Mask-based enhancement of very noisy speech

Leo Lightburn

Communication and Signal Processing Group
Department of Electrical & Electronic Engineering
Imperial College London

A thesis submitted for the degree of
Doctor of Philosophy
November 2019

Copyright Declaration

Statement of Originality

I hereby certify that this thesis is the outcome of the research conducted by myself under supervision from Mike Brookes in the Department of Electrical and Electronic Engineering at Imperial College London. Any work that has been previously published and included in this thesis has been fully acknowledged in accordance with the standard referencing practices of this discipline. I declare that this thesis has not been submitted for any degree at any other University or Institution.

Abstract

When speech is contaminated by high levels of additive noise, both its perceptual quality and its intelligibility are reduced. Studies show that conventional approaches to speech enhancement are able to improve quality but not intelligibility. However, in recent years, algorithms that estimate a time-frequency mask from noisy speech using a supervised machine learning approach and then apply this mask to the noisy speech have been shown to be capable of improving intelligibility.

The most direct way of measuring intelligibility is to carry out listening tests with human test subjects. However, in situations where listening tests are impractical and where some additional uncertainty in the results is permissible, for example during the development phase of a speech enhancer, intrusive intelligibility metrics can provide an alternative to listening tests. This thesis begins by outlining a new intrusive intelligibility metric, Weighted-STOI (WSTOI), that is a development of the existing Short-Time Objective Intelligibility Measure (STOI) metric. WSTOI improves STOI by weighting the intelligibility contributions of different time-frequency regions with an estimate of their intelligibility content. The prediction accuracies of WSTOI and STOI are compared for a range of noises and noise suppression algorithms and it is found that WSTOI outperforms STOI in all tested conditions.

The thesis then investigates the best choice of mask-estimation algorithm, target mask, and method of applying the estimated mask. A new target mask, the High-resolution SWOBM (HSWOBM), is proposed that optimises a modified version of WSTOI with a higher frequency resolution. The HSWOBM is optimised for a stochastic noise signal to encourage a mask estimator trained on the HSWOBM to generalise better to unseen noise conditions. A high frequency resolution version of WSTOI is optimised as this gives improvements in predicted quality compared with optimising WSTOI. Of the tested approaches to target mask estimation, the best-performing approach uses a feed-forward neural network with a loss function based on WSTOI. The best-performing feature set is based on the gains produced by a classical speech enhancer and an estimate of the local voiced-speech-plus-noise to noise ratio in different time-frequency regions, which is obtained with the aid of a pitch estimator.

When the estimated target mask is applied in the conventional way, by multiplying the speech by the mask in the time-frequency domain, it can result in speech with very poor perceptual quality. The final chapter of this thesis therefore investigates alternative approaches to applying the estimated mask to the noisy speech, in order to improve both intelligibility and quality. An approach is developed that uses the mask to supply prior information about the speech presence probability to a classical speech enhancer that minimises the expected squared error in the log spectral amplitudes. The proposed end-to-end enhancer outperforms existing algorithms in terms of predicted quality and intelligibility for most noise types.

Acknowledgements

First and foremost, I would like to thank my supervisor, Mike Brookes, for his continual encouragement, his careful guidance, and the countless insights he gave me into both my own research and the broader field.

I would also like to thank Patrick Naylor, Alastair Moore and Enzo De Sena, who directly contributed to this thesis, in particular to Chapter 6, through their ideas and advice, and to my examiners, Ben Milner and Pier Luigi Dragotti, whose input helped to strengthen this work.

Finally, I would like to thank my parents, for their unconditional support, and Steph, for her fortitude, and her patience.

Contents

List of Figures

List of Tables

List of Acronyms

AI	Articulation Index	64
AMS	Amplitude Modulation Spectrogram	53
AR	Autoregressive	37
BPTT	Back-Propagation Through Time	147
CART	Classification and Regression Tree	54
CASA	Computational Auditory Scene Analysis	44
CFR	Comb Filter Ratio	52
CMA	Constant Modulus Algorithm	205
CSE	Cochlea Scaled Entropy	74
CIRM	Complex Ideal Ratio Mask	48
CHSWOBM	Compact HSWOBM	8
CMA	Conventional Mask Application	205
CMA-CS	Conventional Mask Application with Cepstral Smoothing	205
CMA-MG	Conventional Mask Application with a Minimum Gain	205

DBSTOI	Deterministic Binaural STOI	102
DCT	Discrete Cosine Transform	53
DFT	Discrete Fourier Transform	131
DNN	Deep Neural Network	50
DSOBM	Deterministic STOI-Optimal Binary Mask	93
ERB	Equivalent Rectangular Bandwidth	38
ESTOI	Extended STOI	102
GMM	Gaussian Mixture Model	37
HMM	Hidden Markov Model	37
HSWOBM	High-resolution SWOBM	3
IBM	Ideal Binary Mask	14
IRM	Ideal Ratio Mask	47
KLT	Karhunen-Loéve Transform	37
LC	Local Criterion	44
LPC	Linear Predictive Coding	83
LSA	Log Spectral Amplitude	194
LSTM	Long Short-Term Memory	138
LSTM-RNN	Long Short-Term Memory Recurrent Neural Network	54
LTASS	Long Term Average Speech Spectrum	47
MFCC	Mel-frequency Cepstral Coefficient	53

MMSE	Minimum Mean Squared Error	38
MMSE-MA	Minimum Mean Squared Error Mask Application	203
MOS	Mean Opinion Score	34
OM-LSA	Optimally-Modified Log-Spectral Amplitude	198
PDF	Probability Density Function	73
PEFAC	Pitch Estimation Filter with Amplitude Compression	52
PESQ	Perceptual Evaluation of Speech Quality	62
RAPT	Robust Algorithm for Pitch Tracking	52
RC	Relative Criterion	47
ReLU	Rectified Linear Unit	143
RNN	Recurrent Neural Network	57
sEPSM	speech-based Envelope Power Spectrum Model	64
SII	Speech Intelligibility Index	64
SNR	Signal-to-Noise Ratio	35
SOBM	STOI-Optimal Binary Mask	92
SPP	Speech Presence Probability	194
SS	Spectral Subtraction	38
SSOBM	Stochastic STOI-Optimal Binary Mask	101
STFT	Short Time Fourier Transform	38
STI	Speech Transmission Index	64

STOI	Short-Time Objective Intelligibility Measure	3
SHSWOBM	Smoothed HSWOBM	8
SWOBM	Stochastic WSTOI-Optimal Binary Mask	110
WSTOI	Weighted-STOI	3
SVM	Support Vector Machine	54
TBM	Target Binary Mask	14
TBPTT	Truncated Back-Propagation Through Time	148
TF	Time-Frequency	38
TFGM	Time-Frequency Gain Modification	36
UTBM	Universal Target Binary Mask	47
VSNNR	Voiced-Speech-Plus-Noise to Noise Ratio	52
WOBM	WSTOI-Optimal Binary Mask	51

List of Publications

The following publications were produced during the course of this work:

1. L. Lightburn and M. Brookes. SOBM - a binary mask for noisy speech that optimises an objective intelligibility metric. In *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, 2015.

2. L. Lightburn and M. Brookes. A weighted STOI intelligibility metric based on mutual information. In *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, 2016.

3. L. Lightburn, E. D. Sena, A. Moore, P. A. Naylor, and M. Brookes. Improving the perceptual quality of ideal binary masked speech. In *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, 2017.

4. A. Moore, L. Lightburn, W. Xue, P. Naylor, and M. Brookes. Binaural mask-informed speech enhancement for hearing aids with head tracking. In *Proc. Intl Wkshp Acoustic Signal Enhancement (IWAENC)*, 2018.

Chapter 1

Introduction

Recent decades have seen a proliferation of devices and technologies which aid speech-based communication, including mobile phones, hearing aids and video telephony systems. In these technologies the path between the acoustic signal from the speaker and the output of the loudspeaker can be represented by a signal flow diagram such as Fig. 1.1, which shows a typical speech transmission system. The desired speech signal passes from the speaker through a convolutive acoustic channel before being transduced by the microphone. The signal is then amplified and passes through an electronic channel before arriving at a loudspeaker. Degradations to the speech signal may be introduced at any point in this transmission chain, and can be categorised according to their effect on the speech. For example, degradations may be caused by interfering additive noise signals which are uncorrelated with the desired speech. In Fig. 1.1 the additive noises are introduced in the acoustic domain, though this form of distortion may also be introduced by the electronic channel. Convolutive distortion, which is perceived as colouration and/or reverberation,

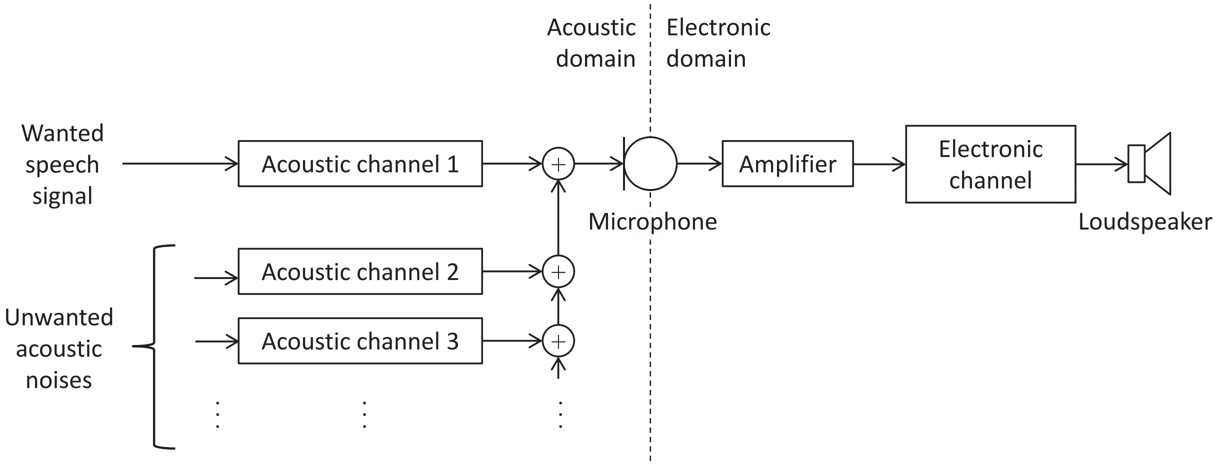


Figure 1.1: Signal flow diagram of a typical speech transmission system.

is typically caused by multiple acoustic reflections of a speech signal arriving at the microphone and results in an interfering signal which is strongly correlated with the desired speech signal. Other forms of distortion are non-linear, such as amplitude limiting or clipping, which may be introduced by a microphone or Coder-Decoder (CODEC).

1.1 Speech quality and intelligibility

The effect of signal degradations on a listener's perception of a speech utterance is to impair both the quality and intelligibility of the speech. The quality or acceptability of speech is highly subjective and encompasses characteristics like 'naturalness' and 'pleasantness' which depend on factors such as the level of background noise and the degree of distortion to the underlying speech signal. It is typically evaluated in terms of a Mean Opinion Score (MOS) which is obtained by asking a group of trained listeners to rate the quality of the speech signal on a scale of 1 (Bad) to 5 (Excellent), and then averaging the responses

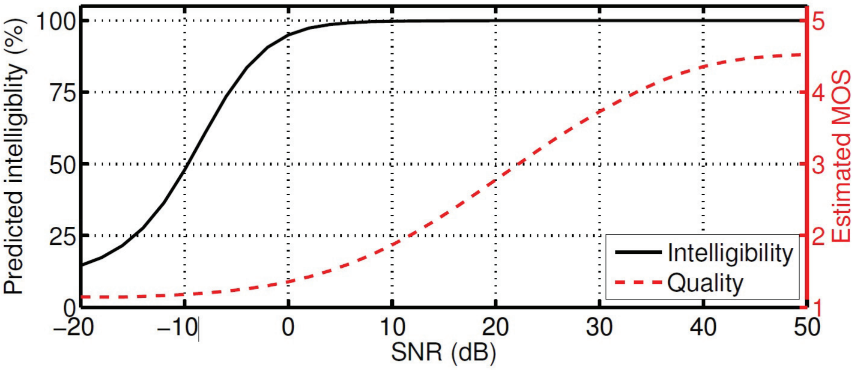


Figure 1.2: Average estimated mean opinion score and predicted intelligibility of more than 100 files of the test set of the TIMIT database [45], where the utterances have been corrupted with white noise at different SNR levels. This figure was taken from [53].

of the listeners [83]. In addition to providing an overall quality rating, listeners may sometimes be asked to separately rate the impact of speech distortion and background noise on the speech quality [86]. The intelligibility of a speech utterance is defined as the percentage of content words in the utterance that a listener is able to correctly identify. Both the quality and the intelligibility of a degraded speech signal depend on the type of degradation. For example, speech intelligibility is resilient to clipping but may be severely damaged by reverberation. The severity of the degradation is also important; low to moderate levels of additive noise can result in speech with unpleasant characteristics but full intelligibility, whilst at very low Signal-to-Noise Ratios (SNRs) both quality and intelligibility are likely to be affected. The relationship between SNR and intelligibility also depends on the nature of the speech and noise. For example, some noises are more detrimental to intelligibility than others. Also, when the vocabulary and/or grammar of the speech is constrained in some way known to the listener, intelligibility will normally increase [3].

Fig. 1.2, taken from [53], shows the estimated MOS and predicted intelligibility of speech signals corrupted with white noise at different SNRs. Conventional speech enhancement algorithms, such as the Time-Frequency Gain Modification (TFGM) methods, subspace methods and model-based methods discussed in Sec. 1.2, are most effective in the region above 0 dB SNR in Fig. 1.2 where, although the speech may have poor quality, it is fully intelligible. The aim of the speech enhancer in this region is to improve the quality of the speech without damaging its intelligibility. The research presented in this thesis, however, focuses on SNRs below 0 dB, where the intelligibility of the speech has been significantly degraded by very high levels of noise. In this range of SNRs, conventional speech enhancement algorithms are usually inadequate; of the studies which have addressed the effects of speech enhancement techniques on intelligibility [13, 157, 7, 76, 111, 113], most have found that noise suppression either had very little positive effect, or had a detrimental effect, on intelligibility.

1.2 Speech enhancement methods

Many techniques for enhancing speech containing uncorrelated additive noise have been proposed in the literature (see [16] for a more complete overview). Whilst a minority of algorithms work in the time-domain, the majority, including subspace enhancement methods and time-frequency domain methods, perform the enhancement in a transform domain where the speech and noise are relatively sparse and can thus be more easily separated.

1.2.1 Subspace enhancement

Subspace enhancement methods, e.g. [39], use the Karhunen-Loéve Transform (KLT) to concentrate the signal energy into a small number of transform components and thereby make it easier to separate from the noise. The transform is typically applied to the noisy speech in frames with a duration of around 20 ms, within which the speech is assumed to lie within a low-order subspace [128]. This is equivalent to assuming that the speech arises from a low-order Autoregressive (AR) process that is time-invariant over intervals of this length, which is a widely used assumption based on acoustic models of the vocal tract. A possible downside of this approach is that, if the additive noise has strong tonal components, it is likely to interpret these as speech components. Using a signal-dependent transform also imposes a high computational cost.

1.2.2 Model-based enhancement

Many speech enhancement approaches use a stochastic model to incorporate prior information about the speech, and in some cases also the noise. A popular approach, e.g. [181, 49, 50], is to combine an AR speech model with a Kalman filter [92]. Several techniques involve modelling the spectral components of the speech using a Gaussian Mixture Models (GMMs), e.g. [105, 29]. In [38] the temporal evolution of the clean speech spectra is modelled using a Hidden Markov Model (HMM) in which the spectra associated with each state is modelled as a Gaussian mixture of AR processes.

1.2.3 Time-frequency domain

A popular domain for enhancement is the Time-Frequency (TF) domain, and the most common way to transform data into the TF domain is with the Short Time Fourier Transform (STFT), which is signal-independent and more computationally efficient than the KLT. The STFT has uniform spacing in the frequency domain and is exactly invertible in the absence of processing. An alternative to the STFT is to pass the signal through a time-domain filterbank (such as a gammatone filterbank [71]) and then divide the output into frames. The filterbank usually uses non-uniform frequency bands that approximately match human frequency resolution, and the TF representation is therefore sometimes referred to as a 'cochleagram'. The frequency band spacing is typically either logarithmic (e.g. third-octave bands), or based on the Equivalent Rectangular Bandwidth (ERB)-rate [119] or Bark [185] scales. Sometimes a gain is applied to the output of the filterbank to account for the mapping between sound pressure level and perceived loudness [81].

Most algorithms operating in the TF domain have a similar general structure; the signal is first transformed into the TF domain where a separate gain is then applied at each TF cell. The enhanced signal is then converted back into the time-domain. This process is known as Time-Frequency Gain Modification (TFGM). Two widely used TFGM algorithms are Spectral Subtraction (SS) [13] and model-based Minimum Mean Squared Error (MMSE) spectral or log-spectral amplitude estimation [36, 37]. These algorithms use different functions to specify the gain [162]; SS methods use an approximation to the Wiener filter which minimises the mean squared error of the complex spectral amplitudes of the clean and noisy speech signals; MMSE methods use a gain function which minimises the mean squared error of the spectral or log-spectral magnitudes. Both methods use a real-valued gain function and leave the phase spectrum uncorrected. This was justified in [168] by demonstrating that using the true phase spectrum of the clean speech signal did not result in a substantial perceptual improvement. In addition, it was shown in [36] that, with appropriate assumptions, the optimum estimator of the speech phase is the phase of the noisy signal.

TFGM methods have a lower computational cost than subspace methods but, since they are typically used in combination with noise estimation techniques which assume that the noise is stationary, they may not perform well on speech containing non-stationary noise. They can also introduce artefacts into the speech including brief tonal components which are perceived as a fluctuating "musical noise". TFGM methods, subspace methods and model-based methods all have a further important limitation: despite being able to substantially improve both the perceived quality and the SNR of noisy speech signals, they normally degrade their intelligibility [68].

1.2.4 Neural network methods

Rather than estimate the TF gain from models of speech and noise, a popular recent alternative is to estimate it directly from the noisy speech. An advantage of this approach is that it does not rely on explicit models of speech and noise. However, this approach requires extensive training and is not robust to speech or noise that differs from that used in training [21, 20, 172].

1.2.5 Binary and ratio mask-based enhancement

In early 2000's the use of a two-valued binary gain mask was proposed as a way of separating a target speaker from interfering noise [164]. The original approach was to make the mask equal to 0 or 1 in TF regions with negative and positive SNR respectively. It was postulated that estimating a binary mask might be easier and more robust than estimating a continuous-valued TF gain. Later works found that, by applying a suitably chosen oracle mask (designed with knowledge of the clean speech) even heavily degraded speech could be made fully intelligible [99].

1.3 Research goals

The goal of the research presented in this thesis is to enhance speech that has been so severely corrupted by additive acoustic noise that its intelligibility has been substantially impaired. The goal is to improve both the quality and the intelligibility of the degraded speech. In Fig. 1.2, the region in which intelligibility has been substantially impaired corresponds to SNRs in the range -20 to 0 dB. Such situations can arise either when the level of unwanted acoustic noise is very high (such as in a crowded pub or restaurant) or when the microphone is a long way from the target speaker. In the latter case, intelligibility may also be degraded by reverberation. However, this research only considers distortion caused by interfering additive noise signals. This research also assumes that only a single microphone is available. The use of microphone arrays can in some circumstances improve the SNR by creating a spatially selective beamformer. However, it is still frequently necessary to apply further enhancement to the single-channel output of the beamformer.

Motivated by [164, 99], the research presented in this thesis aims (a) to identify the binary mask that maximises speech intelligibility, (b) develop a robust technique for estimating the binary mask and (c) develop ways of applying the binary mask to the noisy speech that improve both quality and intelligibility.

1.4 Overview of thesis structure

Chapter 2 presents background material and reviews the literature related to this work. Chapter 3 presents WSTOI, a modified version of the STOI intelligibility metric [151] in which the contribution of each time frame to the metric is weighted by its estimated contribution to intelligibility. Chapter 4 presents a new oracle binary mask, the WSTOI-Optimal Binary Mask (WOBM), that explicitly maximises WSTOI, and describes several variations of this mask that could serve as a target for a mask estimation algorithm. Chapter 5 investigates techniques for estimating these binary masks from noisy speech. Chapter 6 investigates alternative methods of applying estimated binary masks to noisy speech, in order to improve both quality and intelligibility. Finally, Chapter 7 draws conclusions and suggests ways in which the work could be further extended.

Chapter 2

Background to mask-based enhancement and its evaluation

2.1 Introduction

This chapter provides a discussion of existing mask-based enhancers for very noisy speech, followed by an overview of existing methods of evaluating the performance of the enhancers, including algorithms for measuring the quality and intelligibility of the enhanced signals.

2.2 Mask-based intelligibility enhancement

The previous chapter noted that the intelligibility of speech containing additive noise becomes severely degraded when the SNR falls below 0 dB, and that it cannot normally be improved by applying conventional speech enhancement algorithms. A number of studies [5, 110, 17, 167, 98] have shown that the intelligibility of noisy speech can, however, be improved by applying a bounded two-dimensional multiplier denoted a “Time-Frequency mask” to the signal in the TF-domain. In these studies, the mask is constructed using oracle information, i.e. information about the true speech signal, and in many cases also the true noise signal. One approach (discussed in the following subsection) is to set the mask to 1 in TF regions dominated by speech energy and to a low value, such as 0 or 0.1, in TF regions dominated by noise. These studies have inspired the development of enhancement algorithms that have a structure similar to the diagram in Fig. 2.1. Features are extracted from noisy speech and used as inputs to a mask estimation algorithm. During an algorithm training phase, the internal parameters of the estimation algorithm are found by pairing feature vectors extracted from noisy speech with a target output consisting of an oracle mask, i.e. a mask that is obtained with knowledge of the clean speech. After the mask has been estimated it is applied to the noisy speech in the TF domain, and the resulting signal is then converted back to the time-domain.

In the following section, the time-domain speech, noise and noisy speech signals are denoted by x, n and y, respectively, where $y = x + n$. The complex Short Time Fourier Transform (STFT) coefficients of these signals in frequency bin k of frame m are denoted as $X(k, m)$, $N(k, m)$ and $Y(k, m)$, respectively.

2.2.1 Oracle masks

The most widely used oracle mask is the so-called IBM introduced in [164], which is a function of the instantaneous SNR in the corresponding TF cell, indexed by (k, m). The

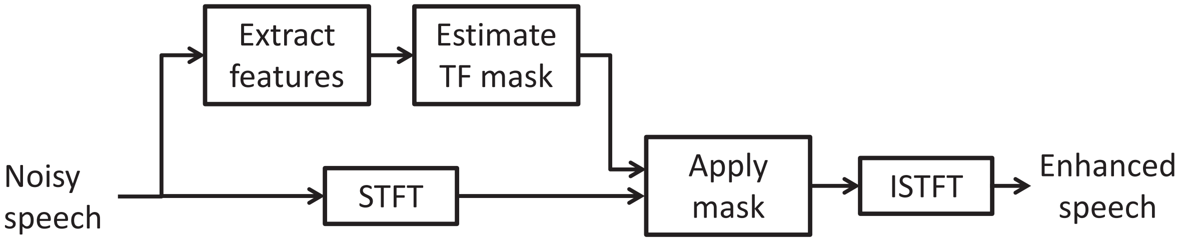


Figure 2.1: Overview of a typical mask-based enhancer.

mask is given by

$$B_{\mathrm{IBM}}(k,\,m) = \begin{cases} 1 & |X(k,\,m)|^2 > \beta\,|N(k,\,m)|^2 \\ 0 & \text{otherwise} \end{cases}. \tag{2.1}$$

The Local Criterion (LC), β, determines the SNR threshold above which the mask will equal 1. The IBM was initially proposed as a goal of Computational Auditory Scene Analysis (CASA) [166] [164], a set of techniques aiming to segregate a target signal from a mixture containing interfering sources. CASA is based on ideas from Auditory Scene Analysis (ASA), a model of human auditory perception in which an acoustic mixture is decomposed into small collections of sensory elements or segments which are then selectively grouped into streams [14]. Several studies have since shown that the IBM can provide improvements in intelligibility [17, 110], and some have suggested possible reasons for this improvement. For example, it has been suggested that by removing TF cells which are dominated by noise the IBM may direct the listener's attention onto TF cells which contain unobstructed "glimpses" of the speech signal, thereby reducing informational masking of the speech by the noise [99, 17]. Another explanation was put forward in [17] whose

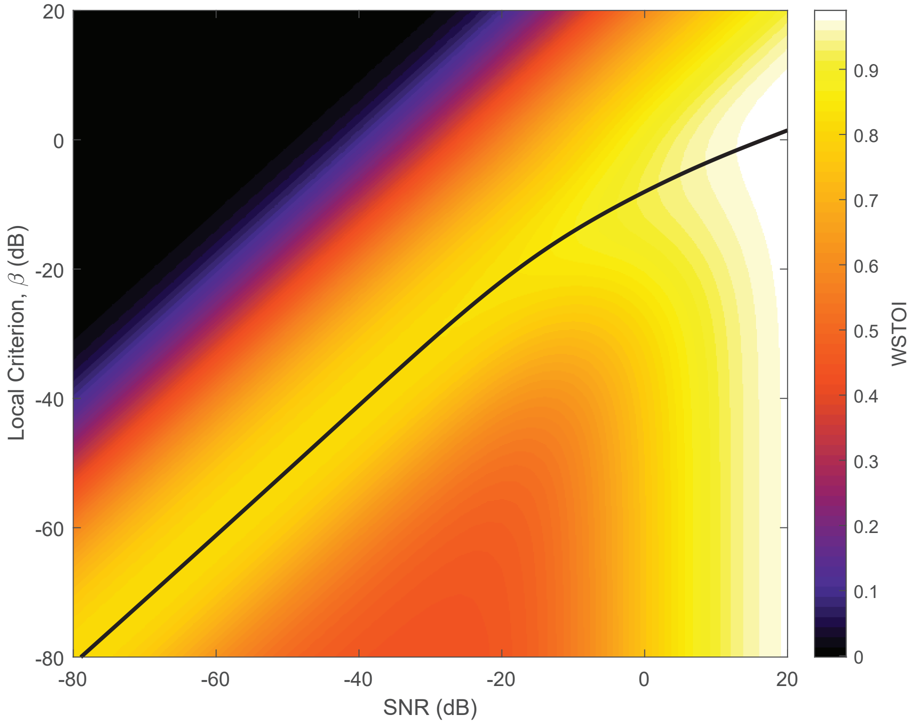


Figure 2.2: Plot showing the value of the WSTOI intelligibility metric computed on noisy speech processed with an oracle IBM, as a function of β and the SNR of the noisy speech. The noisy speech contained speech shaped noise and babble noise. The black curve shows the value of LC, β, that gives the maximum WSTOI at each SNR.

authors observed that the pattern of TF cells in the binary mask had a much greater impact on intelligibility than the underlying local SNR values of these TF cells. The authors proposed that speech perception was limited more by the listener's ability to determine the TF location of the speech energy than by the ability to extract speech information from individual TF cells.

In [99] it was suggested that the masked speech provides two independent speech cues, a noisy speech signal and a vocoded noise signal, and that it is the vocoded component that is responsible for improving the intelligibility. According to this model, the intelligibility gains obtained by applying a binary mask arise from the introduction of spectro-temporal modulation that matches the TF energy distribution of the target speech. This is demonstrated by Fig. 2.2, which shows the value of the WSTOI intelligibility metric (described in Chapter 3) computed on noisy speech processed with an oracle IBM, as a function of the LC, β, and the SNR of the noisy speech utterances. The noisy speech used to generate the plot consisted of 50 utterances from the training set of the TIMIT speech corpus [45] mixed with speech shaped noise and babble noise from the RSG.10 [139] database. It can be seen from the plot that, at every SNR, there is a value of β that results in high predicted intelligibility. The black curve shows the value of the β that gives the maximum WSTOI at each SNR, denoted as $\beta_{opt}\left(\mathrm{SNR}\right)$. At SNRs below around -15 dB, where the primary source of speech information is the vocoded noise signal, the black curve is approximately equal to a straight line with the equation

$$\beta_{opt}\left(\mathrm{SNR}\right) \approx 0.99 \cdot \mathrm{SNR} - 2.2 \approx \mathrm{SNR}. \tag{2.2}$$

Speech at an arbitrarily low SNR can therefore be made fully intelligible by setting β approximately equal to the average SNR of the utterance. If β is too high then the mask is too sparse; for extremely high β the mask is all zeros and intelligibility is 0 %. If β is too low then the mask has too many ones, and for extremely low β the mask is all ones and the masked speech is identical to the noisy speech. At SNRs above 0 dB the primary source of speech information is the noisy speech signal, which has good intelligibility before the mask is applied. In this region, the IBM can damage the otherwise good intelligibility if it is too sparse (i.e. if β is too high), hence the gradient of the line, $\mathrm{d}\beta_{opt}\left(\mathrm{SNR}\right)/\mathrm{dSNR}$, decreases with SNR.

