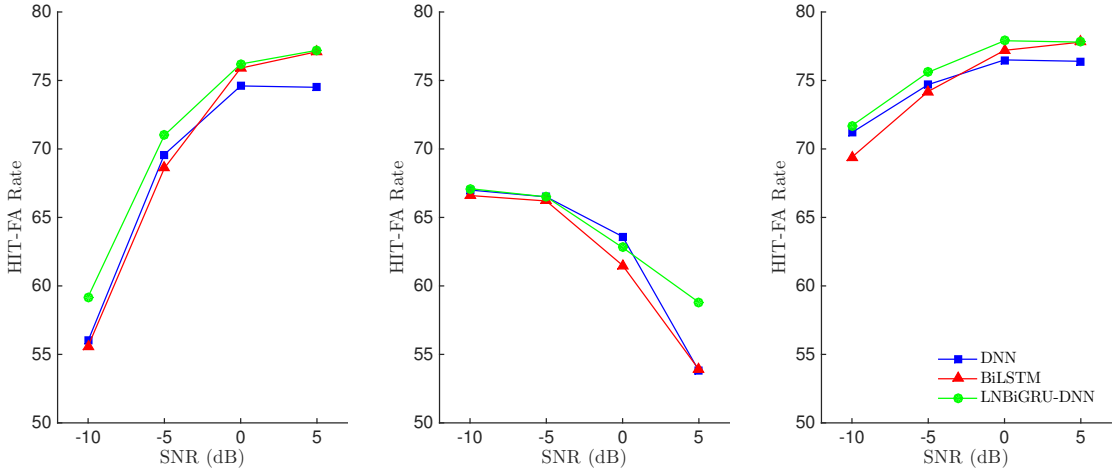
tested are babble and factory noise at SNRs of −10 dB, −5 dB, 0 dB and 5 dB.

Tables 4.3, 4.4 and 4.5 show the full set of objective measures for the test set across all noise type and SNR conditions tested, for audio-only, visual-only and audio-visual models respectively. Objective measures selected are classification accuracy, HIT-FA rate, PESQ and ESTOI. Figures 4.11 to 4.14 provide detailed breakdowns from Tables 4.3, 4.4 and 4.5 for babble noise at −10 dB, −5 dB, 0 dB and 5 dB.



(a) Audio-only (b) Visual-only (c) Audio-visual

Figure 4.11: Effect on mask classification accuracy across SNR for audio-only, visual-only and audio-visual in babble noise for ratio mask estimation.



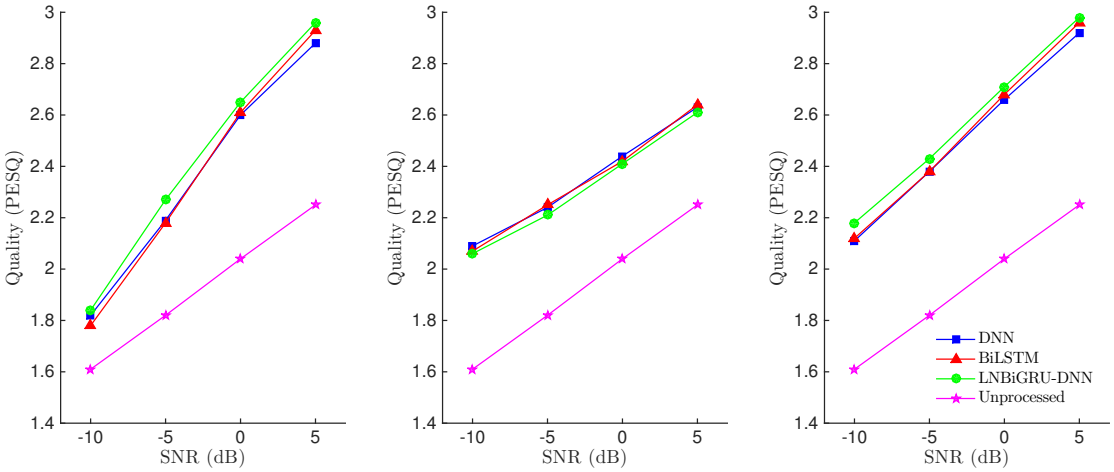
(a) Audio-only (b) Visual-only (c) Audio-visual

Figure 4.12: Effect on mask HIT-FA rate across SNR for audio-only, visual-only and audio-visual in babble noise for ratio mask estimation.

Focusing first on classification accuracy and HIT-FA rate results show that tem-

poral modelling using recurrent neural networks outperform the baseline feed-forward system in most model conditions, for both audio-only and audio-visual models, shown in Figures 4.11 and 4.12 for babble noise at SNRs from −10 dB to 5 dB. The LNBiGRU-DNN outperforms both DNN and BiGRU models, particularly for classification accuracy. The BiLSTM performs similar to the DNN at low SNRs, but shows an increase over DNN models at higher SNRs.

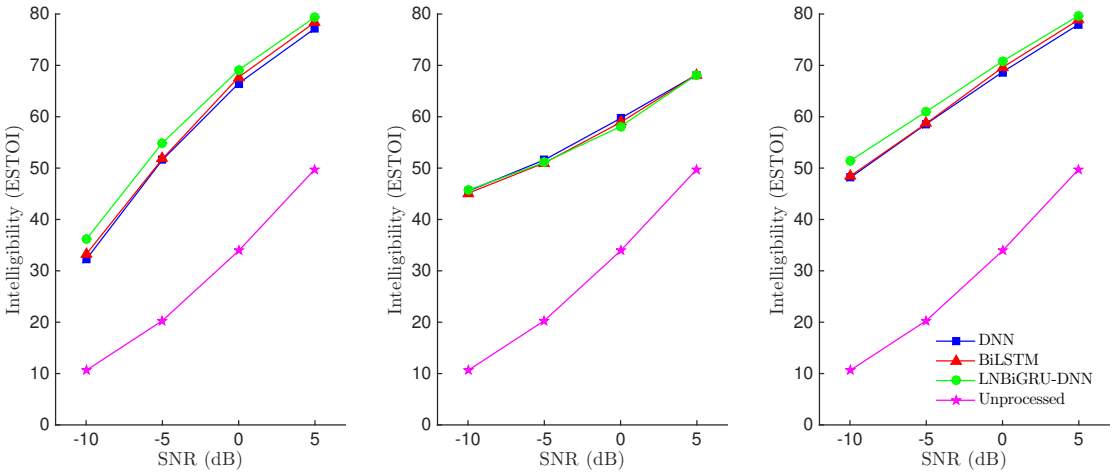
When comparing the LNBiGRU-DNN to the baseline DNN system, an average improvement across both babble and factory noise at −10 dB for classification accuracy of 0.6 and 0.6, and for HIT-FA rate of 1.9 and 1.2 can be found for audio-only and audio-visual models respectively. When comparing the visual-only models, recurrent networks perform worse than DNNs for temporal modelling, except at high SNRs (5 dB) for both classification accuracy and HIT-FA rate.



(a) Audio-only (b) Visual-only (c) Audio-visual

Figure 4.13: Effect on quality through PESQ across SNR for audio-only, visual-only and audio-visual in babble noise for ratio mask estimation.

Looking now at quality scores through PESQ and intelligibility with ESTOI similar trends as with classification accuracy and HIT-FA rate are found, where temporal modelling using recurrent neural networks outperform the baseline feed-forward system most conditions, for both audio-only and audio-visual models, shown in Figures 4.13 and 4.14 for babble noise at SNRs from −10 dB to 5 dB. The performance between all systems is close, with the LNBiGRU-DNN system providing best per-



(a) Audio-only (b) Visual-only (c) Audio-visual

Figure 4.14: Effect on intelligibility through ESTOI across SNR for audio-only, visual-only and audio-visual in babble noise for ratio mask estimation.

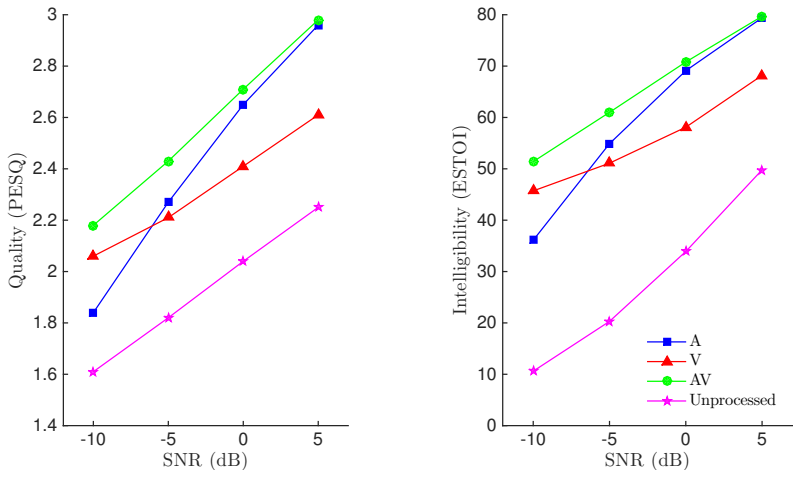
formance, particularly at low SNRs.

When comparing the LNBiGRU-DNN to the baseline DNN system, an average improvement across both babble and factory noise at -10 dB for PESQ of 0.05 and 0.06, and for ESTOI of 2.9 and 2.4 can be found for audio-only and audio-visual models respectively. The performance gained in PESQ is marginal, yet the gain found for ESTOI is relevant, considering how well AAM features perform particularly in this favourable speaker dependent task. At low SNR, gains in PESQ are smaller which is attributed to the increase of noise in the speech, which causes more unavoidable artefacts in the enhanced speech which affect PESQ more than other measures. However, for ESTOI larger gains are found at low SNRs across both audio-only and audio-visual, which is encouraging as the low SNR conditions are more challenging and require more enhancement/noise reduction than compared with high SNRs. When comparing the visual-only models, recurrent networks perform equivalent to DNNs for temporal modelling, showing little variation for both PESQ and ESTOI across all noise types and SNR conditions.

Comparing the performance of the LNBiGRU-DNN to the baseline DNN system across all modalities and SNR conditions, a consistent gain across all measures for

both audio-only and audio-visual models is found. Visual-only models do not benefit from using a recurrent network over a feed-forward network for temporal modelling, although the performance degradation is minimal compared to the gains found in audio-only and audio-visual. With the overall focus being improving intelligibility, large gains are found across all SNR conditions, with larger gains found at the more challenging lower SNR. This improvement at lower SNRs reveals a key benefit from using recurrent networks.

The best performing modality is still audio-visual across all objective measures compared to audio-only and visual-only. Figure 4.15 summarises PESQ and ESTOI scores using the LNBiGRU-DNN in babble noise comparing audio-only, visual-only and audio-visual. Previously, the performance of audio-visual was equivalent to visual-only at an SNR of -10 dB using DNNs, however with the gain found when using the LNBiGRU-DNN models, audio-visual now outperforms the visual-only model. Both audio-only and audio-visual gained benefits from the recurrent architecture, as the SNR increases both models converge performing equally well at 5 dB, as previously found within the DNN architectures.



(a) PESQ (b) ESTOI

Figure 4.15: Effect on quality with PESQ and intelligibility through ESTOI across SNR for audio-only, visual-only and audio-visual in babble noise for ratio mask estimation using the LNBiGRU-DNN architecture.

Table 4.3: (AUDIO-ONLY) Classification accuracy (in %), HIT-FA (in %) PESQ and ESTOI scores for the GRID dataset for audio-only mask estimation with different temporal network architectures in babble and factory noise at −10 dB, −5 dB, 0 dB and 5 dB.

Noise (dB)		Model	Acc	HIT-FA (FA)		PESQ	ESTOI
babble	-10	DNN	86.3	56.0	(6.3)	1.82	32.2
		BiLSTM	85.8	55.6	(7.0)	1.78	33.2
		LNBiGRU-DNN	**87.0**	**59.2**	(6.4)	**1.84**	**36.1**
		unprocessed audio				1.61	10.6
	-5	DNN	89.8	69.6	(5.8)	2.19	51.6
		BiLSTM	89.6	68.6	(5.6)	2.18	51.8
		LNBiGRU-DNN	**90.5**	**71.0**	(4.9)	**2.27**	**54.9**
		unprocessed audio				1.82	20.3
	0	DNN	91.6	74.6	(4.6)	2.60	66.5
		BiLSTM	91.8	75.9	(4.8)	2.61	67.7
		LNBiGRU-DNN	**92.2**	**76.2**	(4.1)	**2.65**	**69.1**
		unprocessed audio				2.04	33.9
	+5	DNN	92.1	74.5	(3.5)	2.88	77.2
		BiLSTM	92.6	77.1	(3.8)	2.93	78.4
		LNBiGRU-DNN	**92.8**	**77.2**	(3.5)	**2.96**	**79.4**
		unprocessed audio				2.25	49.8
factory	-10	DNN	90.3	59.7	(4.1)	**1.95**	32.8
		BiLSTM	90.3	59.0	(3.9)	1.92	34.3
		LNBiGRU-DNN	**91.0**	**62.4**	(3.7)	**1.95**	**37.4**
		unprocessed audio				1.46	10.5
	-5	DNN	93.0	71.9	(3.2)	2.31	52.0
		BiLSTM	93.0	71.9	(3.2)	2.28	52.6
		LNBiGRU-DNN	**93.3**	**73.2**	(3.1)	**2.35**	**54.6**
		unprocessed audio				1.66	20.1
	0	DNN	94.3	77.2	(2.7)	2.64	67.2
		BiLSTM	94.4	77.8	(2.7)	2.65	67.7
		LNBiGRU-DNN	**94.6**	**78.3**	(2.4)	**2.69**	**69.0**
		unprocessed audio				1.87	33.5
	+5	DNN	94.7	77.8	(2.2)	2.93	78.2
		BiLSTM	95.0	**80.0**	(2.3)	2.94	78.7
		LNBiGRU-DNN	**95.1**	79.6	(2.1)	**2.96**	**79.1**
		unprocessed audio				2.09	49.9

Table 4.4: (VISUAL-ONLY) Classification accuracy (in %), HIT-FA (in %) PESQ and ESTOI scores for the GRID dataset for visual-only mask estimation with different temporal network architectures in babble and factory noise at −10 dB, −5 dB, 0 dB and 5 dB.

Noise (dB)		Model	Acc	HIT-FA (FA)		PESQ	ESTOI
babble	-10	DNN	**87.7**	67.0	(8.4)	**2.09**	45.5
		BiLSTM	**87.7**	66.6	(8.1)	2.07	45.1
		LNBiGRU-DNN	87.5	**67.1**	(8.8)	2.06	**45.7**
		unprocessed audio				1.61	10.6
	-5	DNN	**88.0**	**66.5**	(7.6)	2.24	**51.6**
		BiLSTM	87.9	66.2	(7.2)	**2.25**	51.0
		LNBiGRU-DNN	87.7	**66.5**	(8.2)	2.21	51.1
		unprocessed audio				1.82	20.3
	0	DNN	**88.1**	**63.6**	(6.0)	**2.44**	**59.7**
		BiLSTM	88.0	61.5	(5.3)	2.42	58.9
		LNBiGRU-DNN	87.9	62.8	(6.1)	2.41	58.1
		unprocessed audio				2.04	33.9
	+5	DNN	87.3	53.8	(3.2)	2.63	**68.2**
		BiLSTM	87.3	53.9	(3.3)	**2.64**	**68.2**
		LNBiGRU-DNN	**87.7**	**58.8**	(4.6)	2.61	**68.2**
		unprocessed audio				2.25	49.8
factory	-10	DNN	**90.8**	67.5	(5.3)	**2.17**	**47.5**
		BiLSTM	90.5	67.9	(5.8)	2.14	45.6
		LNBiGRU-DNN	90.6	**68.0**	(5.8)	2.14	46.5
		unprocessed audio				1.46	10.5
	-5	DNN	**91.0**	66.6	(4.8)	**2.32**	**53.0**
		BiLSTM	90.9	66.3	(4.9)	2.31	52.1
		LNBiGRU-DNN	90.7	**68.1**	(5.6)	2.28	52.4
		unprocessed audio				1.66	20.1
	0	DNN	**91.0**	64.0	(4.1)	**2.49**	**60.2**
		BiLSTM	90.8	62.8	(4.0)	2.47	59.7
		LNBiGRU-DNN	90.9	**66.4**	(4.9)	2.44	60.1
		unprocessed audio				1.87	33.5
	+5	DNN	**91.0**	**59.9**	(3.1)	**2.65**	**69.3**
		BiLSTM	90.3	52.6	(2.3)	2.63	68.4
		LNBiGRU-DNN	90.7	56.8	(2.8)	2.61	68.0
		unprocessed audio				2.09	49.9

Table 4.5: (AUDIO-VISUAL) Classification accuracy (in %), HIT-FA (in %) PESQ and ESTOI scores for the GRID dataset for audio-visual mask estimation with different temporal network architectures in babble and factory noise at −10 dB, −5 dB, 0 dB and 5 dB.

Noise (dB)		Model	Acc	HIT-FA (FA)		PESQ	ESTOI
babble	-10	DNN	89.4	71.2	(7.0)	2.11	48.1
		BiLSTM	88.8	69.4	(7.3)	2.12	48.4
		LNBiGRU-DNN	**90.0**	**71.7**	(6.3)	**2.18**	**51.4**
		unprocessed audio				1.61	10.6
	-5	DNN	90.8	74.7	(6.0)	2.38	58.6
		BiLSTM	90.7	74.2	(6.1)	2.38	58.7
		LNBiGRU-DNN	**91.5**	**75.6**	(5.2)	**2.43**	**61.0**
		unprocessed audio				1.82	20.3
	0	DNN	91.9	76.5	(4.8)	2.66	68.7
		BiLSTM	92.0	77.2	(4.9)	2.68	69.6
		LNBiGRU-DNN	**92.5**	**77.9**	(4.4)	**2.71**	**70.8**
		unprocessed audio				2.04	33.9
	+5	DNN	92.3	76.4	(4.0)	2.92	78.0
		BiLSTM	92.5	**77.8**	(4.2)	2.96	78.9
		LNBiGRU-DNN	**92.8**	**77.8**	(3.7)	**2.98**	**79.7**
		unprocessed audio				2.25	49.8
factory	-10	DNN	92.5	72.0	(4.0)	2.18	48.4
		BiLSTM	92.4	72.1	(4.1)	2.22	51.3
		LNBiGRU-DNN	**92.8**	**73.7**	(3.9)	**2.25**	**52.6**
		unprocessed audio				1.46	10.5
	-5	DNN	93.6	75.8	(3.3)	2.43	58.8
		BiLSTM	93.6	75.8	(3.4)	2.46	60.3
		LNBiGRU-DNN	**94.0**	**77.1**	(3.1)	**2.50**	**61.7**
		unprocessed audio				1.66	20.1
	0	DNN	94.4	79.0	(2.9)	2.71	69.9
		BiLSTM	94.5	79.5	(3.0)	2.72	70.3
		LNBiGRU-DNN	**94.7**	**79.7**	(2.7)	**2.76**	**71.4**
		unprocessed audio				1.87	33.5
	+5	DNN	94.8	78.5	(2.3)	2.96	79.0
		BiLSTM	95.0	**80.4**	(2.5)	2.97	79.4
		LNBiGRU-DNN	**95.1**	79.8	(2.1)	**3.00**	**79.9**
		unprocessed audio				2.09	49.9

4.6 Conclusions

This work has examined the effect on intelligibility (ESTOI), quality (PESQ) and mask accuracy (classification accuracy and HIT-FA rate) of using recurrent neural networks within ratio mask estimation for speech enhancement. It was found that all recurrent systems provide large gains in intelligibility over our previous feed-forward system (DNN) for audio-only and audio-visual modalities, with largest gains found at lower SNRs. Further gains were also found across all other objective measures for audio-only and audio-visual using our proposed bi-directional feed-forward hybrid network using layer-normalised gated recurrent units (LNBiGRU-DNN). However, for visual-only models, the baseline DNN architecture still outperformed the new recurrent architectures, suggesting the longer temporal modelling found within a recurrent network was unable to extract any additional visual temporal structure over the DNN.

Combining both audio and visual modalities into a single bimodal audio-visual system still provides best performance across all noise types and SNRs, confirming that combining audio and visual features provides a robust complimentary feature set. Previously, the performance of audio-visual was equivalent to visual-only at an SNR of $-10\,\mathrm{dB}$ using a feed-forward system. However with the gain found when using the LNBiGRU-DNN architecture audio-visual now outperforms the visual-only model. Both audio-only and audio-visual gained consistent benefits from the recurrent architecture, as the SNR increases both models converge, and at high SNRs ($5\,\mathrm{dB}$) still perform equally well as each other, as previously found within the DNN systems.

Chapter 5

Ratio masking using convolutional and recurrent neural networks

5.1 Introduction

Previous work in Chapter 4 compared using feed-forward and recurrent neural network architectures for temporal modelling for ratio mask estimation. It was found that using recurrent neural networks outperforms feed-forward neural networks for audio-only and audio-visual models, with a slight degradation for visual-only. In this chapter the focus is on improving the visual feature extraction stage using convolutional neural networks in place of traditional visual feature extraction (AAM) within the speech enhancement framework, prior to temporal modelling, for ratio mask estimation. Specifically comparisons are made between three methods of visual feature extraction, traditional AAM feature extraction, end-to-end trained convolutional neural networks (CNNs) and using pre-trained CNNs for bottleneck feature extraction. The end-to-end trained CNN trains both the CNN and temporal model (RNN) together as a single network. This allows dataset specific features to be learnt, and allows full backpropagation through the recurrent and convolutional layers. Extracting bottleneck features from pre-trained CNNs is similar to traditional visual feature extraction. Features can be extracted and stored prior to

temporal windowing, and can be used to train temporal models just as previously shown with AAM features. This can then be treated as a two network approach, the first network performs feature extraction, but does not necessarily need to be re-trained (pre-trained CNN), and the second network performs temporal modelling (RNN). This allows only the RNN to be trained per experiment, where the CNN can be pre-trained and stored, saving both training time and processing resources by the removal of CNN layers, compared against the end-to-end trained CNN.

The motivation for using convolutional neural networks (CNNs) is from their success in many image processing tasks, such as image classification (Krizhevsky et al. [2012]; Simonyan and Zisserman [2014]; Szegedy et al. [2015]; He et al. [2016a]), object detection (Krizhevsky et al. [2012]) and object localisation (Tompson et al. [2015]). Speech processing has recently utilised CNNs for extracting information of the speakers lips from a video source for many tasks, such as speech recognition through lip-reading (Noda et al. [2014]; Chung and Zisserman [2016]; Assael et al. [2016]), and voice activity detection (Le Cornu and Milner [2015]). Convolutional neural networks are designed to replace traditional feature extraction methods, instead of specifying what the feature is and the method required to extract them, the network can learn features itself from the data within training. Features are extracted by convolving kernels over the input, the kernels weights are updated and learnt through training. This therefore allows the network to determine what features are important in order to learn a suitable mapping to the target output. In some applications learnt features have been shown to extract similar properties to traditional feature extraction, resembling Gabor filters (Gabor [1946]) and edge detection (Krizhevsky et al. [2012]).

Previous work conducted thus far has shown the importance of including visual information within the speech enhancement framework, with large improvements found particularly at low SNRs when combined with audio over an audio-only system across all objective measures. Therefore the focus of this chapter is to improve visual feature extraction instead of acoustic feature extraction, although convolu-

tional neural networks are used more in image processing, they can also be applied to acoustic speech (Van Den Oord et al. [2016]), which is not explored in this work. This chapter considers visual-only and audio-visual models only. Figure 5.1 shows the training pipeline of the proposed audio-visual end-to-end trained CNN & RNN speech enhancement system. For systems using pre-trained CNNs for feature extraction, the CNN is removed from the `CNN & RNN` stage, between `Image Extraction` and `Temporal Windowing`, and is not re-trained as part of training the speech enhancement system. Images are extracted from video and input into either a pre-trained CNN extracting bottleneck features or an end-to-end trained convolutional neural network (CNN), combined with acoustic features extracted from noisy speech (for audio-visual), before input into the recurrent neural network (RNN) for temporal modelling to estimate the ratio mask. For testing purposes, estimated masks are applied to a cochleagram of the noisy speech which suppresses noise-dominated T-F units and the remaining signals are overlapped and summed to produce the enhanced signal, shown in Figure 5.2. The same pipeline is used for all speech enhancement configurations, except the audio stream is removed for visual-only models.

The remainder of this chapter is organised as follows. Section 5.2 provides an overview of the baseline AAM model and acoustic feature extraction method. Section 5.3 introduces image extraction techniques, convolutional neural networks and the final end-to-end trained CNN architecture. In Section 5.4, the process of using pre-trained CNNs for feature extraction is discussed. A review of potential pre-trained architectures is presented, before the final speech enhancement architecture is discussed. Performance evaluations are made in Section 5.5 which first compare the effectiveness of the image extraction techniques outlined in Section 5.3.1 for both visual-only and audio-visual models using an end-to-end trained CNN (Section 5.5.1). Section 5.5.2 compares the performance of traditional AAM features, to our proposed end-to-end trained and pre-trained CNN for visual feature extraction. Experiments are conducted across varying noise type and SNR conditions and used the best performing image extraction method from Section 5.5.1 as input into both end-to-end trained CNNs and pre-trained CNNs. A comparison of features

learnt from end-to-end trained CNNs and features learnt from pre-trained CNNs is provided in Section 5.5.3. Finally, this chapter is concluded in Section 5.6.