Since, under the model proposed in [99], all of the benefit of binary masking comes from the vocoded noise component, it seems appropriate to use a mask that is based on the speech alone, i.e. one that is independent of the noise. It has also been suggested that using an oracle mask which is independent of the noise may help the classifier to generalise to noises that were not present in the training data [171], by focusing the classifier on features that are present in the speech rather than the noise. In [98] the vocoded signal component is created by the Target Binary Mask (TBM) in which the speech energy in each TF cell is compared with $\overline{X}(k)$, the average speech energy in that frequency bin. The TBM is given by

$$B_{\mathrm{TBM}}(k,\,m) = \begin{cases} 1 & X(k,\,m) > \beta'\overline{X}(k) \\ 0 & \text{otherwise} \end{cases} \tag{2.3}$$

where β', the Relative Criterion (RC), typically lies in the range ±5 dB. The Universal Target Binary Mask (UTBM) [56] eliminates the speaker-dependence of the TBM by replacing $\overline{X}(k)$ in (2.3) by $\alpha\overline{\overline{X}}(k)$ where α is the average speech power and $\overline{\overline{X}}(k)$ is a speaker-independent power-normalised Long Term Average Speech Spectrum (LTASS) [18].

More recently, algorithms that estimate a continuous-valued TF gain (still termed a "mask") have been proposed. A popular continuous-valued oracle mask is the so-called Ideal Ratio Mask (IRM) [171] which is given by

$$G_{\mathrm{IRM}}(k,\,m) = \left(\frac{|X(k,\,m)|^{\epsilon}}{|X(k,\,m)|^{\epsilon} + |N(k,\,m)|^{\epsilon}}\right)^{\nu}, \tag{2.4}$$

where ν and ϵ are parameters commonly set to 0.5 and 2, respectively. If the speech and noise are assumed to be uncorrelated stationary stochastic processes then, if $\nu = 1$ and $\epsilon = 2$ then (2.4) gives the gain of the Wiener filter, which minimises the mean square error between the true and estimated speech spectral amplitudes. If $\nu = 0.5$ and $\epsilon = 2$ then (2.4) gives the gain of the square-root Wiener filter, which gives an unbiased estimate of the power spectrum of the desired signal. The IRM has been used as the target mask in several studies [40, 174, 21]. However, since (2.4) depends on the noise, the performance of these algorithms degrades somewhat when the nature of the interfering noise differs from that used to train the neural network [79].

Several studies use masks that take account of phase of the clean and noisy speech STFT coefficients. In [40] the so-called "Phase-Sensitive" Mask (PSM) is tested, which is real-valued but depends on the phase difference between the signals,

$$G_{\mathrm{PSM}}(k,\, m) = \frac{|X(k,\, m)|}{|Y(k,\, m)|}\cos\left(\theta_Y - \theta_X\right)$$

where $\theta_Y = \arg\left(Y(k,\, m)\right)$ and $\theta_X = \arg\left(X(k,\, m)\right)$. The STFT coefficients of the PSM-processed noisy speech, $G_{\mathrm{PSM}}(k,\, m)Y(k,\, m)$, have a phase equal to the phase of the unprocessed coefficients, $Y(k,\, m)$, and a magnitude equal to the scalar projection of $\left(|X(k,\, m)| \,/\, |Y(k,\, m)|\right) Y(k,\, m)$ onto $X(k,\, m)$.

In [178], the target mask, which is complex valued, is based on the so-called Complex Ideal Ratio Mask (CIRM),

$$G_{\mathrm{CIRM}}(k,\, m) = \frac{X(k,\, m)}{Y(k,\, m)},$$

which, when applied as an oracle mask, recovers the complex STFT coefficients of the clean speech exactly. The target mask for the estimation algorithm consists of the concatenated real and imaginary parts of G_{CIRM}, which are compressed with a hyperbolic tangent function to force them into the range [0, 1], as the authors believed this made them more amenable to estimation. Since

$$G_{\mathrm{CIRM}}(k,\, m) = \frac{|X(k,\, m)|}{|Y(k,\, m)|}\cos\left(\theta\right) + i\frac{|X(k,\, m)|}{|Y(k,\, m)|}\sin\left(\theta\right),$$

where $\theta = \theta_Y - \theta_X$, then clearly $G_{\mathrm{PSM}}(k,\, m) = \Re\left\{G_{\mathrm{CIRM}}(k,\, m)\right\}$. Experiments with DNN-based mask estimators using G_{IRM}, G_{PSM} and G_{CIRM} as targets suggest that, although the quality of the processed speech, as predicted by PESQ, is higher for the methods that take account of phase (G_{PSM} and G_{CIRM}), the intelligibility predicted by STOI is similar in each case [178, 176]. This may be due to the imaginary component of the CIRM containing less predictable patterns which are more difficult to estimate [173]. Specifically, the value of $\cos\left(\theta\right)$ is predictable since it takes values close to 1 when $Y(k,\, m)$ is dominated by $X(k,\, m)$, and much less than 1 otherwise, whereas estimating the sign of $\sin\left(\theta\right)$ is more difficult. In [116] the IBM and IRM were used as target masks, but during signal reconstruction the processed STFT magnitudes were combined with an estimate of the clean speech phase. This gave small improvements in PESQ and STOI compared to applying the masks in the conventional way, as real-valued gains with the noisy phase preserved.

An alternative approach is to train a mask estimator to estimate the mask which minimises the error between the true and estimated speech signals, rather than the error between the true and estimated masks. This approach does not require the target mask to be explicitly defined. The error can be measured between STFT magnitudes as in

[174], or between complex STFT coefficients, as in [40]. In the latter study, the mask is constrained to be real-valued and in the range [0, 1]. In [179] the real-valued errors and complex-valued errors are measured separately and the real errors are weighted more heavily. The estimator in [180] minimises the error in the complex STFT coefficients with the two added constraints. First, the enhanced speech and noise must sum to original noisy speech. Second, due to the overlapping frames used to compute the STFT, the STFTs of real signals contain redundancy and the masked speech $G(k,\, m)Y(k,\, m)$ may therefore be "inconsistent", i.e. not the STFT of an actual signal. The second constraint therefore imposes consistency by performing additional inverse and forward STFT steps before computing the error signal.

An alternative to all of the above approaches is to abolish the mask entirely and directly estimate either the clean speech DFT coefficients, $X(k,\, m)$, or their magnitudes, $|X(k,\, m)|$. However, in [171] a Deep Neural Network (DNN)-based mask estimator trained to estimate various target masks was shown to outperform an identical estimator trained to estimate the DFT magnitudes of the clean speech directly. The authors suggested that, since masks are bounded, they are potentially an easier target for estimation than spectral envelopes, which are unbounded. They may also be less speaker-dependent, leading to better generalisation. Another advantage of mask-based estimators is that they have the potential to preserve components of the speech which were not explicitly detected by the mask estimator but which may nonetheless contribute to speech intelligibility and quality, such as the fine detail in the speech.

We have seen that existing oracle masks are able to improve the intelligibility of noisy speech. However, there is evidence that the intelligibility of speech depends not only on the instantaneous spectrum but also on its temporal modulation [8, 32, 97]. The intelligibility of the mask-processed speech will not therefore be maximised if the training target for the mask estimator is a mask, such as those discussed in this chapter, that depends only on the instantaneous spectrum. In Chapter 4 an alternative oracle binary mask, the WSTOI-Optimal Binary Mask (WOBM), will be presented. The WOBM explicitly maximises an objective intelligibility metric, WSTOI, that takes account of spectral modulation. WSTOI will be presented in Chapter 3 and is a development of an existing intelligibility metric called the Short-Time Objective Intelligibility Measure (STOI).

2.2.2 Features for classification

In enhancement algorithms with the structure shown in Fig. 2.1, the inputs to the mask estimator are features extracted from the frames or TF cells of the noisy speech. In several studies, e.g. [135, 56, 170], separate features are used for detecting voiced and unvoiced speech. Voiced speech, produced by vibrations of the vocal chords, is characterised by strong harmonics of the fundamental frequency, or pitch, and features for detecting voiced speech typically exploit these characteristics. For example, a number of studies use features based on pitch estimates obtained from a pitch tracker (see [16] for an overview of pitch trackers). Since most voiced speech energy is concentrated at multiples of the fundamental frequency, the pitch estimate provides an indication of which TF cells are likely to contain speech energy, and can therefore be used directly as a feature. Alternatively, it can be used as a parameter for computing a derived feature. The latter approach is adopted in [60, 170] in which the pitch detector from [89, 90], which identifies multiple pitch candidates in each time-frame and then uses a HMM to join them together into continuous pitch tracks, is used to estimate the pitch from the noisy speech. The cochleagram [166] of the speech is then computed. In each TF cell the autocorrelation function is computed at the time delay corresponding to the estimate of the pitch period in that frame. A large value of the autocorrelation function indicates the presence of a strong periodic signal component at a frequency which is a multiple of the pitch, which signifies the likely presence of voiced speech energy in that TF cell. In [135] a pitch estimate, $f_0(m)$, for each frame, m, is obtained using the Robust Algorithm for Pitch Tracking (RAPT) algorithm [155], which detects peaks in the autocorrelation function of the input signal. The pitch estimate is then used to construct a comb filter, $H_{comb}^{m}(k)$, whose centre frequencies correspond to the estimated voiced speech harmonics, i.e. multiples of $f_0(m)$. A second filter, $H_{combshift}^{m}(k)$, is identical to $H_{comb}^{m}(k)$ but with centre frequencies shifted by 0.5 $f_0(m)$. In each TF cell, the Comb Filter Ratio (CFR) is computed,

$$CFR(k,\, m) = 10\log_{10}\left(\frac{\sum_n H_{comb}^{m}(n)\, Y(n,\, m)}{\sum_n H_{combshift}^{m}(n)\, Y(n,\, m)}\right).$$

Since most voiced speech energy is concentrated at the fundamental frequency and its harmonics, the feature provides an estimate of the Voiced-Speech-Plus-Noise to Noise Ratio (VSNNR). In [56] the pitch estimate for each frame is used directly as a feature alongside the estimated voiced speech probability, with both obtained using the Pitch Estimation Filter with Amplitude Compression (PEFAC) [54, 57] algorithm.

Since unvoiced speech lacks harmonic structure, a different set of techniques are required for its detection. The enhancer proposed in [56] uses a feature set which explicitly detects the aperiodic speech energy of sibilant phones, which are “hissing” sounds created by forcing air through a constricted passage. The feature set includes, for each time-frame, the probability of sibilant speech and a vector containing a normalised estimate of the power spectrum of the sibilant speech in 500 Hz bands. The features are obtained using the algorithm from [55] which detects a sustained increase in power over the duration of a sibilant phone.

In many studies, the same feature set is used to detect both voiced and unvoiced speech. For example, several studies use a feature set based on a cochleagram [64, 21, 20]. Other studies use feature sets which have also been successfully applied to speech recognition, such as RASTA-PLP features [67], Mel-frequency Cepstral Coefficients (MFCCs) [26] and Amplitude Modulation Spectrogram (AMS) features [103]. RASTA-PLP features are used to estimate TF masks in [170, 65, 64]. RASTA-PLP uses Perceptual Linear Prediction (PLP) features [66] computed with an additional Relative Spectral Transform (RASTA) band-pass filtering stage. Perceptual linear prediction is a variant of linear prediction, a method for modelling the spectrum of speech using an autoregressive all-pole model, that incorporates some concepts from psychoacoustics such as equal loudness curves and amplitude compression. The same studies also employ Mel-frequency Cepstral Coefficient (MFCC) features, obtained by computing the Discrete Cosine Transform (DCT) of the log of the mel-frequency-resolution speech power spectrum.

A number of studies [170, 60, 172, 62, 94, 95] use AMS features, which model the frequency content of amplitude modulations in the envelope of the signal over a number of time frames. There is evidence from speech recognition literature that modulation features such as AMS, which vary slowly over time, may be more robust to reverberation than features based on the spectral envelope in a single time frame, such as MFCCs and

RASTA-PLP coefficients [120].

Another approach is to combine several different feature sets together, as in [170, 27, 184, 127, 64]. In [170] the authors proposed a group Lasso approach [182] to select complementary features, resulting in a proposed feature set which included AMS, RASTA-PLP, MFCC and autocorrelation-based features along with some first and second order delta features.

Delta features measure the rate of change of features over time or frequency and are commonly used in speech recognition where it has been found that they significantly improve performance over the use of spectral features alone [43]. Several studies use them as additional features to estimate a TF mask [95, 94, 60, 61]. They are usually computed either as the difference between the value of a feature in one cell and its value in a neighbouring TF cell, or as the gradient of a straight line fitted to a local region around the current frame.

2.2.3 Classifiers

In enhancement algorithms with the structure shown in Fig. 2.1, the objective of the mask estimation algorithm is to estimate the value of the oracle mask in each TF unit from a set of features extracted from the noisy speech. A number of well-understood techniques for estimation and classification have been applied to this problem, including Support Vector Machines (SVMs), Gaussian Mixture Models (GMMs), Classification and Regression Trees (CARTs), Deep Neural Networks (DNNs) and Long Short-Term Memory Recurrent Neural Networks (LSTM-RNNs).

2.2.3.1 Support Vector Machines (SVMs)

Several studies, e.g. [60, 61, 62, 172], employ binary classifiers known as Support Vector Machines (SVMs) to estimate a target binary mask. If the feature vectors belonging to each of the two classes used to train the classifier are linearly separable, the SVM will find the hyperplane that "best" separates the classes; the hyperplane is chosen to maximise the distance from the hyperplane to the nearest data point in each class. During the enhancement phase, the sign of the classifier output, which describes which side of the dividing hyperplane the observed feature vector lies in, is used to classify the vector. If the input feature vectors are not linearly separable (as in [60]) a hyperplane is constructed in a high dimensional feature space in which the transformed data is linearly separable. To prevent the classifier from needing to transform feature vectors to the high dimensional space during either training or enhancement the so-called "kernel trick" is employed, whereby the feature space is chosen such that scalar products between vectors in the high dimensional space can be computed using non-linear functions (known as kernel functions) in the lower dimensional space. In [60] a Gaussian kernel function is employed. The authors found that trained SVMs tended to under-label speech-dominated regions, and proposed to modify the threshold used to binarise the SVM output in order to maximise the HIT-FA performance measure defined in Sec. 2.3.1 below.

2.2.3.2 Gaussian Mixture Models (GMMs)

Several studies, e.g. [135, 95, 94], have used parametric probability density functions known as Gaussian Mixture Models (GMMs) as classifiers. A GMM models the probability distribution of a feature vector as a weighted sum of multiple Gaussian densities. GMMs are popular classifiers in biometric applications such as speaker recognition systems, e.g. [129], as they can form smooth approximations to a wide range of feature distributions. In [135, 95, 94] separate GMMs are used to model feature vectors extracted from different "classes" of TF units. In [135] there are two classes, "speech dominated" and "noise dominated", which correspond to ones and zeros in the IBM, respectively. In [95, 94] these classes are subdivided to give four sub-classes, "very noise dominated", "moderately noise dominated", "moderately speech dominated" and "very speech dominated", with the boundaries between the classes defined by a set of frequency-band-dependent local SNR thresholds. The sub-class division led to faster convergence and higher classification accuracy. During the enhancement phase, TF units are classified according to which GMM gives the highest posterior probability.

2.2.3.3 Feed-forward neural networks

In recent years, many studies have proposed TF mask estimation algorithms based on neural networks, e.g. [65, 64, 21, 20]. This trend mirrors the speech recognition field in which GMMs have been outperformed and largely superseded by neural networks [69, 58]. In a DNN the input signal flows through several "hidden layers" each composed of multiple "hidden units". The layers are "fully-connected", i.e. the output of each unit in layer l is connected to the input of each unit in layer $l + 1$. Within each hidden unit, a weighted sum of the inputs is computed, a bias term applied, and a non-linear "activation function" is applied. A more complete review of neural network architectures is given in Chapter 5.

The IBM estimation algorithm in [65] has two stages. In the first stage, features are extracted from each TF unit of the noisy speech and fed into a subband DNN which estimates the probability that the correct IBM label is one. A separate DNN is trained for each of the 64 frequency bands. In the second stage the probabilities in a rectangular window spanning 5 time frames and 17 frequency bands and centred on each TF unit are concatenated and used as a feature vector for another subband DNN which classifies the unit as a zero or one. The purpose of this window is to improve estimation accuracy by including contextual information from neighbouring TF units into the mask estimate, thereby exploiting the strong correlation between both the speech and the noise in neighbouring frames. The mask estimators in [64, 21] use a single DNN to estimate the mask for all frequency bands. The algorithms use sliding feature and estimation windows: features within a sliding window of length $2V + 1$ frames, extending V frames either side of the current frame, m, are concatenated and used as inputs to the estimator which simultaneously estimates all of the mask values within a window of length $2Q+1$ frames, extending Q frames either side m. At time $m + 1$ the windows shift forward by one frame, and the procedure is repeated. In total, this produces $2Q + 1$ mask estimates for each mask bin, which are then averaged to produced the final mask estimate. The feature context window incorporates contextual information into the mask estimate, while the estimation window is intended to improve performance by lessening the effect of individual mask estimation errors through averaging several estimates.

2.2.3.4 Long Short-Term Memory Recurrent Neural Networks (LSTM-RNN)

Another way of incorporating contextual information into the mask estimate, as an alternative to using delta features and feature context windows, is to use an estimation algorithm, such as a Recurrent Neural Network (RNN), which has an internal memory of previous inputs or outputs and uses it to estimate the mask in frame m. A RNN is similar to a "feed-forward" neural network but contains additional "recurrent" connections which feed the output of each unit back into the input in the following time-step. This gives the algorithm a "short-term" memory in the form of an internal state (in addition to a "long-term" memory in the form of the learned network weights and biases). A popular RNN architecture is the Long Short-Term Memory Recurrent Neural Network (LSTM-RNN) [70], which was introduced to solve the problems of vanishing and exploding gradients during training that exist with standard RNNs. LSTM-RNN have been shown to be effective at many tasks including speech recognition [58] and language modelling [146]. As with feed-forward neural networks, LSTMs can be "stacked" in layers to form "deep" architectures. Each LSTM layer or "cell" contain gates which control when the memory in the internal state is updated. This enables LSTMs to retain information in their memory for longer than standard RNNs, which in turn enables them to model longer dependencies between inputs and outputs. Several TF mask estimators that use LSTM-RNN have been proposed [174, 40, 20]. The algorithms in [174, 20] both outperformed mask estimators that used feed-forward neural networks.

2.2.4 Robustness to new conditions

One advantage of GMMs over neural networks is that they typically use fewer parameters and may therefore be less prone to overfitting. Overfitting damages a mask estimator's ability to generalise to new speakers and noises; this has been a well documented problem for mask estimators and especially for those based on neural networks [21, 20, 172].

A number of strategies have been proposed for improving the robustness of an enhancer to noise types that were not included in the training data set. The most popular strategy, adopted in [21, 20, 172], is to train a single model on a large number of different noise types. Another strategy is to try to adapt the enhancer at test-time to the new noisy environment. In [94] a small amount of noise-only data is gathered from the new noisy environment and used along with some pre-recorded clean speech to produce an initial model whose GMM parameters are then incrementally trained as more noise data becomes available. A similar approach, but based on an SVM classifier, is presented in [62]. In this study, rather than adapting the internal classifier parameters, the threshold used to binarise the SVM output is instead adjusted, with the new threshold chosen to maximise the classification accuracy. A third strategy for generalising to unseen noises, adopted in [78], is to train a number of different models, each on a different noise type. In the enhancement stage, the model that is most appropriate for the environment is selected, either manually by the user or through an automated process.

2.2.5 Mask application

After the mask-estimation procedure has generated a TF mask, enhancement algorithms with the structure shown in Fig. 2.1 apply the mask to the noisy speech and convert the resulting signal back into the time-domain. The conventional way of applying the mask is to multiply the noisy speech by the mask in the TF-domain. Although applying a binary mask in this way can improve the intelligibility of noisy speech, the resulting speech often has very poor perceptual quality. This may be partly due to the fact that the gain changes instantaneously between TF units in neighbouring frames with different mask values. This makes the speech and noise switch on and off abruptly and synchronously, giving a harsh and unnatural quality to the speech. The mask may also contain isolated peaks which give rise to musical noise and classification errors which can introduce distortion artefacts into the speech.

In order to improve the quality of binary masked speech, a number of studies have therefore experimented with modifying the binary mask before applying it to the noisy speech. In [144] the authors evaluated a number of mask modifications including adding dither to the mask and the application of temporal smoothing to the cepstrum of the mask as suggested in [114]. The algorithm from [114] applies different degrees of smoothing to different cepstral coefficients, so that pitch and envelope information is preserved while the random peaks are smoothed. The authors of [144] concluded that the best results were obtained by applying the mask in the conventional way using gains of 1 and 0.1 for the two mask values. In [177], the estimated mask and also its complement were used to obtain intermediate estimates of the speech and noise. These estimates were then combined to derive a continuous-valued TF gain function which was applied to the original noisy speech. A final processing stage then imposed temporal continuity on the sequence of TF spectral magnitudes. The authors found that this processing was able to improve the quality of the enhanced speech while preserving its intelligibility.

2.3 Evaluation metrics

In Sec. 2.2 several techniques for improving the intelligibility of noisy speech signals were discussed. The most reliable way of evaluating these algorithms is to measure the intelligibility and quality of the enhanced signals using listening tests conducted by trained listeners. However, in some situations listening tests may be impractical or too time consuming, for example during the development phase of a speech enhancer. In these situations, an algorithmic metric may provide an adequate alternative to listening tests. Quality metrics and intelligibility metrics estimate the quality and intelligibility of the enhanced signal by comparing the clean and enhanced speech signals, whilst mask-accuracy metrics compare the estimated and oracle masks directly. Before applying a quality or intelligibility metric, the estimated TF mask must be applied to the noisy speech in the TF-domain and the time-domain enhanced speech synthesised from the resulting signal. Comparing the masks directly using a mask-accuracy metric requires fewer computations than assessing the quality or intelligibility of the resynthesised speech and can provide additional insight into the performance of the estimator.

2.3.1 Binary mask-accuracy metrics

One possibility is to evaluate the estimated mask using classification accuracy, defined as the percentage of mask values correctly classified. However, classification accuracy weights misses and false alarms equally. That is, ones in the oracle mask that are incorrectly labelled as zeros in the estimated mask are weighted equally to zeros in the oracle mask that are incorrectly labelled as ones in the estimated mask. It was demonstrated in [110] that intelligibility is more sensitive to the false alarm rate (FA), i.e. the number of false alarms as a percentage of the total number of zeros in the oracle mask, than to the miss rate (MISS), i.e. the number of misses as a percentage of the total number of ones in the oracle mask. Since classification accuracy does not differentiate between the two types of error, [95] proposed to instead use HIT-FA: the hit rate (HIT) minus the false alarm rate (FA), where HIT$= 100-$MISS. The HIT-FA metric was shown to correlate with the intelligibility of the masked speech [95, 94].

2.3.2 Speech quality metrics

One of the most widely used and extensively validated metrics for estimating speech quality is Perceptual Evaluation of Speech Quality (PESQ) [131, 84]. Listening tests have shown that PESQ can predict the quality of noisy speech that has been processed using TFGM-based speech enhancers [77, 132].

The PESQ algorithm comprises a pre-processing stage, an auditory transform stage, a disturbance processing stage and a disturbance aggregation, or cognitive modelling stage. In the pre-processing stage, the levels of the clean and degraded signal are normalised to a standard listening level and are time-aligned. An auditory transform based on a psychoacoustic model is then applied. This first involves grouping together frequency bins according to the Bark scale [185]. Two equalisation steps are then carried out: the first partially compensates for (and limits the effect of) differences in the long-term power spectrum of the speech in active frames caused by linear filtering. The second equalisation step partially compensates for short-term gain variations in the degraded speech. In the final stage of the auditory transform, a gain is applied to account for the mapping between sound pressure level and perceived loudness [141].

After the auditory transform has been applied, the signed error between the clean and degraded signals, termed the disturbance density, is computed in the TF domain. Due to masking, errors that are low in energy compared with the clean and degraded signals are assumed to be inaudible and are set to zero. Errors that increase signal energy are assumed to be more detrimental to speech quality than errors that decrease signal energy. This is modelled using an additional “asymmetrical” disturbance density, which measures only errors which result in an increase in signal energy.

In the final stage of the PESQ algorithm, the disturbance density and asymmetrical disturbance density are aggregated first across frequency (to produce a “frame disturbance”) and then across time, using Lp norms with weights that emphasise disturbances that occur during silences in the clean speech and deemphasise disturbances that occur at the start of the signal, which models the effect of short-term memory on speech quality. The aggregation across time occurs in two stages: the frame disturbance is first averaged over intervals of approximately 320 ms using Lp norms, and these disturbance measures are then averaged across the entire length of the utterance, again with Lp norms. The value of p in the Lp norm is higher for the first time-averaging stage (over 320 ms intervals) to model the fact that, if a small part of a 320 ms interval is distorted, the quality of the entire interval could be considered poor, whereas in a long recording one sentence can be considered distorted and the following sentence considered to have good quality. During the disturbance aggregation stage, additional steps are used to account for possible errors in the estimation of time delays during the pre-processing stage.

After the aggregation of the disturbance densities the final PESQ score is computed as a linear combination of the average disturbance value, D_{av}, and the average asymmetrical disturbance value, DA_{av},

$$\text{PESQ} = 4.5 - 0.1 \cdot D_{av} - 0.0309 \cdot DA_{av}.$$

2.3.3 Speech intelligibility metrics

In the following section several types of intelligibility metric will be discussed. For a more complete overview see [107, 12].

2.3.3.1 SNR-based metrics

Some of the most popular intelligibility metrics are based on measuring the SNR in different frequency bands. The work of [42], originally led to the Articulation Index (AI) [3], as a standardised method of objectively estimating the intelligibility of speech. The AI and its successor, the Speech Intelligibility Index (SII) [4], are computed by measuring the SNR in different frequency bands, adjusting this to account for the masking of higher frequency bands by lower frequency bands, and then computing a weighted sum of these values with weights that reflect the relative importance of the bands.

The SII performs well with stationary additive noise but is unable to predict the effects of fluctuating noises, since the SII is computed from the long-term speech and noise spectra [130]. Studies show that normal-hearing listeners find speech more intelligible in fluctuating noises than in stationary noises [117, 1, 35]. This is thought to be because the listener is able to catch "glimpses" of the speech during periods where the noise energy is low [73, 74]. For this reason, SII was modified in [130] so that it is computed in short-time frames, which are then averaged across time.

2.3.3.2 Modulation-based metrics

Another group of intelligibility metrics are the so-called "modulation-based" metrics, such as the Speech Transmission Index (STI) [80] and the speech-based Envelope Power Spectrum Model (sEPSM) [91]. These metrics measure intelligibility by comparing the temporal amplitudes modulations in the degraded signal with those of the clean speech. In addition to being able to predict the effects of additive noise (as with SII) the STI is able to predict the effect on intelligibility of certain types of non-linear distortions such as peak clipping, or time-domain distortions like reverberation, which affects the signal envelope [140]. STI has a similar structure to SII, but instead of taking clean and degraded speech as inputs it considers the effects of a channel, modelled as a black box, on a special test signal. To create the test signal, noise with the long-term average spectrum of speech is sequentially modulated with a cosine wave at several modulation frequencies which are common in speech. Figure 2.3 shows an example of the test signal envelope at modulation frequency F Hz. The signal on the left, I_{test}, is the test signal envelope at the input of the channel,

$$I_{test}\left(t\right) = \bar{I}_{test}\left(1 + \cos\left(2\pi F t\right)\right)$$

where $\bar{I}_{test}$ is a constant. The noise introduced by the channel is modelled as having a constant envelope, $\bar{I}_{noise}$. The signal envelope at the channel output is

$$I_T\left(t\right) = \bar{I}_T\left(1 + m\cos\left(2\pi F t\right)\right)$$

where

$$m = \frac{\bar{I}_{test}}{\bar{I}_{test} + \bar{I}_{noise}}$$

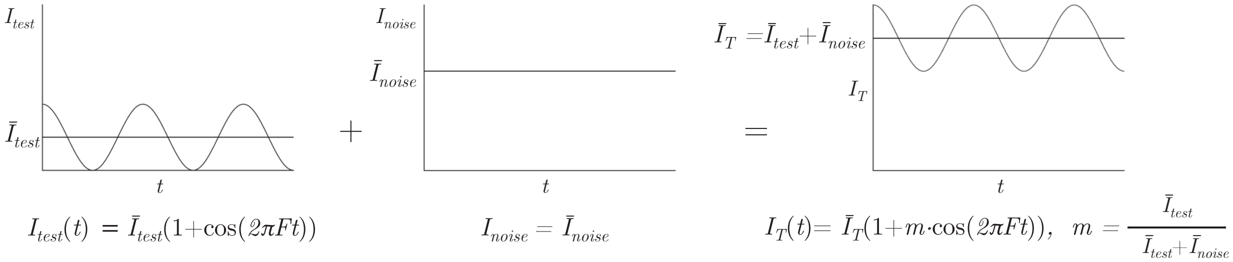

I_{test}
$\bar{I}_{test}$
t
$I_{test}(t) = \bar{I}_{test}(1+\cos(2\pi Ft))$
+
I_{noise}
$\bar{I}_{noise}$
t
$I_{noise} = \bar{I}_{noise}$
=
$\bar{I}_T = \bar{I}_{test}+\bar{I}_{noise}$
I_T
t
$I_T(t) = \bar{I}_T(1+m\cdot\cos(2\pi Ft)),\quad m = \frac{\bar{I}_{test}}{\bar{I}_{test}+\bar{I}_{noise}}$

Figure 2.3: Diagram of an STI test signal envelope at modulation frequency, F. The signal on the left, I_{test}, is the test signal envelope at the input of the channel, where $\bar{I}_{test}$ is a constant. The noise introduced by the channel is modelled as having a constant envelope, $\bar{I}_{noise}$. The signal envelope at the channel output is I_T and m is the modulation index.

is the modulation index, which describes the amount by which the modulated part of the signal varies around its unmodulated level. The effective SNR is obtained from m as

$$\text{SNR} = 10\log_{10}\left(\frac{m}{1-m}\right).$$

The STI is obtained by computing the SNR in different frequency bands and for different modulation frequencies.

The sEPSM [91] metric measures the signal-plus-noise to noise ratio of the envelopes of the speech and noise signals in different frequency bands at the modulation frequencies that are considered important for speech intelligibility. This quantity is combined across all frequency bands and modulation frequency channels and mapped to a predicted intelligibility. sEPSM was shown to predict the effects of additive stationary noise, reverberation and at least one non-linear noise reduction algorithm (spectral subtraction [13]).

It has also been observed in multiple studies that many of the SNR-based and modulation-based metrics correlate poorly with the intelligibility of speech that been processed with non-linear noise reduction algorithms [148, 149, 51, 113], such as TFGM algorithms and binary masks, which are typically unable to improve intelligibility. For example, STI predicts an intelligibility improvement when spectral subtraction is applied [51], which is contradicted by the results of listening experiments.

In order to understand why STI cannot predict the effects of speech enhancement algorithms, it is helpful to consider both the effects of the additive noise on the intelligibility of speech, and why enhancement algorithms do not improve intelligibility. It was proposed in [30, 145] that the noise affects intelligibility through four different mechanisms:

(1) a reduction in the depth of the temporal modulations in the envelope of the speech relative to the level of the noise,

(2) the introduction of modulations from stochastic envelope fluctuations in the noise signal,

(3) the introduction of modulations from phase interactions between the speech and noise signals, and

(4) the corruption of the fine structure of the speech.

The introduction of additional modulations by the noise through effects (2) and (3) creates a challenge for the listener who must separate the speech modulations from those of the interference. In [33] this analysis was extended to explain the intelligibility of noisy speech subject to spectral subtraction [13]. Spectral subtraction, like several other TFGM algorithms including MMSE spectral or log-spectral amplitude estimation [36, 37], involves subtracting a smoothed estimate of the envelope of the noise spectrum from the envelope of the noisy speech spectrum. The authors demonstrated that under “ideal” conditions, where oracle information, i.e. the true speech and noise signals envelopes, are known, if the noise envelope is subtracted from the noisy speech envelope without averaging, then effects (1) and (2) are corrected, but effects (3) and (4) remain, and although the intelligibility increases substantially, it does not reach 100 %. In more realistic conditions, the mean of the envelope of the estimated noise (or a smoothed estimate) is subtracted from the noisy speech, and rectification is applied to the resulting signal envelope, i.e. the parts of the envelope that were made negative-valued are set equal to zero. The result is that effect (1) is compensated for, but effects (2-4) remain, and additional distortion in the form of interfering modulations are introduced to the envelope of the signal, which damages intelligibility. With spectral subtraction, this distortion is introduced via the rectification, which produces “musical noise”. Other TFGM algorithms introduce other forms of distortion to the enhanced envelopes. The result is that, after enhancement, intelligibility either remains the same or is reduced. STI cannot predict this because it only measures effect (1). STI may even confuse the spurious modulations (those introduced by effects (2) and (3), and by the enhancement algorithm) with the speech modulations.

In [34] it was demonstrated that measuring the ratio of the strength of the speech modulations to strength of the spurious modulations (those introduced by the noise and the enhancement algorithm) provided a better prediction of the intelligibility of enhanced speech than the modulation depth. The authors proposed using this measure as the basis of an intelligibility metric.

2.3.3.3 Correlation-based intelligibility metrics

In contrast to the SNR-based and modulation-based metrics, a separate group of “correlation-based” metrics have been shown to be capable of predicting the intelligibility of noisy speech processed with TFGM algorithms [151]. These metrics are based on a correlation-comparison between the spectro-temporal envelopes of the clean and degraded speech signals, and therefore, unlike STI and SII, they account for the effects of spurious modulations introduced by the noise and the enhancement algorithm. The correlation can be computed over either frequency or time. Of the measures which compute correlation over time, some compute it over the entire signal at once, e.g. [51], whilst other methods divide each frequency bin into smaller segments, compute the correlation in each segment and then average the results. The Short-Time Objective Intelligibility Measure (STOI) metric [152], which is in the latter category, compares the spectral amplitude modulation of the clean and degraded speech signals with correlation coefficients computed from overlapping segments. The authors argued that computing the correlation over very long segments (e.g. the entire signal) allows a small number of regions of the clean or degraded speech with high amplitudes to dominate the overall result, whilst using very short segments (20-30 ms) results in a poor modulation frequency resolution which excludes certain important low frequency temporal modulations. After experimenting with segments of between 128 ms and 6.4 seconds, the authors proposed using a segment length of 384 ms. This means that STOI is sensitive to temporal modulations down to 2.6 Hz. The authors found this to be in line with results of several listening experiments [31, 6] which found that temporal amplitude modulations below around 2-3 Hz could be removed without affecting intelligibility, suggesting that a correlation segment length of between 333-500 ms would be adequate.

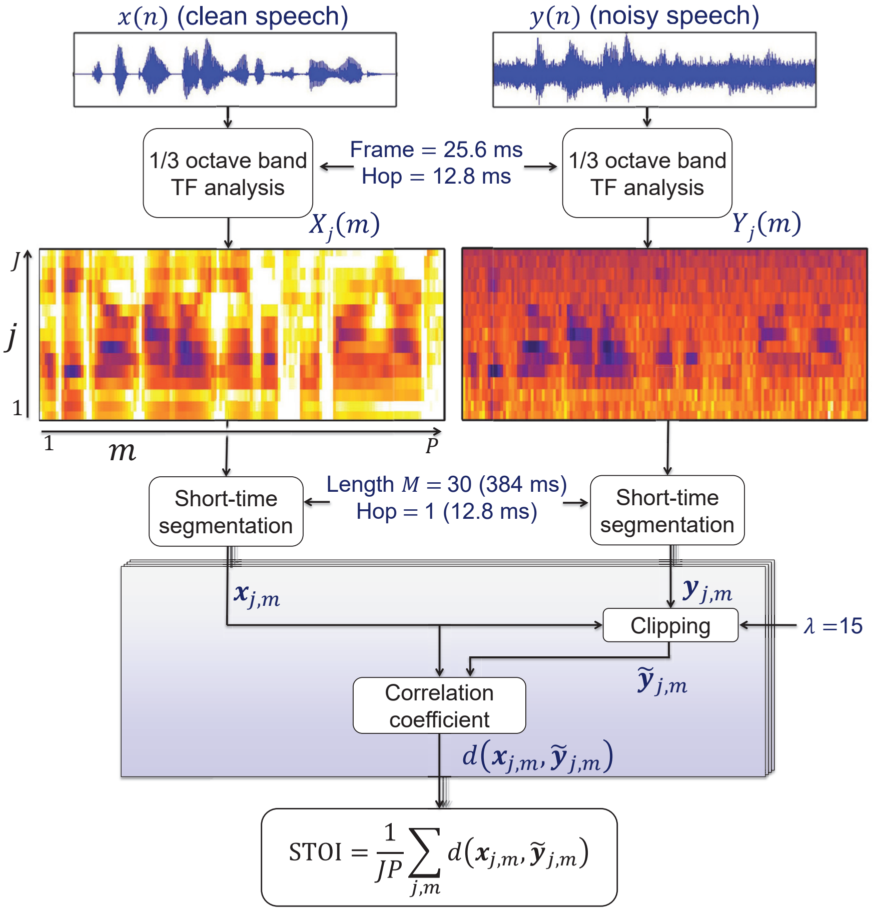


Figure 2.4: Diagram of the computation of the STOI metric [152]

This segment length was also found to be consistent with experiments which suggest that the temporal integration time of the auditory system, which relates the detectability of a brief stimulus to its duration, has an upper bound of a few hundred milliseconds [158].

A brief overview of the STOI metric is now presented, and a block diagram is shown in Fig. 2.4; readers are referred to [151] for a more detailed description. The clean speech is first converted into the STFT-domain using 50%-overlapping Hanning analysis windows of length 25.6 ms. STFT frames whose total energy is 40 dB or more below that of the frame with highest energy are deemed to be silent. These frames are deleted from both the clean and degraded speech signals and are not used in calculating the STOI metric. The resultant complex-valued STFT coefficients, $X(k, m)$, are then combined into J third-octave bands by computing the TF cell amplitudes

$$X_j(m) = \sqrt{\sum_{k=K_j}^{K_{j+1}-1} |X(k,\, m)|^2} \quad \text{for } j = 1,\, \ldots,\, J \tag{2.5}$$

where K_j is the lowest STFT frequency bin within frequency band j. The correlation between clean and degraded speech is performed on vectors of duration 384 ms. For each m, a modulation vector is defined,

$$\mathbf{x}_{j,m} = [X_j(m - M + 1), X_j(m - M + 2),\, \ldots,\, X_j(m)]^T, \tag{2.6}$$

comprising $M = 384/\,(0.5 \times 25.6) = 30$ consecutive TF cells within frequency band j. The same processing is applied to the degraded speech to obtain the corresponding quantities $Y(k,\, m)$, $Y_j(m)$ and $\mathbf{y}_{j,m}$. Before computing the correlation, the degraded speech is clipped to limit the impact of frames containing low speech energy, since it is assumed that degradations in these frames are relatively unimportant to intelligibility. The clipped TF cell amplitudes, denoted by a tilde superscript, are determined as

$$\widetilde{Y}_j(m) = \min\left(Y_j(m),\, \lambda \frac{\|\mathbf{y}_{j,m}\|}{\|\mathbf{x}_{j,m}\|} X_j(m)\right) \tag{2.7}$$

where $\lambda = 6.623$ and $\|\ \|$ is the Euclidean norm. The corresponding modulation vectors are $\tilde{\mathbf{y}}_{j,m}$. The STOI contribution of the TF cell $(j,\, m)$ is then given by

$$d\left(\mathbf{x}_{j,m},\, \tilde{\mathbf{y}}_{j,m}\right) \triangleq \frac{\left(\mathbf{x}_{j,m} - \bar{x}_{j,m}\right)^T \tilde{\mathbf{y}}_{j,m}}{\left\|\mathbf{x}_{j,m} - \bar{x}_{j,m}\right\| \left\|\tilde{\mathbf{y}}_{j,m} - \bar{\tilde{y}}_{j,m}\right\|} \tag{2.8}$$

where $\bar{x}_{j,m}$ denotes the mean of vector $\mathbf{x}_{j,m}$. The overall STOI metric is found by averaging the contributions of TF cells over all bands, j, and all frames, m. That is,

$$\mathrm{STOI} = \frac{1}{JP} \sum_{j=1}^{J} \sum_{m=1}^{P} d\left(\mathbf{x}_{j,m},\, \tilde{\mathbf{y}}_{j,m}\right).$$

Several studies have shown that mapped STOI measurements correlate strongly with subjective intelligibility results in both the case of unenhanced noisy speech and of noisy speech that has been processed with various noise suppression algorithms including binary masks, and that STOI outperforms other intelligibility metrics (including other correlation-based metrics) [52, 151, 152]. It has also been shown that STOI can correctly predict the effect of introducing different types of errors in the IBM [151, 110].

One drawback of STOI is that it performs poorly with additive noise sources that contain strong temporal modulations [87]. Studies have shown that in noises with temporal "dips", i.e. brief moments when the level of the noise is low, the listener can take advantage of these dips to "glimpse" the target speech, which effects intelligibility [163]. Since STOI performs the correlation coefficient over time, rather than frequency, it struggles to account for this effect. STOI is therefore modified in [87] to compute the correlation coefficient over frequency. The TF representation of the signal is divided into blocks covering 384 ms and 5 kHz. For each block, mean and variance normalisation of each band is performed over time and the spectral correlation coefficient of each frame in the block is then computed and averaged over time. The modified STOI, denoted as Extended STOI (ESTOI), was able to accurately predict the intelligibility of speech containing additive noises with strong temporal modulations, as well as noisy speech processed with TFGM algorithms.

2.3.3.4 Intelligibility metrics based on mutual information

Closely related to the correlation-based metrics are a group of metrics based on measuring the mutual information [25] between the spectral envelopes of the clean and degraded signal [88, 154, 159, 160]. The speech communication process is viewed as a transmission system between the speaker and the listener. The mutual information between the signal envelopes measures the information that can be learnt about the clean speech from observing the noisy speech. If the mutual information is high, the intelligibility of the noisy speech is expected to be high. The mutual information depends on the joint Probability Density Function (PDF) of the clean and noisy envelopes. In practice, this is unknown and the mutual information must be estimated from the signals. In [154] this is done using a k-nearest neighbour estimator. The resulting algorithm achieved comparable results to STOI in terms of the normalised correlation coefficient but marginally worse results in terms of root mean squared error. In [153] two metrics were proposed based on two different ways of fitting a GMM to model the joint PDF of the clean and noisy spectral envelopes. In [88], a metric is proposed based on the lower bounds of mutual information, rather than the mutual information itself, in order to simplify the problem. The structure of the algorithm was very similar to that of STOI but with a non-linearity applied to the correlation coefficient before averaging, and it achieved a performance approximately equal to that of STOI. In [159] the authors extend the statistical model from [88] to include the effects of a “production noise”, which they had proposed in [100], where a model of speech communication is outlined for the purposes of speech enhancement. The production noise models the variability in the speech production process, and its effect in the model is to limit the information rate between a hypothetical (and unknown) message signal that the speaker intended to produce before speaking, and the output of the channel. The production noise is estimated from a corpus of speech utterances. Several recordings of each phrase from different speakers were time warped so that the duration of each sound is identical in all the utterances. The production noise was then estimated from the variability between the signals. The proposed metric is based on the upper bound of the information rate over the communication system, measured in bits per second.

2.3.3.5 Estimating intelligibility content of time-frequency cells

A common feature between many of the approaches discussed so far [151, 88, 87, 154, 159] is that, when the intermediate intelligibility measures computed in different TF regions are averaged to obtain the estimated intelligibility, they are all weighted equally. However, it is known that not all portions of a speech signal contain equal quantities of the information required for intelligibility. For example, multiple studies in which parts of a waveform corresponding to consonants and vowels are replaced with noise have observed that vowel phones appear to contribute more to speech intelligibility than consonants [24, 93, 41]. In [143] the authors investigated the link between the relative information carried by different sections of speech and the degree to which the signal in those sections changed as a function of time. Using a quantity they termed Cochlea Scaled Entropy (CSE), which measures how much successive spectral slices differ from preceding slices, the authors showed how, when sentence segments were replaced with noise, intelligibility decreased linearly with the quantity of CSE replaced. This is consistent with the sensitivity to change of human perceptual systems [101] and also with the principle from information theory that the information a signal carries is related to its unpredictability. Encouraged by the results in [143], the authors of [19] compared the intelligibility prediction performance of two metrics after modifying them to exclude segments of speech containing little speech information. The authors found that the best performing segmentation schemes retained most segments corresponding to vowel-consonant transitions and excluded vowel-only or consonant-only segments. They suggested that this was because spectral changes at transitions were more prominent and robust to noise.

Chapter 3

Weighted-STOI

3.1 Introduction

The basis of the widely used Short-Time Objective Intelligibility Measure (STOI) [151] was described in Sec 2.3.3.3. This intrusive intelligibility metric has been extensively validated [52, 152, 110] and has been found to give accurate predictions of the intelligibility gains available from speech enhancement. A drawback of the metric is that it incorrectly assumes that each frame of speech contributes equally to intelligibility. Since this assumption is manifestly inappropriate for silence frames, the STOI metric incorporates an initial step in which silent frames are detected and deleted from the signal before evaluating the metric. This chapter presents Weighted-STOI (WSTOI), a modified version of STOI, in which the contribution of each time frame to the metric is weighted by its estimated contribution to intelligibility. This estimate equals the mutual information between two versions of a hypothetical signal, representing the information bearing component of the clean speech envelope, at either end of a simplified model of human communication. The modification improves STOI by better accounting for the variation in information content of a speech signal with time and frequency. In active speech frames, TF cells containing important speech information are weighted more heavily than those containing less important information. An advantage of this approach is that, since "silent" frames contain little or no information and are therefore downweighted, it is no longer necessary to detect and delete these frames explicitly as in the STOI metric. The result is more physiologically motivated way of handling silences which, unlike STOI, does not require a hard decision on whether a frame is active or silent. This is advantageous since STOI's frame deletion scheme is sensitive to high energy frames and can result in the concatenation of speech segments that are widely separated in time.

3.2 Language models

As indicated above, the WSTOI algorithm requires an estimate of the speech information that is present in different parts of the signal. The proposed approach to this problem, outlined in Sec. 3.4, includes a scale factor, α, whose value is determined by matching the speech information estimated by the WSTOI algorithm with the speech information estimated by a phone-level n-gram language model. Before outlining the proposed modification to STOI, a brief discussion of language models is therefore given here.

In this context, an n-gram is a sequence of n phones, $\epsilon_{i-n+1}\epsilon_{i-n+2}\dots\epsilon_i$, and the objective of the language model is to estimate the speech information rate provided by the final phone, ϵ_i, given the previous phones, $\epsilon_{i-n+1}\epsilon_{i-n+2}\dots\epsilon_{i-1}$. This is the negative log probability of the final phone conditioned on the previous phones, $-\log\left(P(\epsilon_i \mid \epsilon_{i-n+1}\epsilon_{i-n+2}\ldots\epsilon_{i-1})\right)$, divided by the duration of the phone, and is measured in bits per second. There are many different approaches to estimating $P(\epsilon_i \mid \epsilon_{i-n+1}\epsilon_{i-n+2}\ldots\epsilon_{i-1})$, and a detailed discussion of alternative methods is given in [22]. All approaches require a corpus of training data containing a large number of sequences, $\epsilon_{i-n+1}\epsilon_{i-n+2}\ldots\epsilon_i$. The number of times the sequence $\epsilon_{i-n+1}\epsilon_{i-n+2}\ldots\epsilon_i$ appears in the training data (the number of "counts" of the sequence) is denoted as $c\left(\epsilon_{i-n+1}\epsilon_{i-n+2}\ldots\epsilon_i\right)$. The most straightforward approach is to use the maximum likelihood estimate of the probability,

$$P_{ML}\left(\epsilon_i|\epsilon_{i-n+1}\epsilon_{i-n+2}\ldots\epsilon_{i-1}\right) = \frac{c\left(\epsilon_{i-n+1}\epsilon_{i-n+2}\ldots\epsilon_i\right)}{c\left(\epsilon_{i-n+1}\epsilon_{i-n+2}\ldots\epsilon_{i-1}\right)}.$$

However, this approach will assign a probability of zero to any n-gram which did not appear in the training data, and leads to poor performance in many applications [22]. Other approaches therefore aim to "smooth" the language model and make the probabilities more uniform by adjusting the probability of n-grams with many counts downwards, and the probability of n-grams with few or zero counts upwards. In practice, performance can be improved by combining n-gram models with lower order models, e.g. $(n-1)$-gram and $(n-2)$-gram models. So-called "interpolated models" are formed by a linear combination of higher- and lower-order models, whereas "back-off" models use the higher order model if the n-gram has a non-zero count, and the lower order model otherwise.

One example of an interpolated model is the Kneser-Ney model from [22], which is a modified version of the the back-off model from [102]. The algorithm was converted from a back-off model into an interpolation model as this required fewer approximations and yielded better performance. The rationale behind this model is that, for any n-gram, $\epsilon_{i-n+1}\epsilon_{i-n+2}\ldots\epsilon_i$, the $(n-1)$-gram distribution, i.e. $P\left(\epsilon_i|\epsilon_{i-n+2}\epsilon_{i-n+3}\ldots\epsilon_{i-1}\right)$, is only a significant factor in the combined n-gram and $(n-1)$-gram model when the n-gram, $\epsilon_{i-n+1}\epsilon_{i-n+2}\ldots\epsilon_i$, has few or no counts. Therefore, instead of the $(n-1)$-gram distribution being a smoothed version of the maximum likelihood distribution, it should instead reflect the likelihood that the $(n-1)$-gram $\epsilon_{i-n+2}\epsilon_{i-n+3}\ldots\epsilon_i$ will be seen in a new context not seen during training, i.e. preceded by a new value of ϵ_{i-n+1}. The Kneser-Ney distribution for $n=3$ is given by

$$\begin{aligned} P_{KN}\left(\epsilon_i|\epsilon_{i-2}\epsilon_{i-1}\right) = & \frac{\max\left\{c\left(\epsilon_{i-2}\epsilon_{i-1}\epsilon_i\right)-\nu,\, 0\right\}}{\sum_{\epsilon_i} c\left(\epsilon_{i-2}\epsilon_{i-1}\epsilon_i\right)} \\ & + \frac{\nu}{\sum_{\epsilon_i} c\left(\epsilon_{i-2}\epsilon_{i-1}\epsilon_i\right)} N_{1+}\left(\epsilon_{i-2}\epsilon_{i-1}\bullet\right) P_{KN}\left(\epsilon_i|\epsilon_{i-1}\right) \end{aligned}$$

where ν is a fixed “discount” subtracted from each non-zero n-gram count, and

$$N_{1+}\left(\epsilon_{i-2}\epsilon_{i-1}\bullet\right) = \left|\left\{\epsilon_i \,:\, c\left(\epsilon_{i-2}\epsilon_{i-1}\epsilon_i\right) > 0\right\}\right|,$$

where $|\{\cdots\}|$ denotes the number of phones or phone pairs that satisfy the given condition. In words, $N_{1+}\left(\epsilon_{i-2}\epsilon_{i-1}\bullet\right)$ is the number of different phones ϵ_i that follow the phone pair $\epsilon_{i-2}\epsilon_{i-1}$ in the training data. Finally,

$$P_{KN}\left(\epsilon_i|\epsilon_{i-1}\right) = \frac{N_{1+}\left(\bullet\epsilon_{i-1}\epsilon_i\right)}{N_{1+}\left(\bullet\epsilon_{i-1}\bullet\right)},$$

where

$$N_{1+}\left(\bullet\epsilon_{i-1}\epsilon_i\right) = \left|\left\{\epsilon_{i-2} \,:\, c\left(\epsilon_{i-2}\epsilon_{i-1}\epsilon_i\right) > 0\right\}\right|,$$

$$N_{1+}\left(\bullet\epsilon_{i-1}\bullet\right) = \left|\left\{\left(\epsilon_{i-2},\, \epsilon_i\right) \,:\, c\left(\epsilon_{i-2}\epsilon_{i-1}\epsilon_i\right) > 0\right\}\right|.$$

3.3 Overview of WSTOI

This section describes the new intrusive intelligibility metric, Weighted-STOI (WSTOI). In the block diagram of WSTOI shown in Fig. 3.1, the right panel is identical to STOI and calculates the STOI contribution, $d\left(\mathbf{x}_{j,m},\, \tilde{\mathbf{y}}_{j,m}\right)$, of each TF cell, where $\boldsymbol{x}_{j,m}$ denotes the modulation vector of the clean speech in frequency band j that ends at frame m, and $\tilde{\mathbf{y}}_{j,m}$ denotes the clipped modulation vector of the noisy speech. Details of this calculation were given in Sec. 2.3.3.3. The left panel determines the weight, $I_{j,m}$, to apply to each cell and the final metric in the lower block is a weighted sum of the contribution from each TF cell. The weight applied to each cell, $I_{j,m}$, is equal to the mutual information between a hypothetical signal that we assume the speaker intended to produce before speaking, and the version of this signal which is perceived by a listener in an imagined scenario where the listener hears the clean speech signal at a comfortable listening level. To obtain this, a simple model of communication between the speaker and listener is considered.

3.4 WSTOI weights

In the absence of any external speech degradation, the underlying communications channel is represented by the diagram in Fig. 3.2, where $\boldsymbol{s}_m$ is the intended speech in frame m, $\boldsymbol{v}_m$ is the "production noise" proposed in [100], $\boldsymbol{x}_m$ is the actual uttered speech, $\boldsymbol{d}_m$ is the internal ear noise which models the threshold of hearing [4], and $\boldsymbol{u}_m$ the speech perceived by the listener.

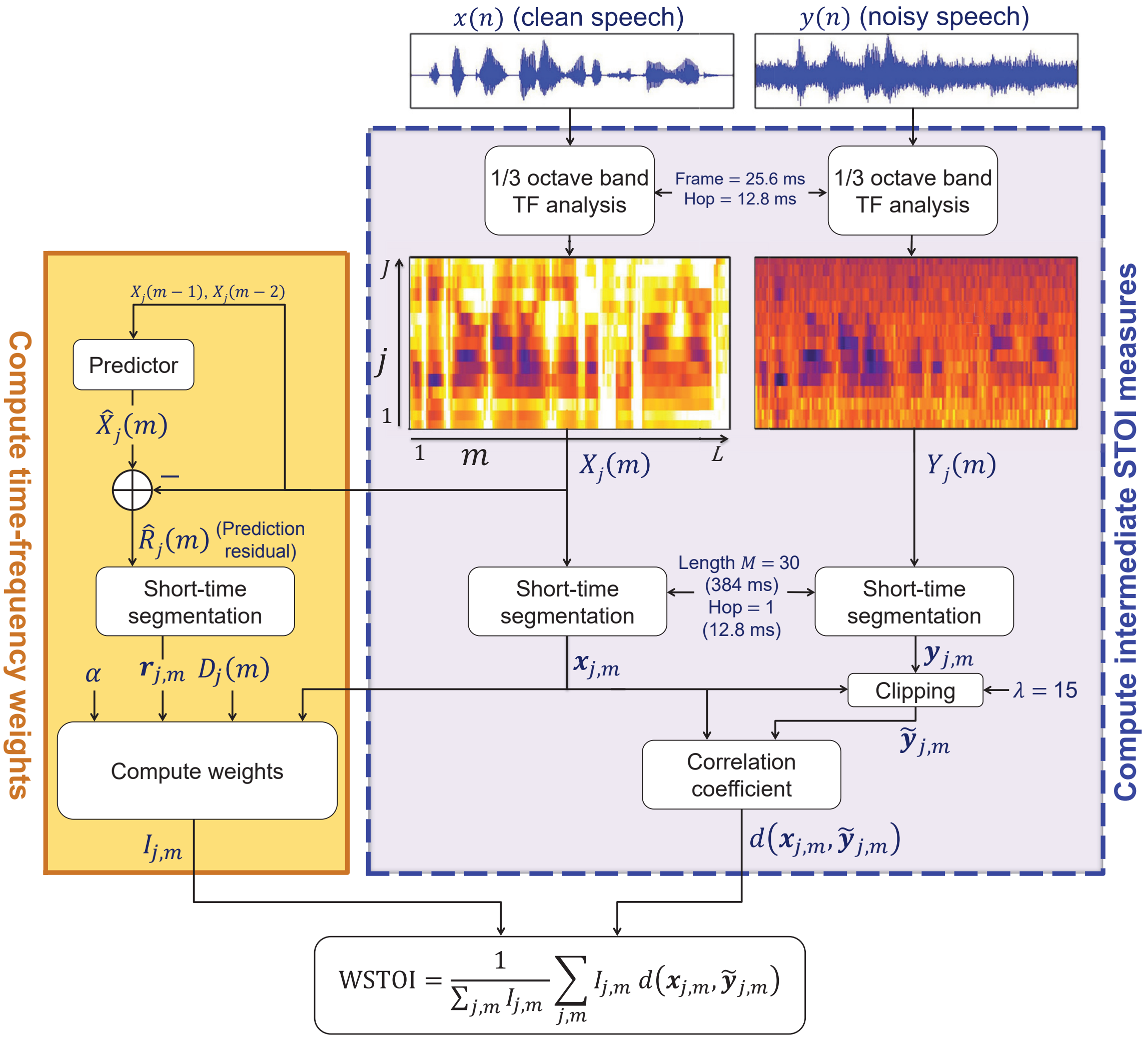


Figure 3.1: Diagram of the WSTOI metric [152]

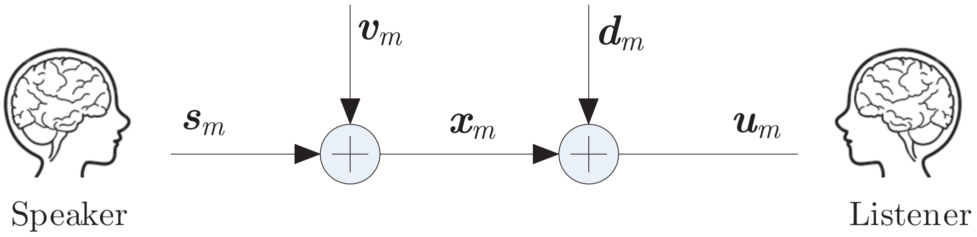


Figure 3.2: Diagram of the underlying communications channel.

In order to assign an appropriate weight to each time frame when calculating the intelligibility metric, we would like to estimate the mutual information, $I\left(\boldsymbol{s}_m; \boldsymbol{u}_m\right)$, that is conveyed to the listener in each modulation vector. We will omit the frequency-band index, j, below since, in common with [3, 150], we assume the frequency bands contribute independently to intelligibility. We assume below that the signals S_m, V_m and D_m are stationary for the duration of the modulation vector (384 ms).