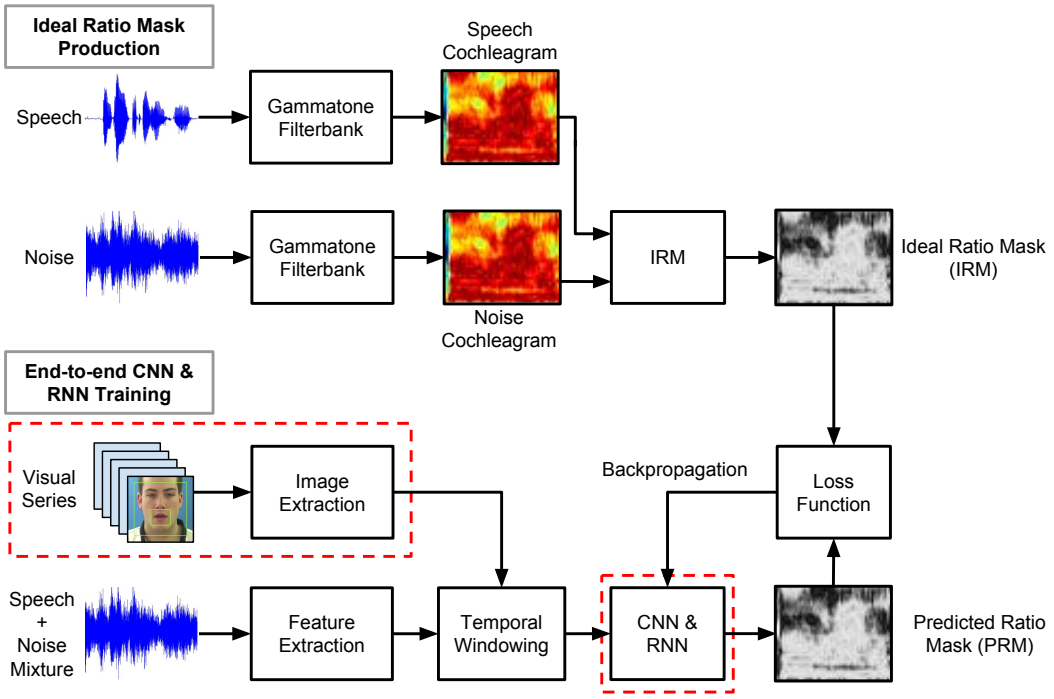


Figure 5.1: Overview of training an end-to-end trained CNN & RNN ratio masking speech enhancement system.

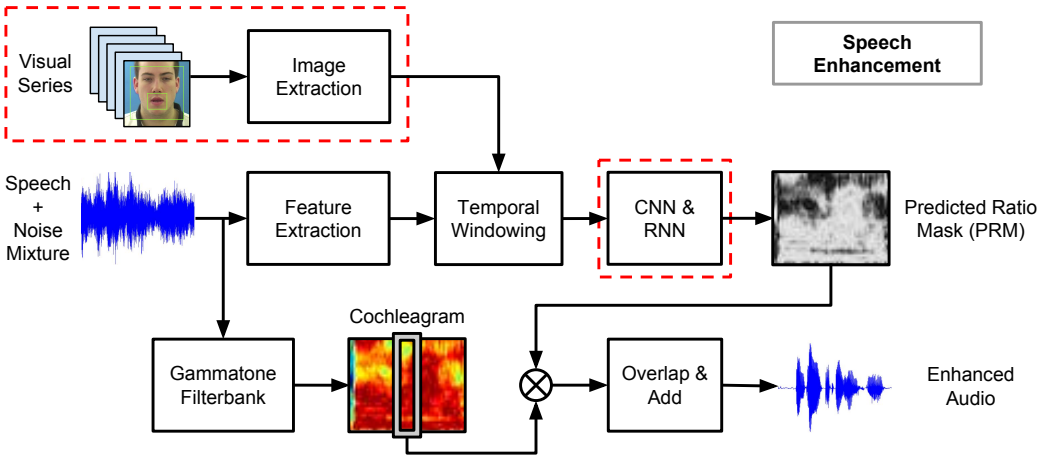


Figure 5.2: Overview of applying the end-to-end trained CNN & RNN predicted ratio mask to noisy speech for speech enhancement testing.

5.2 Baseline AAM based feature extraction model

Previous work in Chapter 4 explored using bi-directional recurrent neural networks as replacements for traditional feed-forward neural networks for temporal modelling, using standard feature extraction methods for input (MRCG for audio and AAM for visual). This found that using the proposed layer normalised bi-directional feed-forward hybrid network using gated recurrent units (LNBiGRU-DNN) provided best performance across all objective measures for audio-only and audio-visual input features and performed equally well as the best performing visual-only model. This forms our baseline model for this work and is the chosen architecture for the temporal model. The LNBiGRU-DNN architecture is shown in Figure 5.3 and comprises 2 pairs of forward and backward recurrent layers containing 256 gated recurrent units (GRU) per layer (512 per pair), 2 further dense layers containing 1024 rectified linear units (ReLU) and a final linear output layer. A skip connection is included combining the input and output from the recurrent layers. Detailed implementations of the LNBiGRU-DNN are in Section 4.3.2.

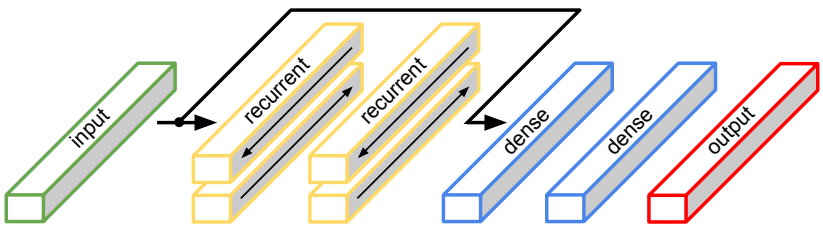


Figure 5.3: Layer normalised bi-directional recurrent feed-forward hybrid (LNBiGRU-DNN) speech enhancement architecture.

From our previous work in Chapters 2 and 3, the acoustic feature MRCG and visual feature AAM was found to perform best and was shown to also perform well in recurrent neural networks (Chapter 4), and as such are selected for the baseline of this work. The multi-resolution cochleagram (MRCG) feature combines 4 different cochleagrams, of both high and low resolution, into a single feature, and was specifically designed for mask estimation based within a cochleagram framework (Chen

et al. [2014]). The active appearance model (AAM) is a model-based combination of shape and appearance, producing a compact feature representation of a mesh fitted to the speaker lips. Details of the implementations of MRCG and AAM feature extraction methods are discussed in Sections 2.3.1.1 and 2.3.2 respectively. For visual-only experiments the input feature $\mathbf{x} = [\mathbf{x}^{\mathrm{AAM}}]$ while for audio-visual experiments the input feature $\mathbf{x} = [\mathbf{x}^{\mathrm{MRCG}}; \mathbf{x}^{\mathrm{AAM}}]$, where ; is a concatenation function.

5.3 End-to-end trained convolutional neural network based feature extraction

This chapter proposes replacing traditional visual feature extraction using AAM with convolutional neural networks (CNNs), which perform and learn feature extraction within the network, instead of pre-extracting features prior to the temporal network. Convolutional neural networks work well with data that has a clear grid structure and can scale to large sizes, and has been most successful for two-dimensional topology. As such CNNs have been applied successfully in many image processing tasks (Krizhevsky et al. [2012]; Tompson et al. [2015]; He et al. [2016a]) and speech processing tasks (Noda et al. [2014]; Le Cornu and Milner [2015]; Assael et al. [2016]). This section explores the process of training an end-to-end trained CNN where both the CNN and temporal model are trained as a single network. The pipeline of the end-to-end trained CNN is shown in Figure 5.4. The image extraction process is discussed in Section 5.3.1, and the CNN architecture is discussed in Section 5.3.2. The same acoustic feature extraction used within the AAM baseline architecture (MRCG features) is used again here.

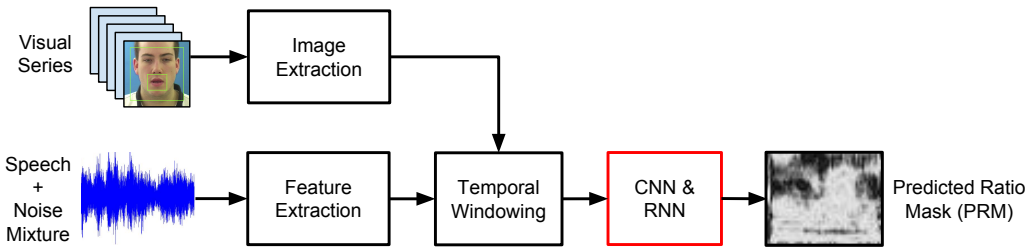


Figure 5.4: Overview of the end-to-end trained CNN & RNN pipeline.

5.3.1 Image extraction for CNN

Instead of producing visual features via AAM feature extraction, convolutional neural networks take images as input, and effectively perform feature extraction through convolving filter kernels within the model architecture (see Section 5.3.2). Mouth-only and full-face ROIs are extracted from raw video frames in RGB colour map. Extracted images are then upsampled across time to the same framerate of the acoustic features (MRCG, 100fps) before input to the models.

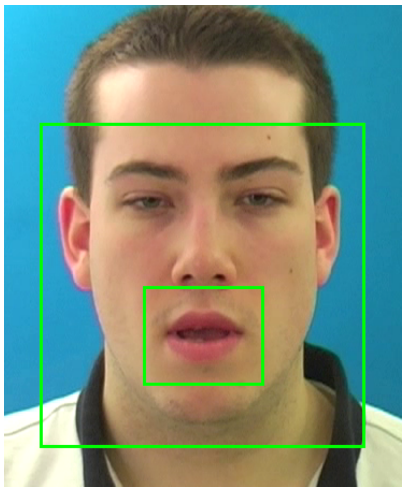
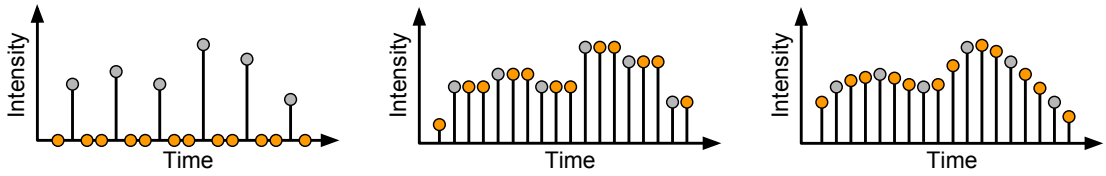


Figure 5.5: Fitted ROIs for mouth-only and full-face image extraction for CNN input.

Images are extracted from the raw video frames using the Viola-Jones (Viola and Jones [2001]) cascade-based object detector. Images are cropped to a fixed box size of 90×110 pixels centred around the mouth for the mouth-only ROI, or a fixed box size of 300×300 pixels for the full-face ROI (see Figure 5.5), both are then

downsampled to 64×64 pixels.

Due to the difference in frame rates between acoustic and video input, input images are upsampled to that of the acoustic features. Two different methods of upsampling are considered, namely interpolation and repetition, which are applied to each individual pixel and RGB channel. Just like with AAM features, upsampling through interpolation is also considered for raw images. However, due to the interpolation process, the upsampled images may introduce additional distortions within the image, producing blurring and inconsistent frames. To compare against this, upsampling through repetition is considered where the previous original raw image is repeated, ensuring no additional processing distortions. Figure 5.6 shows how the pixel intensities from the original (source) framerate are upsampled through repetition and interpolation across time.



(a) Original (b) Repetition (c) Interpolation

Figure 5.6: Techniques for upsampling image pixel values across time, with true source values (grey) and upsampled values (orange).

5.3.2 End-to-end CNN architecture

This section first introduces how traditional convolutional neural networks are implemented, before detailing the specific architecture implementation of the end-to-end trained CNN with temporal modelling. Convolutional neural networks convolve filter kernels over the input. Filter kernels are small in size in comparison to the input, and are used to locate features within a small region of the input. The filter kernels weights are learned through back-propagation during training. The number of kernels and kernel size is specified when initialising the network. The same kernels are used across the whole input as features learnt in one location are likely to be

useful in other locations of the input, i.e edge detectors, this also reduces overheads such as memory and training time, due to less parameters need to be stored and learnt. Generally, CNN layers are constructed of [convolutional, dropout, pooling] stages grouped together, with all three components named as a convolutional layer. A simple one-dimensional CNN is shown in Figure 5.7 showing the stages of a single convolutional layer.

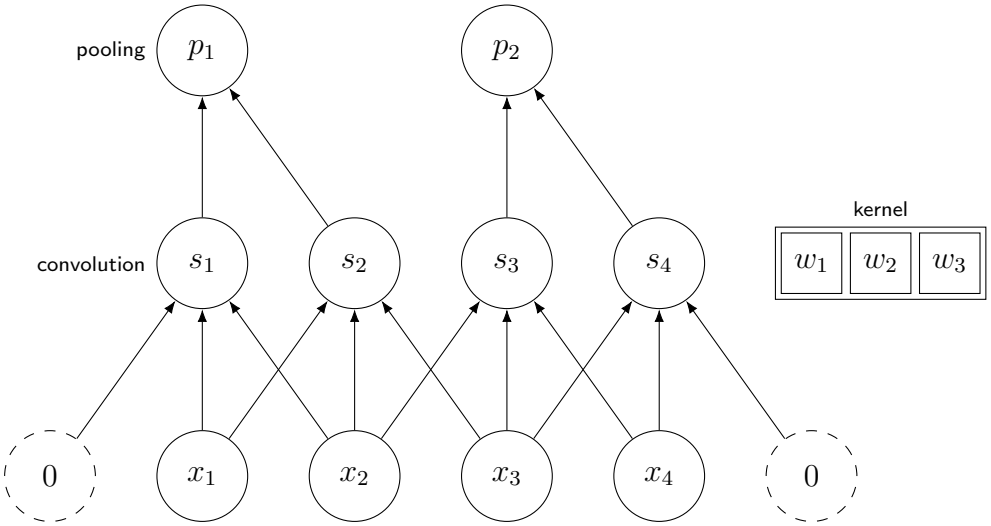


Figure 5.7: Computation of a typical 1-Dimensional convolutional neural network for a single kernel ($K = 3$).

Initially a single filter kernel K is defined with width 3, which is convolved over the input $\{x_1, x_2, ..., x_n\}$. The kernel traverses the input at each location with a kernel stride of 1, the stride length depends on the width of the kernel, and is generally defined as

$$\text{kernel stride} = \frac{\text{kernel width} - 1}{2} \tag{5.1}$$

When kernels are placed at the start and end of the input, the kernels are wider than the input, and as such the input is zero padded to accommodate the kernel (shown in dashed circles). Convolutions, $S(i)$, produced from convolving each kernel over the input sequence $\{x_1, x_2, ..., x_n\}$, are calculated as (without bias and with unit

stride)

$$S(i) = \sum_{m=1}^{k_w} x(i+m) \times K(m) \tag{5.2}$$

where i is an index in the input sequence for each available stride from the kernel and kernel width k_w. At this stage the non-linear activation function is applied to S, which in this work is rectified linear units (ReLU) (Maas et al. [2013]). Dropout is then applied to S (not shown in the figure) to help prevent overfitting within network training before passing to the pooling stage. Pooling, P, is applied to not only reduce feature dimensionality, but also to highlight important features found by the kernels. Pooling is performed similarly to convolving, except now the filter kernels' weights are fixed, and not adjusted through training. Pooling kernels traverse S with a stride generally set to the same size as the kernel width, usually a width and stride of 2 is used. Pooling either takes the form of average pooling or max pooling, average pooling was often used historically but has recently fallen out of favour compared to the max pooling operation (Zhou and Chellappa [1988]), which has been shown to work better in practice, and is defined as

$$P(i) = \mathbf{max}(S(i : i + k_w)) \tag{5.3}$$

which selects the maximum value within the pooling kernel size, at each index i. In this work convolutional neural networks are used to extract visual features from the raw video stream, and as such are applied to cropped images instead of one-dimensional signals. The one-dimensional convolution equation can be extended and applied for two-dimensional inputs (Goodfellow et al. [2016]). Convolutions are applied across the width and height of the input image, and are calculated as (without bias and with unit stride)

$$S(i,j) = \sum_{c=1}^{C} \sum_{m=1}^{k_w} \sum_{n=1}^{k_h} x(c, i+m, j+n) \times K(c,m,n) \tag{5.4}$$

where S is the convolution output, x is the input image with $C = 3$ channels for RGB colourspace, with a width, i, and height j, of 64 pixels each, K is the filter kernel with width k_w and height k_h and the same channel depth of C. Similarly the pooling stage is adjusted to contain filter kernels of size 2×2 and stride 2 in both directions to accommodate the extra dimension.

In this work the convolutional neural network selected is based on the LipNet (Assael et al. [2016]) architecture, which was designed for lip-reading within the GRID corpus. The architecture is constructed of three sets of [convolutional, channel-wise dropout, max-pooling] layers consisting of $[32, 64, 96]$ kernels of size $k_w \times k_h = [5 \times 5, 5 \times 5, 3 \times 3]$ for each layer pairing respectively, followed by a single 256 ReLU unit bottleneck layer for feature reduction, before passing to the temporal network. The temporal network selected is the LNBiGRU-DNN which was used within the baseline AAM architecture (details of the LNBiGRU-DNN can be found in Section 5.2). The full architecture is trained as a single model and is shown in Figure 5.8 (channel-wise dropout omitted) for an audio-visual model, for visual-only the acoustic stream is removed.

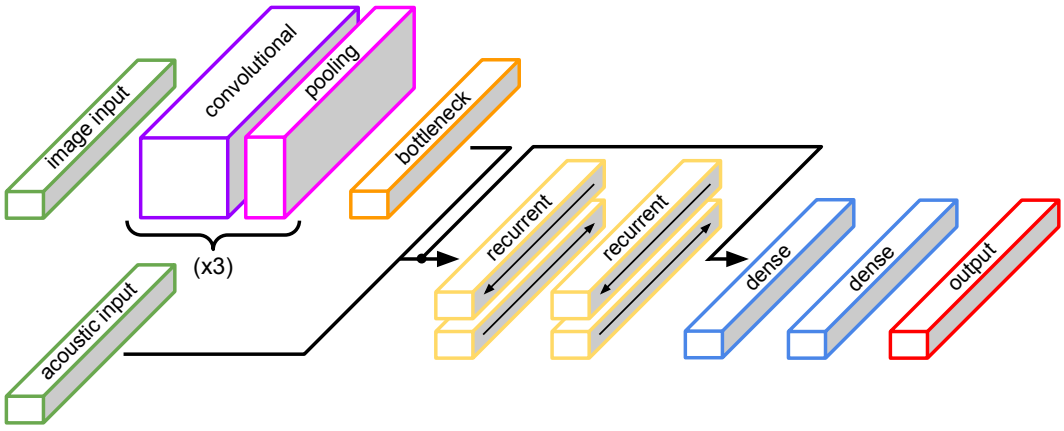


Figure 5.8: Audio-visual convolutional recurrent feed-forward hybrid speech enhancement architecture.

When training either visual-only or audio-visual CNNs the features learnt can vary between the models. For visual-only models the features learnt have to retrieve all relevant information from the input images, whereas audio-visual models can learn complimentary features to combine with the acoustic stream. From previous

experiments conducted, the visual stream is known to be important for audio-visual models at low SNRs, and less so at higher SNRs. Therefore features learnt from the CNN are likely to be more distinctive at lower SNRs, extracting similar information to visual-only, than those at higher SNRs for audio-visual models.

Channel-wise dropout has been shown to provide further performance improvements with respect to CNNs for object localisation (Tompson et al. [2015]) over standard dropout. The bottleneck layer is used simply to reduce the feature size prior to the recurrent network, a size of 256 was chosen to match the feature size of the acoustic MRCG feature. Without a bottleneck layer, the visual stream would dominate the acoustic stream in size, which may force the recurrent layers to favour visual information, and would cause the network to take longer to train due to the increase in number of parameters. The size of input, output and kernel for the production of bottleneck CNN features are shown in Table 5.1.

Table 5.1: CNN bottleneck feature production (channel-wise dropout omitted).

Layer	Kernel: Size	Stride	Padding	Input size
CNN	$C \times 5 \times 5$	$1, 2, 2$	$0, 2, 2$	$C \times 64 \times 64$
Pool	$1 \times 2 \times 2$	$1, 2, 2$		$32 \times 32 \times 32$
CNN	$32 \times 5 \times 5$	$1, 1, 1$	$0, 2, 2$	$32 \times 16 \times 16$
Pool	$1 \times 2 \times 2$	$1, 2, 2$		$64 \times 16 \times 16$
CNN	$64 \times 3 \times 3$	$1, 1, 1$	$0, 1, 1$	$64 \times 8 \times 8$
Pool	$1 \times 2 \times 2$	$1, 2, 2$		$96 \times 8 \times 8$
Bottleneck	256			$96 \times 4 \times 4$

5.3.3 Batch normalisation

Convolutional neural networks require longer processing time to train in comparison to feed-forward and recurrent networks, due to the larger input data size and convolving of filters. A recent approach to improve convergence in convolutional networks is to introduce batch normalisation (Ioffe and Szegedy [2015]), which adds normalisation steps within the network architecture, similar to layer normalisation used in recurrent neural networks (see Section 4.4.3). Normalisation standardises

each input using its mean and standard deviation, similar to z-score normalisation which is computed over the entire training set, is instead computed over batches. Batch normalisation works well when the input is a fixed size, which is required in convolutional networks. Batch normalisation is applied to the convolution layer before the non-linear activation function applied (ReLU) [convolution, batch normalisation, ReLU]. The normalisation function BN is defined as:

$$BN(\boldsymbol{\chi};\boldsymbol{\alpha},\boldsymbol{\beta}) = \frac{(\boldsymbol{\chi}-\boldsymbol{\mu})}{\boldsymbol{\sigma}} \odot \boldsymbol{\alpha} + \boldsymbol{\beta} \tag{5.5}$$

with

$$\mu_{ij} = \frac{1}{N}\sum_{n=1}^{N}\chi_{ij,n}, \quad \sigma_{ij} = \sqrt{\frac{1}{N}\sum_{n=1}^{N}(\chi_{ij,n}-\mu_{ij})^2} \tag{5.6}$$

where μ_{ij} and σ_{ij} are the mean and standard deviation at position i, j of the input $\boldsymbol{\chi}$ (for example 64×64 for the first convolution layer applied to input images), and N is the number of inputs within each mini-batch when training the network. Parameters $\boldsymbol{\alpha}$ (initialised as all 1s) and $\boldsymbol{\beta}$ (initialised as all 0s) are the gain and bias of the same size as $\boldsymbol{\chi}$, and are learnable parameters updated during network training. Batch normalisation, as the name suggests, is calculated over all elements across the batch dimension, unlike layer normalisation, which is calculated over the feature vector dimension. In this work batch normalisation is applied to all convolutional layers, for all models.

5.4 Pre-trained convolutional neural network based feature extraction

Convolutional neural networks are larger than recurrent or feed-forward neural networks, and can be very deep in order to produce peak performance, and as such require large amounts of processing time and hardware power to train. However,

once trained, features can be extracted and used for other tasks. This allows for the previous end-to-end trained CNN system to be split into two separate networks, with one network designated for visual feature extraction (pre-trained CNN) and a second designated for temporal modelling (RNN), a pipeline of this is shown in Figure 5.9.

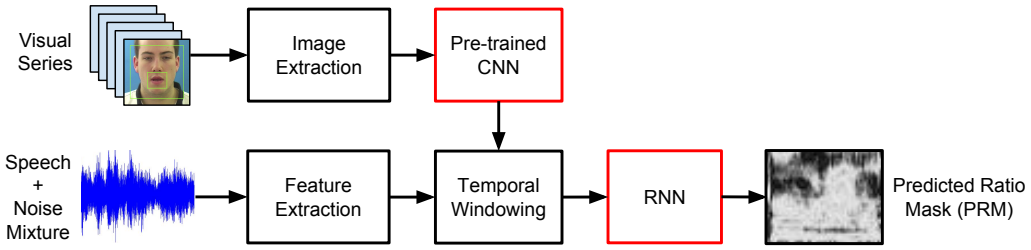


Figure 5.9: Overview of the pre-trained CNN & trained RNN pipeline.

This allows for a single large CNN to be trained *offline*, and allow features to be extracted for use in smaller temporal networks. Images can be passed through a pre-trained network, and visual bottleneck features can be extracted, similar to how traditional visual features are produced. Once features are stored, the second temporal network does not require the original image, only the extracted bottleneck feature, thus convolutional layers can be removed, which speeds up training of the temporal model. This leaves the question of "which pre-trained network should be selected?" A review of pre-trained networks is provided in Section 5.4.1 before the final architecture is discussed in Section 5.4.2.

5.4.1 Review of pre-trained CNNs

This section provides a review of pre-trained networks that can be used for visual bottleneck feature extraction, answering the question of "which pre-trained network should be selected?" One competition that creates large networks is the ImageNet Large Scale Visual Recognition Challenge (ILSVRC). This challenge has become one of the more popular for researchers to compete and advance convolutional networks. From this challenge, four popular networks have emerged with publicly available

pre-trained weights, AlexNet (Krizhevsky et al. [2012]), VGGNet (Simonyan and Zisserman [2014]), GoogLeNet (Szegedy et al. [2015]) and ResNet (He et al. [2016a]).

The AlexNet architecture (Krizhevsky et al. [2012]) was the first CNN system to drastically outperform non-CNN systems in the ILSVRC 2012 competition. The system reduced the top-5 error from 26 % to 15.3 %. The network built upon the LeNet (LeCun et al. [1998]) with more filters per layer, stacked convolutions and a deeper architecture, which was possible due to the improvement in available hardware compared to when LeNet was implemented. The architecture notably used wide filter sizes [11×11, 5×5, 3×3], used ReLU activations for both convolutional and dense layers and included dropout after dense layers to prevent overfitting. The overall size of the AlexNet architecture consisted of 60 million parameters.

The VGGNet (Simonyan and Zisserman [2014]) was runner-up at the ILSVRC 2014 competition scoring a top-5 error of 7.3 % and builds upon the AlexNet architecture. Instead of using large filter kernels, VGGNet replaces them with multiple convolutional layers with small kernels [3×3]. Given a receptive field stacking multiple smaller kernels is better than using a single large kernel because this allows more non-linear layers. This enables the network to learn more complex features at a cost of increasing the depth of the network. It is currently the most preferred choice among researchers for extracting features from images and has been used in many other applications and challenges as a baseline feature extractor. However, VGGNet consists of 138 million parameters, which causes limitations in available hardware.

While the VGGNet achieves very good performance on the ImageNet task, its deployment on easily available hardware is challenging due to its large size for both memory and processing time. Instead, Google developed a network that considers deployment onto other hardware and as such focused on reducing the overall number of parameters, called GoogLeNet (Szegedy et al. [2015]), whilst still providing strong performance, winning the ILSVRC 2014 competition scoring a top-5 error of 6.7 %. To reduce the number of parameters required, GoogLeNet builds upon the

idea that most activations in a deep network are either unnecessary (value of zero) or redundant because of correlations between them. Therefore the most efficient architecture of a deep network will have a sparse connection between activations, which implies that all output channels will not have a connection with all input channels, whereas all output channels in a standard convolutional operation is connected to all input channels. To achieve this, GoogLeNet used a CNN inspired by LeNet, but introduces a new module called the inception module.