From Sec. 10.1 in [25], the mutual information between $\boldsymbol{s}_m$ and $\boldsymbol{u}_m$ is given by $I\left(\boldsymbol{s}_m; \boldsymbol{u}_m\right) = h\left(\boldsymbol{u}_m\right) - h\left(\boldsymbol{z}_m\right)$ where the total additive noise is $\boldsymbol{z}_m = \boldsymbol{v}_m + \boldsymbol{d}_m$, and $h\left(\boldsymbol{u}_m\right)$ denotes the differential entropy of $\boldsymbol{u}_m$. Assuming that all signals are Gaussian, we can use Theorem 9.4.1 from [25] to write

$$I\left(\boldsymbol{s}_m; \boldsymbol{u}_m\right) = \frac{1}{2}\log\frac{\left|\boldsymbol{C}_{u_m}\right|}{\left|\boldsymbol{C}_{\boldsymbol{z}_m}\right|} = \frac{M}{2}\log\frac{\left|\boldsymbol{C}_{\boldsymbol{u}_m}\right|^{\frac{1}{M}}}{\left|\boldsymbol{C}_{\boldsymbol{z}_m}\right|^{\frac{1}{M}}}$$

where $\left|\boldsymbol{C}_{\boldsymbol{u}_m}\right|$ denotes the determinant of the covariance matrix of $\boldsymbol{u}_m$ and M is the length of the modulation vector in frames. Thus, in order to obtain $I\left(\boldsymbol{s}_m; \boldsymbol{u}_m\right)$ it is necessary to estimate $|\boldsymbol{C}_{\boldsymbol{u}_m}|^{\frac{1}{M}}$ and $|\boldsymbol{C}_{\boldsymbol{z}_m}|^{\frac{1}{M}}$. Assuming that the noise terms, V_m and D_m, are uncorrelated both with each other and over time,

$$\boldsymbol{C}_{\boldsymbol{z}_m} = \left(\sigma_{V,m}^2 + \sigma_D^2\right) \boldsymbol{I}$$

from which

$$|\boldsymbol{C}_{\boldsymbol{z}_m}|^{\frac{1}{M}} = \left(\sigma_{V,m}^2 + \sigma_D^2\right)$$

where $\sigma_{V,m}^2$ and σ_D^2 denote the variance of V_m and D_m, respectively. Since the components of $\boldsymbol{u}_m = \boldsymbol{s}_m + \boldsymbol{z}_m$ are uncorrelated, we can write $\boldsymbol{C}_{\boldsymbol{u}_m} = \boldsymbol{C}_{\boldsymbol{s}_m} + \boldsymbol{C}_{\boldsymbol{z}_m}$. The Minkowski determinant inequality (Theorem 7.8.21 in [72]) then implies that

$$|\boldsymbol{C}_{\boldsymbol{u}_m}|^{\frac{1}{M}} \geq |\boldsymbol{C}_{\boldsymbol{s}_m}|^{\frac{1}{M}} + |\boldsymbol{C}_{\boldsymbol{z}_m}|^{\frac{1}{M}} .$$

The temporal evolution of S_m can be modelled as a low-order AR process [136, 169] defined by

$$S_m = R_m - \sum_{k=1}^{K} a_k S_{m-k}$$

where the coefficients, a_k, may be found through Linear Predictive Coding (LPC) analysis and R_m is the LPC residual signal which we assume to be white and zero-mean. Assuming an AR model for the speech helps to account for the temporal correlation between neighbouring frames, m, which reduces $h\left(\boldsymbol{s}_m\right)$ and is neglected in other models of intelligibility based on mutual information, e.g. [88, 154, 160]. The corresponding modulation vectors are related by

$$\mathbf{A}_m \boldsymbol{s}_m = \boldsymbol{r}_m$$

where $\mathbf{A}_m$ is a Toeplitz lower-triangular matrix whose first column is $[1,\, a_1,\, \ldots,\, a_K,\, 0,\, \ldots,\, 0]$. Since R_m is assumed to be zero-mean and white, $\boldsymbol{C}_{\boldsymbol{r}_m} = \sigma_{R,m}^2 \boldsymbol{I}$ and, the speech covariance matrix is given by

$$\boldsymbol{C}_{\boldsymbol{s}_m} = \mathbf{A}_m^{-1} \boldsymbol{C}_{\boldsymbol{r}_m} \mathbf{A}_m^{-T} = \sigma_{R,m}^2 \mathbf{A}_m^{-1} \mathbf{A}_m^{-T}.$$

Because $\mathbf{A}_m$ is lower-triangular with unit diagonal elements,

$$|\mathbf{A}_m| = \left|\mathbf{A}_m^{-1}\right| = 1$$

and so $|\boldsymbol{C}_{\boldsymbol{s}_m}| = \sigma_{R,m}^{2M}$ from which $|\boldsymbol{C}_{\boldsymbol{s}_m}|^{\frac{1}{M}} = \sigma_{R,m}^2$. Combining the previous results,

$$I\left(\boldsymbol{s}_m;\, \boldsymbol{u}_m\right) = \frac{M}{2} \log \frac{|\boldsymbol{C}_{\boldsymbol{u}_m}|^{\frac{1}{M}}}{|\boldsymbol{C}_{\boldsymbol{z}_m}|^{\frac{1}{M}}} \geq \frac{M}{2} \log \frac{|\boldsymbol{C}_{\boldsymbol{s}_m}|^{\frac{1}{M}} + |\boldsymbol{C}_{\boldsymbol{z}_m}|^{\frac{1}{M}}}{|\boldsymbol{C}_{\boldsymbol{z}_m}|^{\frac{1}{M}}},$$

and therefore

$$I\left(\boldsymbol{s}_m;\, \boldsymbol{u}_m\right) \geq \frac{M}{2} \log \frac{\sigma_{R,m}^2 + \sigma_{V,m}^2 + \sigma_D^2}{\sigma_{V,m}^2 + \sigma_D^2}.$$

The variance of the LPC residual is estimated as

$$\sigma_{R,m}^2 \approx M^{-1} \left\|\mathbf{r}_m\right\|^2$$

where $\|\cdot\|$ denotes the Euclidean norm. Within each frequency band, j, the variance of the internal ear noise is taken as the threshold of hearing in that band, $\sigma_D^2 = \theta_j$, (obtained from Table 1 of [4], as described Sec. 3.5). Finally, following [100], the variance of the production noise is taken to be proportional to the speech power in this band, i.e. $\sigma_{V,m}^2 = \alpha M^{-1} \left\|\mathbf{x}_m\right\|^2$ for some constant α that is determined from training as described in

Sec. 3.5. Thus, reinserting the frequency band index, j, and using the lower bound as the estimate of mutual information, we obtain

$$I_{j,m} = I\left(\mathbf{s}_m;\, \mathbf{u}_m\right) \approx 0.5M\log_2\left(1 + \frac{\left\|\mathbf{r}_m\right\|^2}{\left(\alpha\left\|\mathbf{x}_m\right\|^2 + M\theta_j\right)}\right). \tag{3.1}$$

When forming $\mathbf{r}_m$ in (3.1), $R_j(m)$ is approximated as $\hat{R}_j(m) = \hat{X}_j(m) - X_j(m)$ since $S_j(m)$ is unavailable. Using (3.1) as weights, WSTOI is computed as a weighted average of (2.8) over all bands, j, and all frames, m. That is,

$$\text{WSTOI} = \frac{\displaystyle\sum_{j=1}^{J}\sum_{m=1}^{P}\rho_{j,m}}{\displaystyle\sum_{j=1}^{J}\sum_{m=1}^{P}I_{j,m}}$$

where

$$\rho_{j,m} = I_{j,m}d\left(\mathbf{x}_{j,m},\, \tilde{\mathbf{y}}_{j,m}\right). \tag{3.2}$$

3.5 Experimental validation

The TF-dependent weight in (3.1) is a measure of the local information capacity of the communications channel. To determine the free parameter, α, in (3.1), it is assumed that the information content of the speech mirrors the channel capacity. Accordingly, the parameter α was chosen to maximise the correlation between (3.1) summed over all frequency bands, j, and the speech information as estimated from the Kneser-Ney phone-level n-gram language model described in Sec. 3.2 [22]. The training dataset of TIMIT

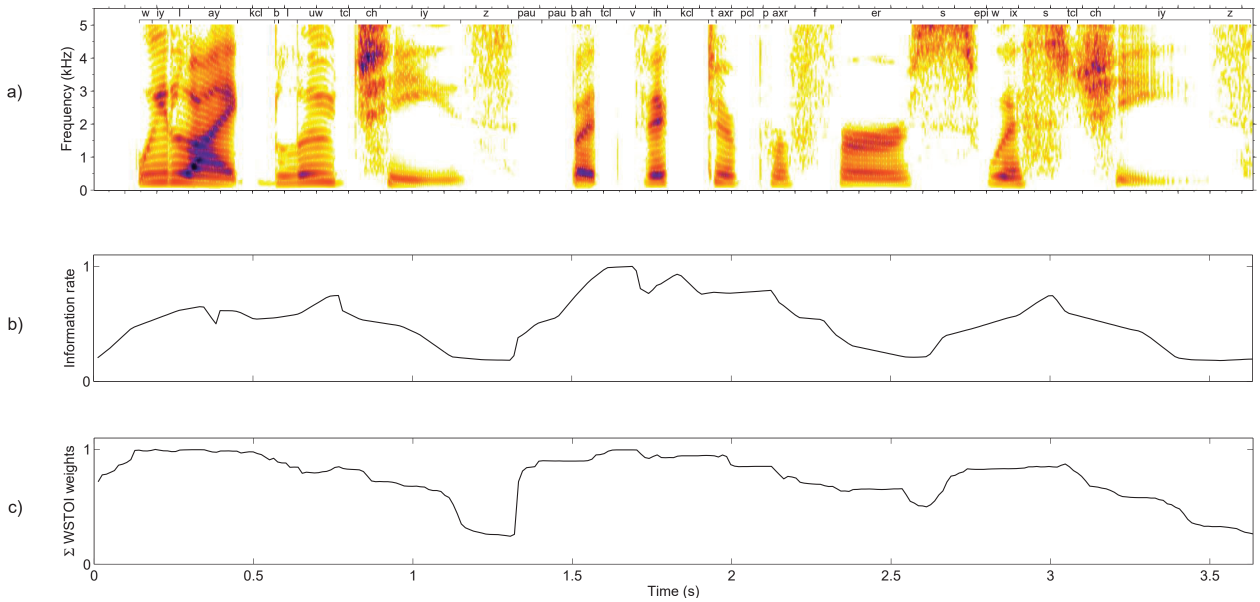


Figure 3.3: a) Spectrogram of the utterance "We like blue cheese but Victor prefers Swiss cheese" with a phonetic transcription shown above. b) Speech information rate predicted by the phone-level n-gram language model from Sec. 3.2. c) WSTOI weights, (3.1), summed over all frequency bands.

[46] was split into a Training and a Validation dataset, each consisting of 1648 utterances. The language model was trained on the phone labels from the Training dataset, with the labels mapped to the reduced set defined in [109]. The n-gram length $n = 3$ was chosen as this maximised the model performance, as measured by perplexity [22], on the Validation set. The AR coefficients, a_k, were obtained using the Training dataset. A prediction order of $K = 3$ was chosen since higher values did not substantially improve the prediction error computed on the Validation set. The optimisation of the parameter α was also performed on the Validation set. The correlation coefficients were computed over the length of each utterance. The output of the language model was the negative log conditional probability of the third phone given the previous two phones, divided by the

duration of the phone. The language model output was smoothed with a moving average of window length $M = 30$, to replicate the smoothing effect of (3.1). The optimum was found to be $\alpha = 2.2 \times 10^{-4}$.

Fig. 3.3 shows a) a spectrogram of the utterance "We like blue cheese but Victor prefers Swiss cheese", b) the smoothed output of the language model and c) the STOI weights from (3.1) summed over all frequency bands. The information rate estimated by the language model is high in time intervals containing many closely spaced phones. The summed weights are high in intervals with frequent changes in the speech spectrum. Since intervals with closely spaced phones coincide with intervals where the spectrum changes frequently, the summed weights mirror the information rate estimated by the language model.

The WSTOI metric was evaluated using the results of the intelligibility tests that were used in [68]. Recordings of the IEEE sentences [133] spoken by a single male speaker combined with babble or car noise were played at one of five SNRs to 60 listeners in either an unprocessed condition or after having been processed using one of three noise suppressors. The number of content words a listener was able to correctly identify in each sentence (between zero and five) was recorded. Intelligibility is defined as the % of content words correctly identified. The responses to a total of 200 sentences were recorded for each combination of noise type, SNR, noise suppressor and suppressor condition (On/Off). For car noise, $\text{SNR} = -\{21, 18, 15, 12, 9\}$ dB, and for babble noise, $\text{SNR} = -\{12, 9, 6, 3, 0\}$ dB. The suppressor algorithms were spectral subtraction (SS) [11], minimum mean squared error log spectral estimation (MMSE) [37] and subspace enhancement (SSA) [75]. STOI scores, d, were mapped to an intelligibility prediction using the logistic function from [151],

$$f(d) = \frac{100}{1 + \exp\left(cd + e\right)}, \tag{3.3}$$

where c and e are free parameters which were fitted to the data using non-linear least squares optimisation. Separate mappings were computed for STOI and WSTOI. The available data was split randomly, with half used to determine the parameters in (3.3) and the remaining half for algorithm evaluation. This process was repeated 1000 times using different splits, with the results from each repetition averaged to compute an overall set of results. The values of θ_j in (3.1) were obtained by integrating the reference internal noise spectrum levels from Table 1 of [4] over the width of each frequency band and then scaling the resulting values for each utterance so that the mean speech-to-internal-noise power ratio of the utterance during active speech periods matched the ratio of the speech and noise spectrum levels for a "normal" vocal effort. Active periods were identified using the procedure in [82].

3.6 Evaluation results

Fig. 3.4a plots the root mean square error (RMSE) in predicted intelligibility against the true intelligibility, for STOI and WSTOI applied to five-sentence segments having the same noise type, SNR and suppressor condition. The histogram is grouped according to the true intelligibility of each segment. It can be seen that both STOI and WSTOI were able to predict the true intelligibility with a root-mean-square error (RMSE) of between 8.7% and 17.7% and that WSTOI gave a lower RMSE at all true-intelligibility levels. Fig. 3.4b

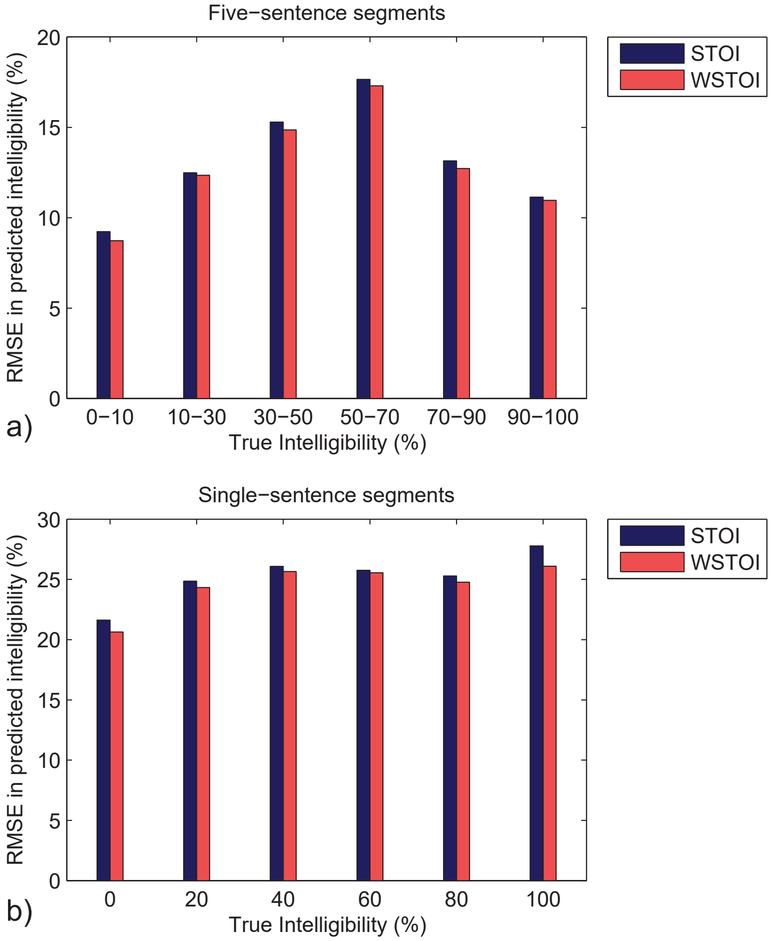


Figure 3.4: Root mean square error in predicted intelligibility against intelligibility for STOI and WSTOI applied to a) five-sentence segments (25 content words) and b) single-sentence segments (5 content words).

shows the performance of STOI and WSTOI on single-sentence segments containing only five content words. Even with these short segments, both STOI and WSTOI were able to predict the intelligibility with an RMSE of between 20.6% and 27.8%. For every one of the 1000 splits the intelligibility prediction performance of WSTOI was significantly better than that of STOI with $p < 10^{-6}$ using a 1-sided sign test.

Fig. 3.5 shows the RMSE in predicted intelligibility for the algorithms applied to single-sentence segments, plotted for each suppressor and noise type. For every combination of suppressor and noise type WSTOI resulted in a lower RMSE than STOI.

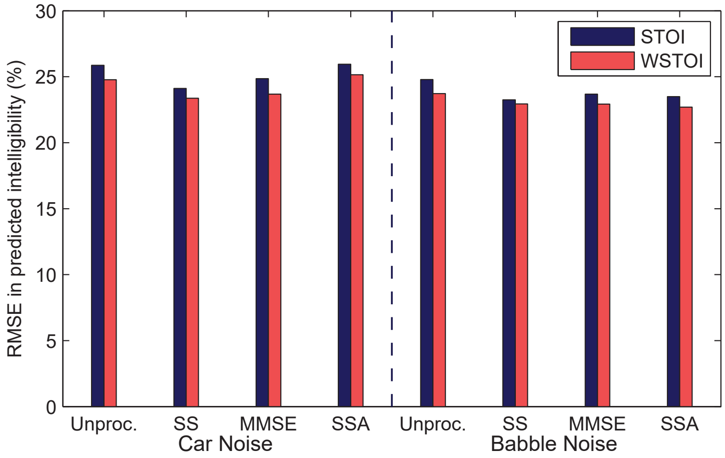


Figure 3.5: Root mean square error in predicted intelligibility for STOI and WSTOI applied to single-sentence segments, plotted for each suppressor and noise type.

3.7 Summary

This chapter has presented the WSTOI intelligibility metric, a modified version of STOI in which the contribution of each TF cell is weighted by an estimate of its intelligibility content. The proposed metric improves on STOI's performance in active speech frames by weighting TF cells containing important speech information more heavily than cells containing less important information. The method avoids the need to detect silent intervals explicitly and hence avoids the discontinuities that result from their removal. Evaluation showed that the modification improved the prediction accuracy of STOI at all performance levels on both long and short utterances. An improvement was observed across all tested combinations of noise type and suppression algorithm.

Chapter 4

STOI-optimal binary masks

4.1 Introduction

In Sec. 2.2.1 a number of existing oracle masks which could serve as a target for a classifier or mask estimation algorithm were described. However, there is evidence that the intelligibility of speech depends not only on the instantaneous spectrum but also on its temporal modulation [8, 32]. This evidence has guided the development of several intelligibility metrics including Speech Transmission Index (STI) [140] and STOI [151]. The intelligibility of the mask-processed speech will not therefore be maximised if the classifier training target uses a mask such as the IBM, TBM, UTBM or IRM since these depend only on the instantaneous spectra of the speech and noise. In this chapter new oracle binary masks are presented, the STOI-Optimal Binary Mask (SOBM) and the WSTOI-Optimal Binary Mask (WOBM), that explicitly maximise intelligibility metrics, STOI and WSTOI, which take account of spectral modulation. The SOBM is derived for two cases: for a deterministic noise signal (DSOBM) and for stochastic noise with a known power spectrum (SSOBM).

4.2 SOBM for Deterministic noise (DSOBM)

In this section the Deterministic STOI-Optimal Binary Mask (DSOBM) is derived; this is the binary mask that maximises STOI in the deterministic noise case. Recall from Chapter 2 that, to compute STOI, the complex-valued STFT coefficients of the clean speech, $X(k,m)$, are combined into J third-octave bands by computing the TF cell amplitudes

$$X_j(m) = \sqrt{\sum_{k=K_j}^{K_{j+1}-1} |X(k,\, m)|^2} \quad \text{for } j = 1,\, \ldots,\, J$$

where K_j is the lowest STFT frequency bin within frequency band j. The correlation between clean and degraded speech is performed on vectors of duration 384 ms. For each time-frame m, a modulation vector is defined,

$$\mathbf{x}_{j,m} = [X_j(m-M+1), X_j(m-M+2),\, \ldots,\, X_j(m)]^T,$$

comprising $M = 384/\,(0.5 \times 25.6) = 30$ consecutive TF cells within frequency band j. A binary mask, $B_j(m) \in \{0,\, 1\}$, is applied to produce the masked signal, $Z_j(m) = B_j(m)Y_j(m)$. Analogously to (2.7), the clipped TF cell amplitudes are determined as

$$\widetilde{Z}_j(m) = \min\left(Z_j(m),\, \lambda \frac{\|\mathbf{z}_{j,m}\|}{\|\mathbf{x}_{j,m}\|} X_j(m) \right)$$

where $\lambda = 6.623$ and $\| \ \|$ is the Euclidean norm. The corresponding modulation vectors are $\tilde{\mathbf{z}}_{j,m}$. The mask is optimised separately in each band, j, by computing

$$B_j(m) = \underset{\{B_j(m):m=1,\,\ldots,\,T\}}{\arg\max} \left(\sum_{m=1}^{T} d\left(\mathbf{x}_{j,m}, \tilde{\mathbf{z}}_{j,m}\right) \right) \tag{4.1}$$

where the function $d\left(\cdot\right)$ is defined in (2.8) and measures the STOI contribution of an individual TF cell. An estimate of (4.1) is obtained efficiently using a dynamic programming approach with three passes: an initial forward pass, a backward pass, and a second forward pass. The initial forward pass and the backward pass produce two independent estimates of $B_j(m)$. The second forward pass uses the results of the first two passes to produce a final, more accurate estimate of $B_j(m)$. The reasons for this three-pass approach are discussed in more detail in Sec. 4.2.1. The subscript j, denoting the STOI frequency band, is omitted in the following description, for clarity.

The initial forward pass involves iterating forwards through frames $m = 1, \ldots, T$, and at each time-step (e.g. time-step m corresponding to frame m) maintaining a list of U_m active states, where each active state corresponds to a unique mask sequence, $\mathbf{b}_m^u$, which starts in frame 1 and ends in frame m, for $u = 1, \ldots, U_m$. Associated with each active state is the STOI sum,

$$k_m^u = \sum_{s=1}^{m} d\left(\mathbf{x}_s, \tilde{\mathbf{z}}_s\right),$$

corresponding to the best mask sequence $\{B(i) : i = 1, \ldots, m\}$ whose final M values match the entries of $\mathbf{b}_m^u$.

At time-step m of the dynamic programming, each of the U_{m-1} active states, b_{m-1}^u, from the previous time-step is used to create two alternative potential active states for the current time-step, $\hat{b}_m^u$ and $\hat{b}_m^{u+U_{m-1}}$ by appending $B(m) = 0$ and $B(m) = 1$ to each of the masks, $\mathbf{b}_{m-1}^u$ for $u = 1, \ldots, U_{m-1}$. This results in a total of $2U_{m-1}$ potential active states for time-step m given by

$$\hat{\mathbf{b}}_m^p = \begin{cases} [\mathbf{b}_{m-1}^p(2),\, \mathbf{b}_{m-1}^p(3),\, \ldots, \mathbf{b}_{m-1}^p(M),\; 0]^T & p = 1, \ldots, U_{m-1} \\ [\mathbf{b}_{m-1}^{p-U_{m-1}}(2),\, \mathbf{b}_{m-1}^{p-U_{m-1}}(3),\, \ldots, \mathbf{b}_{m-1}^{p-U_{m-1}}(M),\; 1]^T & p = U_{m-1}+1, \ldots, 2U_{m-1}. \end{cases}$$

where $\mathbf{b}_{m-1}^p(n)$ is the n^{th} element of the vector $\mathbf{b}_{m-1}^p$. The STOI sum for the potential states, denoted $\hat{k}_m^p$, is computed from the STOI sum of the active states from the previous time-step, k_{m-1}^u, by adding the contribution from frame m. That is

$$\hat{k}_m^p = \begin{cases} k_{m-1,p} + d\left(\mathbf{x}_m, \tilde{\mathbf{z}}_m^p\right) & p = 1, \ldots, U_{m-1} \\ k_{m-1}^{p-U_{m-1}} + d\left(\mathbf{x}_m, \tilde{\mathbf{z}}_m^p\right) & p = U_{m-1}+1, \ldots, 2U_{m-1}, \end{cases}$$

is computed where $\tilde{\mathbf{z}}_m^p$ is the value of $\tilde{\mathbf{z}}_m$ computed using $\hat{\mathbf{b}}_m^p$. Pairs of potential states with identical $\hat{\mathbf{b}}_m^p$ are identified and only the one with the higher value of $\hat{k}_m^p$ is retained, resulting in $\tilde{U}_m$ potential states where $\tilde{U}_m \leq 2U_{m-1}$.

The next step is to create the list of U_m active states for frame m by pruning the list of potential active states. Without pruning, the number of states would approximately double with each time-step. When pruning, the aim is to retain all states that are either in the optimal sequence of states that ends in time-step T, i.e. that satisfy (4.1), or are near-optimal. In order to achieve this, we partition the potential states into subsets according to their mask density (i.e. the number of ones they contain) and prune each subset independently. The set of all mask vectors corresponding to the potential active states are defined as $E = \left\{ \hat{\mathbf{b}}_m^p \, : \, p = 1, \, \ldots, \, \tilde{U}_m \right\}$. The following pruning strategy is employed:

For each possible mask-density, $g = 0, \, \ldots, \, M$ do:

- Form the mask-density subset $E_g = \left\{ \hat{\mathbf{b}}_m^p \in E \, : \, \sum_{i=1}^{M} \hat{\mathbf{b}}_m^p(i) = g \right\}$.
- From E_g, retain the Q potential states corresponding to the Q highest values of $\hat{k}_m^p$.

The purpose of this pruning strategy is to preserve a variety of different mask patterns, since a mask pattern that is sub-optimal at time-step m of the dynamic programming may become optimal by the final time-step. This is discussed further in Sec. 4.2.2.

The potential states retained after pruning become the activate states. For each active state, the preceding state (from frame $m - 1$) is saved, so that the sequence of states ending in each of the active states can be recovered. The algorithm then proceeds to time-step $m+1$. The forward pass is initialised by pre-appending $M-1$ zeros to the start of $X(m)$ and $Y(m)$, and starts at time-step $m = 1$ with $U_0 = 1$ and $\mathbf{b}_0^1$ initialised as an all-zero vector. Once the initial forward pass has finished, the optimal sequence of states is determined for the forward pass: this is the sequence of states that ends in $s_{T,u_{opt}}$where

$$u_{opt} = \underset{\{u \in 1, \, \ldots, \, U_T\}}{\arg\max} \, \left(k_{T,u} \right),$$

and the mask, $B^{F1}(m)$, corresponding to the optimal sequence of states, is determined.

Having determined $B^{F1}(m)$, the backward pass is then initiated. In the backward pass, $M - 1$ zeros are appended to $X(m)$ and $Y(m)$ and the above algorithm runs backwards in time, through frames $m = T + M - 1, \, \ldots, \, M$, with the potential states in frame m

formed from active states in frame $m+1$. This is equivalent to time-reversing the signals X and Y, applying the forward pass, and then time-reversing the resulting masks.

After completing the forward and backward passes, a final forward pass is initiated that is identical to the initial forward pass, except that the mask is constrained to be equal to $B^{F1}(m)$ and $B^{B}(m)$ in TF bins where $B^{F1}(m) = B^{B}(m)$. In time-step m, the list of potential active states, h_m^p, is formed as

$$\hat{\mathbf{b}}_m^p = \begin{cases} [\mathbf{b}_{m-1}^p(2),\, \mathbf{b}_{m-1}^p(3),\, \ldots,\, \mathbf{b}_{m-1}^p(M),\, B^{F1}(m)]^T & p = 1,\, \ldots,\, U_{m-1},\, B^{F1}(m) = B^{B}(m) \\ \tilde{\mathbf{b}} & B^{F1}(m) \neq B^{B}(m) \end{cases}$$

where

$$\tilde{\mathbf{b}} = \begin{cases} [\mathbf{b}_{m-1}^p(2),\, \mathbf{b}_{m-1}^p(3),\, \ldots,\, \mathbf{b}_{m-1}^p(M),\ 0]^T & p = 1,\, \ldots,\, U_{m-1} \\ [\mathbf{b}_{m-1}^{p-U_{m-1}}(2),\, \mathbf{b}_{m-1}^{p-U_{m-1}}(3),\, \ldots,\, \mathbf{b}_{m-1}^{p-U_{m-1}}(M),\ 1]^T & p = U_{m-1}+1,\, \ldots,\, 2U_{m-1}. \end{cases}$$

The second forward pass produces the final mask estimate.

4.2.1 Rationale for three-stage dynamic programming routine

The rationale for carrying out three passes (forwards, backwards, forwards) is that, because the pruning scheme may occasionally prune the optimal sequence, the forward and backward passes do not always result in identical masks. In TF bins where $B^{F1}(m) = B^{B}(m)$, it is assumed that the probability that both of these mask estimates are correct is higher than the probability that both estimates are wrong. In these bins it is therefore assumed that both mask estimates are correct. Although this assumption may not be true for all

TF bins where $B^{F1}(m) = B^{B}(m)$, by assuming that these estimates are correct and doing a final constrained forward pass, more combinations of mask values can be explored in the other TF bins, i.e. those where the first two estimates were unreliable.

4.2.2 Rationale for pruning strategy

A simpler method of pruning is to retain the Q potential states corresponding to the Q highest values of $\hat{k}_m^p$. One problem with this method is that the states which survive pruning at time-step m may all have similar mask patterns. These masks, despite having optimal and near-optimal STOI sums in frame m, may have comparatively low STOI sums by frame T. To illustrate this, Fig. 4.1a shows an example of a segment taken from the STOI modulation-domain representation of a clean speech signal, $X(m)$, in frequency band $j = 14$. The speech was a recording of the phrase "that noise problem grows more annoying each day" from the TIMIT corpus [45]. The high energy frames in the middle of the segment correspond to the affricate /$t\int$/ in the word "each". Fig. 4.1b shows the optimal binary mask, $B^{F1}(m)$, that results from terminating the forward pass at time-step $m = 17$, when the DSOBM was optimised for speech shaped noise with -10 dB SNR. This mask is obtained using only frames of the clean speech signal, $X(m)$, up to and including frame $m = 17$, i.e. the portion of the signal to the left of the dotted line in Fig. 4.1a and Fig. 4.1b. The mask is equal to one in the frames corresponding to the highest values of $X(m)$ within this portion of the signal. Fig. 4.1c shows the optimal binary mask, $B^{F1}(m)$, that results from terminating the forward pass at time-step $m = 35$. This mask is obtained using all frames of $X(m)$ up to and including frame $m = 35$, i.e. all of the signal visible in Fig. 4.1a. As in Fig. 4.1b, the mask is equal to one in the frames corresponding to the highest values of $X(m)$. However, since the region to the right of the dotted line contains much higher values of $X(m)$ than the region to the left, the masks produced by terminating the forward pass at time-steps $m = 17$ and $m = 35$ are not identical for $m \leq 17$. It is therefore important that in time-steps $m \leq 17$ the pruning algorithm preserves the mask pattern that will eventually become the optimal pattern, i.e. the one that is all zeros. Since it is difficult to predict, at time-step $m = 17$, the mask pattern that will go on to become the optimal pattern at time-step $m = 35$, the pruning scheme preserves a variety of different mask patterns.

After experiments involving utterances from the training set of the TIMIT corpus [45] mixed with speech shaped and babble noise from the NOISEX-92 corpus [161], with a variety of different pruning methods and values of Q, it was found that the adopted pruning strategy with $Q = 200$ was sufficient to result in near-full predicted intelligibility on every noisy utterance, with (3.3) from Sec. 3.5 used to map STOI to intelligibility.

4.3 SOBM for Stochastic noise (SSOBM)

There may be advantages to training a mask estimation algorithm with target masks which were optimised for stochastic noise signals, rather than the deterministic noise signals present in the training data. One motivation for this comes from the model in [98, 99], where it was suggested that IBM-masked speech provides two independent speech cues, a noisy speech signal and a vocoded noise signal, and that it is the vocoded component that is responsible for improving the intelligibility. According to this model, the benefit of binary masking comes from the vocoded noise component, and it therefore seems logical to use a

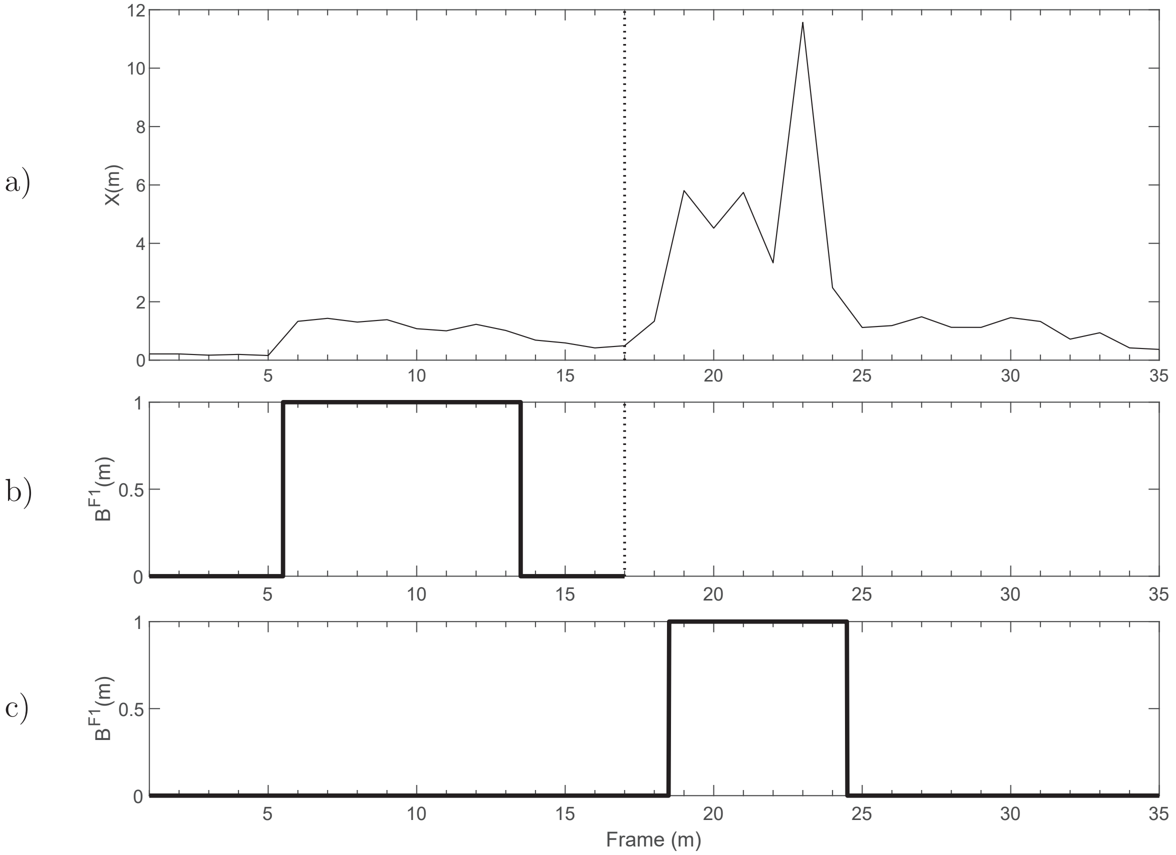


Figure 4.1: Plots of a) a segment taken from the STOI modulation-domain representation of a clean speech signal, $X(m)$, in frequency band $j = 14$, and the optimal binary mask, $B^{F1}(m)$, that results from terminating the forward pass at b) time-step $m = 17$ and c) time-step $m = 35$. The speech was a recording of the phrase "that noise problem grows more annoying each day" from the TIMIT corpus [45]. The high energy frames in the middle of the segment correspond to the affricate /tʃ/ in the word "each". The DSOBM was optimised for speech shaped noise with -10 dB SNR.

mask that is based on the speech alone, i.e. one that is independent of the deterministic noise signals present in the training data. A second motivation is the suggestion in [98] that a mask estimation algorithm is likely to generalise better to new noise conditions if it is trained with a target mask that is independent of the noise, since the estimation algorithm is then more likely to focus on modelling features present in the speech rather than the noise. This may lead to better generalisation since the statistics of noise encountered in a real environment may differ significantly from those in the training set, whereas the features in the speech are likely to be more consistent between the training and testing data sets. We will see in Sec. 4.4 that the STOI-optimal masks are anyway largely independent of the noise, meaning that stochastic variants of the STOI or WSTOI optimal masks which have been optimised for white noise at a fixed SNR can be used as noise-independent and SNR-independent masks with little loss in intelligibility relative to the optimal deterministic masks.

In this section we derive the Stochastic STOI-Optimal Binary Mask (SSOBM), the binary mask that maximises STOI in case of a stochastic noise signal with a known power spectrum. We wish to determine the mask that maximises the expected value of STOI when $X(k, m)$ is known and the noise, $D(k, m) = Y(k, m) - X(k, m)$, is a stationary zero-mean complex Gaussian random variable with variance

$$\langle D(k, m)D^*(k, m)\rangle = \sigma_j^2 \quad \text{for } K_j \leq k < K_{j+1} \tag{4.2}$$

where $\langle\ \rangle$ denotes the expected value and σ_j^2 is assumed to have the same value for all k in frequency band j. We now wish to find the $B_j(m)$ that maximises the expected value of the sum given in (4.1). To make the analysis tractable, it is assumed that clipping is very rare in the stochastic noise case, so that $\widetilde{Y}_j(m) \approx Y_j(m)$ in (2.7), where

$$Y_j(m) = \sqrt{\sum_{k=K_j}^{K_{j+1}-1} |Y(k,\, m)|^2} \quad \text{for } j = 1,\, \ldots,\, J.$$

The clipping stage has similarly been omitted in two modified versions of STOI, Extended STOI (ESTOI) [87] and Deterministic Binaural STOI (DBSTOI) [2]. It is shown in Appendix B that $\sqrt{2}\sigma_j^{-1} Y_j(m)$ has a non-central χ distribution with mean [124, 123] given by

$$\left\langle \sqrt{2}\sigma_j^{-1} Y_j(m) \right\rangle = 2^{-0.5}\pi^{0.5} L_{0.5}^{(0.5\nu_j - 1)} \left(-0.5 R_j(m)\right) \tag{4.3}$$

and second moment [104] given by

$$\left\langle 2\sigma_j^{-2} Y_j^2(m) \right\rangle = \nu_j + R_j(m) \tag{4.4}$$

where $L_n^{(\alpha)}(z)$ is a generalised Laguerre polynomial, $\nu_j = 2\left(K_{j+1} - K_j\right)$ is the degrees of freedom and

$$R_j(m) = 2\sigma_j^{-2} \sum_{k=K_j}^{K_{j+1}-1} |X(k,\, m)|^2$$

is the non-centrality parameter.

Defining the length-M non-centrality vector, $\boldsymbol{r}_{j,m}$, analogous to $\mathbf{x}_{j,m}$ in (2.6), we can write

$$\left\langle \mathbf{z}_{j,m} \right\rangle = 2^{-0.5}\pi^{0.5} \mathbf{b}_{j,m} \circ L_{0.5}^{(0.5\nu_j - 1)} \left(-0.5 \boldsymbol{r}_{j,m}\right) \tag{4.5}$$

where $\circ$ denotes elementwise multiplication and $L_n^{(\alpha)}(\ )$ acts elementwise on a vector argument. Under the assumption that $Y_j(m)$ and $Y_j(n)$ are independent for $m \neq n$, it is shown in Appendix B that

$$\begin{aligned}
\left\langle \|\mathbf{z}_{j,m} - \bar{z}_{j,m}\|^2 \right\rangle &= \left\langle \|\mathbf{z}_{j,m}\|^2 \right\rangle - M \left\langle \bar{z}_{j,m}^2 \right\rangle \\
&= 0.5\sigma_j^2 \frac{M-1}{M} \mathbf{b}_{j,m}^T \left(\nu_j + \mathbf{r}_{j,m}\right) \\
&\quad - \frac{\pi\sigma_j^2}{4M} \left(\mathbf{b}_{j,m}^T L_{0.5}^{(0.5\nu_j - 1)} \left(-0.5\mathbf{r}_{j,m}\right) \right)^2 \\
&\qquad + \frac{\pi\sigma_j^2}{4M} \left\| \mathbf{b}_{j,m} \circ L_{0.5}^{(0.5\nu_j - 1)} \left(-0.5\boldsymbol{r}_{j,m}\right) \right\| .^2
\end{aligned} \tag{4.6}$$

Finally, combining (2.8), (4.5) and (4.6), we can calculate

$$\left\langle d\left(\mathbf{x}_{j,m},\, \mathbf{z}_{j,m}\right) \right\rangle \approx \frac{\left(\mathbf{x}_{j,m} - \bar{x}_{j,m}\right)^T \left\langle \mathbf{z}_{j,m} \right\rangle}{\|\mathbf{x}_{j,m} - \bar{x}_{j,m}\| \sqrt{\left\langle \|\mathbf{z}_{j,m} - \bar{z}_{j,m}\|^2 \right\rangle}}. \tag{4.7}$$

4.4 Results & Evaluation

The SOBM was evaluated using a subset of 80 TIMIT utterances [46] and seven noise types from the NOISEX-92 corpus [161]. Fig. 4.2a shows the average STOI plotted against SNR for speech degraded with each noise type. Most noise types give similar curves, with the exceptions of two noise types: Volvo car noise and machine gun noise. With Volvo car noise, most of the signal energy is concentrated at very low frequencies (see the spectrogram in Fig. A.1 of the Appendix). This means that, at very low SNRs (e.g. $-40 \leq \text{SNR} \leq 0$ in Fig. 4.2a), the amplitude modulation in the speech in most STOI frequency bands is better preserved than with the other noise types, resulting in a higher predicted intelligibility.

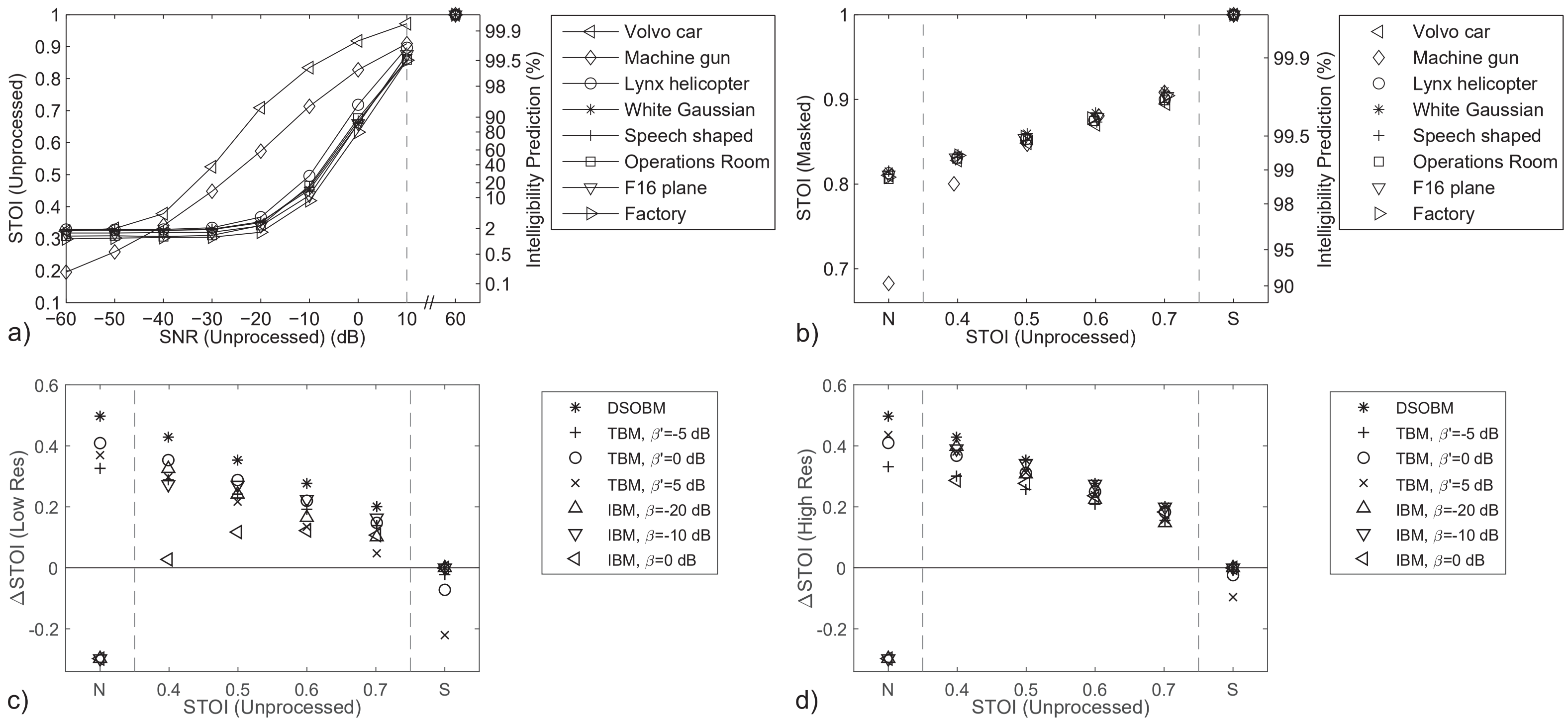


Figure 4.2: a) STOI against SNR for the 8 tested noise types. b) Average STOI of masked speech against STOI before processing for the deterministic algorithm, DSOBM, applied to speech containing different noise types. Average improvement in STOI across all noise types due to application of DSOBM, TBM or IBM masks. The TBMs and IBMs have c) third-octave band resolution and d) full STFT resolution. "N" and "S" denote "noise-only" and "clean speech" input signals, respectively.

Machine gun noise, by contrast, is highly intermittent (see the spectrogram in Fig. A.1 in the Appendix). Since there is effectively no noise in the gaps between the machine gun bursts, the amplitude modulation in the speech in these regions is also well preserved, and the predicted intelligibility of the unprocessed noisy speech also remains fairly high, even at very low SNRs. The right hand axis gives the mapping from STOI to predicted intelligibility from [151] for previously unheard sentences. This mapping from STOI to intelligibility is very task-dependent and so is included in these results for guidance only.

Fig. 4.2b plots the average STOI of the masked speech against the STOI before processing, for the DSOBM applied to speech degraded with different noise types. The symbols “N” and “S” on the horizontal axis denote “noise-only” and “clean speech” input signals, respectively. The DSOBM resulted in a large improvement in STOI for all noise types, at all noise levels except for “S” for which STOI was unchanged from a unprocessed value of 1. With the exception of machine gun noise at very poor SNRs, the DSOBM resulted in an improvement in STOI that was largely independent of noise type and, for all noise levels including “N”, gives an average STOI above 0.8 (corresponding to >98% intelligibility).

Fig. 4.2c shows the average improvement in STOI across all noise types against the STOI before processing, for the DSOBM, and for selected IBMs and TBMs, where the masks all use identical third-octave band frequency resolutions. The DSOBM outperformed all of the tested TBMs and IBMs at all input noise levels other than the clean speech condition, “S”. After the DSOBM, the best performing mask was the TBM with β'=0 dB. The TBMs gave consistently good results for noisy speech, but degraded the intelligibility of clean speech. The IBMs preserved the intelligibility of clean speech, but performed worse than the TBMs with very noisy speech.

In Fig. 4.2d the IBMs and TBMs used the full STFT resolution; this is much higher than that of the DSOBM. For test samples with unprocessed STOIs below 0.6, the DSOBM still gave the greatest improvement in STOI of all tested masks. For unprocessed STOIs of 0.6 and above, the improvement in STOI given by the DSOBM and the IBM with β=-10 dB was approximately equal.

Fig. 4.3 plots the improvement in STOI for different SSOBMs relative to the DSOBM averaged over all noises except machine gun noise, which is plotted separately. The SSOBM gives about 0.02 less STOI improvement than the DSOBM at all noise levels except in the case of clean speech ("S"). To assess the effect of mismatch, we determined the SSOBMs for white-noise at SNRs of -60 and $-10\,\text{dB}$ and applied these masks to all test signals ($\triangleright$, $\triangleleft$ in Fig. 4.3). It can be seen that, except for "S", the STOI improvement differs by less than 0.025 from that of the SSOBM that used a matched noise spectrum and SNR. Even for the "S" case, the difference is <0.06 which corresponds to a negligible difference in intelligibility. This demonstrates that it is possible to use the SSOBM for $-60\,\text{dB}$ white noise as a noise-independent and SNR-independent mask with little loss in intelligibility compared to the optimum DSOBM. The highly non-stationary machine gun noise is plotted separately in Fig. 4.3; its intermittent nature means that the SSOBM performs significantly worse than the DSOBM.

The spectrograms in Fig. 4.4 compare the effect of applying the IBM with $\beta = -10$ dB (plot (c)) and the SSOBM (plot (d)) to speech containing white Gaussian noise at -10 dB SNR. The SSOBM was optimised for the correct SNR and noise type. The speech is part of an utterance of the phrase "a big goat idly ambled through the farmyard". All the

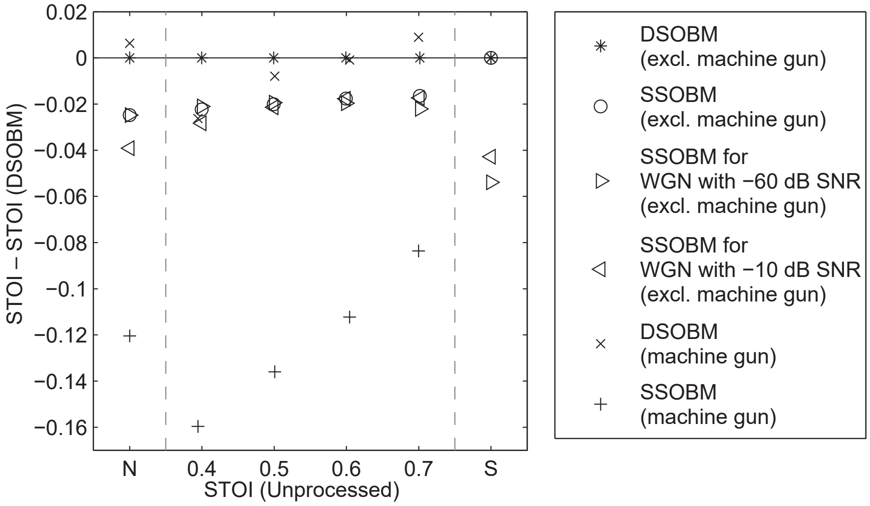


Figure 4.3: Improvement in STOI for different masks relative to the DSOBM averaged over all noises other than machine gun noise, which is plotted separately.

spectrograms have STOI's frequency resolution, i.e. 15 third-octave bands with the centre frequency of the first band equal to 150 Hz. Plot (e) shows the difference between the intermediate STOI measure, $d\left(\mathbf{x}_{j,m},\, \tilde{\mathbf{y}}_{j,m}\right)$, from (2.8), computed on corresponding pairs of TF cells in signals (d) and (c); in this plot a positive value (coloured blue) indicates that the Oracle SSOBM mask outperforms the Oracle IBM mask. Two regions of the spectrogram (A and B) are highlighted in plot (e), and all TF cells, j, m, contributing to $d\left(\right)$ in these regions are highlighted in plots a-d. In both the high speech energy (A) and low speech energy (B) regions the SSOBM-masked speech contains a temporal modulation pattern which is closer to the modulation pattern in the clean speech than the modulation pattern in the IBM-masked speech. In region A, the speech energy is high, and the IBM is almost all ones, which means the IBM-masked speech (plot (c)) in this region is similar to the noisy speech. In contrast, the SSOBM-masked speech (plot (d)) has a modulation pattern more similar to the clean speech. In region B, the speech energy is low, and the

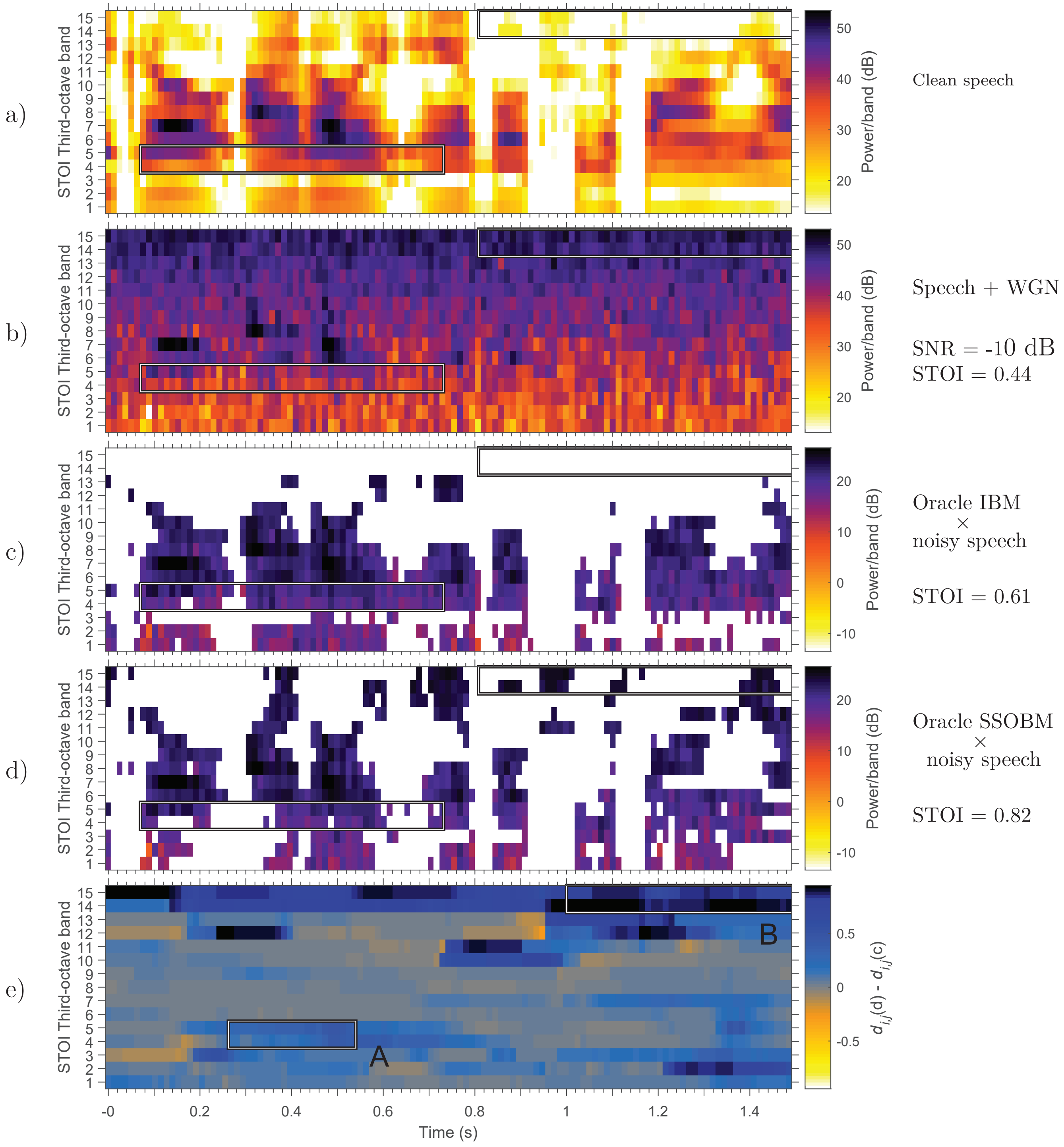


Figure 4.4: Spectrograms of a) part of an utterance of the phrase "a big goat idly ambled through the farmyard", b) the utterance after adding white Gaussian noise (WGN) with an SNR of -10 dB, and the noisy speech spectra after applying c) the IBM with $\beta = -10$ dB, d) the SSOBM optimised for the correct SNR and noise type, and e) the difference between the intermediate STOI measure, $d\left(\mathbf{x}_{j,m}, \tilde{\mathbf{y}}_{j,m}\right)$, computed on corresponding pairs of TF cells in the signals produced by applying masks (d) and (c) to the noisy speech. Two regions (A and B) are highlighted in plot (e), and all TF cells, j, m, contributing to $d\left(\right)$ in these regions are highlighted in plots a-d.

IBM is all zeros, which means the IBM-masked speech in this region has no modulation pattern, whilst the SSOBM-masked speech again has a modulation pattern similar to the clean speech. The better-matching modulation pattern produced by the SSOBM-masked speech in both regions is confirmed by plot (e), which is mostly blue and black in these regions, indicating higher STOI contributions, $d\,()$, from the SSOBM-masked speech than the IBM-masked speech.

Fig. 4.5 shows the distribution of the difference between the intermediate STOI measure, $d\,()$, computed on corresponding pairs of modulation vectors in the SSOBM-processed and IBM-processed noisy speech signals shown in 4.4d and 4.4c. In 86% of TF cells $d\,()$ computed on the SSOBM-processed speech was higher than $d\,()$ computed on the IBM-processed speech, and in a significant number of cells it was much higher.

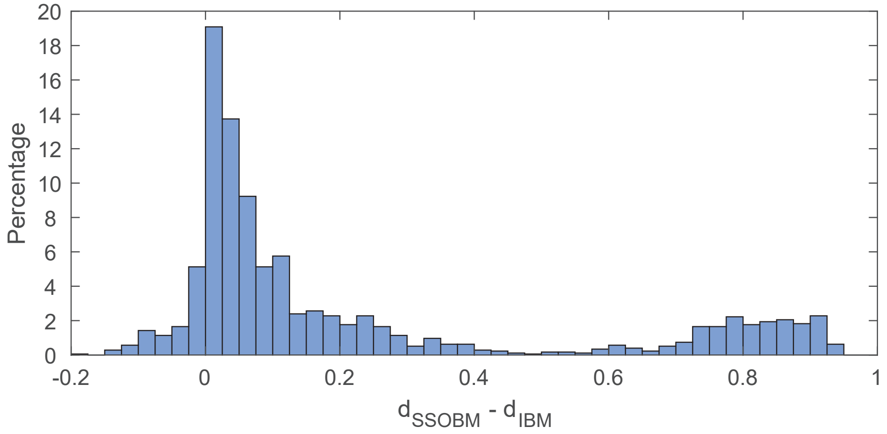


Figure 4.5: Distribution of the difference between the intermediate STOI measure, $d\,()$, from (2.8), computed on corresponding pairs of modulation vectors in the SSOBM-processed and IBM-processed noisy speech signals shown in Fig. 4.4d and 4.4c.