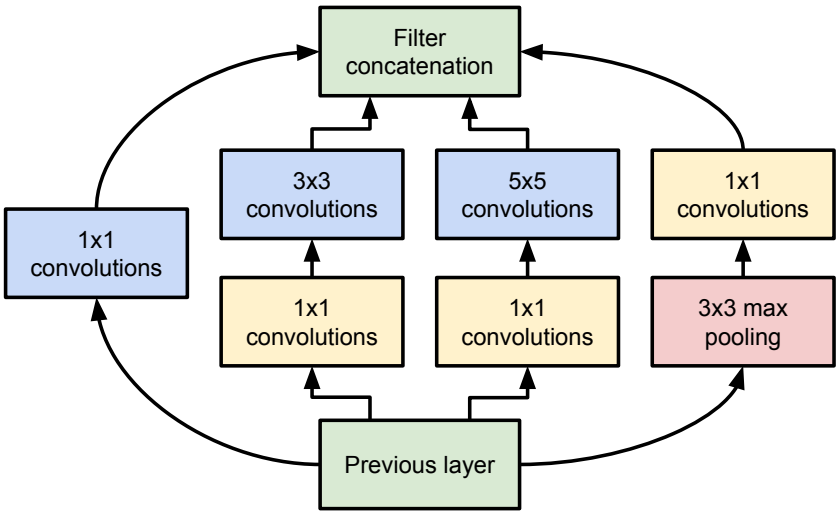
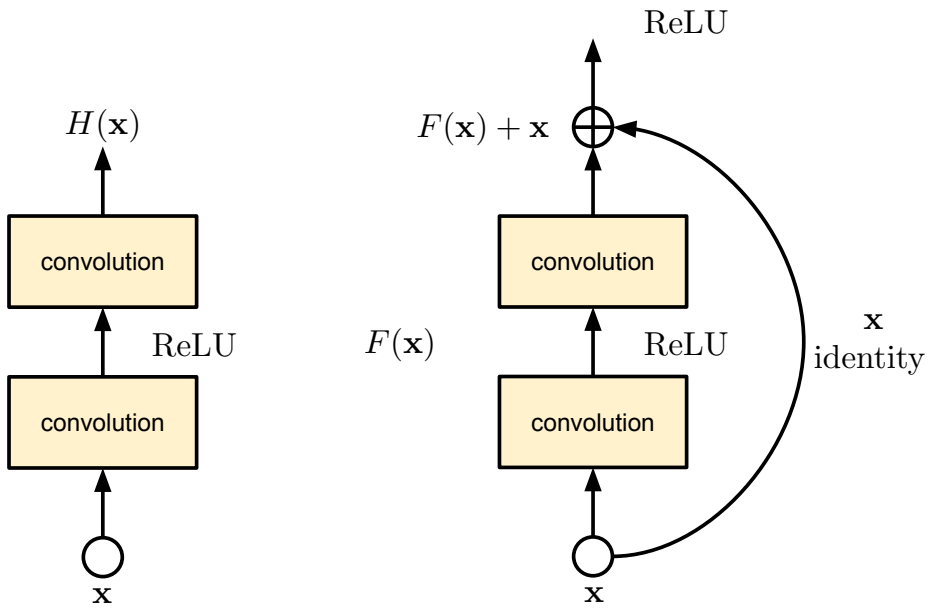


Figure 5.10: GoogLeNet inception module with dimensionality reduction.

The inception module approximates a sparse CNN using standard convolutional layers, shown in Figure 5.10. Since only a small number of activations are effective (i.e non-zero), the number of kernels and kernel size is kept small. The inception module utilises convolutions with varying kernel sizes to capture information at different scales $[5 \times 5,\ 3 \times 3,\ 1 \times 1]$. The module also uses $[1 \times 1]$ convolutions as bottlenecks (shown in yellow) to further reduce the number of parameters, with application before applying convolutions with large kernels to reduce the channel dimension of the input. A further change implemented within GoogLeNet was to replace the feed-forward layers at the end of the network with global average pooling which averages applied across the channel dimension. This drastically reduces the number of parameters as feed-forward layers generally use a large percentage of parameters within CNN architectures. This resulted in the overall size of the

GoogLeNet architecture to contain only 22 million parameters, giving a 6 times reduction when compared with VGGNet.

The newest and best performing pre-trained network is the Residual Neural Network or ResNet (He et al. [2016a]), winner of the ILSVRC 2015 competition scoring a top-5 error of 3.57 %, with a network containing 152 layers whilst having less parameters than the VGGNet. Problems with increasing the number of layers is that the signal required to change network weights during training is relatively small at the start of the network compared to the end of the network, causing earlier layers to be almost removed an not updated, known as vanishing gradients. Residual networks allow training of such deep networks by constructing the network through modules called residual blocks shown in Figure 5.11.



(a) Standard layers (b) Residual block

Figure 5.11: Difference between standard convolutional blocks, and residual blocks introduced by ResNet.

The residual block allows the network to learn the residual, $F(\mathbf{x}) = H(\mathbf{x}) - \mathbf{x}$, instead of learning $F(\mathbf{x}) = H(\mathbf{x})$ as with standard convolutional connections, forcing the network to update earlier layers, as only the residual is transmitted. Learning the residual helps to alleviate the vanishing gradient, allowing very deep networks to be learnt. ResNets continue to grow in popularity and size, with a recent

implementation containing 1001 layers was used on the CIFAR-10/100 (Krizhevsky and Hinton [2009]) achieving best performance for this task (He et al. [2016b]). The use of skip connections is similar to the RNN-DNN architecture proposed in Section 4.3.2.

Overall, taking into considerations the size of the network and shown performance given the limited hardware resource available, the GoogLeNet architecture is selected as the pre-trained network for visual features to be extracted from. The process of extracting bottleneck features from GoogLeNet and subsequent final architecture is discussed in Section 5.4.2.

5.4.2 Pre-trained CNN architecture

This section details how bottleneck features are extracted from the GoogLeNet architecture and how subsequently can be used to train temporal networks (as previously shown in Figure 5.9). Bottleneck features are simply an extraction of an internal state within a network, which is representative of the input passed into the network. States are extracted for every different input image, and can be stored for use in training other networks. For CNNs the internal state extracted is generally the state of the final (deepest) pooling layer. For the GoogLeNet architecture the final pooling layer is called *pool5*, and is used to extract bottleneck features. Figure 5.12 shows a basic implementation of the GoogLeNet architecture, with the prior convolutional, pooling and inception blocks shown simply as the GoogLeNet block (grey), only a single feed-forward layer and output layer trails the *pool5* (pink) layer.

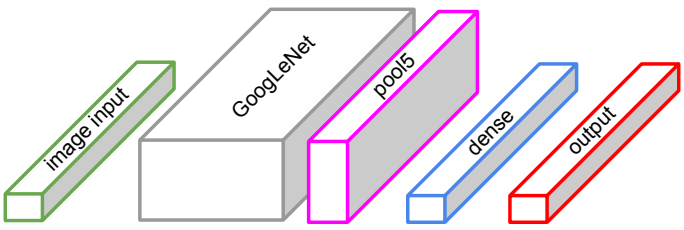


Figure 5.12: Simplified GoogLeNet architecture for feature extraction.

Input images are required to be of size 224×224 pixels and normalised by the provided mean and standard deviation, for convenience the previously extracted 64×64 images used in the end-to-end trained system is simply upsampled to the correct size (image extraction details can be found in Section 5.3.1). Extracted bottleneck features are of size 1024 and are stored before training the temporal model (RNN) specifically for speech enhancement.

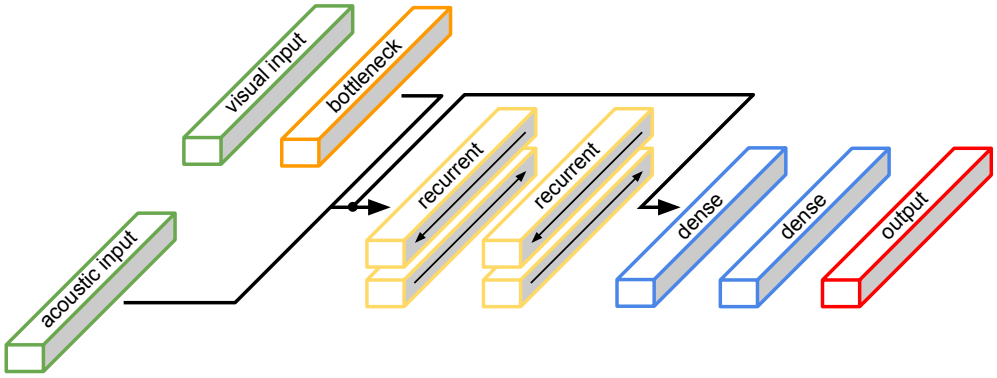


Figure 5.13: Audio-visual recurrent feed-forward hybrid speech enhancement architecture with bottleneck layer for visual feature reduction.

The temporal model is shown in Figure 5.13 and follows a similar implementation of the LNBiGRU-DNN as used within the baseline AAM and end-to-end trained architectures (details of the LNBiGRU-DNN can be found in Section 5.2). However, in this implementation the visual and acoustic streams have separate inputs, allowing the visual stream to pass through a bottleneck layer prior to the recurrent layers. Similar to the end-to-end system, the bottleneck layer is used to reduce feature dimensionality, and again a 256 layer with ReLU units is selected. This then allows a fairer comparison between the end-to-end trained system and this pre-trained architecture, and does not overpower the acoustic stream. The same temporal network architecture is selected as the end-to-end trained system and is equivalent to the baseline architecture, a layer normalised bi-directional recurrent feed-forward hybrid network using gated recurrent units (LNBiGRU-DNN) which was found to be optimal in Chapter 4.

5.5 Experimental results

The performance of using traditional AAM, end-to-end trained convolutional neural networks and pre-trained convolutional neural networks for visual feature extraction is compared. Firstly, the image extraction methods outlined in Section 5.3.1 are optimised within an end-to-end trained CNN for both visual-only and audio-visual models. The best performing image extraction method is then used to compare the performance between AAM, end-to-end trained CNNs and pre-trained CNNs in varying noise type and SNR conditions.

The first experiment compares image extraction techniques, combining using mouth-only or full-face ROIs with repetition and interpolating upsampling methods. This experiment is conducted using the end-to-end trained CNN architecture in babble noise at $-5\,\text{dB}$, for both visual-only and audio-visual models. The best performing image extraction technique (through objective measures) is selected for further analysis.

The second experiment is to compare the performance of traditional AAM features, to our proposed end-to-end trained and pre-trained CNN for visual feature extraction. Experiments are conducted across varying noise type and SNR conditions, specifically babble and factory noise at SNRs of $-10\,\text{dB}$, $-5\,\text{dB}$, $0\,\text{dB}$ and $5\,\text{dB}$ is used, again in visual-only and audio-visual models.

The CNN and RNN models were implemented within the Lasagne framework (Dieleman et al. [2015]) with the Theano (Theano Development Team [2016]) backend. Input data was z-score normalised and grouped into mini-batches of 256. To prevent overfitting, dropout of 0.5 was applied between all convolutional layers, and dropout of 0.2 applied to all other layers and early stopping (Prechelt [1998]) was used when the validation score did not improve after 5 further epochs. Training used backpropagation with the Adam optimiser (Kingma and Ba [2014]) and a learning rate 0.001 for CNN and RNN, minimising the MSE loss function. All experiments use a single speaker (speaker 12) from the GRID dataset (details provided in Section

A.1), containing 1000 utterances which are allocated into 640, 160 and 200 for the training, validation and test sets respectively.

5.5.1 Image extraction using end-to-end trained CNNs

An initial comparison is made between the alternative ROIs and upsampling methods for extracting images used as input to the CNNs, outlined in Section 5.3.1. Images are extracted with mouth-only or full-face ROIs, and upsampled via repetition or interpolation before input to the end-to-end CNN in both visual-only and audio-visual models.

All experiments are performed in babble noise at $-5\,\text{dB}$, using the end-to-end trained CNN architecture outlined in Section 5.3.2. A window width of 31 with $K = 15$ (320 ms) is chosen, which was found to be optimal in Chapter 4. Table 5.2 shows the classification accuracy, HIT-FA rate, PESQ and ESTOI scores for all models and image extraction methods across the validation set.

Table 5.2: Classification accuracy (in %), HIT-FA (in %), PESQ and ESTOI scores for the GRID dataset in babble noise at $-5\,\text{dB}$ with varying CNN input features.

Feature			Acc	HIT-FA (FA)	PESQ	ESTOI
V	Mouth	rep	87.8	67.3 (8.3)	2.22	52.3
		int	**88.1**	67.1 (7.9)	**2.24**	**52.9**
	Face	rep	87.9	67.2 (8.1)	**2.24**	52.3
		int	87.6	**67.4** (8.7)	2.23	50.6
AV	Mouth	rep	**91.7**	**76.8** (5.3)	2.43	61.6
		int	**91.7**	76.1 (4.9)	**2.46**	**62.1**
	Face	rep	91.4	75.3 (5.3)	2.41	59.8
		int	91.6	75.7 (5.1)	2.44	61.2
unprocessed audio					1.82	22.0

Focusing first on visual-only, all models perform similarly across classification accuracy, HIT-FA rate and PESQ scores with the mouth-only with interpolation upsampling performing slightly better across all measures. For intelligibility through ESTOI, more variation is seen across the models, with mouth-only with interpolation

upsampling providing gains of 0.6 compared to the second best methods. Across all measures the mouth-only with interpolation upsampling generally provides best performance.

Looking now at the audio-visual models, similar trends are found to visual-only, but the difference between each model is larger. Classification accuracy and PESQ scores are similar between all models, with mouth-only with repetition upsampling providing highest HIT-FA rate, and mouth-only with interpolation providing best intelligibility scores with a gain of 0.5 in terms of ESTOI over the second best performing method. As with visual-only, mouth-only with interpolation upsampling generally provides best performance across all measures.

Overall, comparing both visual-only and audio-visual models, the mouth-only ROI outperforms the full-face ROI across all measures, and upsampling through interpolation outperforms upsampling with repetition. Upsampling through interpolation was likely to outperform repetition due to providing a smoother transition between time-steps, which helps the recurrent layers within the temporal model. The only downside could have been the potential to introduce image distortions through processing errors such as blurring, however this does not seem to have affected network performance.

The mouth-only outperforming the full-face is interesting due to the full-face containing all the information of the mouth-only plus information about the jaw line and cheeks, which should provide more information about speech production and should therefore provide more information to the network. However, this degradation in performance is attributed to over-downsizing the full-face ROI, such that important information, particularly the mouth, is now too small to capture important features. Both ROIs are downsized to 64×64 pixels, this causes the mouth information to be relatively small in the full-face condition. The mouth is likely to be the most important aspect to capture as the visual feature. If the full-face ROI was downsized such that the mouth was of similar size to that found in the mouth-only condition, it is expected the full-face ROI would outperform the mouth-only, due

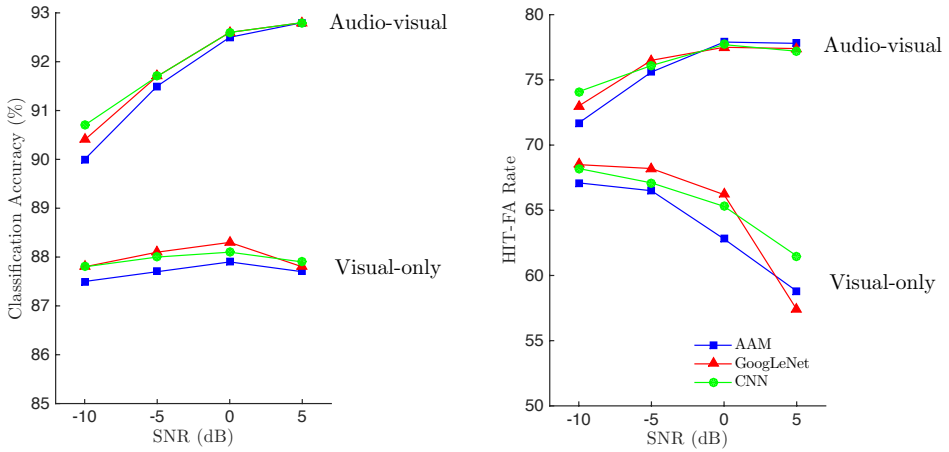
to the additional facial information captured. However, two downsides to increasing the size of the input image to the network are larger memory required to store larger images (both HDD and RAM) and a larger CNN would be required, in terms of layers to downsize the increased image within the network before the temporal model, which therefore increases the number of trainable parameters. Increasing the number of trainable parameters would cause both an increase in memory usage and the time required to train the network. This introduces a trade-off between available resource in terms of hardware and processing time, and with network performance. With the experiments conducted and the limitations outlined, the mouth-only ROI with interpolation upsampling is chosen for further analysis in varying noise types and SNR.

5.5.2 Comparison of visual feature extraction methods across noise type and SNR – AAMs, end-to-end trained CNNs and pre-trained CNNs

In Section 5.5.1 different input image ROIs and upsampling techniques were tested using an end-to-end trained CNN in babble noise at -5 dB only, which found mouth-only ROIs upsampled via interpolation to perform best. In this experiment a comparison between AAM features, end-to-end trained CNNs and pre-trained CNNs for visual feature extraction is tested in varying noise type and SNR conditions. The noise types and SNR conditions tested are babble and factory noise at SNRs of -10 dB, -5 dB, 0 dB and 5 dB. Both the end-to-end trained CNN (shown in Section 5.3.2) and pre-trained CNN (shown in Section 5.4.2) use mouth-only images upsampled via interpolation as input.

Tables 5.3 and 5.4 show the full set of objective measures for the test set across all noise type and SNR conditions tested, for visual-only and audio-visual models. Objective measures selected are classification accuracy, HIT-FA rate, PESQ and ESTOI. Figures 5.14 and 5.15 provide detailed breakdowns from Tables 5.3 and 5.4

for babble noise at −10 dB, −5 dB, 0 dB and 5 dB.

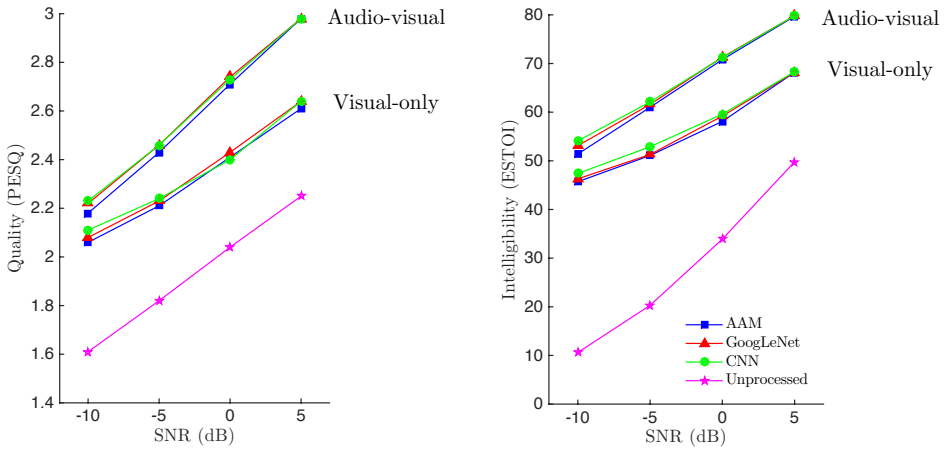


(a) Classification Accuracy (b) HIT-FA

Figure 5.14: Effect on mask classification accuracy and HIT-FA rate across SNR for visual-only and audio-visual in babble noise for ratio mask estimation using convolutional neural networks.

Focusing first on classification accuracy and HIT-FA rate results show that extracting features using both end-to-end (CNN) and pre-trained (GoogLeNet) CNNs outperform the baseline AAM feature in most conditions, for both visual-only and audio-visual models, shown in Figure 5.14 for babble noise at SNRs from −10 dB to 5 dB. The CNN and GoogLeNet systems perform similarly across all conditions, with CNN performing slightly better in audio-visual models, and GoogLeNet performing slightly better in visual-only models. Largest gains are found at low SNRs, with little gain found at high SNRs. When comparing the CNN system against AAM, an average improvement across both babble and factory noise at −10 dB for classification accuracy of 0.35 and 0.6, and for HIT-FA rate of 1.4 and 1.95 can be found for visual-only and audio-visual models respectively.

Looking now at quality scores through PESQ and intelligibility with ESTOI similar trends as with classification accuracy and HIT-FA rate are found, where extracting features using both end-to-end (CNN) and pre-trained (GoogLeNet) CNNs outperform the baseline AAM feature in most conditions, for both visual-only and audio-visual models, shown in Figure 5.15 for babble noise at SNRs from −10 dB to 5 dB. The performance between all systems is close, with the end-to-end CNN



(a) PESQ (b) ESTOI

Figure 5.15: Effect on quality through PESQ and intelligibility through ESTOI across SNR for visual-only and audio-visual in babble noise for ratio mask estimation using convolutional neural networks.

system providing best performance, particularly at low SNRs and in ESTOI.

When comparing the CNN system against AAM, an average improvement across both babble and factory noise at -10 dB for PESQ of 0.03 and 0.05, and for ESTOI of 2.0 and 2.6 can be found for visual-only and audio-visual models respectively. The performance gained in PESQ is marginal, yet the gain found for ESTOI is relevant, considering how well AAM features perform particularly in this favourable speaker dependent task.

Comparing the performance of the end-to-end CNN to the baseline AAM system across all modalities and SNR conditions, consistent gains are found across all measures for both visual-only and audio-visual models. With our focus being improving intelligibility, large gains are found across all SNR conditions, with larger gains found at the more challenging lower SNR. This improvement at lower SNRs reveals a key benefit from using convolutional neural networks for feature extraction. When comparing the end-to-end CNN with the pre-trained GoogLeNet feature, the difference becomes smaller than found with AAM, however, training a bespoke end-to-end CNN for the specific task does provide best performance across most measures and conditions, although marginal.

Table 5.3: (VISUAL-ONLY) Classification accuracy (in %), HIT-FA (in %) PESQ and ESTOI scores for the GRID dataset for visual-only mask estimation with different CNN architectures in babble and factory noise at −10 dB, −5 dB, 0 dB and 5 dB.

Noise (dB)		Model	Acc	HIT-FA (FA)	PESQ	ESTOI
babble	-10	AAM	87.5	67.1 (8.8)	2.06	45.7
		GoogLeNet	**87.8**	**68.5** (8.8)	2.08	46.3
		CNN	**87.8**	68.2 (8.6)	**2.11**	**47.4**
		unprocessed audio			1.61	10.6
	-5	AAM	87.7	66.5 (8.2)	2.21	51.1
		GoogLeNet	**88.1**	**68.2** (8.3)	2.23	51.3
		CNN	**88.1**	67.1 (7.9)	**2.24**	**52.9**
		unprocessed audio			1.82	20.3
	0	AAM	87.9	62.8 (6.1)	2.41	58.1
		GoogLeNet	**88.3**	**66.2** (6.9)	**2.43**	59.2
		CNN	88.1	65.3 (6.9)	2.40	**59.6**
		unprocessed audio			2.04	33.9
	+5	AAM	87.7	58.8 (4.6)	2.61	68.2
		GoogLeNet	87.8	57.4 (3.9)	**2.64**	**68.3**
		CNN	**87.9**	**61.5** (5.6)	**2.64**	68.2
		unprocessed audio			2.25	49.8
factory	-10	AAM	90.6	68.0 (5.8)	2.14	46.5
		GoogLeNet	90.9	**69.8** (5.8)	**2.17**	48.1
		CNN	**91.0**	69.7 (5.6)	2.14	**48.8**
		unprocessed audio			1.46	10.5
	-5	AAM	90.7	68.1 (5.6)	2.28	52.4
		GoogLeNet	**91.0**	**68.4** (5.2)	**2.30**	53.2
		CNN	**91.0**	68.0 (5.2)	2.29	**53.9**
		unprocessed audio			1.66	20.1
	0	AAM	90.9	66.4 (4.9)	2.44	60.1
		GoogLeNet	**91.1**	66.3 (4.6)	**2.45**	60.2
		CNN	**91.1**	**67.2** (4.7)	2.43	**60.8**
		unprocessed audio			1.87	33.5
	+5	AAM	90.7	56.8 (2.8)	2.61	68.0
		GoogLeNet	91.0	60.1 (3.2)	**2.64**	68.7
		CNN	**91.3**	**66.1** (4.0)	2.63	**70.1**
		unprocessed audio			2.09	49.9

Table 5.4: (AUDIO-VISUAL) Classification accuracy (in %), HIT-FA (in %) PESQ and ESTOI scores for the GRID dataset for audio-visual mask estimation with different CNN architectures in babble and factory noise at −10 dB, −5 dB, 0 dB and 5 dB.

Noise (dB)		Model	Acc	HIT-FA (FA)	PESQ	ESTOI
babble	-10	AAM	90.0	71.7 (6.3)	2.18	51.4
		GoogLeNet	90.4	73.0 (6.0)	2.22	53.1
		CNN	**90.7**	**74.1** (6.0)	**2.23**	**54.1**
		unprocessed audio			1.61	10.6
	-5	AAM	91.5	75.6 (5.2)	2.43	61.0
		GoogLeNet	**91.7**	**76.5** (5.3)	**2.46**	61.6
		CNN	**91.7**	**76.5** (4.9)	**2.46**	**62.1**
		unprocessed audio			1.82	20.3
	0	AAM	92.5	**77.9** (4.4)	2.71	70.8
		GoogLeNet	**92.6**	77.5 (4.0)	**2.74**	**71.3**
		CNN	**92.6**	77.7 (4.1)	2.73	71.2
		unprocessed audio			2.04	33.9
	+5	AAM	**92.8**	**77.8** (3.7)	**2.98**	79.7
		GoogLeNet	**92.8**	77.4 (3.4)	**2.98**	**79.8**
		CNN	**92.8**	77.2 (3.4)	**2.98**	**79.8**
		unprocessed audio			2.25	49.8
factory	-10	AAM	92.8	73.7 (3.9)	2.25	52.6
		GoogLeNet	93.1	74.6 (3.7)	**2.29**	53.8
		CNN	**93.3**	**75.2** (3.6)	**2.29**	**55.0**
		unprocessed audio			1.46	10.5
	-5	AAM	94.0	77.1 (3.1)	2.50	61.7
		GoogLeNet	**94.1**	77.5 (3.0)	**2.54**	**63.0**
		CNN	**94.1**	**77.8** (3.0)	2.52	62.8
		unprocessed audio			1.66	20.1
	0	AAM	94.7	**79.7** (2.7)	**2.76**	71.4
		GoogLeNet	**94.8**	79.2 (2.5)	**2.76**	71.5
		CNN	**94.8**	**79.7** (2.5)	**2.76**	**72.0**
		unprocessed audio			1.87	33.5
	+5	AAM	95.1	79.8 (2.1)	**3.00**	79.9
		GoogLeNet	**95.2**	79.7 (2.1)	2.99	**80.1**
		CNN	95.1	**79.9** (2.2)	2.98	**80.1**
		unprocessed audio			2.09	49.9

5.5.3 Comparison of features learnt between end-to-end trained and pre-trained CNNs

In this section a comparison of features learnt from an end-to-end trained CNN is compared with features learnt within the pre-trained GoogLeNet architecture. The end-to-end trained CNN was specifically trained for this task (lip-reading) and for the GRID dataset, whereas the GoogLeNet architecture was specifically trained for image classification on a different dataset. To extract the features learnt, the internal state of the first convolutional layer is extracted, which learns features directly from the input image, similar to how bottleneck features were extracted in Section 5.4.2. For comparison, a visual-only end-to-end trained CNN in babble noise at $-5\,\mathrm{dB}$ was chosen to compare against the GoogLeNet architecture. A visual-only model was chosen over an audio-visual model due to fairer comparison with the GoogLeNet system, which was also trained purely on visual information. Also, the features learnt within an audio-visual model can vary from visual-only due to learning complimentary features between visual and acoustic information, which may produce different features compared to visual-only models.

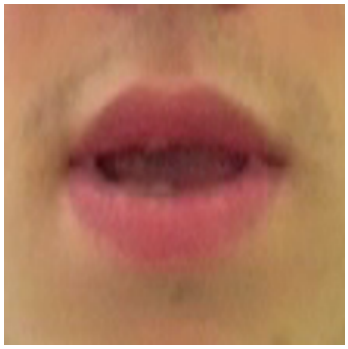


Figure 5.16: Example input image of a mouth-only ROI upsampled via interpolation from the GRID dataset.

Figure 5.16 shows an example input image used for training the end-to-end CNN and for extracting bottleneck features from GoogLeNet. This is applied to both architectures to extract the internal state of the first convolutional layer representing the features learnt within each architecture. Figures 5.17 and 5.18 show the activations of each filter kernel learnt within the first convolutional layer of the end-to-end

CNN and GoogLeNet architectures respectively.

Focusing first on Figure 5.17, which shows features learnt within an end-to-end trained CNN, strong activations are found across all kernels. The majority of activations are focused on the internal mouth across all kernels, specifically where teeth, tongue and inner lip contours are. This area is where most of the visual articulation occurs, and shows that the CNN was able to learn this itself from the training data. Other areas where activations are prominent are on the lips themselves, again an important visual cue. All kernels also provide some activation to the skin area surrounding the lips. Surprisingly, many of the kernels show similar activations to other kernels, extracting the same information. This could be due to the dropout used within training, where this information was found to be critical and as such multiple copies are extracted to ensure the information is not lost via dropout.

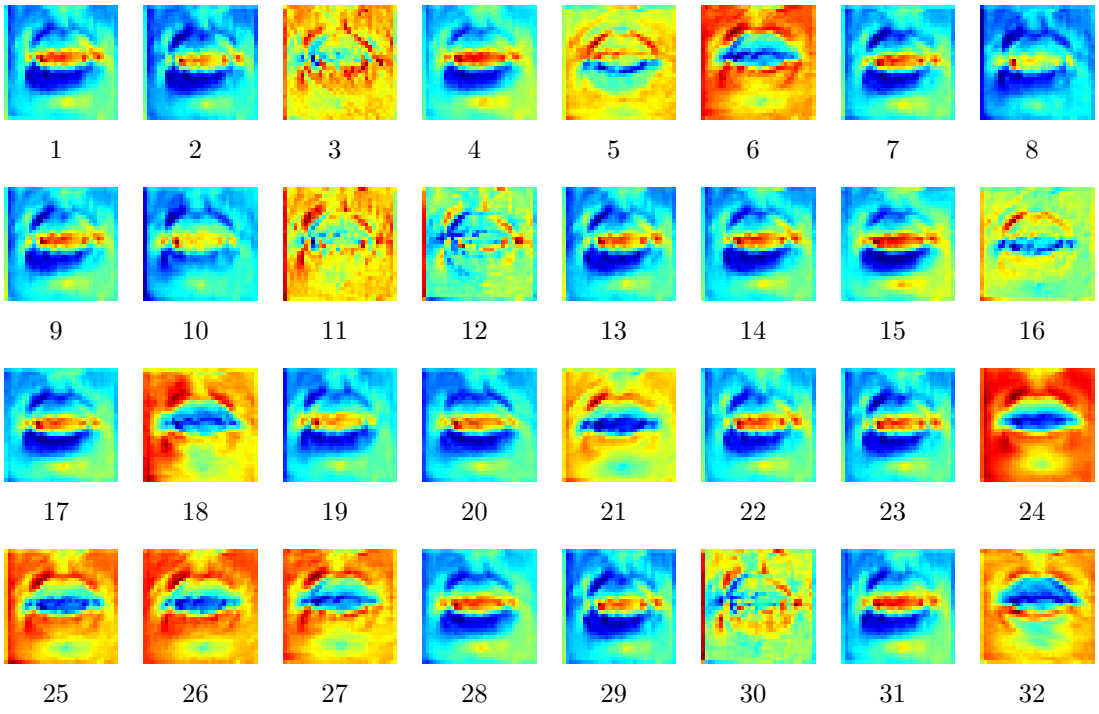


Figure 5.17: Activations of the filter kernels learnt from the first convolutional layer of the end-to-end trained CNN.

Looking now at Figure 5.18, which shows features learnt within the GoogLeNet architecture, strong activations are found for some kernels, yet many show little to no activations. This is due from the GoogLeNet model being trained for image

classification on another dataset, and not on the GRID dataset as with the end-to-end system. Many of the kernels have learnt features which are not present within an image of a mouth, and therefore show no activation when provided with mouth images. Those kernels which do show activations are again focused on the inner mouth, lip area or skin area around the lips, which is known to provide the majority of visual cues.

Overall, it is clear that training CNNs specifically on the dataset used within the application enables the CNN to learn more appropriate features. This is why the end-to-end trained CNN outperformed the pre-trained CNN using features extracted from GoogLeNet within experiments in Section 5.5.2. However, features learnt within the GoogLeNet model show similarities to the dataset specific end-to-end model. This enabled models using the GoogLeNet bottleneck feature to still perform well, and were able to beat traditional AAM feature extraction, which are also dataset specific. Therefore, using pre-trained CNNs can be beneficial for applications if training a dataset specific CNN is not possible or is impractical in terms of training time or available processing hardware.

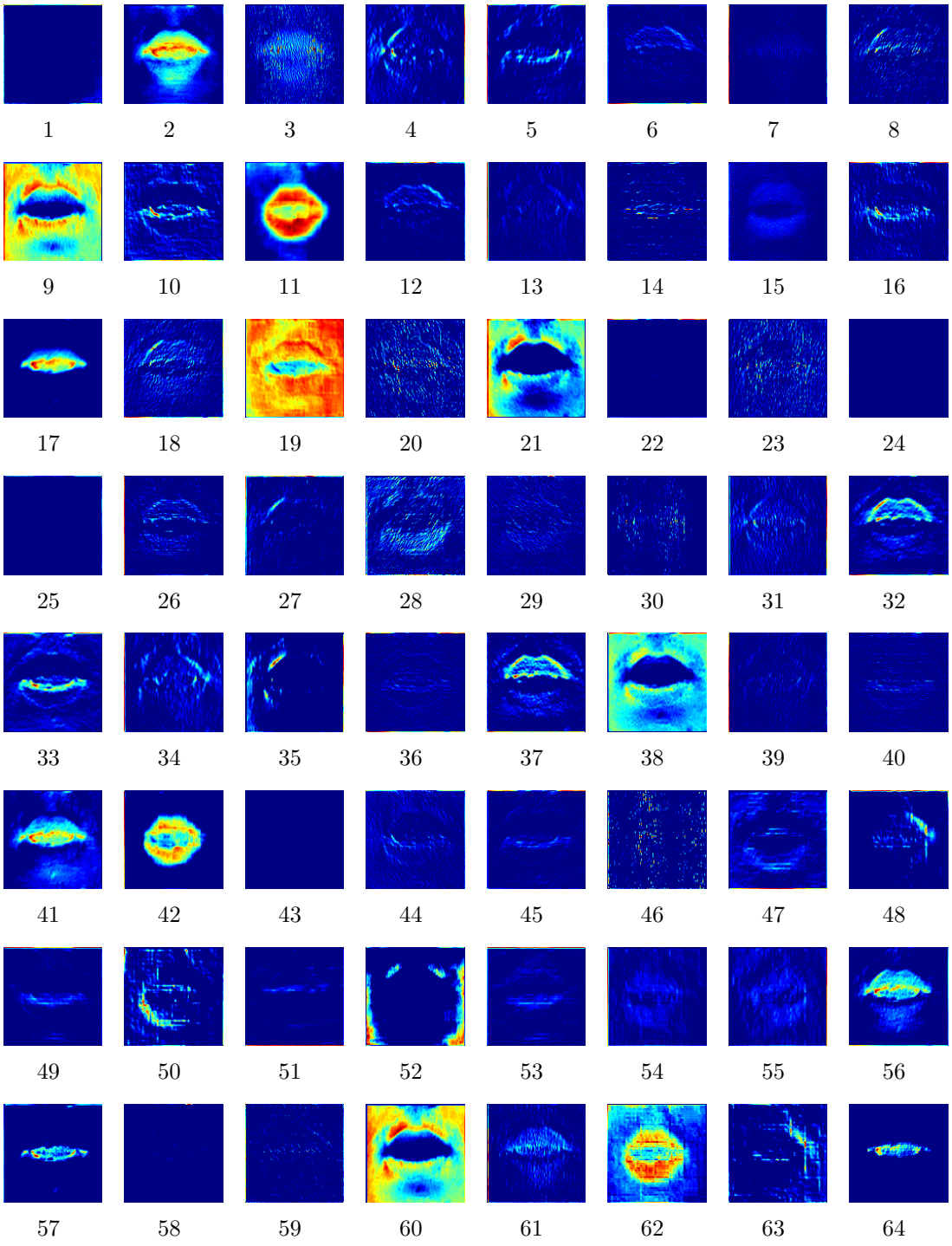


Figure 5.18: Activations of the filter kernels learnt from the first convolutional layer of the pre-trained GoogLeNet CNN.

5.6 Conclusions

This work has examined the effect on intelligibility (ESTOI), quality (PESQ) and mask accuracy (classification accuracy and HIT-FA rate) of using convolutional neural networks for feature extraction within ratio mask estimation for speech enhancement. It was found that extracting ROIs focusing on the mouth-only outperformed extracting full-face ROIs across both visual-only and audio-visual models. Upsampling through interpolation also outperformed upsampling with repetition, the variation produced for each time-step for interpolation over repetition likely aids the temporal recurrent network when traversing through time.

Using mouth-only ROIs with interpolation upsampling in both pre-trained CNNs and end-to-end CNNs provide large gains in intelligibility over the traditional AAM feature extraction method, in both visual-only and audio-visual models. Further gains were found across the other measures, with largest gains found for intelligibility. Peak performance was found when training an end-to-end CNN over using pre-trained networks, training a model specifically for the task (lip-reading) should outweigh adapting a previously trained model targeted for a different task (image classification), although the time-saved from training and ease of testing network variations could outweigh the drop in performance compared to the end-to-end system. Features learnt from the pre-trained GoogLeNet architecture showed similarities to the dataset specific end-to-end trained CNN, which were sufficient to enable the pre-trained models to outperform models using traditional AAM features.

Combining both audio and visual modalities into a single bimodal audio-visual system still provides best performance across all noise types and SNRs, confirming that combining audio and visual features provides a robust complimentary feature set. Extracting features from CNNs not only improves performance over AAM, but also removes many of the time-expensive processes required to extract AAM features, such as hand labelling detailed landmarks, instead only a cropped box around a tracked ROI is required to be extracted.

Chapter 6

Evaluation of model generalisation to unseen noise conditions and dataset size

6.1 Introduction

Previous work has explored and maximised the performance of audio-only, visual-only and audio-visual speech enhancement systems in matched noise type and SNR dependent conditions, with constraints on noise type and SNR conditions and dataset size being applied to aid in the development of the speech enhancement systems. However such constraints are impractical for application in real-world environments where noise conditions will change over time. This chapter explores removing these constraints by first developing systems for unconstrained noise type and SNR conditions, before applying the best performing methods to a larger unconstrained vocabulary dataset.

All evaluations so far have been conducted in matched noise and SNR conditions, where individual models are trained for each noise condition. This approach works well if all noise conditions for the application deployment environment are known

in advance, allowing noise type and SNR dependent models to be implemented per known condition. The speech enhancement system can select the noise dependent model trained for the current environment. If the environment changes, the system can simply select the new environment-dependent model. In unconstrained real-world environments, where noise type and SNR conditions can vary greatly, having noise dependent models becomes impractical and infeasible. Instead, having a noise independent system, where a single speech enhancement system is trained which can generalise to both known and unknown conditions would provide a more practical solution for real-world environments. This is achieved by training a single model with multiple and varying noise and SNR conditions, allowing the model to learn from more noise sources, in an attempt to provide generalisation.

This work considers applying our best performing audio-only, visual-only and audio-visual architectures for noise independent training. Figure 6.1 shows the training pipeline of the audio-visual speech enhancement system, of which the implementation is based from combining the architectures used in Sections 5.3.2 and 5.4.2. Images are extracted from video and input into an end-to-end trained convolutional neural network (CNN), bottleneck features extracted from the CNN are then combined with acoustic features extracted from noisy speech, before input into the recurrent neural network (RNN) for temporal modelling to estimate the ratio mask. Noise conditions used in training are no longer condition dependent, and instead cover a wide range of varying noise conditions. For testing purposes, estimated masks are applied to a cochleagram of the noisy speech which suppresses noise-dominated T-F units and the remaining signals are overlapped and summed to produce the enhanced signal, shown in Figure 6.2. The same pipeline is used for all speech enhancement configurations, except the visual stream is removed for audio-only and the audio stream is removed for visual-only. To confirm the models are able to learn generalisation, and not operate only in noise conditions used in training, both noise conditions used in training (seen conditions) and noise conditions not seen in training (unseen conditions) are tested.

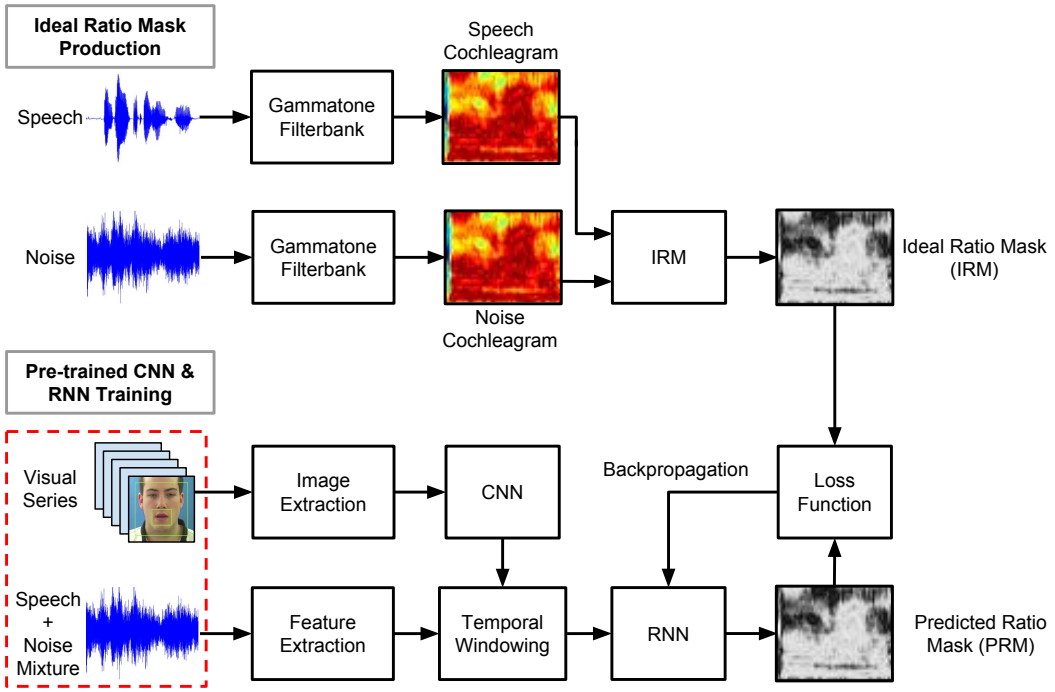


Figure 6.1: Overview of training the CNN & RNN ratio masking speech enhancement system for noise type and SNR independent conditions.

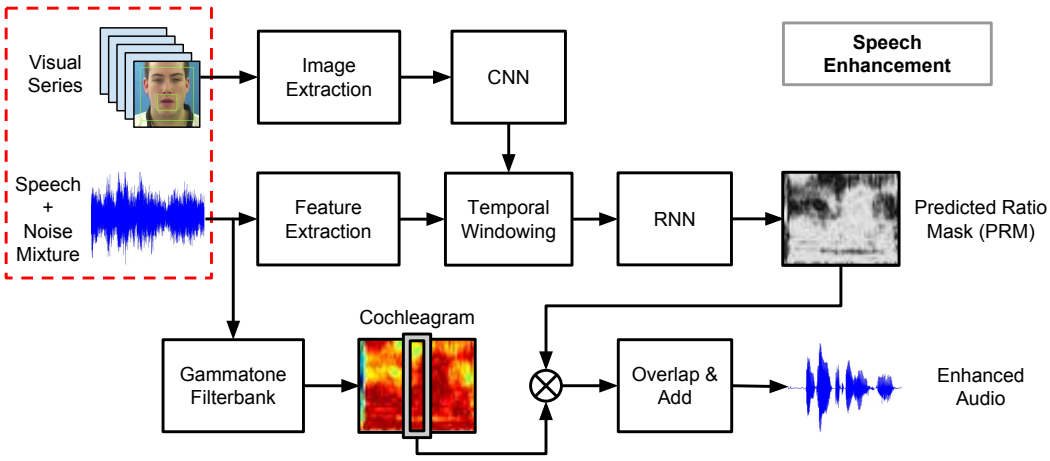


Figure 6.2: Overview of applying the CNN & RNN predicted ratio mask to noisy speech for speech enhancement testing in noise type and SNR independent conditions.

Previous chapters have performed evaluations within a small constrained vocabulary dataset, GRID (Cooke et al. [2006]). Using the GRID dataset has allowed for easier development and optimisation of the various speech enhancement models. However, the techniques and models developed are not specific for constrained

speech, and can be extended and applied to an unconstrained dataset, RM-3000 (Howell et al. [2016]), using the same pipeline as used for GRID (details of the RM-3000 dataset are provided in Section A.3). The key difference is between the number of words within each dataset. Only 51 words are in GRID whereas RM-3000 contains 1000. From experiments conducted so far, largest gains found over audio-only models in intelligibility are from audio-visual models at low SNRs, where the acoustic information is most corrupted and the audio-visual model is using lip-reading to improve performance. However, due to the large increase in words for RM-3000, lip-reading specifically becomes more difficult giving a far more challenging task for the speech enhancement systems, and will further test the generalisation of the proposed work.

The remainder of this chapter is organised as follows. Section 6.2 provides an overview of the model architectures and feature extraction methods used for noise independent training. Also shown is an adaptation of the visual-only and audio-visual models through alternative visual feature extraction compared to the previously best performing noise dependent architectures, required to perform noise independent training within a reasonable time frame. Performance evaluations are made in Section 6.3 which compare the effectiveness of audio-only, visual-only and audio-visual models for generalisation in noise independent conditions for both small scale (GRID) and large scale (RM-3000) datasets. Firstly, Section 6.3.1 evaluates the generalisation to both seen and unseen noise conditions within the GRID dataset. Section 6.3.2 compares the difference in performance when training in noise independent conditions compared to noise dependent models. The generalisation of noise independent models to larger vocabulary unconstrained speech is evaluated in Section 6.3.3, before the effect of applying speech enhancement in small scale or large scale datasets are compared in Section 6.3.4. Finally, this chapter is concluded in Section 6.4.

6.2 Neural network architectures

Previous work has explored and maximised the performance of audio-only, visual-only and audio-visual speech enhancement models for noise dependent conditions. This work now evaluates those models in unconstrained and noise independent conditions, focusing on generalisation. This section introduces an alternative approach for extracting visual features for noise independent training than used previously for noise dependent models, which is subsequently used for visual-only and audio-visual models. This was required to enable the noise independent models to be trained within a reasonable amount of time, which would be impractical using the previous noise dependent approach. An overview of the acoustic feature extraction and audio-only, visual-only and audio-visual architectures is provided. The same model and architecture designs are used for both GRID and RM-3000 datasets.

6.2.1 Audio-only

The best performing audio-only model found is the layer normalised bi-directional recurrent feed-forward hybrid network using gated recurrent units (LNBiGRU-DNN) from Chapter 4, shown in Figure 6.3. The architecture comprises 2 pairs of forward and backward recurrent layers containing 256 gated recurrent units (GRU) per layer (512 per pair), 2 further dense layers containing 1024 rectified linear units (ReLU) and a final linear output layer. A skip connection is included combining the input and output from the recurrent layers. Detailed implementations of the LNBiGRU-DNN are in Section 4.3.2.

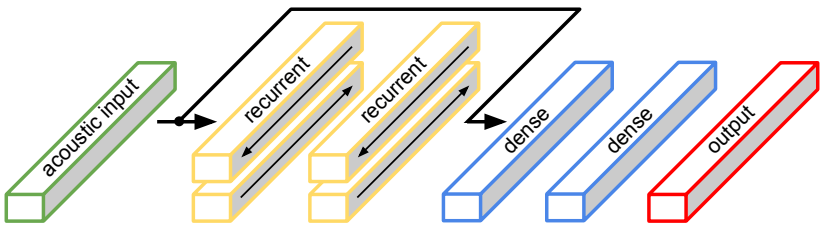


Figure 6.3: Audio-only layer normalised bi-directional recurrent feed-forward hybrid (LNBiGRU-DNN) speech enhancement architecture.

The acoustic feature extracted was the multi-resolution cochleagram (MRCG), which was found to perform best for both binary masking (Chapter 2) and ratio masking (Chapter 3). The MRCG feature combines 4 different cochleagrams, of both high and low resolution, into a single feature, this feature was specifically designed for mask estimation based within a cochleagram framework (Chen et al. [2014]). Detailed implementations of the MRCG feature are in Section 2.3.1.1.

6.2.2 Visual-only

The best performing visual-only model found is an end-to-end trained CNN for feature extraction with a layer normalised bi-directional recurrent feed-forward hybrid network using gated recurrent units (LNBiGRU-DNN) temporal architecture, from Chapter 5. In Chapter 5, an end-to-end trained CNN was compared against using a pre-trained CNN which had been trained for image classification on an image dataset (ImageNet). The end-to-end trained CNN system outperformed the pre-trained CNN system due mainly from being specifically trained on the GRID dataset, however, using features extracted from a pre-trained network was faster to train.

In this work a single model is now trained under multiple noise conditions which causes implications on the time taken to train models. For each additional noise condition, the time taken to train a network increases linearly. Therefore, a compromise between using end-to-end trained CNNs and pre-trained networks is proposed. Instead of training an end-to-end CNN on all noise conditions, a single noise dependent model is trained and used to extract bottleneck features as a replacement to using the GoogLeNet model in Section 5.4.2. This provides a speed up in training, as only the temporal network is trained in multiple noise conditions, and the performance gain from using dataset specific features. The bottleneck features learnt from a noise dependent system are unlikely to differ from one learnt in multiple noise conditions. This is due to the fact the visual stream remains unchanged even when the acoustic signal is mixed in different noise conditions, therefore the input

image ROI is also the same across varying noise conditions.

The implementation of a visual-only noise independent speech enhancement architecture is split into two networks. The first network is trained to extract visual bottleneck features from raw video input using CNNs within a noise dependent environment. The second network trains a noise independent temporal architecture using the extracted bottleneck features, without the need to train additional convolutional layers. To extract visual bottleneck features the end-to-end trained CNN previously shown to perform best in matched noise dependent conditions (Section 5.5.2) was selected. This architecture follows the same implementation as in Section 5.3.2, and is shown in Figure 6.4. The architecture comprises of three sets of [convolutional, channel-wise dropout, max-pooling] layers consisting of [32, 64, 96] kernels of size [5×5, 5×5, 3×3] followed by a single 256 ReLU unit bottleneck layer for feature reduction, before passing to the temporal network. The temporal network selected is a layer normalised bi-directional recurrent feed-forward hybrid network using gated recurrent units (LNBiGRU-DNN) and follows the same implementation as shown for audio-only (Section 6.2.1).

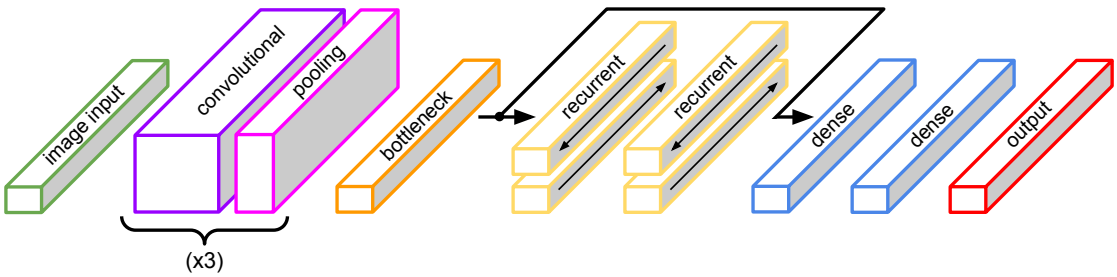


Figure 6.4: Visual-only noise dependent convolutional layer normalised bi-directional recurrent feed-forward hybrid speech enhancement architecture used for bottleneck feature extraction.