4.5 High-resolution Stochastic WOBM (HSWOBM)

Just as the SSOBM is the binary mask that optimises STOI, we define the Stochastic WSTOI-Optimal Binary Mask (SWOBM) to be the binary mask that optimises WSTOI. The SWOBM is computed in a similar manner to the SSOBM, using dynamic programming as described in Sec. 4.2. For the SWOBM, analogously to (4.1) we compute

$$B_j(m) = \underset{\{B_j(m):m=1,\,\ldots,\,T\}}{\arg\max} \left(\sum_{m=1}^{T} I_{j,m} \left\langle d\left(\mathbf{x}_{j,m},\, \mathbf{z}_{j,m}\right)\right\rangle \right). \tag{4.8}$$

We have seen that by applying oracle STOI-optimal masks we can obtain large improvements in STOI. However, the quality of the resulting speech, as predicted by PESQ, is very poor. One possible reason for the poor speech quality may be the low frequency resolution of STOI-optimal masks, which have only $J = 15$ frequency bands, and are therefore unable to resolve the fine detail in the speech, such as the harmonics of the fundamental frequency of the speaker. Since this detail cannot be resolved by the mask, it will not be reintroduced into the noisy speech when the mask is applied. Furthermore, if the harmonics cannot be resolved, noise cannot be rejected in the spaces between the harmonics, which may further reduce the speech quality. In this section we therefore present a modified version of the SWOBM, denoted the High-resolution SWOBM (HSWOBM). The HSWOBM is identical to the SWOBM except that it optimises a version of the WSTOI metric that has been modified to have a higher frequency resolution. In the modified WSTOI metric the correlation comparison is computed in bands which occupy only a single STFT frequency bin, rather than in third-octave bands. In other words, instead of (2.5) we have

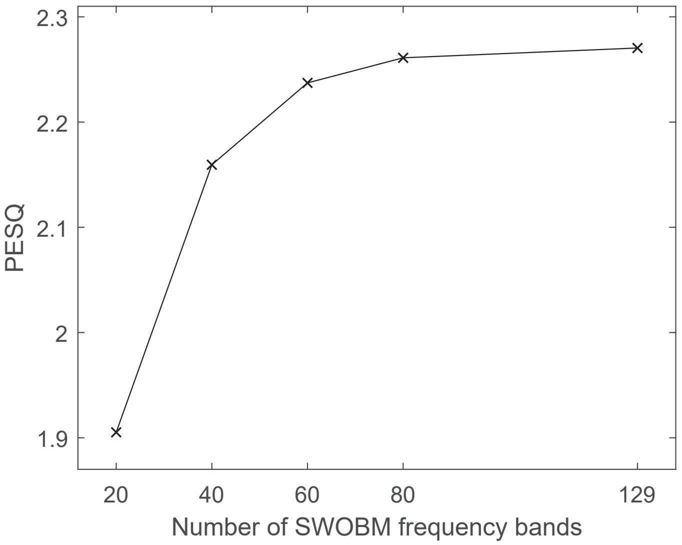
$$X_j(m) = |X(j, m)| \quad \text{for } j = 1, \ldots, J \tag{4.9}$$

where $J = 256/2 + 1 = 129$.

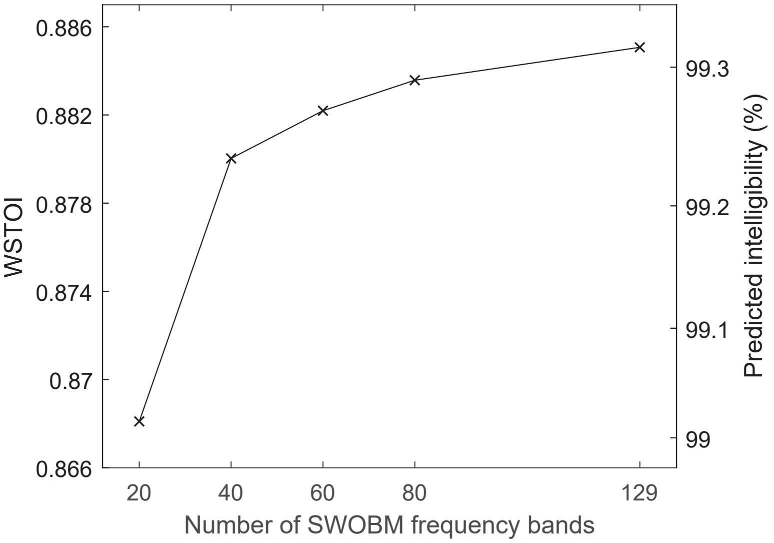
Figures 4.6a and 4.6b show the PESQ and WSTOI metrics computed on noisy speech processed with a modified version of the SWOBM against the frequency resolution of the masks. The SWOBMs used to compute Fig. 4.6 were modified to optimise a version of WSTOI that computed the correlation comparison with modulation vectors formed within $J = \{20, 40, 60, 80\}$ ERB-spaced bands rather than the $J = 15$ third-octave bands used in STOI and WSTOI. The J bands had centre frequencies equally spaced on the ERB scale with the centre frequencies of the first and last bands equal to 100 Hz and 5 kHz, respectively. The Equivalent Rectangular Bandwidth (ERB) scale, whose transformation is denoted by $\Phi(f)$ and whose inverse derivative, $\frac{df}{d\Phi(f)}$, approximates the bandwidths of the human auditory filters, can be approximated as

$$\Phi(f) = 11.17268 \cdot \ln\left(1 + \frac{46.06538 \cdot f}{f + 14678.49}\right) \tag{4.10}$$

between 0.1 and 6.5 kHz [15, 119]. The HSWOBM (with $J = K/2 + 1 = 129$ bands where K is the DFT length) is also included in the figure. The masks were optimised for white Gaussian noise at -5 dB SNR. To generate the noisy speech utterances 100 TIMIT [46] utterances were mixed with extracts of babble and speech shaped (SS) noise from the RSG.10 [139] database. The noisy utterances had WSTOI scores corresponding to predicted intelligibilities of {60, 70, 80, 90} % using the mapping, (3.3), between WSTOI and predicted



(a)



(b)

Figure 4.6: Plots of a) PESQ and b) WSTOI for noisy speech processed with the SWOBM against the number of frequency bands in the SWOBM. The HSWOBM, with the full DFT resolution of 129 bands, is also included in the figures.

intelligibility from Chapter 3, which correspond to SNRs of $\{-2.7, -1.8, -0.6, 1.1\}$ dB for babble noise and $\{-4.0, -3.0, -1.7, 0.2\}$ dB for SS noise. From Fig. 4.6a it can be seen that, as the frequency resolution of the masks increases, PESQ predicts that the quality of the resulting speech also increases. The largest PESQ occurred when the modified SWOBM used the full STFT resolution (i.e. when it was identical to the HSWOBM). WSTOI also increases with frequency resolution, although this corresponds to a very small increase in predicted intelligibility as can be seen from the right hand axis of Fig. 4.6b.

Fig. 4.7 shows spectrograms of the SWOBM with $J = 20$ frequency bands and 129 frequency bands (the HSWOBM) for a single speech utterance. In comparison with the SWOBM the HSWOBM appears to capture more fine detail in the speech spectra such as the harmonics of the fundamental frequency of the speaker. This may have the effect of increasing the noise rejection in the spaces between the harmonics and hence increasing the speech quality at higher resolutions, as predicted by PESQ in Fig. 4.6. The HSWOBM also appears to capture more information about the position of the formant frequencies, around which the width of the harmonics is increased. This may also contribute to the increased PESQ at higher resolutions.

4.6 Smoothed HSWOBM (SHSWOBM)

We have seen from Fig. 4.6 that the oracle HSWOBM performs well in terms of PESQ and WSTOI. The upper plot of Fig. 4.7 shows the spectrogram of a clean speech utterance of part of the phrase “or borrow some money from someone and go home by bus”, the middle plot shows the corresponding HSWOBM for white Gaussian noise at -5 dB SNR, and the

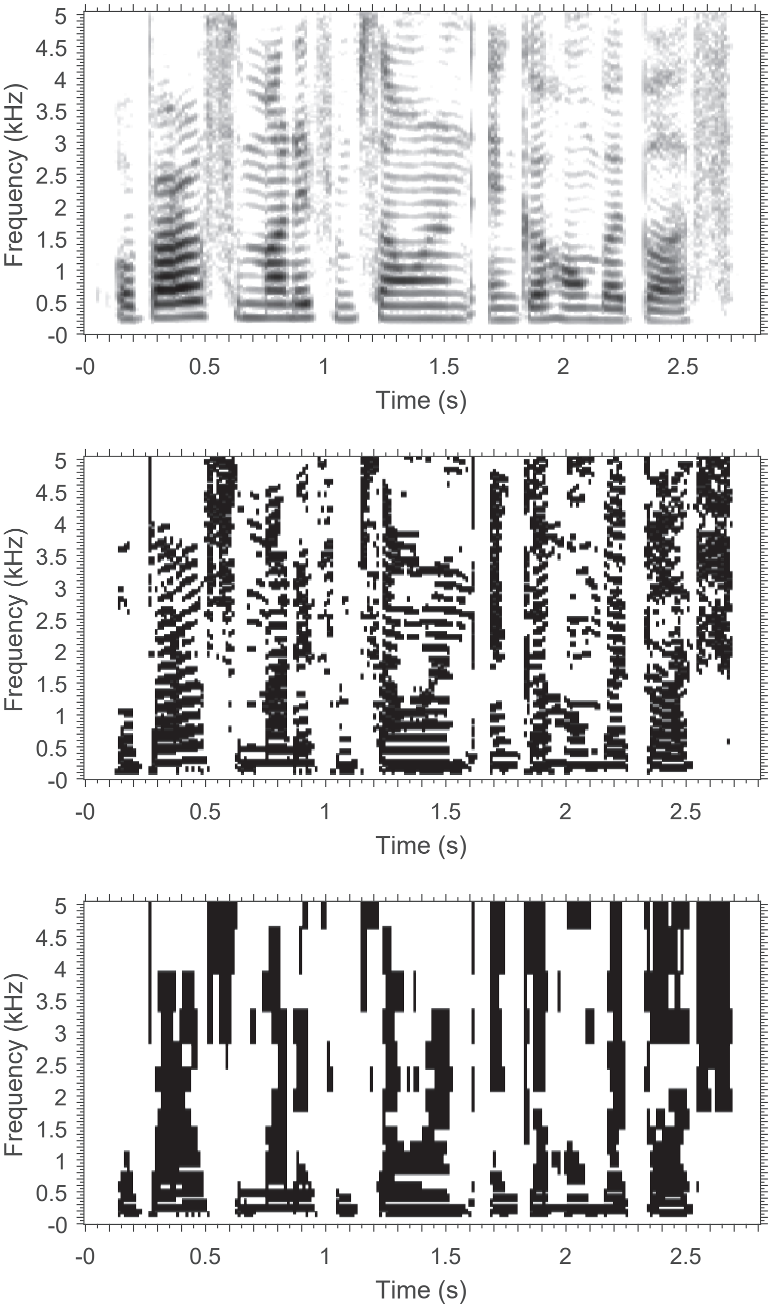


Figure 4.7: Spectrograms of (upper plot) a clean speech utterance of part of the phrase "or borrow some money from someone and go home by bus", (middle plot) the HSWOBM of the utterance for white Gaussian noise at -5 dB SNR, and (lower plot) the SWOBM formed in ERB bands with $J = 20$ for the same stochastic noise signal.

lower plot shows the corresponding SWOBM formed in ERB bands with $J = 20$ for the same stochastic noise signal. It can be seen that the HSWOBM contains a lot of detail including fundamental frequency harmonics. A mask estimator trained on the HSWOBM will therefore have to account for the pitch-dependency of the mask. Because estimating the pitch of a very noisy speech signal is difficult to do reliably, we decided to evaluate an additional smoothed version of the HSWOBM in which the pitch information has been largely removed. In this section, we therefore investigate the effect of removing detail from the HSWOBM, in order to create a target mask that will be easier to estimate. Detail in the HSWOBM which can only be resolved at higher frequency resolutions is discarded. The resulting mask is termed the Smoothed-HSWOBM (SHSWOBM).

To obtain the SHSWOBM we optimise a modified version of WSTOI. As with the HSWOBM, the modified WSTOI computes the correlation comparison on modulation vectors computed in 129 bands centred on each STFT frequency bin. However, unlike the HSWOBM which uses 1 STFT bin per band, the SHSWOBM uses 50% overlapping triangular bands encompassing a number of neighbouring STFT bins. The width of the triangular bands was chosen to be 13 STFT bins, which corresponds to 508 Hz. Since the fundamental frequency of most voiced speech is below $504/2 = 252$ Hz [156, 9], at least 2 harmonics of the fundamental frequency will normally be contained within each window.

The upper plot in Fig. 4.9 shows the HSWOBM for a speech utterance in which the harmonic structure is clearly visible. The middle plot shows the result after smoothing in frequency to give the SHSWOBM. It can be seen that the overall structure of the mask has been preserved but the fine detail of the harmonics has been largely eliminated.

Fig. 4.8 shows the effect on PESQ and WSTOI of multiplying the noisy speech by the

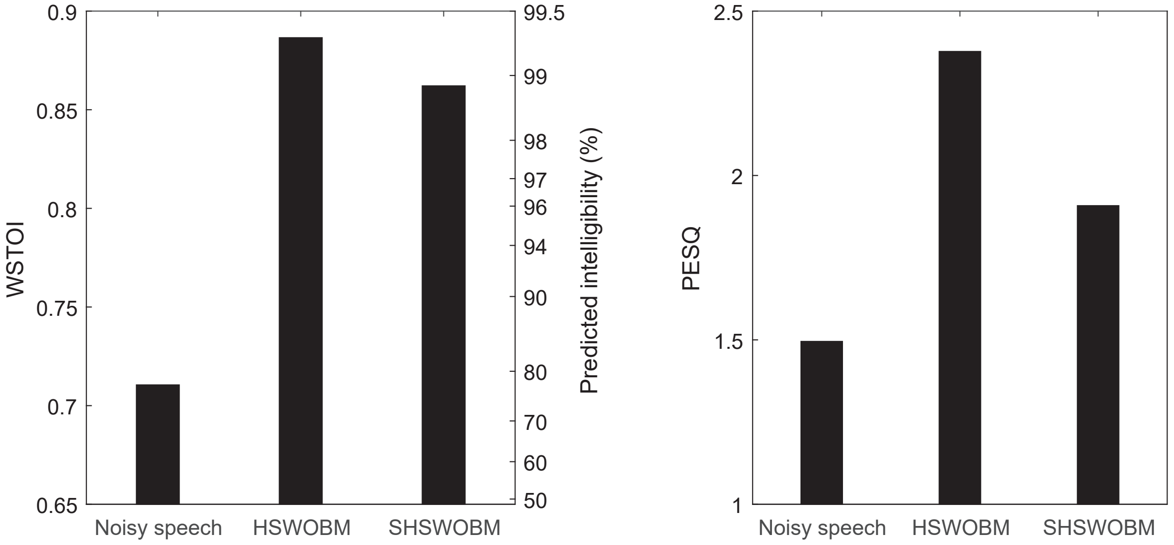


Figure 4.8: Effect on a) PESQ and b) WSTOI of applying the HSWOBM and SHSWOBM to speech containing babble noise with SNRs of $\{-2.7, -1.8, -0.6, 1.1\}$ dB and SS noise with SNRs of $\{-4.0, -3.0, -1.7, 0.2\}$ dB.

HSWOBM and SHSWOBM. To generate the plots, 100 TIMIT [46] utterances were mixed with extracts of babble and speech shaped noise from the RSG.10 [139] database to form the noisy speech. The noisy utterances had WSTOI scores corresponding to predicted intelligibilities of {60, 70, 80, 90} % using the mapping between WSTOI and predicted intelligibility, (3.3), from Chapter 3, which correspond to SNRs of $\{-2.7, -1.8, -0.6, 1.1\}$ dB for babble noise and $\{-4.0, -3.0, -1.7, 0.2\}$ dB for SS noise. The HSWOBM caused a significant increase in both WSTOI and PESQ. With the SHSWOBM, most of this improvement in WSTOI is preserved. Using (3.3) the reduction in WSTOI between the speech processed with the HSWOBM and the SHSWOBM corresponds to a difference in predicted intelligibility of only 0.52 %. The difference in predicted quality as predicted by PESQ between the speech processed with the HSWOBM and the SHSWOBM is, however, substantial. Comparing Fig. 4.8 and Fig. 4.6, it can be seen that applying the HSWOBM results in similar PESQ and WSTOI scores to applying the SWOBM with $J = 20$ frequency bands. This may be due to the fact that, even with $J = 20$ bands, the SWOBM has greater frequency resolution at most frequencies: 16 of the 20 bands have fewer than 13 STFT bins (the width of the SHSWOBM bands). Nonetheless, despite the lower WSTOI and PESQ scores resulting from applying the SHSWOBM compared with higher resolution targets, we believe a mask estimator trained on the SHSWOBM may outperform estimators trained on targets which are pitch-dependent and therefore more challenging to estimate.

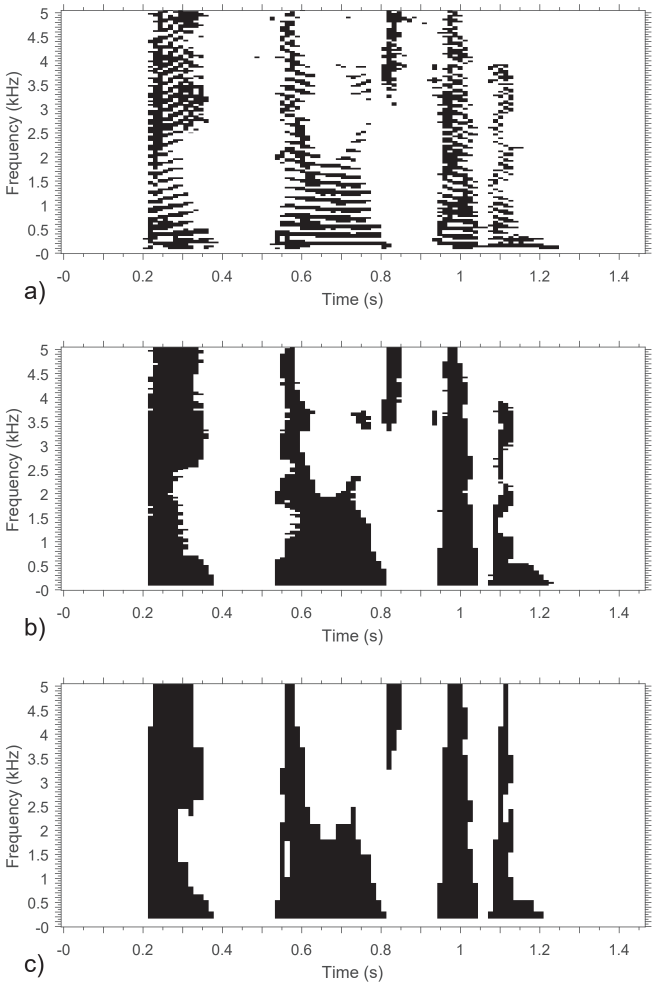


Figure 4.9: Spectrograms of a) an oracle HSWOBM, b) the SHSWOBM, c) the CHSWOBM formed using a library of $D = 100$ patterns.

4.7 Compact-HSWOBM (CHSWOBM)

In this section we present another modified version of the HSWOBM, denoted the Compact-HSWOBM (CHSWOBM). In the CHSWOBM, the information in the HSWOBM that is important for speech intelligibility is compressed into a more compact form. This is done by reconstructing the HSWOBM using a library of mask patterns. We believe that, if the important information is presented to the mask estimation algorithm in a more compact form, it may be easier for the mask estimator to learn the mapping between features and mask. A further motivation for the CHSWOBM is that, since the library of mask vectors used to construct the estimated masks was obtained from oracle HSWOBMs, which do not depend on any particular realisation of noise and therefore do not contain noise artefacts, we expect the approach to be less prone to introducing distorting artefacts into the processed speech than other approaches.

We believe that the HSWOBM is suitable for compression due to the following observations:

> (i) The HSWOBMs of real speech signals contain recurring patterns across time and frequency.
>
> (ii) WSTOI is insensitive to the mask value in some TF cells. In these cells, the impact on WSTOI of swapping the mask value is negligible.

We know from (i) that many of the mask vectors which occur in real speech signals have a very similar binary pattern, and from (ii) it follows that some of these masks can likely be interchanged with little loss of WSTOI or intelligibility. To exploit these observations we can therefore use a library of D mask patterns where $D \ll 2^J$.

4.7.1 Overview of steps in computing the CHSWOBM

The first step in computing an oracle CHSWOBM is to compute corresponding oracle SHSWOBM. The CHSWOBM is then obtained by selecting the optimal sequence of mask patterns from a library of D mask patterns that contains a subset of all 2^J possible patterns. The library is the set

$$\zeta = \left\{ \boldsymbol{v}_i \, : \, i = 1, \, \ldots, \, D, \, \boldsymbol{v}_i \in \{0, \, 1\}^J \right\},$$

where each entry, $\boldsymbol{v}_i$, is a binary vector representing the mask values in each of J frequency bands for a single instance in time. The SHSWOBM in time frame m can be represented by a binary vector $\boldsymbol{r}_m = [B_S(0, m), \, \ldots, \, B_S(K/2, m)]$, where $B_S(k, m)$ is the SHSWOBM in frequency bin k of frame m, and K is the DFT length, so that $J = K/2 + 1$. To form the CHSWOBM, in each frame, $m = 1, \, \ldots, \, T$, the SHSWOBM vector, $\boldsymbol{r}_m$, is replaced with the optimal library mask vector, $\boldsymbol{v}^m_{opt}$, where

$$\boldsymbol{v}^m_{opt} = \underset{\{\boldsymbol{v}_i \in \zeta\}}{\arg\min} \left(\Omega\left(\boldsymbol{\rho}_m, \, \boldsymbol{r}_m, \, \boldsymbol{v}_i\right) = \sum_{j=1}^{J} \boldsymbol{\rho}_m(j) \left| \boldsymbol{r}_m(j) - \boldsymbol{v}_i(j) \right| \right), \tag{4.11}$$

where $\boldsymbol{\rho}_m \in \mathbb{R}^J$ are weights. The loss function, Ω, measures the approximate reduction in the expected value of the high resolution version of WSTOI that is caused by substituting the SHSWOBM in frame m, i.e. $\boldsymbol{r}_m$, with the optimal library mask vector, $\boldsymbol{v}^m_{opt}$. This loss function takes account of observation (ii) from Sec. 4.7 and replaces each $\boldsymbol{r}_m$ with the $\boldsymbol{v}_i$ which minimises the damage to WSTOI under the assumption that WSTOI is not impacted by the library-based reconstruction in the neighbouring frames. The weights,

$\boldsymbol{\rho}_m$, are the product of two separate weights: a band importance weighting, $\boldsymbol{\alpha}$, and a WSTOI sensitivity weighting, $\boldsymbol{\beta}_m$. The band importance weighting is included to account for fact that the high resolution version of WSTOI weights all STFT bins equally, despite their contribution to intelligibility being unequal. We define

$$\boldsymbol{\rho}_m = \boldsymbol{\alpha} \circ \boldsymbol{\beta}_m,$$

where the symbol $\circ$ denotes the Hadamard product,

$$\boldsymbol{\alpha} = \left[\boldsymbol{\alpha}\left(0\right), \boldsymbol{\alpha}\left(2\right), \ldots, \boldsymbol{\alpha}\left(K/2\right)\right]^{\mathrm{T}},$$

where $\alpha\left(k\right)$ is the importance of frequency bin k using the band importance function from Table 3 of [4], and

$$\boldsymbol{\beta}_m = \left[\beta\left(0, m\right), \beta\left(1, m\right), \ldots, \beta\left(K/2, m\right)\right]^{\mathrm{T}},$$

where $\beta\left(k, m\right)$ measures, for time-frequency bin (k, m), the expected reduction in the high resolution version of WSTOI that would result from applying the oracle SHSWOBM with the mask value in that bin inverted (i.e. the effect of a single error in the oracle mask at (k, m)), in the stochastic noise case. The oracle SHSWOBM with TF unit (a, b) inverted is defined as

$$B_S^{a,b}\left(k, m\right) = \begin{cases} 1 - B_S\left(k, m\right) & (k, m) = (a, b) \\ B_S\left(k, m\right) & \text{otherwise} \end{cases}.$$

We have

$$\beta\left(k,m\right)=\sum_{n=m}^{m+M-1}I_{k,n}\left(\left\langle d\left(\mathbf{x}_{k,n},\,\boldsymbol{b}_{k,n}\circ\mathbf{y}_{k,n}\right)\right\rangle-\left\langle d\left(\mathbf{x}_{k,n},\,\boldsymbol{b}_{k,n}^{k,m}\circ\mathbf{y}_{k,n}\right)\right\rangle\right),$$

where $\boldsymbol{b}_{k,m}$ is the modulation vector ending in (k,m), from the high resolution of WSTOI, formed from the mask B_S, i.e.

$$\boldsymbol{b}_{k,m}=\left[B_S\left(k,m-M+1\right),B_S\left(k,m-M+2\right),\,\ldots,\,B_S\left(k,m\right)\right]^{\mathrm{T}},$$

$\boldsymbol{b}_{k,n}^{k,m}$ is formed in the same way from $B_S^{k,m}\left(k,n\right)$, $\left\langle d\left(\cdot\right)\right\rangle$ is computed using (4.7), and $I_{k,m}$ is the WSTOI weight as defined in (3.1).

4.7.2 CHSWOBM Library

To construct the library of mask patterns, ζ, we apply a clustering procedure to a corpus of training utterances. The purpose of the procedure is to group together similar masks vectors, $\boldsymbol{r}_m$, so that they can be represented by a single vector, $\boldsymbol{v}_i$. The procedure employs k-means clustering with the loss function, Ω, from (4.11). The clusters form the set

$$\Pi=\left\{\boldsymbol{\kappa}_i\,:\,i=1,\,\ldots,\,D,\,\boldsymbol{\kappa}_i\in\{0,\,1\}^J\right\},$$

where $\boldsymbol{\kappa}_i$ are the cluster centres. The training utterances are concatenated to create one extended utterance with a corresponding oracle SHSWOBM, $B_S\left(k,m\right)$, training mask vectors, $\boldsymbol{r}_m=\left[B_T\left(0,m\right),\,\ldots,\,B_S\left(K/2,m\right)\right]$ and corresponding values of $\boldsymbol{\rho}_m$. The training data for the k-means clustering is the set of pairs $\Psi=\left\{\left(\boldsymbol{r}_m,\,\boldsymbol{\rho}_m\right)\,:\,m=1,\,\ldots,\,T\right\}$.

The following steps are used to construct the CHSWOBM library:

1. The cluster centres, Π, are initialised as unique binary vectors selected randomly from the training vectors, Ψ.

2. The following steps are repeated until convergence:

   (a) The training pair subsets Ψ_i are formed where

   $$\Psi_i = \left\{(\boldsymbol{r}_m,\, \boldsymbol{\rho}_m) \,:\, \boldsymbol{\kappa}_{opt}^m = \boldsymbol{\kappa}_i \,,\, \Psi_i \in \Psi\right\} \quad \text{for}\, i = 1,\, \ldots,\, D$$

   and

   $$\boldsymbol{\kappa}_{opt}^m = \underset{\{\boldsymbol{\kappa}_i \in \Pi\}}{\arg\min} \left(\Omega\left(\boldsymbol{\rho}_m,\, \boldsymbol{r}_m,\, \boldsymbol{\kappa}_i\right)\right).$$

   (b) The set of cluster centres, Π, are recomputed as

   $$\boldsymbol{\kappa}_i = \underset{\left\{\boldsymbol{\kappa}_i \in \{0,\,1\}^J\right\}}{\arg\min} \left(\sum_{(\boldsymbol{r}_m,\, \boldsymbol{\rho}_m) \in \Psi_i} \Omega\left(\boldsymbol{\rho}_m,\, \boldsymbol{r}_m,\, \boldsymbol{\kappa}_i\right)\right) \quad \text{for}\, i = 1,\, \ldots,\, D. \tag{4.12}$$

   The solutions to (4.12) can be obtained independently for each band, $j = 1,\, \ldots,\, J$, by evaluating both possibilities, i.e. $\boldsymbol{\kappa}_i(j) = 0$ and $\boldsymbol{\kappa}_i(j) = 1$.

3. The final set of cluster centres becomes the library of mask patterns, i.e. $\zeta = \Pi$.

Convergence occurs when the cluster centres, Π, are unchanged for two successive iterations.