As discussed previously, the visual information does not change even when the interfering noise condition does, and as such the visual stream remains unchanged across all noise conditions. Thus, the choice of noise condition and SNR used to train the dependent model should have minimal impact on the features learnt, any variation introduced can be accounted for within training the second temporal network. Therefore, the noise dependent model previously trained in Section 5.5.2 for bab-

ble noise at −5 dB was selected. The dependent model requires images extracted of mouth-only ROIs upsampled via interpolation as input, which was found to perform best in Section 5.5.1.

Mouth-only ROIs are extracted from raw video frames using the Viola-Jones (Viola and Jones [2001]) cascade-based object detector. Images are cropped to a fixed box size of 90×110 pixels centred around the mouth in RGB colourspace before downsampling to 64×64 pixels. Due to the difference in frame rates between acoustic and video input, input images are upsampled through interpolation, for each pixel and RGB channel, to that of the acoustic features. Detailed implementations of mouth-only ROI extraction are in Section 5.3.1. For the RM-3000 dataset, a new noise dependent model is trained in babble noise at −5 dB to extract bottleneck features specific for RM-3000.

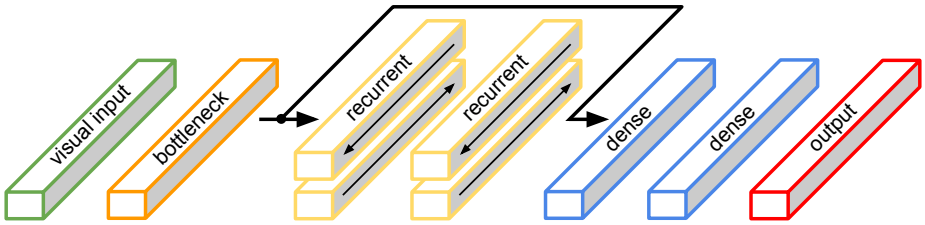


Figure 6.5: Visual-only noise independent layer normalised bi-directional recurrent feed-forward hybrid speech enhancement architecture.

The second network is trained using the visual bottleneck features extracted from the first network for noise independent conditions. The noise independent architecture follows the same implementation as used for the temporal architecture using extracted GoogLeNet features in Section 5.4.2, shown in Figure 6.5. The only difference between this architecture and that used for audio-only, is the additional bottleneck feature before the recurrent layers, this was to account for the potential variation between noise dependent and noise independent features. Although in initial experiments the inclusion of the additional bottleneck layer had minimal performance difference to networks without this layer, but was kept for consistency with the GoogLeNet implementation.

6.2.3 Audio-visual

The best performing audio-visual model found is an end-to-end trained CNN for feature extraction with a layer normalised bi-directional recurrent feed-forward hybrid network using gated recurrent units (LNBiGRU-DNN) temporal architecture, from Chapter 5. As discussed in Section 6.2.2, using an end-to-end system for noise independent models provides additional constraints than that for noise dependent. The time taken to train a noise independent model increases linearly for each additional noise condition over the noise dependent model. Instead, a pre-trained noise dependent end-to-end trained CNN is used to extract bottleneck features before training a noise independent temporal network. The same pipeline is used to build an audio-visual noise independent model. Figure 6.6 shows the temporal architecture used to train the audio-visual noise independent speech enhancement system.

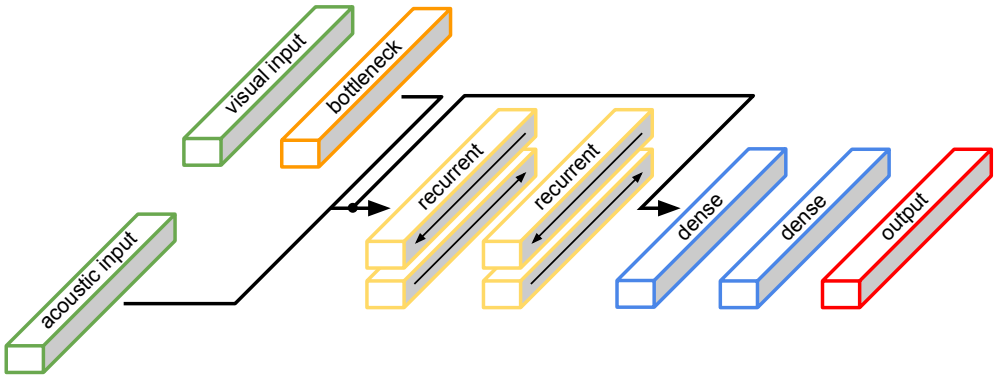


Figure 6.6: Audio-visual noise independent layer normalised bi-directional recurrent feed-forward hybrid speech enhancement architecture.

The same visual-only noise dependent end-to-end trained CNN from Section 6.2.2 is used to extract bottleneck features for the GRID and RM-3000 datasets respectively, i.e the visual bottleneck features used in audio-visual is the same as used in visual-only. The acoustic input used is the multi-resolution cochleagram, which was previously used in Chapter 5 and is the same as used in audio-only (Section 6.2.1).

6.3 Experimental Results

The performance of audio-only, visual-only and audio-visual models (outlined in Section 6.2) for noise generalisation are evaluated first using the small constrained GRID dataset, before applying to the larger unconstrained RM-3000 dataset. Comparisons are made between the performance of noise dependent and noise independent models and between small scale and large scale datasets.

Models are trained in babble, factory and speech shape noise at SNRs of $-10\,\text{dB}$, $-5\,\text{dB}$, $0\,\text{dB}$ and $5\,\text{dB}$, giving 12 seen noise conditions. To evaluate noise generalisation, models are tested in all seen conditions, and then in unseen conditions, namely cafeteria babble and street noise at SNRs of $-10\,\text{dB}$, $-5\,\text{dB}$, $0\,\text{dB}$ and $5\,\text{dB}$, giving an additional 8 unseen noise conditions. Results for the GRID and RM-3000 datasets are in Sections 6.3.1 and 6.3.3 respectively.

The CNN and RNN models were implemented within the Lasagne framework (Dieleman et al. [2015]) with the Theano (Theano Development Team [2016]) backend. Input data was z-score normalised and grouped into mini-batches of 256. To prevent overfitting, dropout of 0.5 was applied between all convolutional layers, and dropout of 0.2 applied to all other layers and early stopping (Prechelt [1998]) was used when the validation score did not improve after 5 further epochs. Training used backpropagation with the Adam optimiser (Kingma and Ba [2014]) and a learning rate 0.001 for CNN and RNN, minimising the MSE loss function.

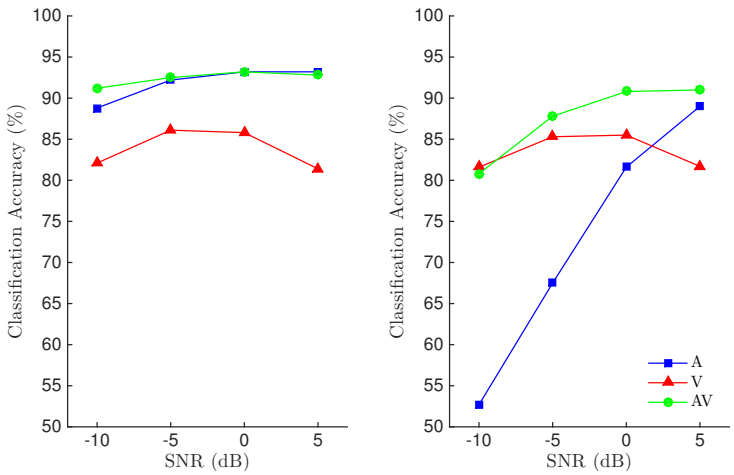
6.3.1 Noise independent speech enhancement – GRID

This experiment compares the generalisation of audio-only, visual-only and audio-visual architectures across both seen and unseen noise conditions for the GRID dataset. Previous experiments have shown the models to perform well in noise dependent conditions, whereas here the focus is noise independence. All experiments use a single speaker (speaker 12) from the GRID dataset (details provided in Section A.1), containing 1000 utterances which are allocated into 640, 160 and 200 for the

training, validation and test sets respectively.

Table 6.1 shows the full set of objective measures for the test set across all 20 noise conditions (12 seen and 8 unseen) for audio-only, visual-only and audio-visual models. Objective measures selected are classification accuracy, HIT-FA rate, PESQ and ESTOI. Figures 6.7 to 6.10 provide detailed breakdowns from Table 6.1 for babble (seen) and street (unseen) noise conditions at −10 dB, −5 dB, 0 dB and 5 dB.

Focusing first on classification accuracy and HIT-FA rate results show that the audio-visual model outperforms both audio-only and visual-only in most seen conditions and all unseen conditions. Figures 6.7 and 6.8 show classification accuracy and HIT-FA rate for babble (seen) and street (unseen) noise.

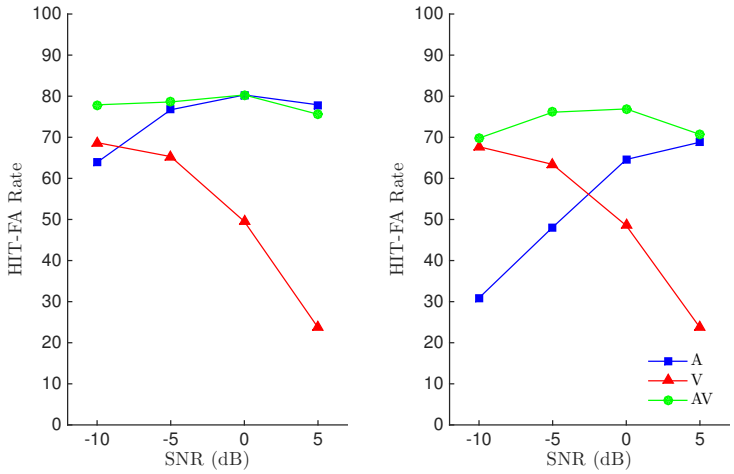


(a) babble (b) street

Figure 6.7: Effect on mask classification accuracy across SNR for audio-only, visual-only and audio-visual models in seen (babble) and unseen (street) noise conditions for ratio mask estimation for the GRID dataset.

In seen conditions, audio-only performs almost equivalently to audio-visual at high SNRs, dropping below audio-visual at low SNRs, particularly at −10 dB. Visual-only performs consistently worse than both audio-only and audio-visual across all conditions except at −10 dB for HIT-FA rate where visual-only outperforms audio-only.

In unseen conditions, the performance of audio-only reduces drastically, particularly at low SNRs. At 5 dB, audio-only performs as well as audio-visual, but at



(a) babble (b) street

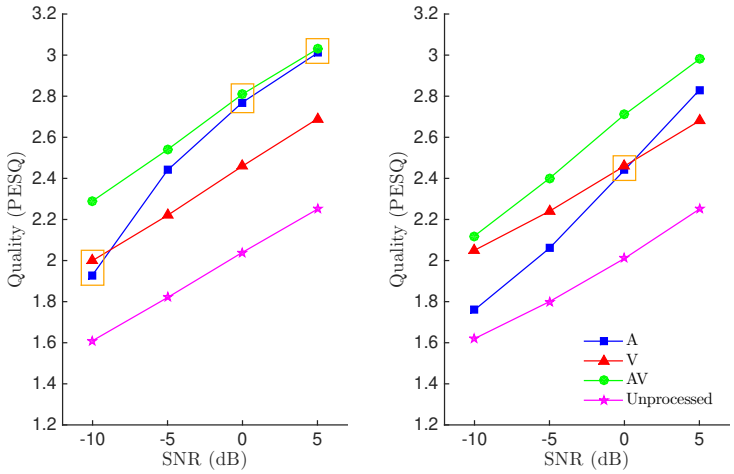
Figure 6.8: Effect on mask HIT-FA rate across SNR for audio-only, visual-only and audio-visual models in seen (babble) and unseen (street) noise conditions for ratio mask estimation for the GRID dataset.

−10 dB, the difference between audio-only and audio-visual increases from 2.4 and 14.0 in seen conditions (babble) to 28.1 and 38.9 in unseen conditions (street) for classification accuracy and HIT-FA rate respectively.

The performance of visual-only stays consistent across all conditions, for both classification accuracy and HIT-FA rate, showing no sign of degradation in unseen noise conditions. Audio-visual also performs similarly in unseen conditions, only at −10 dB does performance drop below that of seen conditions, matching the performance of visual-only. This clearly shows how the audio-visual model adapts the weighting between visual information and acoustic information as the SNR varies, even though the model is not told what the SNR is, only from what is provided by the acoustic features. At low SNRs more weight is provided to the visual stream, whereas at high SNRs more weight is applied to the acoustic stream. When comparing visual-only in seen and unseen noise types across SNR for classification accuracy the performance peaks between −5 dB and 0 dB and tails at both −10 dB and 5 dB, yet for HIT-FA rate performance peaks at −10 dB before dropping rapidly towards 5 dB. This is due to the visual-only model not having access to the acoustic information in training, instead only the visual information is presented. As mentioned in

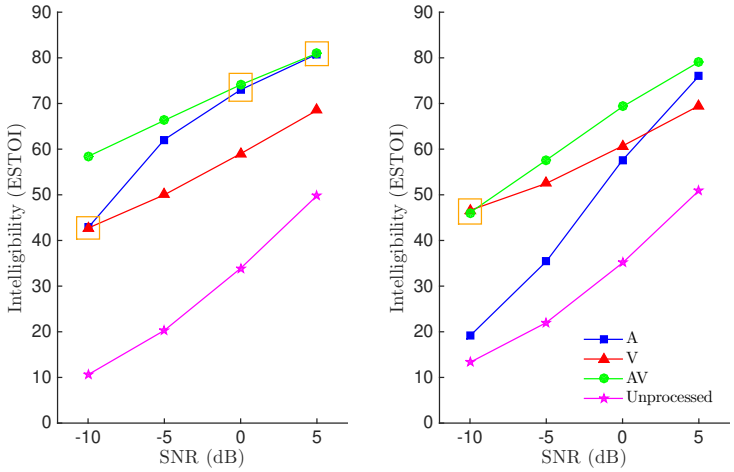
Section 6.2.2, the visual information does not change even when the audio is mixed in different conditions, but the target ratio masks do change for different noise conditions. The visual-only model can only learn a mapping from the provided visual input, and is instead forced to learn a mean ratio mask for each given input sequence. Therefore the mask for an utterance mixed in a particular noise condition is equivalent to the same utterance mixed in all other noise conditions. Here models are trained with SNRs of $-10\,\text{dB}$, $-5\,\text{dB}$, $0\,\text{dB}$ and $5\,\text{dB}$, producing a mean SNR of $-2.5\,\text{dB}$, where the peak is found for classification accuracy, suggesting the mask produced is equivalent to an SNR of $-2.5\,\text{dB}$. As for the trend found in HIT-FA rate, this is due to calculating HITs and FAs for each SNR. At SNRs below $-2.5\,\text{dB}$, the output mask is amplified compared to the target mask, introducing higher HITs and higher FAs, for example babble at $-10\,\text{dB}$ has 88.5 HITS and 19.8 FAs. At SNRs above $-2.5\,\text{dB}$, the output mask is attenuated compared to the target mask, introducing lower HITs and lower FAs, for example babble at $5\,\text{dB}$ has 24.6 HITS and 0.9 FAs.

Looking now at quality scores through PESQ and intelligibility with ESTOI similar trends as with classification accuracy and HIT-FA rate are found, where the audio-visual outperforms both audio-only and visual-only in most seen conditions and all unseen conditions. All models are shown to outperform unprocessed audio across all noise and SNR conditions tested. Two-way analysis of variance (ANOVA) over all model pairs (A, V, AV and unprocessed) per noise type and SNR is applied with multiple comparison tests according to Tukey's HSD (Tukey [1949]; Ghosh and Sharma [1963]). This is performed for each combination of model pairs (giving 6 comparisons) within each noise type and SNR for both PESQ and ESTOI, by comparing objective scores from the test set for each model, i.e for all 200 utterances for GRID, and are statistically significantly different if $p < 0.05$. Figures 6.9 and 6.10 show PESQ and ESTOI scores for babble (seen) and street (unseen) noise for the A, V, AV systems and for unprocessed audio. Two scores, where the difference between them is not statistically significant, are highlighted by being enclosed in an orange box. All other scores are measured to be statistically different.



(a) babble (b) street

Figure 6.9: Effect on quality through PESQ across SNR for audio-only, visual-only and audio-visual models in seen (babble) and unseen (street) noise conditions for ratio mask estimation for the GRID dataset. Model pair's within orange boxes are not statistically significantly different through two-way ANOVA testing (i.e $p \geq 0.05$).



(a) babble (b) street

Figure 6.10: Effect on intelligibility through ESTOI across SNR for audio-only, visual-only and audio-visual models in seen (babble) and unseen (street) noise conditions for ratio mask estimation for the GRID dataset. Model pair's within orange boxes are not statistically significantly different through two-way ANOVA testing (i.e $p \geq 0.05$).

In seen conditions, audio-only performs almost equivalent to audio-visual at high SNRs (no statistically significant difference), dropping below audio-visual at low SNRs (below 0 dB), particularly at -10 dB. Visual-only consistently performs worse than both audio-only and audio-visual across all conditions except at -10 dB, where

high levels of noise is present, visual-only performs similarly to audio-only for both PESQ and ESTOI with no statistically significant difference between them.

In unseen conditions, the performance of audio-only again reduces drastically, particularly at low SNRs. Unlike for classification accuracy and HIT-FA rate, even at high SNRs (5 dB), audio-only still performs worse than audio-visual. For PESQ the performance difference at −10 dB stays consistent at 0.36 for both babble and street noise respectively, however this is due to a reduction for both audio-only and audio-visual, with audio-only providing little gain over unprocessed audio. For ESTOI the performance difference at −10 dB increases from 15.4 for babble to 27.0 for street noise even with the degradation of audio-visual, again audio-only offers little gain over unprocessed audio. A larger difference is seen across higher SNRs (above −10 dB) across PESQ and ESTOI where audio-visual performance stays closer to seen conditions, whereas audio-only drops from seen conditions. A difference of 0.04 and 1.1 for babble and 0.27 and 11.7 for street is seen at 0 dB between audio-visual and audio-only for PESQ and ESTOI scores respectively.

The performance of visual-only stays consistent across all conditions, for both PESQ and ESTOI, showing no sign of degradation in unseen noise conditions. Audio-visual also performs similarly in unseen conditions, only at −10 dB does performance drop below that of seen conditions, matching the performance of visual-only (no statistically significant difference). Both trends were also found for classification accuracy and HIT-FA rate, confirming the adaptability of audio-visual for varying SNRs and the importance of visual information at low SNRs.

Comparing the performance of audio-only, visual-only and audio-visual across all noise conditions and SNRs, audio-visual consistently outperforms both audio-only and visual-only across all objective measures. In seen conditions, the performance gain from audio-visual over audio-only is only found at low SNRs (below 0 dB), with similar performance found at higher SNRs, both outperforming visual-only across all SNRs. In unseen conditions again audio-visual performs best across all objective measures. Visual-only performed equally as well in unseen conditions as to seen con-

ditions, showing no sign of degradation. On the other hand, audio-only performed particularly poorly in unseen conditions, particularly at low SNRs, where performance was marginally better than unprocessed. When considering generalisation, visual information is shown to be critical, allowing both visual-only and audio-visual to perform well in unseen conditions, with the combination of audio-visual still providing best overall performance at all SNRs and noise types evaluated.

Table 6.1: (GRID) Classification accuracy (in %), HIT-FA (in %) PESQ and ESTOI scores for the GRID dataset in seen (babble, factory, speech shape) and unseen (cafeteria babble, street) noise at −10 dB, −5 dB, 0 dB and 5 dB.

Noise (dB)		Model	Acc	HIT-FA	(FA)	PESQ	ESTOI
babble	-10	A	88.8	63.9	(5.2)	1.93	42.8
		V	82.1	68.7	(19.8)	2.00	42.7
		AV	**91.2**	**77.9**	(6.7)	**2.29**	**58.4**
		unprocessed audio				1.61	10.6
	-5	A	92.2	76.7	(4.4)	2.44	62.1
		V	86.1	65.3	(10.8)	2.22	50.0
		AV	**92.5**	**78.6**	(4.6)	**2.54**	**66.3**
		unprocessed audio				1.82	20.3
	0	A	**93.2**	**80.3**	(4.0)	2.77	73.0
		V	85.8	49.5	(4.2)	2.46	59.0
		AV	**93.2**	**80.3**	(4.0)	**2.81**	**74.1**
		unprocessed audio				2.04	33.9
	+5	A	**93.2**	**77.9**	(3.0)	3.01	80.8
		V	81.4	23.7	(0.9)	2.69	68.5
		AV	92.8	75.6	(2.6)	**3.03**	**81.0**
		unprocessed audio				2.25	49.8

Continued on next page

Table 6.1 – continued from previous page

Noise (dB)		Model	Acc	HIT-FA (FA)	PESQ	ESTOI
factory	-10	A	91.8	65.7 (3.4)	1.98	41.9
		V	79.1	69.4 (24.0)	1.94	44.5
		AV	**93.6**	**77.9** (3.8)	**2.32**	**58.7**
		unprocessed audio			1.46	10.5
	-5	A	94.2	76.5 (2.6)	2.46	60.7
		V	86.0	71.4 (13.8)	2.15	51.2
		AV	**94.6**	**78.9** (2.7)	**2.59**	**66.7**
		unprocessed audio			1.66	20.1
	0	A	**95.2**	**81.0** (2.4)	2.79	72.7
		V	89.4	59.8 (5.4)	2.37	59.8
		AV	95.1	80.7 (2.4)	**2.82**	**74.2**
		unprocessed audio			1.87	33.5
	+5	A	**95.4**	**80.0** (1.9)	3.02	81.1
		V	87.3	31.0 (1.2)	2.60	68.8
		AV	95.1	77.9 (1.7)	**3.04**	**81.4**
		unprocessed audio			2.09	49.9
speech shape	-10	A	92.6	76.1 (4.1)	2.21	53.9
		V	81.8	71.6 (21.0)	2.02	45.1
		AV	**93.1**	**79.9** (4.6)	**2.36**	**61.4**
		unprocessed audio			1.60	12.3
	-5	A	**94.2**	82.2 (3.6)	2.54	67.3
		V	87.2	70.6 (11.4)	2.22	52.2
		AV	94.1	**81.3** (3.5)	**2.58**	**68.7**
		unprocessed audio			1.73	22.6
	0	A	**94.8**	**83.8** (3.1)	2.81	**75.9**
		V	88.0	54.9 (4.4)	2.46	61.4
		AV	94.6	82.4 (3.0)	**2.83**	75.8
		unprocessed audio			1.93	37.2
	+5	A	**94.9**	**82.0** (2.3)	**3.06**	**83.2**
		V	84.0	26.5 (0.9)	2.68	70.9
		AV	94.4	79.0 (2.0)	**3.06**	83.0
		unprocessed audio			2.17	53.6

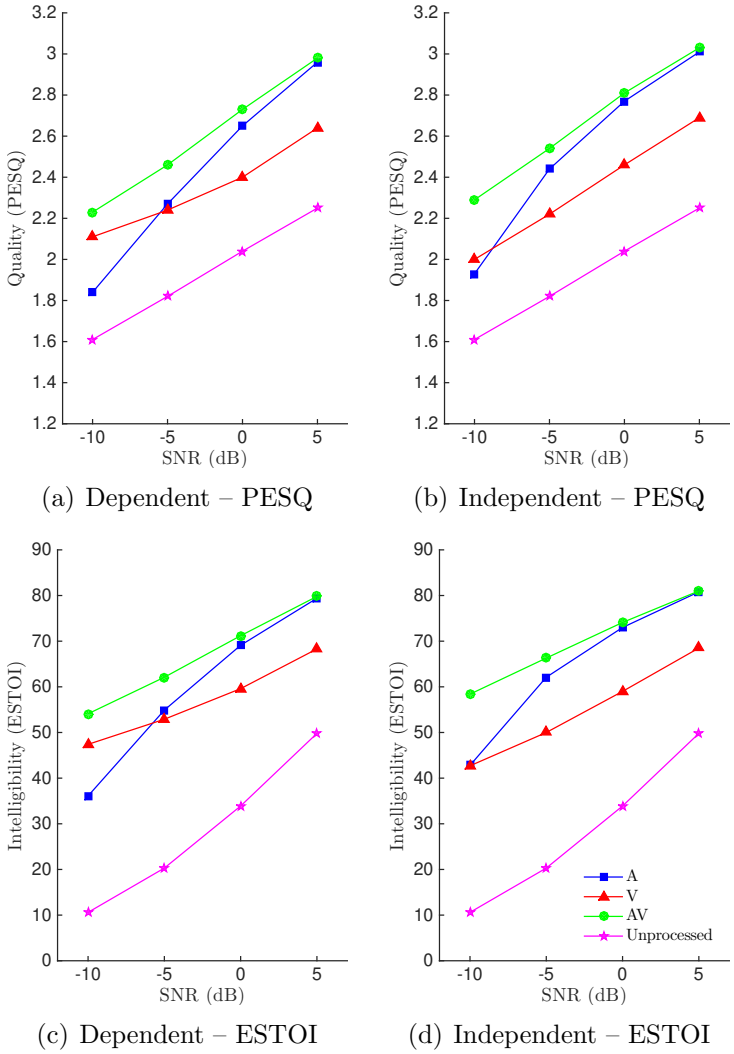
Continued on next page

Table 6.1 – continued from previous page

Noise (dB)		Model	Acc	HIT-FA (FA)	PESQ	ESTOI
cafeteria babble	-10	A	59.0	31.5 (45.7)	1.47	15.0
		V	80.6	69.3 (22.1)	1.91	42.4
		AV	**88.2**	**76.8** (12.1)	**2.07**	**46.8**
		unprocessed audio			1.51	9.3
	-5	A	71.9	49.7 (30.1)	1.83	33.0
		V	86.2	68.4 (12.5)	2.13	50.1
		AV	**91.7**	**79.5** (7.0)	**2.38**	**58.8**
		unprocessed audio			1.68	19.1
	0	A	83.4	66.0 (16.3)	2.28	53.9
		V	87.7	54.0 (5.0)	2.35	58.9
		AV	**93.3**	**79.4** (4.3)	**2.68**	**70.1**
		unprocessed audio			1.86	32.3
	+5	A	90.9	72.4 (5.9)	2.67	71.4
		V	84.6	26.7 (1.1)	2.58	67.9
		AV	**93.2**	**73.7** (2.5)	**2.94**	**79.4**
		unprocessed audio			2.10	47.5
street	-10	A	52.7	30.9 (58.4)	1.76	19.1
		V	**81.6**	67.7 (20.4)	2.05	**46.6**
		AV	80.8	**69.8** (22.8)	**2.12**	46.1
		unprocessed audio			1.62	13.3
	-5	A	67.5	48.1 (38.3)	2.06	35.4
		V	85.3	63.4 (11.6)	2.24	52.5
		AV	**87.8**	**76.1** (12.5)	**2.40**	**57.6**
		unprocessed audio			1.80	22.0
	0	A	81.6	64.6 (19.0)	2.44	57.6
		V	85.5	48.5 (4.7)	2.46	60.7
		AV	**90.8**	**76.9** (7.1)	**2.71**	**69.3**
		unprocessed audio			2.01	35.1
	+5	A	89.0	68.8 (7.2)	2.83	76.0
		V	81.7	23.7 (1.0)	2.68	69.5
		AV	**91.0**	**70.7** (4.1)	**2.98**	**79.1**
		unprocessed audio			2.25	50.9

6.3.2 Noise dependent verses noise independent models

In Section 6.3.1 the best performing audio-only, visual-only and audio-visual architectures for noise dependent architectures were evaluated for generalisation within noise independent conditions. Now, the difference in performance between training noise dependent and noise independent models is compared. Figure 6.11 shows a comparison between quality using PESQ and intelligibility using ESTOI for noise dependent and noise independent trained models in babble noise at −10 dB, −5 dB, 0 dB and 5 dB. Similar results and trends are also found across factory noise.



(a) Dependent – PESQ (b) Independent – PESQ

(c) Dependent – ESTOI (d) Independent – ESTOI

Figure 6.11: Comparison of the effect on quality through PESQ and intelligibility ESTOI across SNR for audio-only, visual-only and audio-visual noise dependent and noise independent models in babble and factory noise conditions for ratio mask estimation for the GRID dataset.

Focusing first at quality through PESQ, the overall performance between noise dependent and noise independent models are similar, with large increases found over unprocessed audio across all models. For audio-only and audio-visual models, performance is marginally better for noise independent models than noise dependent, whereas for visual-only performance is again marginally better at higher SNRs but is slightly degraded at lower SNRs for noise independent compared to noise dependent. An average performance difference of 0.11, -0.05 and 0.07 can be found for audio-only, visual-only and audio-visual noise independent models compared to noise dependent models across all SNRs for babble noise.

Looking now at intelligibility through ESTOI, a larger difference between noise dependent and noise independent models are found, where noise independent models provide performance gains for audio-only and audio-visual models across all SNRs. However, for visual-only models, performance stays equal at higher SNRs between noise independent and noise dependent models, but degrades at lower SNRs where a degradation of 4.7 at $-10\,\mathrm{dB}$ in babble noise is seen. An average performance difference of 4.8, -2.0 and 3.2 can be found for audio-only, visual-only and audio-visual noise independent models compared to noise dependent models across all SNRs for babble noise.

Overall, both audio-only and audio-visual models have gains in quality and large gains in intelligibility for noise independent models over noise dependent models. This is attributed to training with more data across more noise and SNR conditions, as although dependent models are trained at specific noise type and SNR conditions, the SNR still varies throughout the utterance. For visual-only models, a small performance degradation was found for training in noise independent over noise dependent conditions at low SNRs. The model only has access to visual information, and as such cannot take advantage of the SNR variation within the acoustic stream. Instead, the task becomes more challenging as the training data contains more examples of similar input mouth shapes mapping to varying target masks per additional noise condition and SNR. Audio-only and audio-visual models do have ac-

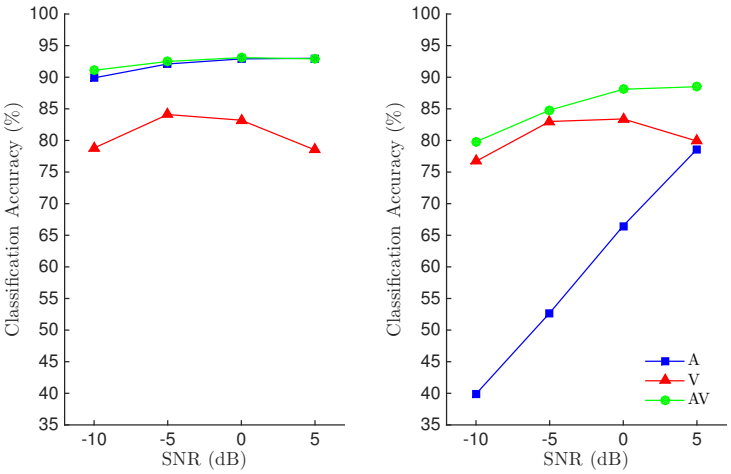
cess to the acoustic stream, and as such can utilise the increased amount of noise and SNR varying training data. Therefore training with more instances of varying SNR conditions is more beneficial in terms of performance for models containing acoustic information as input, and more beneficial for models containing visual information as input due to increased noise generalisation as shown in Section 6.3.1.

6.3.3 Noise independent speech enhancement – RM-3000

In Section 6.3.1 the generalisation of audio-only, visual-only and audio-visual architectures in noise type and SNR independent conditions was evaluated using the GRID dataset. Now, the GRID dataset is replaced with the large vocabulary unconstrained RM-3000 dataset. The RM-3000 dataset increases the vocabulary over GRID from 51 words to 1000 words, increases the number of utterances from 1000 to 3000, which vary in length ranging between 2 and 12s, with an average of 5s, compared to GRIDs 3s utterances, and removes the constrained grammar. This provides a more realistic real-world dataset to evaluate the performance of the proposed models from Section 6.2. The same 20 noise conditions (12 seen and 8 unseen) as used for the GRID experiment are used here. All experiments use a single male speaker from the RM-3000 dataset (details can be found in Section A.3), containing 3000 utterances which are allocated into 1920, 480 and 600 for the training, validation and test sets respectively.

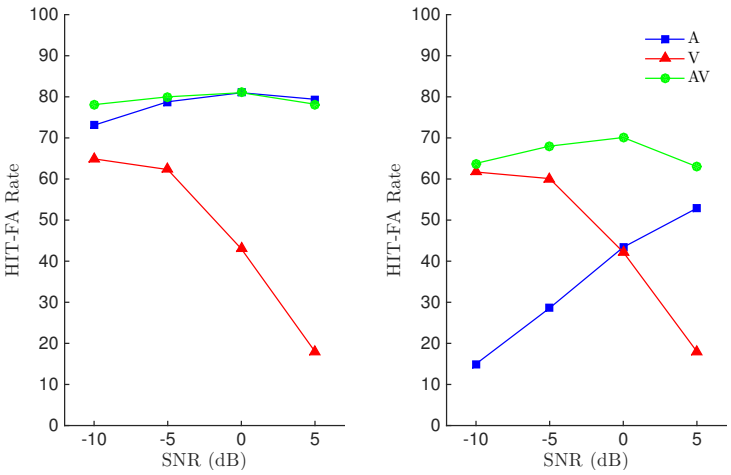
Table 6.2 shows the full set of objective measures for the test set across all 20 noise conditions (12 seen and 8 unseen) for all audio-only, visual-only and audio-visual models. Objective measures selected are classification accuracy, HIT-FA rate, PESQ and ESTOI. Figures 6.12 to 6.15 provide detailed breakdowns from Table 6.2 for babble (seen) and street (unseen) noise conditions at $-10\,\text{dB}$, $-5\,\text{dB}$, $0\,\text{dB}$ and $5\,\text{dB}$.

Focusing first on classification accuracy and HIT-FA rate results show that the audio-visual model outperforms both audio-only and visual-only in most seen conditions and all unseen conditions. Figures 6.12 and 6.13 show classification accuracy and HIT-FA rate for babble (seen) and street (unseen) noise.



(a) babble (b) street

Figure 6.12: Effect on mask classification accuracy across SNR for audio-only, visual-only and audio-visual models in seen (babble) and unseen (street) noise conditions for ratio mask estimation for the RM-3000 dataset.



(a) babble (b) street

Figure 6.13: Effect on mask HIT-FA rate across SNR for audio-only, visual-only and audio-visual models in seen (babble) and unseen (street) noise conditions for ratio mask estimation for the RM-3000 dataset.

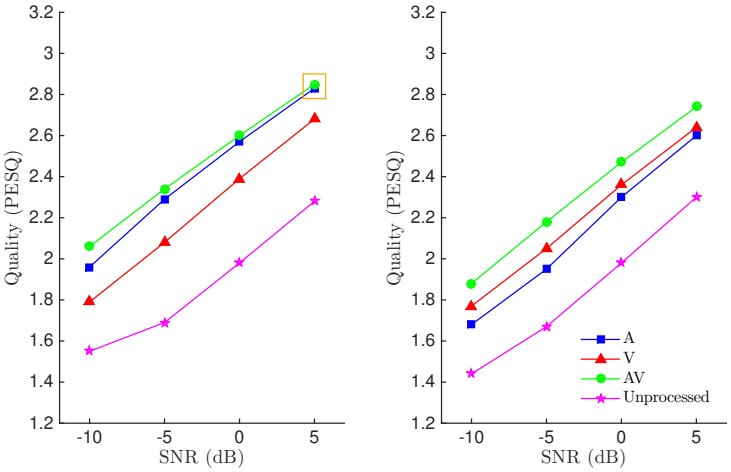
In seen conditions, audio-only performs almost equivalent to audio-visual across most SNRs, dropping just below audio-visual only at $-10\,\mathrm{dB}$. Visual-only consistently performs considerably worse than both audio-only and audio-visual across all conditions and SNRs, unlike for GRID where visual-only could outperform audio-only at $-10\,\mathrm{dB}$ for HIT-FA rate.

In unseen conditions, the performance of audio-only reduces drastically across all SNRs, performing considerably worse than audio-visual and visual-only in classification accuracy. At 5 dB, audio-only performs as well as visual-only in classification accuracy, and outperforms visual-only in HIT-FA rate at 5 dB. The largest difference between audio-only and audio-visual is found at low SNRs, at −10 dB the difference increases from 1.2 and 5.0 in seen conditions (babble) to 39.9 and 48.8 in unseen conditions (street) for classification accuracy and HIT-FA rate respectively.

The performance of visual-only stays consistent across all conditions, for both classification accuracy and HIT-FA rate, showing no sign of degradation in unseen noise conditions. Audio-visual also performs similarly in unseen conditions to seen conditions, with performance gradually reducing as the SNR lowers, before matching the performance of visual-only at −10 dB in HIT-FA rate. Just as with GRID, this shows clearly the benefit of using visual information over audio-only models. The performance of visual-only in classification accuracy and HIT-FA rate follow the same trends as seen for the GRID dataset, and as discussed in Section 6.3.1, this is due from training on multiple noise conditions, producing a mean mask at roughly −2.5 dB which is the mean SNR used in training, causing classification accuracy to peak between −5 dB and 0 dB. For HIT-FA rate, at SNRs below −2.5 dB the output mask is amplified compared to the target mask, introducing higher HITs and higher FAs, for example babble at −10 dB has 90.1 HITS and 25.2 FAs. At SNRs above −2.5 dB the output mask is attenuated compared to the target mask, introducing lower HITs and lower FAs, for example babble at 5 dB has 18.8 HITS and 0.8 FAs.

Looking now at quality scores through PESQ and intelligibility with ESTOI similar trends as with classification accuracy and HIT-FA rate are found, where the audio-visual outperforms both audio-only and visual-only in all seen and unseen conditions. Two-way analysis of variance (ANOVA) over all model pairs (A, V, AV and unprocessed) per noise type and SNR is applied. This is performed for each combination of model pairs (giving 6 comparisons) within each noise type and SNR for both PESQ and ESTOI, by comparing objective scores from the test set for

each model, i.e for all 600 utterances for RM-3000, and are statistically significantly different if $p < 0.05$. Figures 6.14 and 6.15 show PESQ and ESTOI scores for babble (seen) and street (unseen) noise for the A, V, AV systems and for unprocessed audio. Two scores, where the difference between them is not statistically significant, are highlighted by being enclosed in an orange box. All other scores are measured to be statistically different.

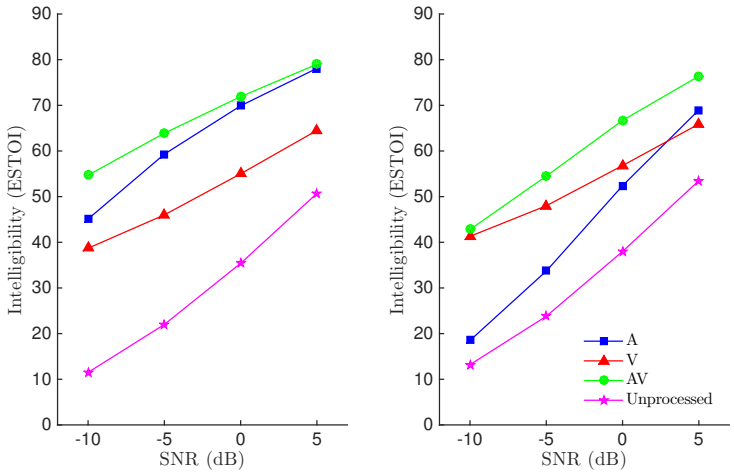


(a) babble (b) street

Figure 6.14: Effect on quality through PESQ across SNR for audio-only, visual-only and audio-visual models in seen (babble) and unseen (street) noise conditions for ratio mask estimation for the RM-3000 dataset. Model pair's within orange boxes are not statistically significantly different through two-way ANOVA testing (i.e $p \geq 0.05$).

In seen conditions, audio-only performs almost equivalently to audio-visual at high SNRs (no statistically significant difference at 5 dB for PESQ), dropping below audio-visual at low SNRs (below 0 dB), particularly at -10 dB, the difference in performance is closer in PESQ than found in ESTOI. Visual-only consistently performs worse than both audio-only and audio-visual across all conditions.

In unseen conditions, the performance of audio-only again drastically reduces, particularly at low SNRs. The performance in PESQ falls just below visual-only across all SNRs (with statistically significant difference), but falls far below visual-only for ESTOI across most SNRs, only at 5 dB can audio-only outperform visual-only. When comparing audio-only to audio-visual, the performance drop in PESQ is consistent across all SNRs, with an average drop of 0.19 found within street noise,



(a) babble (b) street

Figure 6.15: Effect on intelligibility through ESTOI across SNR for audio-only, visual-only and audio-visual models in seen (babble) and unseen (street) noise conditions for ratio mask estimation for the RM-3000 dataset. Model pair's within orange boxes are not statistically significantly different through two-way ANOVA testing (i.e $p \geq 0.05$).

whereas the performance drop in ESTOI is greater at lower SNRs. At -10 dB, the performance difference increases from 9.4 for babble to 24.3 for street noise even with the degradation of audio-visual, as found with GRID audio-only offers little gain over unprocessed audio.

The performance of visual-only stays consistent across all conditions, for both PESQ and ESTOI, showing no sign of degradation in unseen noise conditions. Audio-visual also performs similarly in unseen conditions to seen conditions, with performance gradually reducing as the SNR lowers, before matching the performance of visual-only at -10 dB in PESQ. Both trends were also found for classification accuracy and HIT-FA rate, confirming the importance of visual information at low SNRs.

Comparing the performance of audio-only, visual-only and audio-visual across all noise conditions and SNRs, the same trends are found within RM-3000 as with GRID. Audio-visual consistently outperforms both audio-only and visual-only for both seen and unseen conditions. Visual-only performed equally as well in unseen conditions as to seen conditions, showing no sign of degradation. Audio-only per-

formed almost as well as audio-visual in seen conditions, but performed drastically worse in unseen conditions, falling below visual-only and offering little benefit in intelligibility over unprocessed audio from ESTOI results. When considering generalisation, visual information is again shown to be critical, allowing both visual-only and audio-visual to perform well in unseen conditions.

Table 6.2: (RM-3000) Classification accuracy (in %), HIT-FA (in %) PESQ and ESTOI scores for the RM-3000 dataset in seen (babble, factory, speech shape) and unseen (cafeteria babble, street) noise at −10 dB, −5 dB, 0 dB and 5 dB.

Noise (dB)		Model	Acc	HIT-FA (FA)	PESQ	ESTOI
babble	-10	A	89.9	73.1 (6.7)	1.96	45.1
		V	78.8	64.9 (25.2)	1.79	38.7
		AV	**91.1**	**78.1** (6.9)	**2.06**	**54.7**
		unprocessed audio			1.55	11.5
	-5	A	92.1	78.8 (5.2)	2.29	59.3
		V	84.1	62.3 (12.8)	2.08	46.0
		AV	**92.5**	**80.0** (4.9)	**2.34**	**63.9**
		unprocessed audio			1.69	22.0
	0	A	92.9	**81.0** (4.6)	2.57	69.9
		V	83.2	43.0 (4.3)	2.39	55.0
		AV	**93.1**	**81.0** (4.3)	**2.60**	**71.9**
		unprocessed audio			1.98	35.4
	+5	A	**93.0**	**79.4** (3.6)	2.83	78.1
		V	78.5	18.0 (0.8)	2.68	64.5
		AV	92.9	78.2 (3.3)	**2.85**	**79.0**
		unprocessed audio			2.28	50.7

Continued on next page

Table 6.2 – continued from previous page

Noise (dB)		Model	Acc	HIT-FA (FA)	PESQ	ESTOI
factory	-10	A	92.3	70.6 (4.1)	2.02	43.7
		V	73.1	61.8 (31.1)	1.79	41.3
		AV	**93.1**	**75.9** (4.2)	**2.14**	**55.5**
		unprocessed audio			1.45	12.3
	-5	A	94.1	77.0 (3.0)	2.36	59.0
		V	82.0	63.1 (17.7)	2.10	48.4
		AV	**94.4**	**78.2** (2.9)	**2.42**	**64.8**
		unprocessed audio			1.57	22.5
	0	A	94.8	78.8 (2.4)	2.64	70.4
		V	86.7	49.3 (6.9)	2.40	57.1
		AV	**94.9**	**79.1** (2.3)	**2.67**	**73.0**
		unprocessed audio			1.85	35.9
	+5	A	**95.0**	**76.7** (1.6)	2.91	78.8
		V	85.8	23.7 (1.3)	2.68	65.5
		AV	94.9	76.3 (1.5)	**2.92**	**79.8**
		unprocessed audio			2.15	50.3
speech shape	-10	A	92.2	75.9 (4.8)	2.02	49.9
		V	76.2	64.4 (28.2)	1.74	49.9
		AV	**92.4**	**77.4** (4.9)	**2.08**	**56.3**
		unprocessed audio			1.36	12.7
	-5	A	93.6	**80.7** (4.0)	2.31	62.8
		V	83.8	64.7 (15.1)	2.02	48.2
		AV	**93.7**	**80.7** (3.9)	**2.34**	**65.5**
		unprocessed audio			1.57	23.7
	0	A	**94.3**	**82.1** (3.3)	2.57	72.7
		V	85.7	47.2 (5.4)	2.33	57.5
		AV	**94.3**	81.5 (3.2)	**2.59**	**73.9**
		unprocessed audio			1.87	38.4
	+5	A	**94.4**	**80.0** (2.4)	2.82	80.4
		V	82.6	20.6 (1.1)	2.63	66.6
		AV	94.2	78.6 (2.2)	**2.83**	**80.8**
		unprocessed audio			2.18	53.9

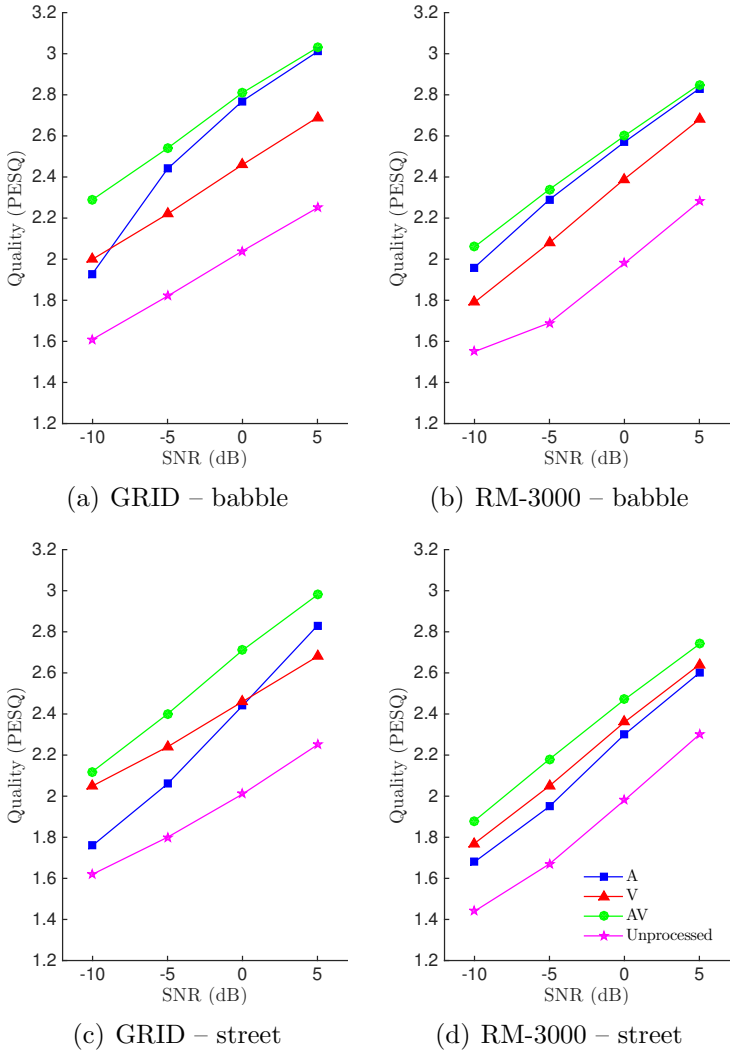
Continued on next page

Table 6.2 – continued from previous page

Noise (dB)		Model	Acc	HIT-FA (FA)	PESQ	ESTOI
cafeteria babble	-10	A	42.9	20.0 (70.7)	1.67	15.8
		V	**76.7**	**63.8** (27.6)	1.73	**39.4**
		AV	72.8	56.9 (31.8)	**1.78**	36.3
		unprocessed audio			1.55	11.3
	-5	A	57.2	33.9 (31.8)	1.86	31.8
		V	**83.4**	62.2 (14.9)	2.02	47.0
		AV	82.1	**65.6** (18.4)	**2.10**	**51.1**
		unprocessed audio			1.58	21.2
	0	A	73.2	48.0 (27.4)	2.22	51.0
		V	84.6	44.4 (5.5)	2.33	55.9
		AV	**87.9**	**67.1** (8.7)	**2.41**	**65.3**
		unprocessed audio			1.84	34.2
	+5	A	84.3	55.6 (10.5)	2.57	67.3
		V	81.4	19.2 (1.1)	2.62	64.4
		AV	**89.2**	**61.1** (3.9)	**2.71**	**75.4**
		unprocessed audio			2.15	48.1
street	-10	A	39.9	15.0 (76.1)	1.68	18.5
		V	76.7	61.7 (27.1)	1.77	41.3
		AV	**79.8**	**63.8** (22.3)	**1.88**	**42.8**
		unprocessed audio			1.44	13.2
	-5	A	52.7	28.6 (58.0)	1.95	33.7
		V	83.0	60.1 (14.4)	2.05	48.0
		AV	**84.8**	**68.0** (14.6)	**2.18**	**54.5**
		unprocessed audio			1.67	23.8
	0	A	66.5	43.4 (38.4)	2.30	52.4
		V	83.4	42.2 (5.3)	2.36	56.8
		AV	**88.1**	**70.1** (9.2)	**2.47**	**66.7**
		unprocessed audio			1.98	38.0
	+5	A	78.6	52.9 (19.5)	2.60	69.0
		V	79.9	18.0 (1.1)	2.64	65.9
		AV	**88.5**	**63.0** (5.1)	**2.74**	**76.4**
		unprocessed audio			2.30	53.5

6.3.4 Effect on dataset size on speech enhancement

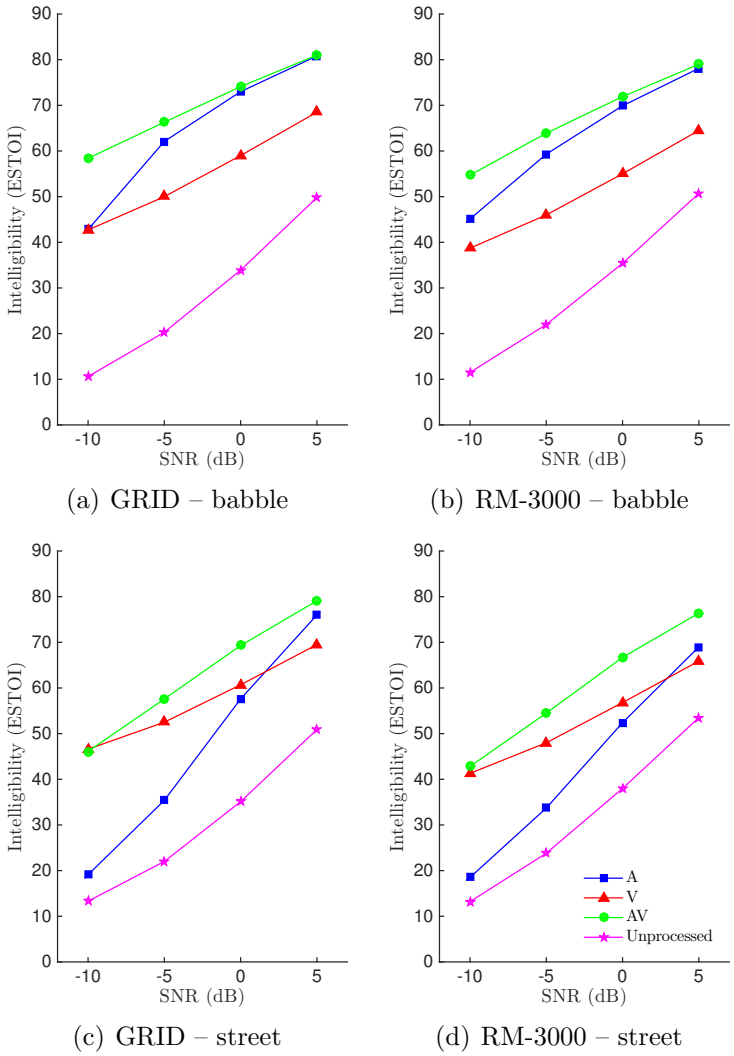
In Sections 6.3.1 and 6.3.3 the generalisation of audio-only, visual-only and audio-visual architectures in noise type and SNR independent conditions was evaluated for the GRID and RM-3000 datasets respectively. Now, the difference in performance between using small constrained vocabulary speech (GRID) and large unconstrained vocabulary speech (RM-3000) is compared. Figures 6.16 and 6.17 show a comparison between GRID and RM-3000 for quality using PESQ and intelligibility using ESTOI in babble (seen) and street (unseen) noise at $-10\,\text{dB}$, $-5\,\text{dB}$, $0\,\text{dB}$ and $5\,\text{dB}$.