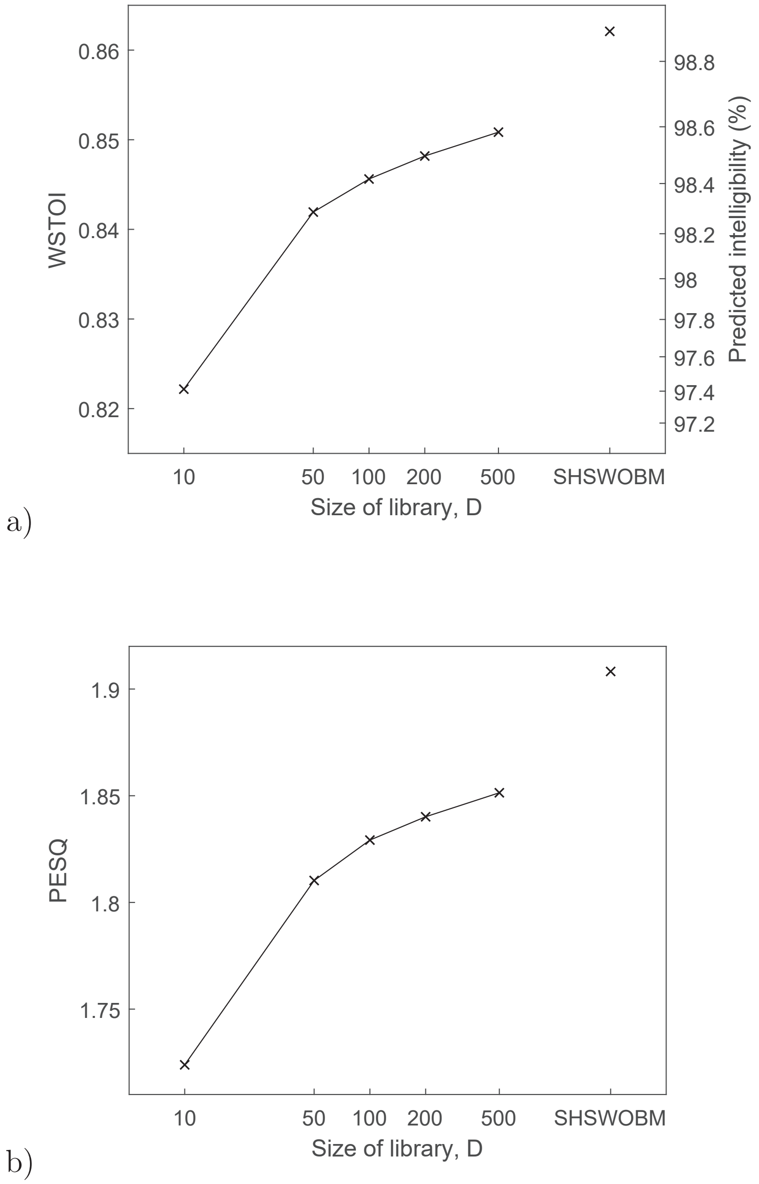


Figure 4.10: a) WSTOI and b) PESQ of the noisy speech after processing with CHSWOBMs as a function of the size of the library used to compute the CHSWOBM (plotted on a log axis). WSTOI and PESQ are also plotted for the SHSWOBM.

4.7.3 Experiments

As training data for the k-means clustering, 200 utterances from the TIMIT training set [46] were randomly selected. As test data, a further 200 utterances were selected and mixed with extracts of babble and speech shaped noise from the RSG.10 [139] database. As in the previous section, the noisy utterances had WSTOI scores corresponding to predicted intelligibilities of $\{60, 70, 80, 90\}$ % which correspond to SNRs of $\{-2.7, -1.8, -0.6, 1.1\}$ dB for babble noise and $\{-4.0, -3.0, -1.7, 0.2\}$ dB for SS noise. The training masks were optimised for white Gaussian noise at -5 dB SNR.

Fig 4.9c shows the effect of approximating the HSWOBM in Fig 4.9a by the CHSWOBM. It can be seen that the harmonic structure has been completely eliminated and that there are some distortions to the shape of the mask although its overall structure has been preserved. Fig. 4.10a and Fig. 4.10b respectively show WSTOI and PESQ as a function of the size of the library, D. It can be seen that below $D = 50$, WSTOI decreases rapidly with decreasing D. For $D = 100$, predicted intelligibility is within 1% of the predicted intelligibility of speech obtained by applying masks which have not been reconstructed from the library. Hence $D = 100$ was chosen for the library size. For this value of D, the PESQ score is about 0.8 less than that of the SHSWOBM.

4.8 Summary

This chapter presented a new oracle mask, the SOBM, that explicitly maximises the STOI objective intelligibility metric. For deterministic additive noise, the DSOBM always results in a higher STOI value than other oracle masks. By assuming a stochastic noise signal, the SSOBM achieves a performance close to the DSOBM for a wide range of SNRs and noise types, even when the noises used for mask optimisation and testing are mismatched. Analogously to the SSOBM we then defined the SWOBM which optimises the WSTOI intelligibility metric for stochastic noise signals. An extension to the SWOBM is the HSWOBM which has an increased frequency resolution and results in speech with a higher predicted quality. The SHSWOBM is a smoothed version of the HSWOBM in which the pitch information, which is difficult for a mask estimation algorithm to reliably estimate from noisy speech, has been largely removed. The CHSWOBM is a modified version of the SHSWOBM in which the information that is important for speech intelligibility is compressed into a more compact form. These modified version of the HSWOBM largely preserve its intelligibility benefits but, in oracle form, result in significantly lower quality. However, it is expected that they will be easier to estimate from noisy speech.

In the next chapter we propose an algorithm for estimating an oracle mask from noisy speech. We then compare the effect of using different oracle masks, features and estimation algorithms on the predicted intelligibility of the speech produced by applying the estimated mask.

Chapter 5

Optimal mask estimation

5.1 Introduction

In this chapter we present a technique for estimating a binary mask from noisy speech. This technique encompasses the “Extract features” and “Estimate TF mask” modules of the mask-based enhancer shown in Fig. 5.1 (repeated from Fig. 2.1). We will begin by discussing the “Extract features” module in the figure. As target masks we will evaluate the stochastic variants of the WSTOI-optimal binary masks, HSWOBM, SHSWOBM and CHSWOBM, presented in the previous chapter.

5.2 Features for mask estimation

In this section we discuss the feature sets that are used to estimate the oracle mask from noisy speech. The features are extracted from the noisy speech and passed as inputs to a noise estimation algorithm, as shown in Fig. 5.1. Since the target masks depend on the

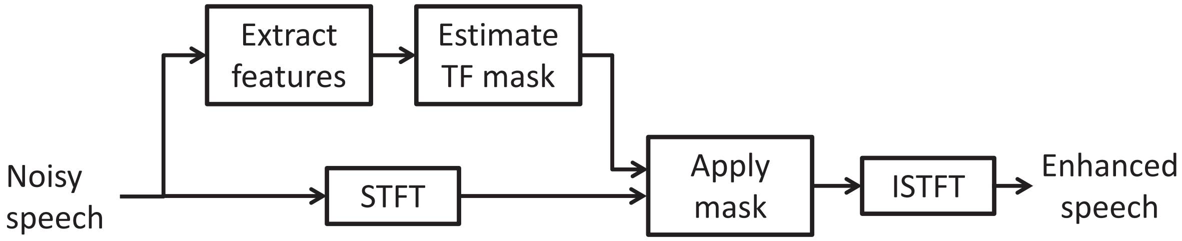


Figure 5.1: Overview of a typical mask-based enhancer.

speech alone, i.e. they are independent of the noise, the features selected are intended to help identify TF cells containing speech energy. A diagram is presented in Fig. 5.2 showing the procedure for computing the feature set which comprises three subsets. The first step in the computation of all three subsets is to normalise the noisy speech to an estimated active level of 0 dB using the PEFAC algorithm from [57]. The reason for normalising the active level of the noisy speech is to ensure that the mask estimation algorithm is level-independent. The feature set is then computed at time intervals matching the intervals between the mask bins, i.e. once per STFT frame.

Each of the three feature subsets has Ψ features, giving $\Omega = 3\Psi$ features in total. The first subset is formed from the TF gains estimated by the classical Log-MMSE speech enhancement algorithm from [37]. Since this algorithm incorporates a noise estimation algorithm, we expect that these features will help the mask estimator to generalise to unseen noise types. The second feature subset is formed from the enhanced speech produced by applying the Log-MMSE gains from the first feature subset to the noisy speech. The third feature subset is an estimate of the local voiced-speech-plus-noise to noise ratio in different TF regions and is obtained with the aid of a pitch estimator. This feature subset is included as we believe that the presence of voiced speech in a TF region gives a strong indication of the mask value.

5.2.1 Feature subset 1: classical enhancer gains

The first feature subset is formed from the TF gains estimated by the classical Log-MMSE speech enhancement algorithm from [37], which minimises the mean-squared error in the log-spectral amplitudes. The motivation for including these features is that the gain is expected to be high when speech is present and low when speech is absent. Here we present a brief overview of this algorithm whose constituent blocks are enclosed by a dashed line in Fig. 5.2. The noisy speech is first converted into the STFT domain using overlapping Hamming analysis windows. Recall that $X(k,\, m)$, $N(k,\, m)$ and $Y(k,\, m)$ denote the complex STFT coefficients of the clean speech, the noise and noisy speech respectively in frequency bin k of frame m. The STFT coefficients of the speech and noise are modelled as statistically independent complex Gaussian random variables. We need to determine the gain function $G(k,\, m)$ in frequency bin k of frame m that satisfies

$$G(k,\, m)\, |Y(k,\, m)| = \exp\left\{E\left[\log|X(k,\, m)| \mid Y(k,\, m)\right]\right\}$$

where $E\left[\cdot\right]$ is the expectation operator. It is shown in [37] that

$$G(k,\, m) = \frac{\xi(k,\, m)}{1+\xi(k,\, m)} \exp\left(\frac{1}{2}\int_{v(k,\, m)}^{\infty} \frac{e^{-t}}{t} dt\right)$$

where

$$\xi(k,\, m) \triangleq E\left[|X(k,\, m)|^2\right] / E\left[|N(k,\, m)|^2\right]$$

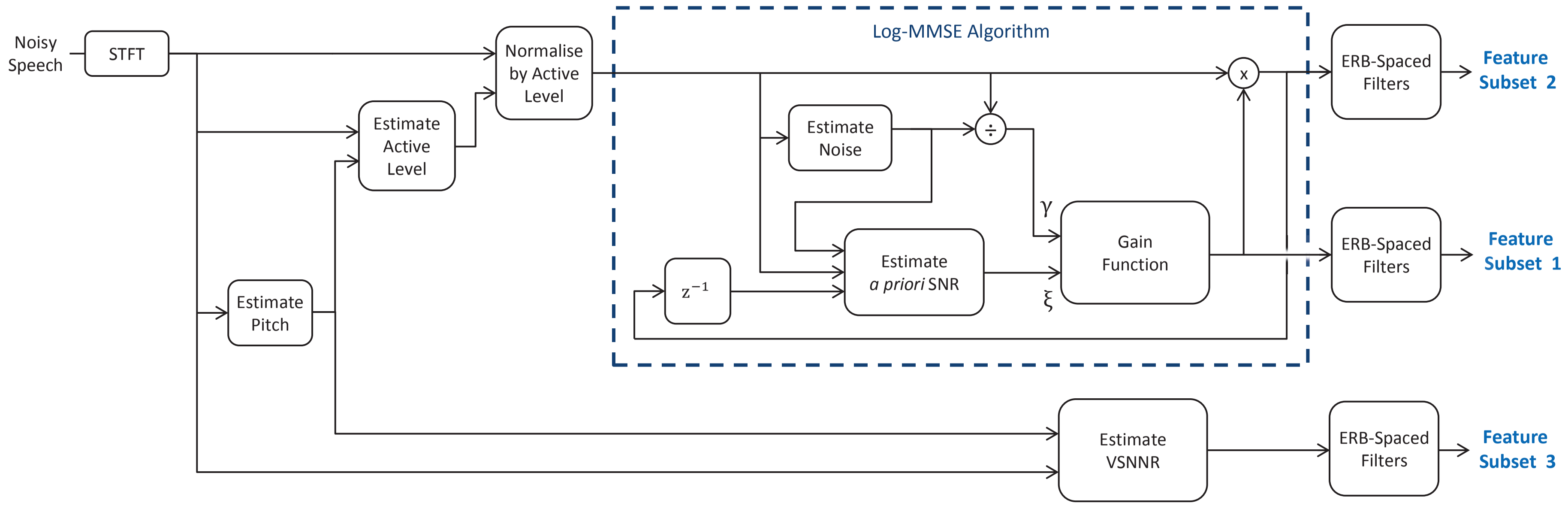


Figure 5.2: Diagram of the procedure for computing the proposed feature set.

is the *a priori* SNR, and

$$v(k,\, m) \triangleq \gamma(k,\, m)\xi(k,\, m)/\left(1 + \xi(k,\, m)\right),$$

where

$$\gamma(k,\, m) \triangleq \left|Y(k,\, m)\right|^2 / E\left[\left|N(k,\, m)\right|^2\right]$$

is the *a posteriori* SNR. An estimate $\hat{\xi}(k,\, m)$ of $\xi(k,\, m)$ is obtained using the "decision-directed" approach from [36],

$$\hat{\xi}(k,\, m) = \alpha G^2(k, m-1)\gamma(k, m-1)$$

$$+(1-\alpha)\max\left\{\gamma(\mathrm{k,\, m}) - 1, 0\right\}$$

where α is a smoothing parameter. A noise estimator is used to provide an estimate of $E\left[\left|N(k,\, m)\right|^2\right]$. In experiments we used the noise estimator from [47].

The first feature subset, $\boldsymbol{\mu}_m^{(1)} = \left[\mu_{1,m}^{(1)},\, \ldots,\, \mu_{\Psi,m}^{(1)}\right]^{\mathrm{T}}$, is a $\Psi \times 1$ vector found by averaging the gain, $G(k,\, m)$, in Ψ triangular windows, $w_i\left(k\right)$, with 50% overlap between windows and centre frequencies equally spaced on the Equivalent Rectangular Bandwidth (ERB) scale, then computing the natural logarithm, i.e.

$$\mu_{i,m}^{(1)} = \ln\left\{\frac{\sum\limits_{k=0}^{K/2} w_i\left(k\right) G(k,\, m)}{\sum\limits_{k=0}^{K/2} w_i\left(k\right)}\right\} \quad \text{for } i = 1,\, \ldots,\, \Psi.$$

where K is the Discrete Fourier Transform (DFT) length. The ERB scale, whose transformation is denoted by $\varPhi\left(f\right)$ and whose inverse derivative, $\frac{df}{d\varPhi(f)}$, approximates the bandwidths of the human auditory filters, can be approximated as

$$\varPhi\left(f\right)=11.17268\cdot\ln\left(1+\frac{46.06538\cdot f}{f+14678.49}\right)$$

between 0.1 and 6.5 kHz [15, 119], with an inverse, $\varPhi^{-1}\left(\cdot\right)$, given by

$$\varPhi^{-1}\left(a\right)=\frac{676170.4}{47.06538-e^{0.08950404\cdot a}}-14678.49.$$

The triangular windows are defined in the ERB domain as

$$W_i\left(a\right)=\Lambda\left(\frac{2\left(a-\upsilon_i\right)}{B}\right)\quad\text{for}\, i=1,\,\ldots,\,\Psi,$$

where B is the width of each window measured in ERBs, $\Lambda\left(\cdot\right)$ is the triangular function, i.e.

$$\Lambda\left(x\right)\triangleq\begin{cases}1-|x| & |x|<1\\ \\ 0 & \text{otherwise}\end{cases},$$

and υ_i is the centre frequency of the i^{th} triangular window measured in ERBs. The DFT-domain triangular windows, $w_i\left(k\right)$, are illustrated in the central plot of Fig 5.3 and are obtained by sampling $W_i\left(a\right)$,

$$w_i\left(k\right)=W_i\left(\varPhi\left(\frac{kf_s}{K}\right)\right)\quad\forall k. \tag{5.1}$$

where f_s is the sample rate. The centre frequencies of the lowest and highest windows are

F_l Hz and F_h Hz, respectively. The centre frequencies, υ_i, in ERBs are at

$$\upsilon_i = \mathit{\Phi}\left(F_l\right) + (i-1) \times \left(\frac{B}{2}\right) \quad \text{for}\, i = 1, \ldots, \Psi,$$

and

$$B = \frac{2 \times \left(\mathit{\Phi}\left(F_h\right) - \mathit{\Phi}\left(F_l\right)\right)}{\Psi - 1}.$$

5.2.2 Feature subset 2: Classically enhanced speech

The second subset of features, $\boldsymbol{\mu}_m^{(2)} = \left[\mu_{1,m}^{(2)}, \ldots, \mu_{\Psi,m}^{(2)}\right]^{\mathrm{T}}$, is an estimate of the level-normalised enhanced speech amplitude in each frequency band. These features are included since they provide a direct estimate of speech presence in each time-frequency cell. The subset is obtained in a similar way to feature subset 1, by averaging the processed noisy speech, $G(k, m)\,|Y(k, m)|$, with the overlapping triangular windows, i.e.

$$\mu_{i,m}^{(2)} = \ln\left\{\frac{\displaystyle\sum_{k=0}^{K/2} w_i\left(k\right) G(k, m)\,|Y(k, m)|}{\displaystyle\sum_{k=0}^{K/2} w_i\left(k\right)}\right\} \quad \text{for}\, i = 1, \ldots, \Psi.$$

5.2.3 Feature subset 3: VSNNR estimate

The third subset of features, $\boldsymbol{\mu}_m^{(3)} = \left[\mu_{1,m}^{(3)}, \ldots, \mu_{\Psi,m}^{(3)}\right]^{\mathrm{T}}$, is used to detect the presence of voiced speech energy in local TF regions. These features provide an independent way to detect speech presence that may be more robust in conditions of high noise. The pitch

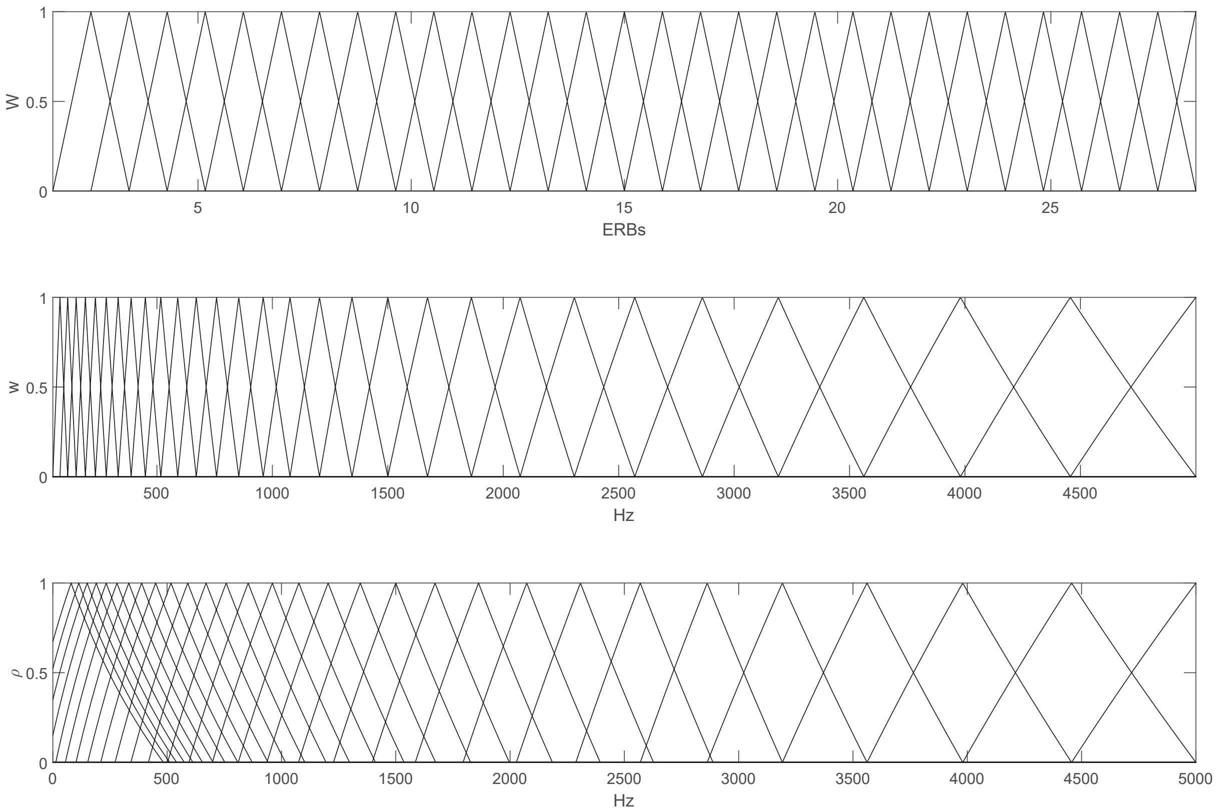


Figure 5.3: Plot of the overlapping windows used to compute feature subsets 1 and 2, with (upper plot) an ERB frequency scale and (middle plot) a linear frequency scale. (lower plot) windows used to compute feature subset 3. In this example, $\Psi = 30$, $F_l = 80$ Hz, $F_h = 5000$ Hz and $b_{min} = 600$ Hz.

estimation algorithm from [57] is first used to estimate the fundamental frequency, $f_0\left(m\right)$, of the speech in each time frame, m. In each frame, the Voiced-Speech-Plus-Noise to Noise Ratio (VSNNR) is then estimated within Ψ frequency bands by comparing the energy at harmonics of the fundamental frequency with the energy mid-way between consecutive harmonics. This is obtained by first multiplying the noisy speech power, $\left|Y(k,\, m)\right|^2$, by a set of triangular gains with peaks at multiples of the estimated fundamental frequency. A second signal is then produced by multiplying $\left|Y(k,\, m)\right|^2$ by an identical set of gains, but offset by half of the estimated fundamental frequency. These quantities are then averaged within Ψ overlapping triangular frequency bands which are equally spaced on the ERB scale, and the ratio of the resulting quantities are computed. That is,

$$\mu_{i,m}^{(3)} = \ln\left\{\frac{\displaystyle\sum_{k=0}^{K/2}\rho_i\left(k\right)h_p\left(k\right)\left|Y(k,\, m)\right|^2}{\displaystyle\sum_{k=0}^{K/2}\rho_i\left(k\right)h_t\left(k\right)\left|Y(k,\, m)\right|^2}\right\} \text{ for } i = 1,\, \ldots,\, \Psi. \tag{5.2}$$

where

$$h_p\left(k\right) = \sum_j \Lambda\left(\frac{2\left(f - f_0\left(m\right)\cdot j\right)}{f_0\left(m\right)}\right)\Bigg|_{f=\frac{kfs}{\mu}}$$

and

$$h_t\left(k\right) = \sum_j \Lambda\left(\frac{2\left(f - f_0\left(m\right)\cdot\left(j+0.5\right)\right)}{f_0\left(m\right)}\right)\Bigg|_{f=\frac{kfs}{\mu}}.$$

The $\rho_i(k)$ in (5.2) are identical to $w_i(k)$ in (5.1) except that we impose a minimum width b_{min} on the triangular windows, measured in Hertz. The triangular windows become

$$\Omega_i(e) = \Lambda\left(\frac{2(e-\upsilon_i)}{\widetilde{B_i}}\right) \quad \text{for}\, i = 1, \ldots, \Psi,$$

$$\rho_i(k) = \Omega_i(a)\big|_{a=\varPhi\left(\frac{kfs}{K}\right)} \quad \forall k,$$

where

$$\widetilde{B_i} = \begin{cases} B & f_i(B) > b_{min} \\ f_i^{-1}(b_{min}) & \text{otherwise} \end{cases}, \tag{5.3}$$

$$f_i(x) = \varPhi^{-1}\left(\upsilon_i + \frac{x}{2}\right) - \varPhi^{-1}\left(\upsilon_i - \frac{x}{2}\right).$$

Solutions to $f_i^{-1}(b_{min})$ are obtained numerically.

The b_{min} parameter was introduced to ensure that each ERB-spaced window will be wide enough such that, if voiced speech is present, the window will include at least one pitch harmonic for the voiced speech detector to detect.

The complete feature set for frame m is obtained by concatenating the three subsets to obtain

$$\boldsymbol{\mu}_m = \begin{bmatrix} \boldsymbol{\mu}_m^{(1)} \\ \boldsymbol{\mu}_m^{(2)} \\ \boldsymbol{\mu}_m^{(3)} \end{bmatrix}.$$

5.2.4 Cochleagram-based feature set

For comparison, some of the evaluations include an alternative feature set based on the power $^1/_{15}$ cochleagram feature set from [21], using the code from [165]. This feature set was chosen as several studies have used enhancers based on cochleagram features to successfully improve the intelligibility of noisy speech, e.g. [64, 21, 20]. To compute this feature set, the noisy speech is passed through a bank of $\mathit{\Omega}$ overlapping Gammatone filters [126] with centre frequencies evenly spaced on the ERB scale from F_l to F_h Hz. The impulse response of filter i is

$$g_i(t) = \alpha t^{n-1} e^{-2\pi b_i t} \cos\left(2\pi \upsilon_i t + \phi\right)$$

where α is the amplitude, n is the filter's order, $b_i = f_i\left(B\right)$ is the bandwidth of filter i in Hz, and ϕ is the phase of the carrier. A gain is applied to account for the mapping between sound pressure level and perceived loudness [81]. The power of each bandpass filtered signal is then computed, and the resulting signals are divided into frames. This results in a feature vector of length $\mathit{\Omega}$ for each frame, m. Finally, the cochleagram is compressed by raising it to the power $^1/_{15}$.

5.2.5 Delta features

The effect of including delta features in the feature sets described above was also evaluated. Their inclusion is motivated by their use in speech recognition where they significantly improve performance over the use of spectral features alone [43]. Delta features approximate the time-derivative of the features by computing the gradient of a straight line fitted in a local region around the current frame. We can fit an $\mathit{\Omega}$-dimensional linear fit to $2\Theta + 1$ consecutive feature vectors $\boldsymbol{\mu}_m$, from frame $i = m - \Theta$ to frame $i = m + \Theta$. Assuming a linear model of the form

$$\boldsymbol{\mu}_i = \boldsymbol{\nu}_m + \boldsymbol{\delta}_m\left(i - m\right) + \boldsymbol{\epsilon}_i,$$

where $\boldsymbol{\epsilon}_i$ is the error vector, $\boldsymbol{\nu}_m$ is the y-axis intercept, and $\boldsymbol{\delta}_m$ is the gradient of the line, we want to find the pair of parameters ($\boldsymbol{\nu}_m$, $\boldsymbol{\delta}_m$) which minimises the power of the noise,

$$\underset{\{\boldsymbol{\nu}_m, \boldsymbol{\delta}_m\}}{\arg\min} \left(Q = \sum_{i=m-\Theta}^{m+\Theta} \boldsymbol{\epsilon}_i^2 = \sum_{i=m-\Theta}^{m+\Theta} \left(\boldsymbol{\mu}_i - \boldsymbol{\nu}_m - \boldsymbol{\delta}_m (i-m) \right)^2 \right).$$

The solution to this linear regression problem is given by [43],

$$\boldsymbol{\delta}_m = \frac{\sum_{\theta=1}^{\Theta} \theta \left(\boldsymbol{\mu}_{m+\theta} - \boldsymbol{\mu}_{m-\theta} \right)}{2 \sum_{\theta=1}^{\Theta} \theta^2}.$$

Although several speech recognition systems compute the regression over short intervals of 40-60 ms, longer intervals of 120 ms or more have been found to be optimal with noisy speech [108, 63]. Longer intervals risk invalidating the linear model, whilst shorter intervals may compromise estimation accuracy.

5.3 Estimation algorithms

In order to estimate the mask, the features described in Sec. 5.2 are applied as the input to a neural net estimator. In this section we describe two alternative neural net architectures that are based on feed forward (DNN) and recursive (Long Short-Term Memory (LSTM)) neural nets respectively. Both the DNN and LSTM used sliding feature and estimation windows similar to those used in [21], which are illustrated in Fig. 5.4. Features within a sliding window of length $2V + 1$ frames, extending V frames either side of the current frame, m, are concatenated and used as inputs to the estimator which simultaneously

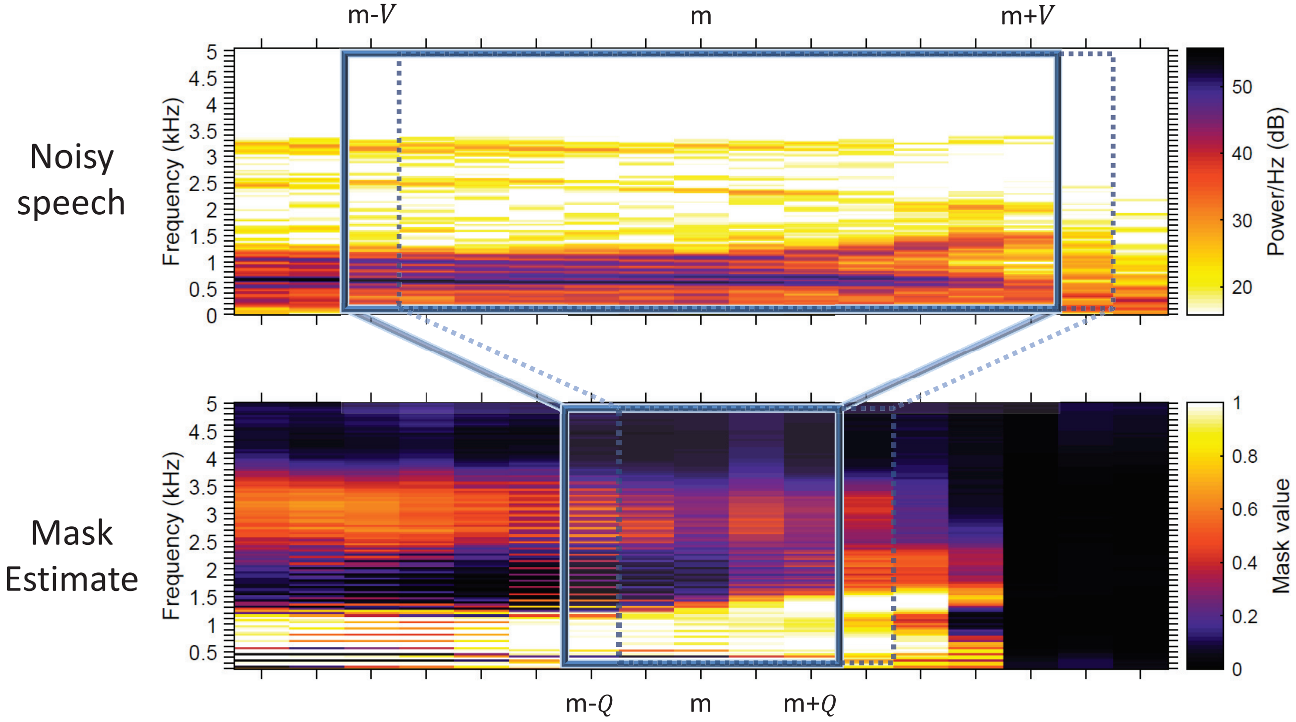


Figure 5.4: Diagram of the sliding feature and estimation windows. Features within a sliding window of length $2V + 1$ frames, extending V frames either side of the current frame, m, are concatenated and used as inputs to the estimator which simultaneously estimates all of the mask values within a window of $2Q + 1$ frames, extending Q frames either side m. At time $m + 1$ the windows shift forward by one frame to the position shown by the dotted line, and the procedure is repeated. In total, this produces $2Q + 1$ mask estimates for each mask bin, which are then averaged to produced the final mask estimate. This procedure results in an algorithmic delay equal to $\max(Q, V)$ frames. In Sec. 6.9 we discuss a modified version of the algorithm which has no algorithmic delay, in which the feature window includes no future frames and $Q = 0$.

estimates all of the mask values within a window of length $2Q+1$ frames, extending Q frames either side m. At time $m+1$ the windows shift forward by one frame to the position shown by the dotted line, and the procedure is repeated. In total, this produces $2Q+1$ mask estimates for each mask bin, which are then averaged to produce the final mask estimate. The feature context window is intended to improve the performance of the estimator by exploiting the strong correlation between both the speech and the noise in neighbouring frames. The estimation window is intended to improve performance and lessen the effect of individual mask estimation errors by averaging several estimates. This procedure results in an algorithmic delay equal to $\max(Q, V)$ frames. In Sec. 6.9 we additionally evaluate a modified version of the algorithm which has no algorithmic delay, in which the feature window includes only past frames and $Q=0$.