(a) GRID – babble (b) RM-3000 – babble

(c) GRID – street (d) RM-3000 – street

Figure 6.16: Comparison of the effect on quality through PESQ across SNR for audio-only, visual-only and audio-visual models in seen (babble) and unseen (street) noise conditions for ratio mask estimation for the GRID and RM-3000 datasets.



(a) GRID – babble (b) RM-3000 – babble

(c) GRID – street (d) RM-3000 – street

Figure 6.17: Comparison of the effect on intelligibility through ESTOI across SNR for audio-only, visual-only and audio-visual models in seen (babble) and unseen (street) noise conditions for ratio mask estimation for the GRID and RM-3000 datasets.

Focusing first at quality through PESQ, the overall performance for GRID is consistently higher than RM-3000, and the performance difference between models is reduced for RM-3000 compared to GRID. In seen conditions, average performance gains of audio-only, visual-only and audio-visual models for babble noise over unprocessed audio across all SNRs of 0.61, 0.41 and 0.74 are found for GRID, whereas gains of 0.54, 0.36 and 0.59 are found for RM-3000 respectively. The performance reduction for RM-3000 compared to GRID stays consistent across all models, with

an average reduction of 0.6. In unseen conditions, average gains over unprocessed audio in street noise for audio-only, visual-only and audio-visual models of 0.35, 0.44, and 0.63 are found for GRID, whereas 0.29, 0.36 and 0.47 are found for RM-3000 respectively. Now the performance difference between audio-only models stay consistent at 0.6, but is increased for both visual-only and audio-visual, raising to 0.16 for audio-visual. This was due to the RM-3000 models containing visual information reduced generalisation to unseen conditions, as discussed in Section 6.3.3.

Looking now at intelligibility through ESTOI, again the overall performance for GRID is consistently higher than RM-3000, however the spread of performance between each model is similar between datasets. In seen conditions, average performance gains of audio-only, visual-only and audio-visual models for babble noise over unprocessed audio across all SNRs of 36.0, 26.4 and 41.3 are found for GRID, whereas gains of 33.2, 21.2 and 37.5 are found for RM-3000 respectively. The performance reduction for RM-3000 compared to GRID stays fairly consistent across all models, with an average reduction of 3.9. In unseen conditions, average gains over unprocessed audio in street noise for audio-only, visual-only and audio-visual models of 16.7, 27.0, and 32.7 are found for GRID, whereas 11.3, 20.1 and 28.0 are found for RM-3000 respectively. Unlike PESQ results, the difference between models for intelligibility stays consistent in unseen conditions, raising to just 5.7 in unseen from 3.9 in seen conditions.

The drop in performance for visual-only and audio-visual models is relatively small across all conditions for intelligibility, raising slightly for unseen conditions within PESQ. This is surprising considering how challenging lip-reading is for large vocabulary speech. Audio-only performs similarly within both datasets, with little distinction between them. Audio-visual performs slightly worse for RM-3000, due mainly from the increased confusability within the visual information as seen from visual-only. Overall, the application of noise independent speech enhancement to an unconstrained, large vocabulary dataset has been successful showing similar performance to the GRID dataset.

6.4 Conclusions

This work has examined the effect on intelligibility (ESTOI), quality (PESQ) and mask accuracy (classification accuracy and HIT-FA rate) of using audio-only, visual-only and audio-visual models within noise type and SNR independent conditions for estimating ratio masks for speech enhancement. Firstly, an alternative method for extracting CNN bottleneck features was proposed to allow visual-only and audio-visual models to be trained under multiple noise conditions within a reasonable amount of time. These noise independent models were subject to experiments in both seen and unseen noise conditions to evaluate generalisation, using both small scale (GRID) and large scale (RM-3000) datasets.

It was found that both audio-only and audio-visual models perform well in seen conditions. Audio-only performed drastically worse in unseen conditions, particularly for intelligibility, where little benefit was found over unprocessed audio. Visual-only performed equally well in both seen and unseen conditions, outperforming audio-only in unseen conditions. Audio-visual provided peak performance across all measures and conditions, and generalised well to unseen conditions. The inclusion of visual information within audio-visual models over audio-only models not only provides large gains in performance at low SNRs, but is critical for allowing the model to generalise to unseen noise conditions, proven by the audio-only and visual-only results.

A comparison between training noise dependent and noise independent systems revealed that both audio-only and audio-visual models show performance gains in terms of quality and intelligibility in noise independent conditions over noise dependent. Visual-only models showed a small degradation in performance at low SNRs when trained in noise independent conditions. The improvements for audio-only and audio-visual models, and reductions for visual-only is attributed to training with more data across more noise and SNR conditions, as although dependent models are trained at specific noise type and SNR conditions, the SNR still varies throughout the utterance.

Visual-only models only have access to visual information on input, and as such cannot take advantage of the SNR variation within the acoustic stream. Instead, the task becomes more challenging as the training data contains more examples of similar input mouth shapes mapping to varying target masks per additional noise condition and SNR. Audio-only and audio-visual models do have access to the acoustic stream, and as such can utilise the increased amount of noise and SNR varying training data. Therefore training with more instances of varying SNR conditions is more beneficial in terms of performance for models containing acoustic information as input, and more beneficial for models containing visual information as input due to increased noise generalisation.

Performance using the large, unconstrained RM-3000 dataset provided surprisingly similar results to the GRID dataset. Even though the RM-3000 dataset is considerably larger and more challenging due to the increased vocabulary size, the models were still able to provide substantial gains across all measures. The overall strong performance of all models for RM-3000 shows that the model architectures and approaches developed can generalise to larger vocabulary unconstrained speech across noise independent conditions and can be considered for monaural speaker dependent real-world applications.

Combining both audio and visual modalities into a single bimodal audio-visual system still provides best performance across all noise types and SNRs, confirming that combining audio and visual features provides a robust complimentary feature set. Experiments clearly showed that the audio-visual model adapts its weighting between visual information and acoustic information as the SNR varies, even though the model is not told what the SNR is, only from what is provided by the acoustic features. At low SNRs more weight is provided to the visual stream, whereas at high SNRs more weight is applied to the acoustic stream. Visual information was shown to be critical for generalisation to unseen noise conditions in both small scale (GRID) and large scale (RM-3000) datasets.

Chapter 7

Conclusions

7.1 Restatement of the problem

Speech enhancement is concerned with improving some perceptual aspect of speech that has been degraded by noise. The introduction of noise affects both the perceived quality and perceived intelligibility of the speech for the listener. The aim of this thesis has been to explore monaural speech enhancement focusing on improving intelligibility, although the effect on quality is also considered.

The key focus is on how visual speech information can be used within speech enhancement systems. Visual speech has been applied successfully to other areas of speech processing when the acoustic stream is degraded by noise, such as ASR for both lip-reading (visual-only) and as a combination of visual and acoustic information (audio-visual), as the information provided from the visual speech is not corrupted from the interfering noise. This property of visual speech is therefore well suited for speech enhancement, and as such audio-only, visual-only and audio-visual speech enhancement models were evaluated throughout the thesis.

The methods presented in this work were developed using speech from a male speaker from the GRID audio-visual dataset. Objective evaluations of the predicted masks and the quality and intelligibility of the enhanced speech signal within varying

noise type and SNR conditions were performed to determine the performance of the various methods and configurations explored. The speech enhancement system can be segmented into three main areas: speech enhancement algorithm selection, feature extraction of input data for model input, and the model architecture used within supervised learning.

7.2 Summary

Masking algorithms rely on a criterion function to determine the values contained within the mask used to suppress noise and retain speech information, which can take the form of either binary masking or ratio masking. Chapters 2 and 3 evaluated and compared the performance of binary and ratio masking within a deep feed-forward neural network (DNN) architecture.

In Chapter 2, binary masking was evaluated and formed the baseline system of this thesis. Novel loss functions were proposed to improve the mapping of input data to target masks within training the DNN model, focusing on improving the resulting intelligibility of the enhanced signal. The proposed loss functions were found to increase intelligibility over standard classification loss functions, particularly at higher SNRs, with the binary cross-entropy HIT-FA hybrid (CEHF) loss function performing best.

In Chapter 3, ratio masking was explored and compared against binary masking. It was found that across all objective measures, ratio masking outperformed binary masking for all models evaluated. This is attributed to the constrained nature of binary masking, as a binary mask is a representation of a ratio mask that has been quantised into two classes, speech dominant and noise dominant. This quantisation removes and reduces the resolution found within the ratio mask, which produces the degradation in performance. The largest improvement was for quality through PESQ, which was found to provide large gains over unprocessed audio for ratio masking, yet for binary masking unprocessed audio scored higher than the enhanced

signals, particularly at low SNRs. Binary masking introduces musical noise into the enhanced signal which reduces quality, however ratio masking does not introduce these distortions, resulting in improved quality over binary masking and unprocessed audio. The large gain in quality and improvements for intelligibility and mask accuracy resulted in ratio masking being selected for the remaining experiments of this thesis.

In Chapter 4, the DNN used in Chapter 3 was replaced with recurrent neural networks (RNNs) for temporal modelling. RNNs are specifically designed for modelling sequences with temporal structure and as such are well suited for speech processing tasks. It was found that RNNs outperform standard DNNs across all objective measures for audio-only and audio-visual models, but visual-only models found minimal performance difference between architectures. The proposed bi-directional recurrent feed-forward hybrid network using gated recurrent units and layer normalisation (LNBiGRU-DNN) was the best performing RNN architecture and was used in subsequent experiments.

Feature extraction from input data plays a key role in the ability of the speech enhancement models to learn a mapping to the target output masks. Without robust features extracted, the model is unable to accurately predict target masks, resulting in poor enhanced speech signals. In Chapters 2 and 3 traditional acoustic feature extraction techniques were optimised within binary masking and ratio masking frameworks. It was found that extracting multi-resolution cochleagram (MRCG) acoustic features outperform an ensemble of traditional complimentary acoustic features (ARPMG) for both binary and ratio masking frameworks. Although the ARPMG features work well for other speech processing applications, the MRCG is more similar and closer related to the masking framework used in this work (cochleagram based).

In Chapter 5, traditional visual feature extraction using active appearance models (AAM) was compared against using CNNs which perform feature extraction directly on the raw image, for visual-only and audio-visual models. It was found that us-

ing mouth-only regions-of-interests (ROIs) outperformed using full-face regions-of-interest within CNN archtictures. This was attributed to the mouth area containing most of the important information of the visual articulators, such as lips and teeth, which although are captured within the full-face ROI, due to the downsizing prior to input into the CNN, the information is degraded and lost. Upsampling images across time via interpolation over repetition also provided performance gains for both the mouth-only and full-face ROI configurations. For both visual-only and audio-visual models, using CNNs on mouth-only ROIs outperformed traditional AAM features across most objective measures, with large gains found for intelligibility.

For real-world applications, the noise type and SNR condition is likely to change over time, and therefore dependent models are less effective. One approach could be to train and store dependent models for all possible noise conditions, however this approach is unfeasible and scales poorly. Instead, Chapter 6 presents noise independent models, where a single model is trained within multiple noise conditions (seen) and tested in both seen and unseen noise conditions to evaluate generalisation.

Evaluations found similar trends to dependent models in conditions which were previously seen in training, with audio-visual models outperforming audio-only and visual-only across all seen conditions. Audio-only still performed well at high SNRs and visual-only performed well at low SNRs. However, in unseen conditions, the performance of audio-only models significantly degraded, particularly at low SNRs falling to equivalent performance of unprocessed audio. Both visual-only and audio-visual performed well in unseen conditions, providing similar performance to seen conditions, with only audio-visual performance degrading slightly at low SNRs. This degradation in performance is attributed to the inability to extract useful information from the acoustic stream, as found within the audio-only model. Audio-visual provides best performance across all measures, generalising to both seen and unseen noise conditions at varying SNRs, and with the stable performance also found with visual-only reveals that visual information is critical for noise condition generalisation.

Noise independent models were also applied to large unconstrained vocabulary speech (RM-3000). Evaluations provided surprisingly similar results to the GRID dataset. Even though the RM-3000 dataset is considerably larger and more challenging due to the increased vocabulary size, the models were still able to provide substantial gains across all measures. The overall strong performance of all models for RM-3000 shows that the model architectures and approaches developed can generalise to unconstrained speech across noise independent conditions and can be considered for monaural speaker dependent real-world applications.

7.3 Key findings

Considering the aims of the project set out in Section 1.2, and the work carried out, the following key findings were found.

7.3.1 Key finding #1 – Including visual speech information for speech enhancement

Across Chapters 2 to 5 the performance of audio-only, visual-only and audio-visual models were optimised in noise type and SNR dependent conditions. It was found that all models provide large gains in intelligibility and quality over unprocessed audio, and combining audio and visual speech into a single bimodal audio-visual model outperformed both audio-only and visual-only models across all objective measures, noise types and SNRs. Largest gains over audio-only was found at low SNRs, where the acoustic signal is most corrupted from interfering noise, at high SNRs audio-only models performed equally to audio-visual. At low SNRs, visual-only models was able to outperform audio-only models, but performed poorly at high SNRs compared to audio-only and audio-visual models. The audio-visual model was able to utilise information from both acoustic and visual streams, where visual information is more important at low SNRs, and acoustic information is more important at high SNRs.

7.3.2 Key finding #2 – Visual feature extraction

In Chapter 5 visual feature extraction using both pre-trained and end-to-end trained CNNs was shown to outperform traditional AAM feature extraction. Pre-trained CNNs train the CNN and temporal model in isolation from each other, whereas an end-to-end trained CNN is trained with the temporal model in a single network with backpropagation applied through all network layers. This work used the GoogLeNet model trained for image classification as the pre-trained network used to extract bottleneck features from passing images cropped to mouth-only ROIs, before feeding into the temporal model. Training an end-to-end CNN outperformed extracting bottleneck features from pre-trained networks across all objective measures for both visual-only and audio-visual models. This was attributed to the dataset and task dependent features learnt within an end-to-end trained CNN, whereas pre-trained networks are more generalised.

In order to train both visual-only and audio-visual models for noise independent conditions, a pre-trained visual-only noise dependent end-to-end system is used to extract visual bottleneck features instead of training a fully end-to-end CNN system (which performed best in dependent conditions) on all noise conditions. This provides the benefit of producing dataset dependent features and the time saving found from pre-trained networks. This visual bottleneck feature was used for training both visual-only and audio-visual models in noise independent conditions. Using an end-to-end trained CNN for noise independent models could provide increased performance over using pre-trained CNNs, however the substantial increase in training time and processing requirements favours using dataset specific pre-trained CNNs.

7.3.3 Key finding #3 – Deep learning architecture for temporal modelling

The use of neural networks has recently provided large gains in performance for various supervised learning tasks, and as such were used as the model architecture

throughout this thesis. Neural networks were used for modelling the relationship between input speech data and target masks generated from the speech enhancement algorithms criteria function. This work has explored using deep feed-forward neural networks (DNN) and recurrent neural networks (RNN) for temporal modelling of the input speech data. Chapter 4 explored replacing the DNN used in Chapter 3 with RNNs, which are specifically designed for modelling temporal sequences and as such are well suited for speech processing tasks. It was found that RNNs outperform standard DNNs across all objective measures for audio-only and audio-visual models, but visual-only models found minimal performance difference between all architectures. The proposed bi-directional recurrent feed-forward hybrid network using gated recurrent units and layer normalisation (LNBiGRU-DNN) was the best performing RNN architecture and was used in subsequent experiments.

7.3.4 Key finding #4 – Generalisation to unseen noise conditions

For real-world applications, the noise type and SNR condition is likely to change over time, and therefore dependent models are less effective. One approach could be to train and store dependent models for all possible noise conditions, however this approach is unfeasible and scales poorly. Instead, Chapter 6 presents noise independent models, where a single model is trained within multiple noise conditions (seen) and tested in both seen and unseen noise conditions to evaluate generalisation. The best performing audio-only, visual-only and audio-visual dependent architectures were selected for evaluation in noise independent conditions.

Evaluations found both audio-visual and visual-only models to generalise to both seen and unseen conditions, however audio-only models only performs well in seen conditions and significantly degrade in unseen conditions, particularly at low SNRs falling to equivalent performance of unprocessed audio. The stable performance of audio-visual and visual-only models showed the importance of visual information, revealing visual information is critical for noise condition generalisation.

7.3.5 Key finding #5 – Training noise dependent and noise independent models

When comparing training in noise dependent or noise independent conditions, it was found both audio-only and audio-visual models had large gains in intelligibility for noise independent models over noise dependent models. This was attributed to training with more data across more noise conditions, as although dependent models are trained at specific noise type and SNR conditions, the SNR still varies throughout the utterance. For visual-only, a small performance degradation was found for training in noise independent conditions over noise dependent conditions. The model only has access to visual information, and as such cannot take advantage of the SNR variation within the acoustic stream. Instead, the task becomes more challenging as the training data contains more examples of similar input mouth shapes mapping to varying target masks per additional noise condition and SNR. Audio-only and audio-visual models do have access to the acoustic stream, and as such can utilise the increased amount of noise and SNR varying training data. Therefore training with more instances of varying SNR conditions is beneficial for models containing acoustic information as input.

7.3.6 Key finding #6 – Application to large unconstrained vocabulary speech

The best performing models developed for the GRID dataset were applied on the RM-3000 dataset to evaluate the generalisation of the developed models to large unconstrained vocabulary speech. The same approach and model architectures used for the GRID dataset was applied to RM-3000, with models trained in multiple varying noise conditions and tested in both seen and unseen conditions. Evaluations found the same trends as seen with the GRID dataset across all models for both seen and unseen conditions. Audio-only continued to perform well in seen conditions, but performed poorly for unseen showing poor generalisation. Visual-only performed

equally well in both seen and unseen conditions showing strong generalisation, but was consistently worse than audio-visual models. The audio-visual model continued to provide best performance and generalisation across all conditions tested. When comparing RM-3000 with GRID results, audio-only models performed equally well, but visual-only and audio-visual models performed slightly worse for RM-3000 than GRID. This performance degradation is attributed to the larger vocabulary set within RM-3000, containing more confusable words within the visual domain. It is well known that phonemes can have the same visual articulation, and as such was more challenging for the model to distinguish them. The overall strong performance of all models for RM-3000 shows that the model architectures and approaches developed can generalise to unconstrained speech across noise independent conditions and can be considered for monaural speaker dependent real-world applications.

7.4 Future work

This thesis has presented audio-only, visual-only and audio-visual methods for monaural speaker dependent speech enhancement. In this section, potential future work is outlined focusing on three main areas: i) Further improvements for the best performing architecture are outlined. ii) Modifications for real-time applications are considered. iii) Performance implications and potential datasets for speaker independent applications.

7.4.1 Further model improvements

Feature extraction has been shown to be key for the performance of speech enhancement algorithms. It was shown that using convolutional neural networks (CNN) for visual feature extraction outperformed traditional feature extraction methods. Future work could explore the use of CNNs for improving acoustic feature extraction. This could be achieved in two forms, either from applying 1-dimensional CNNs directly on the raw noisy audio signal, or by applying 2-dimensional CNNs on the

noisy audio transformed into cochleagram or spectrogram representation. Both approaches are worth consideration, approach one would be an ideal scenario and would remove the additional transform needed for approach two, however the network may struggle to learn appropriate features due to the interfering noise degrading spectral structure. Approach two would provide structure which may be necessary to aid the network in training to learn features, at the expense of speed.

7.4.2 Real-time applications

Speech enhancement is also required for real-time applications such as hearing-aids, cochlear implants and communication systems. Such applications also have security concerns regarding personal data, which subsequently must be encrypted to prevent cybersecurity attacks, when the speech enhancement model located within the cloud instead of a personal device. Work performed in (Adeel et al. [2018]) has developed encryption techniques for both the audio-visual input data and the transmitted enhanced signal for hearing-aid applications. The approaches developed within this thesis can be modified to work within real-time constraints. Two main factors affect real-time performance, amount of future acoustic windowing used as input and processing time. For acoustic windowing, traditionally symmetric windows of speech are extracted, containing both past and future contexts. As discussed in Chapter 4, both past and future contexts are important in order to model carry-over and anticipatory coarticulation. For real-time applications, the amount of future context affects the availability to be performed in real-time, yet reducing future context reduces the ability of the model to learn anticipatory coarticulation.

Work performed for real-time automatic speech animation (Websdale et al. [2018]) has shown using asymmetric acoustic windows as input can still provide realistic performance as fully symmetric windows. The same approach used can be applied for speech enhancement and requires minimal modification to the architectures developed within this thesis which use recurrent neural networks for the temporal model. The amount of processing time required depends on the available hardware and

the neural network model. Using deep networks with convolutional and recurrent networks does impact the time taken to make predictions, however the amount of future context has a larger impact. The available hardware could limit the size of the network model used and is application specific. Therefore, a balance between future context, model architecture and hardware is required to fit the desired real-time application.

7.4.3 Speaker independent audio-visual speech enhancement

For real-world applications it is likely that speaker independent speech enhancement would be more beneficial than speaker dependent, such as systems used for automatic speech recognition or for use within hearing impairment applications. Subsequently, the speaker dependent models developed within this thesis have been shown to generalise to both noise independent conditions and larger vocabulary unconstrained datasets. However, this was achieved by training in multiple noise conditions (seen conditions), and testing in unseen noise conditions (unseen conditions). To achieve speaker and noise generalisation required for speaker independent applications, the models could be trained on a number of speakers (seen speakers), and tested with unseen speakers within a noise generalised framework. The same architectures and approaches developed for speaker dependent applications can be extended for use in speaker independent applications.

It is expected similar amounts of performance degradation found within speaker independent ASR applications would also be found within speaker independent speech enhancement. Visual-only applications (lip-reading) are particularly affected when moving from speaker dependent to speaker independent conditions. This is due to the highly speaker dependent visual stream, and although speakers may use similar mouth movements and visual articulators to produce the same phoneme, the acoustic signal produced can vary greatly in terms of frequency and pitch. Therefore, when considering visual-only application for speech enhancement, the models are likely to perform well for seen speakers, but generalise poorly to unseen speak-

ers. However, for audio-only and audio-visual models, the performance should stay stable between seen and unseen speakers and show strong generalisation due to the acoustic stream. With these considerations, it is expected that audio-visual models should continue to outperform audio-only and visual-only models, with the acoustic information providing generalisation to speakers, and visual information providing generalisation to noise type and SNR conditions.

To develop audio-visual speaker independent speech enhancement models, datasets containing both audio and visual streams of multiple speakers are required. The currently used GRID dataset can provide small vocabulary constrained speech, and the TCD-TIMIT (Harte and Gillen [2015]) dataset could be selected for large vocabulary unconstrained speech (details of the TCD-TIMIT dataset are shown in Section A.4). The number of speakers and noise conditions used within training will impact the time taken to train models, and as such the available hardware and processing power should also be considered.

Appendix A

Datasets

A.1 GRID

The GRID audio-visual speech dataset (Cooke et al. [2006]) contains low-resolution and high-definition video, and audio recordings of 34 speakers, of which 18 are male and 16 are female. Speaker 12 is selected in this work. For each speaker there are recordings of 1000 utterances each with a length of three seconds, giving 50 minutes of data in total. The ages of the speakers range from 18 to 49, with all but two of the speakers having British accents. Sentences take the form:

$$< \text{command} >< \text{colour} >< \text{preposition} >< \text{letter} >< \text{digit} >< \text{adverb} >$$

and follow the grammar as displayed in A.1.

Table A.1: GRID sentence grammar.

command	colour	preposition	letter	digit	adverb
bin	blue	at	A-Z	1-9	again
lay	green	by	minus W	zero	now
place	red	in			please
set	white	with			soon

The video has a frame rate of twenty-five frames per second, giving seventy-five

frames per three-second video. The high-resolution frame size is 720×576 pixels, and the low-resolution frame size is 360×288. Both sets of video contains full red-green-blue (RGB) colour information. Accompanying the dataset are word time-alignment files for each utterance that describe the start and end points for each word, including periods of silence. Separately recorded audio, sampled at 50 kHz, accompanies the video stream. Furthermore, two sets of imaged-based 2D-DCT visual features are provided. One set contains features extracted from a region of interest that is stationary throughout the video, and the other from a region of interest located about a tracked point localised to the mouth of the speaker.

A.2 NOIZEUS

A noisy speech corpus (NOIZEUS) (Hu and Loizou [2007]) was developed to facilitate comparison of speech enhancement algorithms among research groups. The noisy database contains 30 IEEE sentences (produced by three male and three female speakers) corrupted by eight different real-world noises at different SNRs. The noise was taken from the AURORA database and includes suburban train noise, babble, car, exhibition hall, restaurant, street, airport and train-station noise. Noises are provided at 16 kHz and is the sampling rate used in this work.

A.3 RM-3000

The RM-3000 audio-visual speech dataset (Howell et al. [2016]) was collected for performing confusion modelling for lip-reading, where it was found that other large-vocabulary audio-visual datasets contained too few data. The corpus contains 3000 utterances spoken by a native English male speaker, with sentences selected from the Resource Management corpus (Price et al. [1988]). The vocabulary contains 1000 words, and lends itself well for continous audio-visual speech processing applications. The sentence length ranges between 2 and 12s, with an average of 5s.

The video information was captured at 60 frames per second with a resolution of 1920 × 1080 pixels. The camera was placed in front of the speaker to record a full-frontal pose. A clip-on microphone was used to record the audio with a sampling frequency of 48 kHz. Pre-extracted AAM features of the inner and outer-lip are provided, having been extracted from the video re-sampled to a resolution of 640×360 pixels. The AAM visual feature vector dimensionality was chosen to retain 95 % of the shape variation, and 90 % of the appearance variation. Furthermore, phoneme transcriptions are provided.

A.4 TCD-TIMIT

The TCD-TIMIT audio-visual speech dataset (Harte and Gillen [2015]) was collected for performing speaker independent audio-visual speech recognition, where it was found that other audio-visual datasets contained too few speakers. In total, there are 62 speakers in TCD-TIMIT, which use sentences from the TIMIT corpus (Garofolo et al. [1993]). Three of these are professional lipspeakers, and each reciting almost 400 sentences each. All the lipspeakers are female. The average age of the lipspeakers is 60. The other 59 TCD-TIMIT speakers are non-lipspeakers and were recruited from volunteers around the local University, each reciting 98 sentences. Of these, 32 are male and 27 are female. The average age is 24, with the minimum age 16 and the maximum age 57. One speaker has a Spanish accent, and two have British accents. The rest of the accents in the database are Irish accents, the majority being "neutral" Dublin accents. The data was shot in front of a green screen for possible speaker segmentation applications.

The video information was captured from two cameras, one camera recorded the speaker from directly in front, while the other recorded at an angle of 30° to the speaker's right. Video is captured at 30 frames per second with a resolution of 1920 × 1080 pixels. A clip-on microphone was used to record the audio with a sampling frequency of 48 kHz.

Bibliography

Adeel, A., Ahmad, J., and Hussain, A. (2018). Real-Time Lightweight Chaotic Encryption for 5G IoT Enabled Lip-Reading Driven Secure Hearing-Aid. *arXiv preprint arXiv:1809.04966*.

Ahmadi, M., Gross, V. L., and Sinex, D. G. (2013). Perceptual learning for speech in noise after application of binary time-frequency masks. *The Journal of the Acoustical Society of America*, 133(3):1687–1692.

Almajai, I. and Milner, B. (2008). Using audio-visual features for robust voice activity detection in clean and noisy speech. In *16th European Signal Processing Conference (EUSIPCO)*, pages 1–5. IEEE.

Almajai, I. and Milner, B. (2009). Enhancing Audio Speech using Visual Speech Features. In *10th Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pages 1959–1962.

Assael, Y. M., Shillingford, B., Whiteson, S., and De Freitas, N. (2016). LipNet: End-to-End Sentence-level Lipreading. *arXiv preprint arXiv:1611.01599*.

Ba, J. L., Kiros, J. R., and Hinton, G. E. (2016). Layer Normalization. *arXiv preprint arXiv:1607.06450*.

Benesty, J., Morgan, D. R., and Sondhi, M. M. (1998). A Better Understanding and an Improved Solution to the Specific Problems of Stereophonic Acoustic Echo Cancellation. *IEEE Transactions on Speech and Audio Processing*, 6(2):156–165.

Berthommier, F. (2004). Characterization and Extraction of Mouth Opening Parameters available for Audiovisual Speech Enhancement. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, volume 3, pages 789–792. IEEE.

Boll, S. (1979). Suppression of Acoustic Noise in Speech Using Spectral Subtraction. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 27(2):113–120.

Brown, G. J. and Cooke, M. (1994). Computational auditory scene analysis. *Computer Speech and Language*, 8(4):297–336.

Brungart, D. S., Chang, P. S., Simpson, B. D., and Wang, D. (2006). Isolating the energetic component of speech-on-speech masking with ideal time-frequency segregation. *The Journal of the Acoustical Society of America*, 120(6):4007–4018.

Chen, J. and Wang, D. (2018). DNN Based Mask Estimation for Supervised Speech Separation. In *Audio Source Separation*, pages 207–235. Springer.

Chen, J., Wang, Y., and Wang, D. (2014). A Feature Study for Classification-Based Speech Separation at Low Signal-to-Noise Ratios. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 22(12):1993–2002.

Chen, J., Wang, Y., and Wang, D. (2016). Noise perturbation for supervised speech separation. *Speech Communication*, 78:1–10.

Chen, S., Mulgrew, B., and Grant, P. M. (1993). A Clustering Technique for Digital Communications Channel Equalization Using Radial Basis Function Networks. *IEEE Transactions on Neural Networks*, 4(4):570–590.

Cho, K., Van Merriënboer, B., Bahdanau, D., and Bengio, Y. (2014). On the Properties of Neural Machine Translation: Encoder-Decoder Approaches. *arXiv preprint arXiv:1409.1259*.

Chung, J., Gulcehre, C., Cho, K., and Bengio, Y. (2014). Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. *arXiv preprint arXiv:1412.3555*.

Chung, J. S. and Zisserman, A. (2016). Lip Reading in the Wild. In *Asian Conference on Computer Vision*, pages 87–103. Springer.

Cooke, M., Barker, J., Cunningham, S., and Shao, X. (2006). An audio-visual corpus for speech perception and automatic speech recognition. *The Journal of the Acoustical Society of America*, 120(5):2421–2424.

Cootes, T. F., Edwards, G. J., and Taylor, C. J. (2001). Active Appearance Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(6):681–685.

Dean, D. and Sridharan, S. (2010). Dynamic visual features for audio-visual speaker verification. *Computer Speech and Language*, 24(2):136–149.

Dendrinos, M., Bakamidis, S., and Carayannis, G. (1991). Speech enhancement from noise: A regenerative approach. *Speech Communication*, 10(1):45–57.

Dieleman, S., Schlüter, J., Raffel, C., Olson, E., Sønderby, S. K., Nouri, D., Maturana, D., Thoma, M., Battenberg, E., Kelly, J., Fauw, J. D., Heilman, M., de Almeida, D. M., McFee, B., Weideman, H., Takács, G., de Rivaz, P., Crall, J., Sanders, G., Rasul, K., Liu, C., French, G., and Degrave, J. (2015). Lasagne: First release.

Ephraim, Y. and Malah, D. (1984). Speech Enhancement Using a Minimum Mean-Square Error Short-Time Spectral Amplitude Estimator. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 32(6):1109–1121.

Ephraim, Y. and Van Trees, H. L. (1995). A Signal Subspace Approach for Speech Enhancement. *IEEE Transactions on Speech and Audio Processing*, 3(4):251–266.

Erdogan, H., Hershey, J. R., Watanabe, S., and Le Roux, J. (2015). Phase-sensitive and recognition-boosted speech separation using deep recurrent neural networks. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 708–712. IEEE.

ETSI (2002). Speech Processing, Transmission and Quality Aspects (STQ); Distributed speech recognition; Advanced front-end feature extraction algorithm; Compression algorithms. ES 202 050 version 1.1.1, ETSI STQ-Aurora DSR Working Group.

Fan, Y., Qian, Y., Xie, F.-L., and Soong, F. K. (2014). TTS Synthesis with Bidirectional LSTM based Recurrent Neural Networks. In *15th Annual Conference of the International Speech Communication Association (INTERSPEECH)*.

Fox, N. and Reilly, R. B. (2003). Audio-Visual Speaker Identification Based on the Use of Dynamic Audio and Visual Features. In *International Conference on Audio- and Video-Based Biometric Person Authentication*, pages 743–751. Springer.

Furui, S. (1986). Speaker-Independent Isolated Word Recognition Using Dynamic Features of Speech Spectrum. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 34(1):52–59.

Gabor, D. (1946). Theory of communication. Part 1: The analysis of information. *Journal of the Institution of Electrical Engineers-Part III: Radio and Communication Engineering*, 93(26):429–441.

Garofolo, J. S., Lamel, L. F., Fisher, W. M., Fiscus, J. G., and Pallett, D. S. (1993). DARPA TIMIT acoustic-phonetic continuous speech corpus CD-ROM. NIST speech disc 1-1.1. *NASA STI/Recon technical report*, 93.

Gers, F. A. and Schmidhuber, J. (2000). Recurrent Nets that Time and Count. In *IEEE-INNS-ENNS International Joint Conference on Neural Networks*, volume 3, pages 189–194. IEEE.

Ghosh, M. and Sharma, D. (1963). Power of Tukey's Test for Non-Additivity. *Journal of the Royal Statistical Society: Series B (Methodological)*, 25(1):213–219.

Girin, L., Schwartz, J.-L., and Feng, G. (2001). Audio-visual enhancement of speech in noise. *The Journal of the Acoustical Society of America*, 109(6):3007–3020.

Gogate, M., Adeel, A., Marxer, R., Barker, J., and Hussain, A. (2018). DNN Driven Speaker Independent Audio-Visual Mask Estimation for Speech Separation. *19th Annual Conference of the Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pages 2723–2727.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT press.

Goodfellow, I. J., Warde-Farley, D., Mirza, M., Courville, A., and Bengio, Y. (2013). Maxout Networks. *arXiv preprint arXiv:1302.4389*.

Graves, A. (2013). Generating Sequences With Recurrent Neural Networks. *arXiv preprint arXiv:1308.0850*.

Graves, A. and Jaitly, N. (2014). Towards End-to-End Speech Recognition with Recurrent Neural Networks. In *International Conference on Machine Learning (ICML)*, pages 1764–1772.

Graves, A., Jaitly, N., and Mohamed, A. (2013a). Hybrid speech recognition with deep bidirectional LSTM. In *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pages 273–278. IEEE.

Graves, A., Mohamed, A.-r., and Hinton, G. (2013b). Speech recognition with deep recurrent neural networks. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6645–6649. IEEE.

Graves, A. and Schmidhuber, J. (2005). Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks*, 18(5-6):602–610.

Greff, K., Srivastava, R. K., Koutník, J., Steunebrink, B. R., and Schmidhuber, J. (2017). LSTM: A Search Space Odyssey. *IEEE Transactions on Neural Networks and Learning Systems*, 28(10):2222–2232.

Han, K. and Wang, D. (2012). A classification based approach to speech segregation. *The Journal of the Acoustical Society of America*, 132(5):3475–3483.

Hanson, B. and Applebaum, T. (1990). Robust speaker-independent word recognition using static, dynamic and acceleration features: Experiments with Lombard and noisy speech. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 857–860. IEEE.

Harding, P. and Milner, B. (2012). Enhancing Speech by Reconstruction from Robust Acoustic Features. In *13th Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pages 709–712.

Harding, P. and Milner, B. (2015). Reconstruction-based speech enhancement from robust acoustic features. *Speech Communication*, 75:62–75.

Harte, N. and Gillen, E. (2015). TCD-TIMIT: An Audio-Visual Corpus of Continuous Speech. *IEEE Transactions on Multimedia*, 17(5):603–615.

He, K., Zhang, X., Ren, S., and Sun, J. (2016a). Deep Residual Learning for Image Recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778.

He, K., Zhang, X., Ren, S., and Sun, J. (2016b). Identity Mappings in Deep Residual Networks. In *European Conference on Computer Vision (ECCV)*, pages 630–645. Springer.

Healy, E. W., Delfarah, M., Vasko, J. L., Carter, B. L., and Wang, D. (2017). An algorithm to increase intelligibility for hearing-impaired listeners in the presence of a competing talker. *The Journal of the Acoustical Society of America*, 141(6):4230–4239.

Healy, E. W., Yoho, S. E., Chen, J., Wang, Y., and Wang, D. (2015). An algorithm to increase speech intelligibility for hearing-impaired listeners in novel segments of the same noise type. *The Journal of the Acoustical Society of America*, 138(3):1660–1669.

Healy, E. W., Yoho, S. E., Wang, Y., and Wang, D. (2013). An algorithm to improve speech recognition in noise for hearing-impaired listeners. *The Journal of the Acoustical Society of America*, 134(4):3029–3038.

Heckmann, M., Berthommier, F., and Kroschel, K. (2002). Noise Adaptive Stream Weighting in Audio-Visual Speech Recognition. *EURASIP Journal on Applied Signal Processing*, 2002(1):1260–1273.

Hermansky, H. and Morgan, N. (1994). RASTA Processing of Speech. *IEEE Transactions on Speech and Audio Processing*, 2(4):578–589.

Holube, I. and Kollmeier, B. (1996). Speech intelligibility prediction in hearing-impaired listeners based on a psychoacoustically motivated perception model. *The Journal of the Acoustical Society of America*, 100(3):1703–1716.

Howell, D., Cox, S., and Theobald, B. (2016). Visual units and confusion modelling for automatic lip-reading. *Image and Vision Computing*, 51:1–12.

Hu, G. and Wang, D. (2004). Monaural Speech Segregation Based on Pitch Tracking and Amplitude Modulation. *IEEE Transactions on Neural Networks*, 15(5):1135–1150.

Hu, Y. and Loizou, P. (2007). Subjective comparison and evaluation of speech communication algorithms. *Speech Communication*, 49(7–8):588–601.

Hu, Y. and Loizou, P. (2008). Evaluation of Objective Quality Measures for Speech Enhancement. *IEEE Transactions on Audio, Speech and Language Processing*, 16(1):229–238.

Ikeda, S. and Sugiyama, A. (1999). An Adaptive Noise Canceller with Low Signal Distortion for Speech Codecs. *IEEE Transactions on Signal Processing*, 47(3):665–674.

Ioffe, S. and Szegedy, C. (2015). Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In *International Conference on Machine Learning (ICML)*, pages 448–456.

Jensen, J. and Taal, C. H. (2016). An Algorithm for Predicting the Intelligibility of Speech Masked by Modulated Noise Maskers. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 24(11):2009–2022.

Jin, Z. and Wang, D. (2009). A Supervised Learning Approach to Monaural Segregation of Reverberant Speech. *IEEE Transactions on Audio, Speech, and Language Processing*, 17(4):625–638.

Jozefowicz, R., Zaremba, W., and Sutskever, I. (2015). An Empirical Exploration of Recurrent Network Architectures. In *International Conference on Machine Learning (ICML)*, pages 2342–2350.

Kates, J. and Arehart, K. (2005). Coherence and the speech intelligibility index. *The Journal of the Acoustical Society of America*, 117(4):2224–2237.

Kato, A. and Milner, B. (2016). HMM-based speech enhancement using sub-word models and noise adaptation. *17th Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pages 3748–3752.

Khan, F. and Milner, B. (2013). Speaker Separation using Visually-Derived Binary Masks. In *12th International Conference on Auditory-Visual Speech Processing (AVSP)*.

Khan, F. and Milner, B. (2015). Using Audio and Visual Information for Single Channel Speaker Separation. In *16th Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pages 1517–1521.

Khan, F., Milner, B. P., and Le Cornu, T. (2018). Using Visual Speech Information in Masking Methods for Audio Speaker Separation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 26(10):1742–1754.

Kim, G., Lu, Y., Hu, Y., and Loizou, P. C. (2009). An algorithm that improves speech intelligibility in noise for normal-hearing listeners. *The Journal of the Acoustical Society of America*, 126(3):1486–1494.

Kingma, D. P. and Ba, J. (2014). Adam: A Method for Stochastic Optimization. *arXiv preprint arXiv:1412.6980*.

Kiros, R., Salakhutdinov, R., and Zemel, R. S. (2014). Unifying Visual-Semantic Embeddings with Multimodal Neural Language Models. *arXiv preprint arXiv:1411.2539*.

Kollmeier, B. and Koch, R. (1994). Speech enhancement based on physiological and psychoacoustical models of modulation perception and binaural interaction. *The Journal of the Acoustical Society of America*, 95(3):1593–1602.

Krizhevsky, A. and Hinton, G. (2009). Learning Multiple Layers of Features from Tiny Images. Technical report, Citeseer.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 1097–1105.

Lan, Y., Harvey, R., Theobald, B., Ong, E.-J., and Bowden, R. (2009). Comparing Visual Features for Lipreading. In *8th International Conference on Auditory-Visual Speech Processing (AVSP)*, pages 102–106.

Lan, Y., Theobald, B.-J., Harvey, R., Ong, E.-J., and Bowden, R. (2010). Improving Visual Features for Lip-reading. In *9th International Conference on Auditory-Visual Speech Processing (AVSP)*, pages 7–3.

Le Cornu, T. and Milner, B. (2015). Voicing classification of visual speech using convolutional neural networks. In *1st International Joint Conference on Facial Analysis, Animation and Auditory-Visual Speech Processing (FAAVSP)*, pages 103–108.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-Based Learning Applied to Document Recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

Li, N. and Loizou, P. C. (2008). Factors influencing intelligibility of ideal binary-masked speech: Implications for noise reduction. *The Journal of the Acoustical Society of America*, 123(3):1673–1682.

Liang, L., Liu, X., Zhao, Y., Pi, X., and Nefian, A. V. (2002). Speaker independent audio-visual continuous speech recognition. In *IEEE International Conference on Multimedia and Expo (ICME)*, volume 2, pages 25–28. IEEE.

Lim, J. and Oppenheim, A. (1978). All-Pole Modeling of Degraded Speech. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 26(3):197–210.

Lim, J. S. and Oppenheim, A. V. (1979). Enhancement and Bandwidth Compression of Noisy Speech. *Proceedings of the IEEE*, 67(12):1586–1604.

Liu, Q., Aubrey, A. J., and Wang, W. (2014). Interference Reduction in Reverberant Speech Separation With Visual Voice Activity Detection. *IEEE Transactions on Multimedia*, 16(6):1610–1623.

Liu, Q., Wang, W., Jackson, P. J., Barnard, M., Kittler, J., and Chambers, J. (2013). Source Separation of Convolutive and Noisy Mixtures Using Audio-Visual Dictionary Learning and Probabilistic Time-Frequency Masking. *IEEE Transactions on Signal Processing*, 61(22):5520–5535.

Liu, Q., Wang, W., Jackson, P. J., and Tang, Y. (2017). A Perceptually-Weighted Deep Neural Network for Monaural Speech Enhancement in Various Background Noise Conditions. In *25th European Signal Processing Conference (EUSIPCO)*, pages 1270–1274. IEEE.

Loizou, P. and Kim, G. (2011). Reasons why Current Speech-Enhancement Algorithms do not Improve Speech Intelligibility and Suggested Solutions. *IEEE Transactions on Audio, Speech and Language Processing*, 19(1):47–56.

Ma, J., Hu, Y., and Loizou, P. (2009). Objective measures for predicting speech intelligibility in noisy conditions based on new band-importance functions. *The Journal of the Acoustical Society of America*, 125(5):3387–3405.

Maas, A. L., Hannun, A. Y., and Ng, A. Y. (2013). Rectifier Nonlinearities Improve Neural Network Acoustic Models. In *International Conference on Machine Learning (ICML)*, volume 30, pages 1–6.

Matthews, I., Cootes, T. F., Bangham, J. A., Cox, S., and Harvey, R. (2002). Extraction of Visual Features for Lipreading. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(2):198–213.

Murphy, Kevin, P. (2012). *Machine Learning, A Probabilistic Perspective*. MIT Press.

Narayanan, A. and Wang, D. (2013). Ideal ratio mask estimation using deep neural networks for robust speech recognition. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7092–7096. IEEE.

Noda, K., Yamaguchi, Y., Nakadai, K., Okuno, H. G., and Ogata, T. (2014). Lipreading using Convolutional Neural Network. In *15th Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pages 1149–1153.

Patterson, R., Nimmo-Smith, I., Holdsworth, J., and Rice, P. (1987). An efficient auditory filterbank based on the gammatone function. In *Meeting of the IOC Speech Group on Auditory Modelling at RSRE*, volume 2, pages 1–33.

Perkins, C., Hodson, O., and Hardman, V. (1998). A Survey of Packet loss Recovery Techniques for Streaming Audio. *IEEE Network*, 12(5):40–48.

Potamianos, G., Neti, C., Gravier, G., Garg, A., and Senior, A. (2003). Recent Advances in the Automatic Recognition of Audiovisual Speech. In *Proceedings of the IEEE*, volume 91, pages 1306–1326.

Potamianos, G., Neti, C., Luettin, J., and Matthews, I. (2004). Audio-Visual Automatic Speech Recognition: An Overview. In *Issues in Visual and Audio-visual Speech Processing*. MIT Press.

Prechelt, L. (1998). Early Stopping – but when? In *Neural Networks: Tricks of the trade*, pages 55–69. Springer.

Price, P., Fisher, W. M., Bernstein, J., and Pallett, D. S. (1988). The DARPA 1000-Word Resource Management Database for Continuous Speech Recognition. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 651–654. IEEE.

Rappaport, T. S. et al. (1996). *Wireless Communications: Principles and Practice*, volume 2. Prentice Hall PTR New Jersey.

Rivet, B., Girin, L., and Jutten, C. (2007). Mixing Audiovisual Speech Processing and Blind Source Separation for the Extraction of Speech Signals From Convolutive Mixtures. *IEEE Transactions on Audio, Speech, and Language Processing*, 15(1):96–108.

Rivet, B., Wang, W., Naqvi, S. M., and Chambers, J. A. (2014). Audiovisual Speech Source Separation: An overview of key methodologies. *IEEE Signal Processing Magazine*, 31(3):125–134.

Rix, R., Beerands, J., Hollier, M., and Hekstra, A. (2001). Perceptual evaluation of speech quality (PESQ) – a new method for speech quality assessment telephone networks and codecs. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 749–752. IEEE.

Rubinstein, R. Y. and Kroese, D. P. (2013). *The cross-entropy method: a unified approach to combinatorial optimization, Monte-Carlo simulation and machine learning*. Springer Science and Business Media.

Simonyan, K. and Zisserman, A. (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv preprint arXiv:1409.1556*.

Sodoyer, D., Rivet, B., Girin, L., Schwartz, J.-L., and Jutten, C. (2006). An analysis of visual speech information applied to voice activity detection. In *IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP)*, volume 1, pages 601–604. IEEE.

Srivastava, N., Hinton, G. E., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2015). Going Deeper with Convolutions. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–9.

Tchorz, J. and Kollmeier, B. (2003). SNR Estimation Based on Amplitude Modulation Analysis With Applications to Noise Suppression. *IEEE Transactions on Speech and Audio Processing*, 11(3):184–192.

Thangthai, K., Bear, H. L., and Harvey, R. (2018). Comparing phonemes and visemes with DNN-based lipreading. *arXiv preprint arXiv:1805.02924*.

Thangthai, K., Harvey, R. W., Cox, S. J., and Theobald, B.-J. (2015). Improving Lip-reading Performance for Robust Audiovisual Speech Recognition using DNNs. In *1st International Joint Conference on Facial Analysis, Animation and Auditory-Visual Speech Processing (FAAVSP)*, pages 127–131.

Theano Development Team (2016). Theano: A Python framework for fast computation of mathematical expressions. *arXiv preprint arXiv:1605.02688*.

Tompson, J., Goroshin, R., Jain, A., LeCun, Y., and Bregler, C. (2015). Efficient Object Localization Using Convolutional Networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 648–656.

Tukey, J. W. (1949). Comparing Individual Means in the Analysis of Variance. *Biometrics*, 5(2):99–114.

Valentini-Botinhao, C., Wu, Z., and King, S. (2015). Towards Minimum Perceptual Error Training for DNN-Based Speech Synthesis. In *16th Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pages 869–873.

Van Den Oord, A., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., Kalchbrenner, N., Senior, A. W., and Kavukcuoglu, K. (2016). WaveNet: A Generative Model for Raw Audio. In *SSW*, pages 1–15.

Vinyals, O., Kaiser, Ł., Koo, T., Petrov, S., Sutskever, I., and Hinton, G. (2015a). Grammar as a Foreign Language. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 2773–2781.

Vinyals, O., Toshev, A., Bengio, S., and Erhan, D. (2015b). Show and Tell: A Neural Image Caption Generator. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3156–3164. IEEE.

Viola, P. and Jones, M. (2001). Rapid Object Detection Using a Boosted Cascade of Simple Features. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 1–9. IEEE.

Wang, D. and Brown, G. J. (2006). *Computational Auditory Scene Analysis: Principles, Algorithms, and Applications*. Wiley-IEEE Press.

Wang, D., Kjems, U., Pedersen, M. S., Boldt, J. B., and Lunner, T. (2009). Speech Intelligibility in Background Noise with Ideal Binary Time-Frequency Masking. *The Journal of the Acoustical Society of America*, 125(4):2336–2347.

Wang, W., Cosker, D., Hicks, Y., Saneit, S., and Chambers, J. (2005). Video assisted speech source separation. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume 5, pages 425–428. IEEE.

Wang, Y., Narayanan, A., and Wang, D. (2014). On Training Targets for Supervised Speech Separation. *IEEE/ACM Transactions on Audio, Speech and Language Processing*, 22(12):1849–1858.

Wang, Y. and Wang, D. (2013). Towards Scaling Up Classification-Based Speech Separation. *IEEE Transactions on Audio, Speech, and Language Processing*, 21(7):1381–1390.

Websdale, D., Le Cornu, T., and Milner, B. (2015). Objective Measures for Predicting the Intelligibility of Spectrally Smoothed Speech with Artificial Excitation. In *16th Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pages 638–642.

Websdale, D. and Milner, B. (2015). Analysing the importance of different visual feature coefficients. In *1st International Joint Conference on Facial Analysis, Animation and Auditory-Visual Speech Processing (FAAVSP)*, pages 137–142.

Websdale, D. and Milner, B. (2017a). A comparison of perceptually motivated loss functions for binary mask estimation in speech separation. *18th Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pages 2003–2007.

Websdale, D. and Milner, B. (2017b). Using visual speech information and perceptually motivated loss functions for binary mask estimation. *14th International Conference on Auditory-Visual Speech Processing (AVSP)*, pages 41–46.

Websdale, D., Taylor, S., and Milner, B. (2018). The Effect of Real-Time Constraints on Automatic Speech Animation. *19th Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pages 2479–2483.

Weintraub, M. (1985). *A Theory and Computational Model of Auditory Monaural Sound Separation.* PhD thesis, Stanford University.

Weiss, M., Aschkenasy, E., and Parsons, T. (1975). *Study and Development of the INTEL Technique for Improving Speech Intelligibility.* Technical report, DTIC.

Weninger, F., Hershey, J. R., Le Roux, J., and Schuller, B. (2014). Discriminatively Trained Recurrent Neural Networks for Single-Channel Speech Separation. In *IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pages 577–581. IEEE.

Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., Zemel, R., and Bengio, Y. (2015a). Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. In *International Conference on Machine Learning (ICML)*, pages 2048–2057.

Xu, Y., Du, J., Dai, L.-R., and Lee, C.-H. (2015b). A Regression Approach to Speech Enhancement Based on Deep Neural Networks. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 23(1):7–19.

Yu, Y., Wang, W., and Han, P. (2016). Localization based stereo speech source separation using probabilistic time-frequency masking and deep neural networks. *EURASIP Journal on Audio, Speech, and Music Processing*, 7(1):1–18.

Zhao, Y., Wang, D., Merks, I., and Zhang, T. (2016). DNN-based enhancement of noisy and reverberant speech. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6525–6529. IEEE.

Zhou, Y. and Chellappa, R. (1988). Computation of Optical Flow Using A Neural Network. In *IEEE International Conference on Neural Networks*, volume 2, pages 71–78.