The input feature vector of the estimation algorithm in frame m is

$$\boldsymbol{\kappa}_m = \operatorname{vec}\left(\left[\boldsymbol{\mu}_{m-V}, \boldsymbol{\mu}_{m-V+1}, \ldots, \boldsymbol{\mu}_{m+V}, \boldsymbol{\delta}_{m-V}, \boldsymbol{\delta}_{m-V+1}, \ldots, \boldsymbol{\delta}_{m+V}\right]\right),$$

where $\operatorname{vec}(A)$ denotes the vectorisation of the matrix A. The estimation algorithm produces input-output pairs $(\boldsymbol{\kappa}_m, \boldsymbol{y}_m)$ where

$$\boldsymbol{y}_m = \operatorname{vec}\left(\begin{bmatrix} \bar{B}_{2Q+1}(0, m-Q) & \bar{B}_{2Q}(0, m-Q+1) & \cdots & \bar{B}_1(0, m+Q) \\ \bar{B}_{2Q+1}(1, m-Q) & \bar{B}_{2Q}(1, m-Q+1) & \cdots & \bar{B}_1(1, m+Q) \\ \vdots & \vdots & \ddots & \vdots \\ \bar{B}_{2Q+1}(K/2, m-Q) & \bar{B}_{2Q}(K/2, m-Q+1) & \cdots & \bar{B}_1(K/2, m+Q) \end{bmatrix}\right),$$

which results in $2Q+1$ mask estimates, $\bar{B}_q(k, m)$ for $q = 1, \ldots, 2Q+1$, for each mask bin. These estimates are then averaged to produced the overall mask estimate,

$$B(k,\, m) = \frac{1}{2Q+1} \sum_{q=1}^{2Q+1} \bar{B}_q\left(k, m\right).$$

5.3.1 Feed-forward Deep Neural Networks

The upper plot in Fig. 5.5 shows a diagram of a feed-forward DNN with Z “hidden” layers and $W = 6$ units in each hidden layer. In each time frame m, each element of the input feature vector, $\boldsymbol{\kappa}_m = [\kappa_{m,1},\, \ldots,\, \kappa_{m,N}]$, is connected to one “unit” in the input layer, whose structure is shown in the lower plot. The signal flows from the input layer, through several hidden layers, to an output layer which outputs the vector $\boldsymbol{y}_m = [y_{m,1},\, \ldots,\, y_{m,R}]$. The hidden layers are “fully-connected” or “dense” since the inputs, $\boldsymbol{x} = [x_1,\, \ldots,\, x_G]$, of each unit include the outputs of every unit in the previous layer and the output, h, of each unit is the input to each unit in the following layer. The output, h, of each unit in the hidden and output layers is obtained by computing a weighted sum of its inputs, adding a bias parameter, b, and then applying a non-linear “activation” function, $\psi\left(\cdot\right)$, to the result, i.e.

$$h = \psi\left(b + \sum_{i=1}^{G} w_i x_i\right),$$

where $\boldsymbol{w} = [w_1,\, \ldots,\, w_G]$ are the weights. In the example hidden unit shown in Fig. 5.5 (lower plot) the activation function is a rectifier, i.e.

$$\psi\left(x\right) = \begin{cases} x & x > 0 \\ 0 & otherwise \end{cases},$$

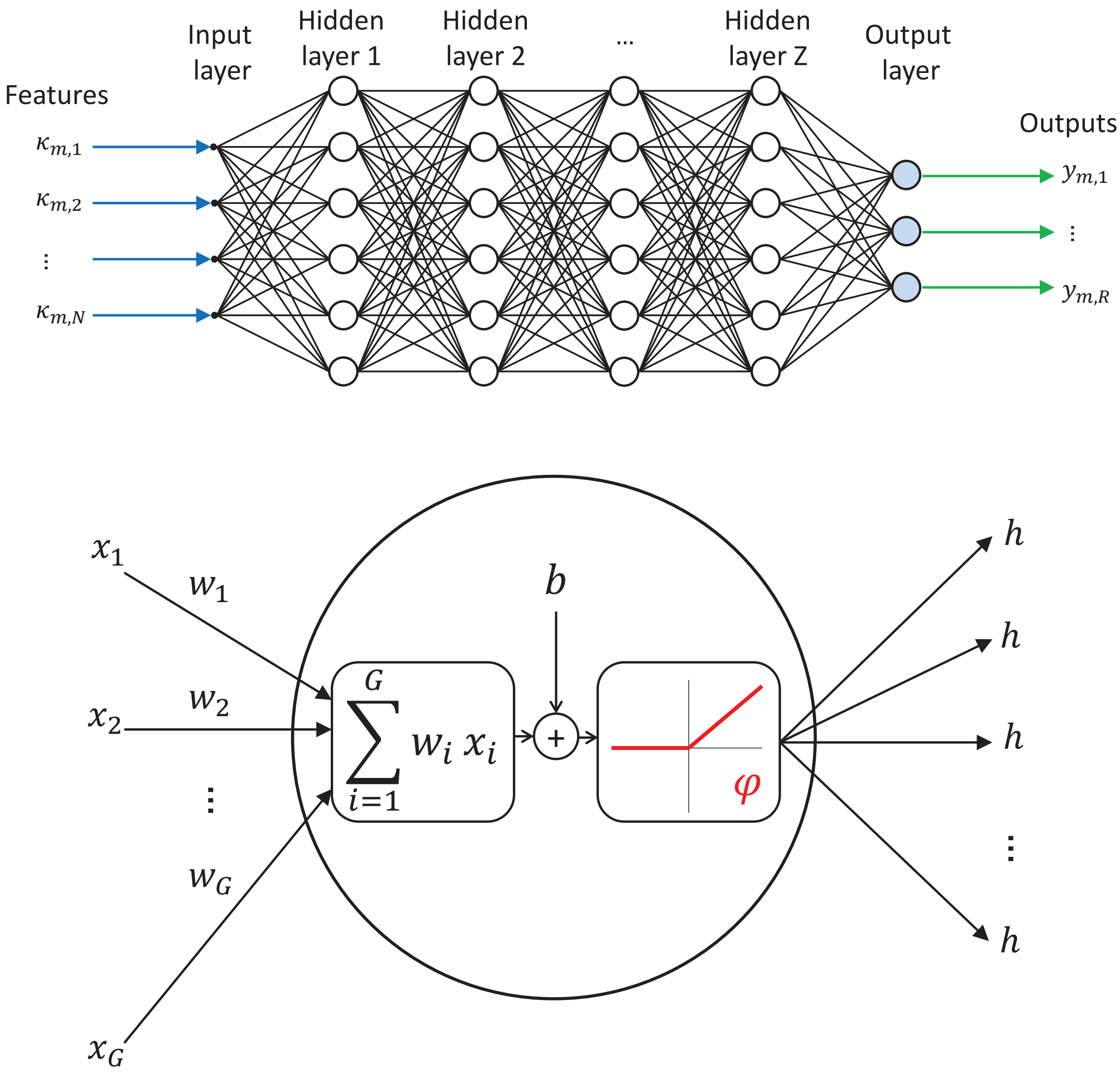


Figure 5.5: Diagram of a feed-forward deep neural network (upper plot) with Z “hidden” layers and $W = 6$ units in each hidden layer, and a Rectified Linear Unit (ReLU) from a hidden layer (lower plot).

and the unit is referred to as a Rectified Linear Unit (ReLU) [122]. Networks constructed from ReLUs have been demonstrated to train several times faster than their equivalents constructed using a hyperbolic tangent activation function [106]. Each unit in the hidden and output layers of the DNN has its own set of weights, $\boldsymbol{w}$, and bias term, b, which are learnt during an algorithm training phase. The combined weights and biases of all units in the DNN form the set of DNN parameters, θ.

The typical way to train a DNN is to use a gradient descent-based optimisation algorithm such as the algorithm from [96], paired with the back-propagation algorithm [134]. The training data consists of $\mathit{\Delta}$ pairs each comprising a feature vector and a corresponding label, $(\boldsymbol{\kappa}_j, \boldsymbol{\varsigma}_j)$, for $j = 1, ..., \mathit{\Delta}$, where $\boldsymbol{\varsigma}_j = [\varsigma_{j,1}, \ldots, \varsigma_{j,R}]$. The optimiser minimises a loss function, such as the mean-squared error between the output of the DNN and the corresponding labels, computed on the training data. The effect of the DNN, parameterised by θ, on the inputs $\boldsymbol{\kappa}_j$ can be represented as a function $\zeta_\theta(\cdot)$, so that $\boldsymbol{y}_j = \zeta_\theta(\boldsymbol{\kappa}_j)$ where $\boldsymbol{y}_j$ is the output of the DNN corresponding to input $\boldsymbol{\kappa}_j$. The mean-squared error loss function is

$$J(\theta) = \frac{1}{\mathit{\Delta} R} \sum_{j=1}^{\mathit{\Delta}} \sum_{r=1}^{R} \left(y_{j,r} - \varsigma_{j,r}\right)^2, \tag{5.4}$$

where $\| \, \|$ is the Euclidean norm. We wish to find the set of parameters, θ, which minimise 5.4. In the standard gradient descent algorithm we update each parameter, θ_i, iteratively as

$$\theta_i = \theta_i - \eta \nabla_{\theta_i} J(\theta_i), \tag{5.5}$$

where ∇_x is the partial derivative with respect to x and η is a step-size parameter often referred to as the "learning rate". To obtain partial derivatives for parameters in the hidden layers, the chain rule is applied to "back-propagate" partial derivatives from each layer to the previous layer, starting with the output layer. It is common to perform parameter updates on "mini-batches" formed from B training samples instead of using the entire training set. Varying B allows for trade-offs in computational speed and estimation error [118].

5.3.2 Recurrent Neural Networks

In a feed-forward neural network, the outputs at time-step m depend only on the features at time-step m. We have seen how we can exploit the correlation between the speech and noise in neighbouring frames by using a sliding feature window. An alternative approach to capturing these correlations is to use a Recurrent Neural Network (RNN). A simple way to construct a RNN is to use the structure from Fig. 5.5 (upper plot) but with additional "recurrent" connections which feed the output of each unit back into the input in the following time-step. This gives the algorithm a "short-term" memory in the form of an internal state (in addition to a "long-term" memory in the form of the learned network weights and biases).

Unfortunately, it has been widely observed that this type of recurrent neural network becomes more difficult to train as the duration of the dependences to be captured increases [10]. This is due to the gradients having a tendency to either explode or vanish as they are back-propagated through time [70, 10, 125], which is a consequence of the temporal evolution of the back-propagated loss function gradients depending exponentially on the size of the weights. Exploding gradients may lead to oscillating weights, while vanishing gradients make training over long time lags take a prohibitive amount of time, or fail [70].

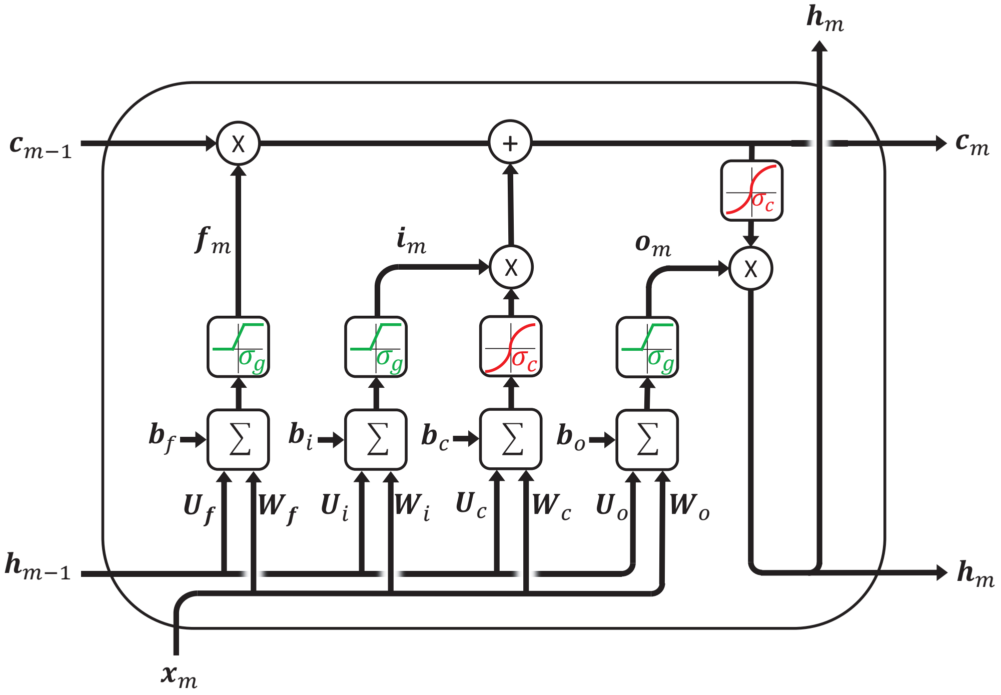


Figure 5.6: Diagram of an LSTM cell, which makes up one of the layers in Fig. 5.7.

5.3.3 Long short-term memory

Recurrent neural networks using Long Short-Term Memory (LSTM) [70] architectures were introduced to solve the problems of vanishing and exploding gradients that exist with standard RNNs. Several variants of the LSTM have been proposed [59]. We will consider the architecture illustrated by the block diagram in Fig. 5.6 of a single LSTM "cell". The inputs and outputs to the cell at time-step m are the vectors $\boldsymbol{x}_m \in \mathbb{R}^G$, and $\boldsymbol{h}_m \in \mathbb{R}^W$, respectively, where G is the number of dimensions in the input vector and W is a parameter which is referred to as the number of "units" in the LSTM and is analogous to the number of feed-forward units in each layer of a DNN. Each cell contains internal memory in the form of a cell state vector, $\boldsymbol{c}_m \in \mathbb{R}^W$, and can also access the output from the previous time-step, $\boldsymbol{h}_{m-1}$. At each time-step, the input vector, $\boldsymbol{x}_m$, and the output from the previous time-step, $\boldsymbol{h}_{m-1}$, are used to update the cell state vector, $\boldsymbol{c}_m$. The first part of the update is applying a "forget gate", which varies between 0 and 1 and controls the amount of information to discard from each element of the previous cell state vector, $\boldsymbol{c}_{m-1}$. A new candidate for $\boldsymbol{c}_m$ is then generated from $\boldsymbol{x}_m$ and $\boldsymbol{h}_{m-1}$. An "input gate" controls the degree to which each element in the cell state is updated with the candidate values. After the cell state has been updated it is used to generate a new candidate for the output vector, $\boldsymbol{h}_m$, with an "output gate" then used to control the update of the output vector.

The equations governing the LSTM are

$$\boldsymbol{f}_m = \sigma_g\left(\boldsymbol{W}_f\boldsymbol{x}_m + \boldsymbol{U}_f\boldsymbol{h}_{m-1} + \boldsymbol{b}_f\right),$$

$$\boldsymbol{i}_m = \sigma_g\left(\boldsymbol{W}_i\boldsymbol{x}_m + \boldsymbol{U}_i\boldsymbol{h}_{m-1} + \boldsymbol{b}_i\right),$$

$$\boldsymbol{o}_m = \sigma_g\left(\boldsymbol{W}_o\boldsymbol{x}_m + \boldsymbol{U}_o\boldsymbol{h}_{m-1} + \boldsymbol{b_o}\right),$$

$$\boldsymbol{c}_m = \boldsymbol{f}_m \circ \boldsymbol{c}_{m-1} + \boldsymbol{i}_m \circ \sigma_c\left(\boldsymbol{W}_c\boldsymbol{x}_m + \boldsymbol{U}_c\boldsymbol{h}_{m-1} + \boldsymbol{b}_c\right),$$

$$\boldsymbol{h}_m = \boldsymbol{o}_m \circ \sigma_c\left(\boldsymbol{c}_m\right),$$

where $\circ$ denotes elementwise multiplication. The weight matrices, $\boldsymbol{W}_f$, $\boldsymbol{W}_i$, $\boldsymbol{W}_o$,$\boldsymbol{W}_c$, $\boldsymbol{U}_f$, $\boldsymbol{U}_i$, $\boldsymbol{U}_o$, $\boldsymbol{U}_c$, and bias vectors, $\boldsymbol{b}_f$, $\boldsymbol{b_i}$, $\boldsymbol{b}_o$, $\boldsymbol{b}_c$, are learned during training. The vectors $\boldsymbol{f}_m$, $\boldsymbol{i}_m$, and $\boldsymbol{o}_m$ are the activation vectors of the forget gate, the input gate and the output gate. To reduce computation cost, the activation function, σ_g , used to generate the three signals $\boldsymbol{f}_m$, $\boldsymbol{i}_m$ and $\boldsymbol{o}_m$ uses a piecewise-linear approximation to the sigmoid function called a “hard-sigmoid”,

$$\sigma_g\left(x\right)=\begin{cases}0 & x<-\mathit{\Phi}\\ x/\left(2\times\mathit{\Phi}\right)+0.5 & -\mathit{\Phi}\leq x\leq\mathit{\Phi}\\ 1 & x>\mathit{\Phi}\end{cases},$$

where $\mathit{\Phi}=2.5$ is a typical value. The other activation function, σ_c, is the hyperbolic tangent function,

$$\sigma_c\left(x\right)=\frac{e^x-e^{-x}}{e^x+e^{-x}}.$$

From the LSTM equations it can be observed that, when the forget gate is “on” ($\boldsymbol{f}_m=1$), and the input gate is “off” ($\boldsymbol{i}_m=0$), the cell state, $\boldsymbol{c}_m$, remains unchanged over successive frames. This enables LSTMs to retain information in their memory for longer than standard RNNs, thereby enabling them to model longer dependencies between inputs and outputs. Equivalently, the back-propagated error gradient is protected from vanishing or exploding over time, as is common in the standard RNN.

Fig. 5.7 shows an LSTM architecture consisting of Z hidden layers, each with recurrent connections, followed by one dense layer with no recurrent connections. Each LSTM layer is comprised of a single LSTM cell as shown in Fig. 5.6, with the outputs of each layer forming the inputs to the following layer. LSTMs which “stack” multiple layers in this way have been shown to outperform single-layer LSTMs on many tasks, e.g. [147].

To train the LSTM, a modified version of the back-propagation algorithm for sequence data is used, called Back-Propagation Through Time (BPTT) [121]. In BPTT the gradient of the error function is back-propagated not only through each layer but also backwards through time to account for the accumulated effect of each network weight over all past time steps on the error at time m. Since the required number of partial derivative computations becomes untenable for large m, in practise the back-propagation is halted after a fixed number of time steps, T. This is referred to as Truncated Back-Propagation Through Time (TBPTT) [175].

The TBPTT training procedure used in our experiments involves first segmenting the training pairs, $(\boldsymbol{\kappa}_j, \boldsymbol{\varsigma}_j)$ for $j = 1, ..., \mathit{\Delta}$, into contiguous non-overlapping sequences of length T. These sequences are then grouped into batches of size B, arranged such that the beginning of the n^{th} sequence of the q^{th} batch continues on from where the n^{th} sequence of the $(q-1)^{th}$ batch ended. Each sequence has its own set of internal states reflecting the history of that sequence up to and including the current batch. A single parameter update is performed on each batch, in each epoch. After performing a parameter update on the q^{th} batch, the B sets of internal states corresponding to the B sequences in the q^{th} batch become the initial internal states of the $(q+1)^{th}$ batch, and so on.

5.3.4 Mask estimators based on neural networks

For the DNN and LSTM, the number of units in the input layer was

$$N = \begin{cases} (2V+1)\,\mathit{\Omega} & V \geq 1, \text{ no delta features} \\ 2\mathit{\Omega} & V = 0, \text{ including delta features} \end{cases}$$

since in our experiments delta features were only considered for the $V = 0$ case. The DNN used rectified linear units in the hidden layers. The LSTM used hyperbolic tangent and

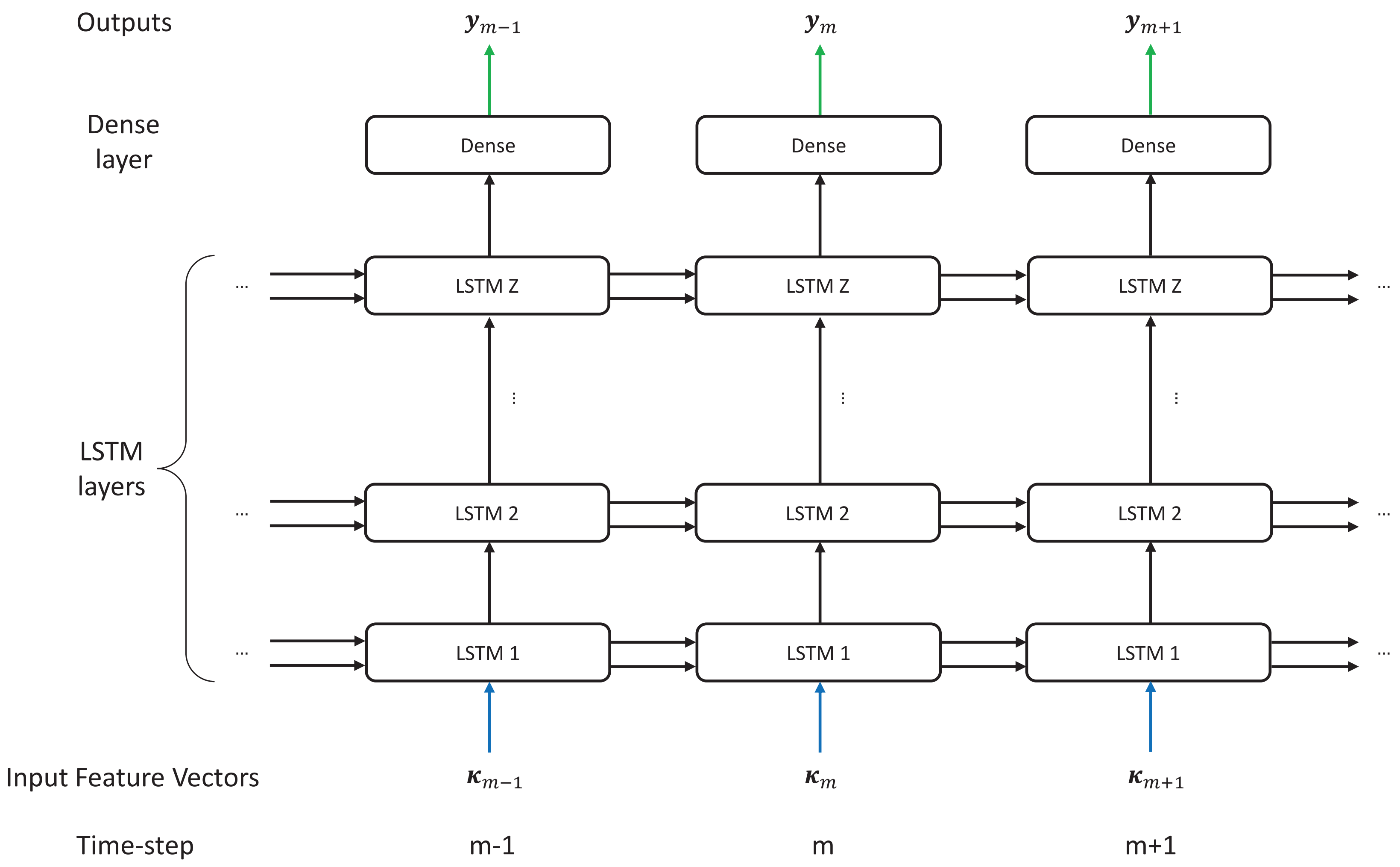


Figure 5.7: Diagram of a “stacked” long short-term memory recurrent neural network with Z LSTM layers followed by 1 dense layer. Each layer is comprised of one of the LSTM cells shown in Fig. 5.6.

hard sigmoid activation functions.

We will test the effect of two different approaches to mask estimation: a) estimation of the mask directly without a library, and b) estimation using a library of mask patterns.

5.3.4.1 Direct Estimator

With the first approach, the algorithm estimates the target mask directly without the use of the library. We denote this the “Direct Estimator”. In this case, the LSTM or DNN has $R = L \times Q$ units in the final dense layer with a sigmoidal activation function, i.e.

$$\psi\left(x\right) = \frac{1}{1 + e^{-x}}.$$

5.3.4.2 Library Estimator

With the second approach, the algorithm constructs the estimated mask as a linear combination of the D mask vectors from the library used to construct the CHSWOBM. We will term this the “Library Estimator”. In the case of the Library Estimator, the final dense layer of the DNN and LSTM had $R = D\left(2Q + 1\right)$ units and a softmax activation function,

$$\psi\left(x_j\right) = \frac{e^{x_j}}{\sum_{i=1}^{D} e^{x_i}}.$$

The output vector, $\boldsymbol{y}_m$, satisfies

$$\sum_{i=1}^{D} y_{m,i} = 1.$$

The estimated mask is obtained by summing the D library mask vectors, with the DNN outputs as weights. If $B_T\left(k, m\right)$ is our oracle target mask in time-frequency bin $\left(k, m\right)$, and we define $\boldsymbol{r}_m = [B_T(0,m), \ldots, B_T(K/2,m)]$, then our estimate $\hat{\boldsymbol{r}}_m$ of $\boldsymbol{r}_m$ is

$$\hat{\boldsymbol{r}}_m = \sum_{i=1}^{D} y_{m,i} \boldsymbol{v}_i$$

where $\boldsymbol{v}_i \in \{0, 1\}^L$ is the i^{th} mask vector in the library used to construct the CHSWOBM, and $L = K/2 + 1$.

5.3.4.3 Neural network loss function

Several studies which use neural networks to estimate a TF mask (e.g. [21, 20]) use a mean square error loss function, i.e. (5.4). In these studies, equal weighting is applied to all errors in the loss function. However, the impact of a mask error in a TF bin on the intelligibility of the mask processed speech is not uniform across different TF bins. That is, an error in one TF bin may be more detrimental to intelligibility than an identical error in a different TF bin. To take account of this, we propose weighting each error with an estimate of the importance of the mask value in that TF cell to the intelligibility of the processed speech. By doing this we hope to encourage the learning algorithm to place greater emphasis on correctly estimating the mask bins that are more significant in terms of intelligibility, thereby improving the intelligibility of the mask-processed speech. The weighted mean square error loss function is

$$J(\theta) = \frac{\displaystyle\sum_{j=1}^{\mathit{\Delta}} \sum_{r=1}^{R} \rho_{j,r} \left(y_{j,r} - \varsigma_{j,r}\right)^2}{\displaystyle \mathit{\Delta} R \sum_{j=1}^{\mathit{\Delta}} \sum_{r=1}^{R} \rho_{j,r}}. \tag{5.6}$$

where the weights are the elements of the vector $\boldsymbol{\rho}_j = [\rho_{j,r}, \ldots, \rho_{j,R}]$. The weights are the product of two separate weights: a band importance weighting and a WSTOI sensitivity weighting. We define

$$\boldsymbol{\rho}_j = \boldsymbol{\alpha} \circ \boldsymbol{\beta}_j,$$

where the symbol $\circ$ denotes the Hadamard product,

$$\boldsymbol{\alpha} = \mathrm{vec}\left(\underbrace{\begin{bmatrix} \alpha(0) & \alpha(0) & \cdots & \alpha(0) \\ \alpha(1) & \alpha(1) & \cdots & \alpha(1) \\ \vdots & \vdots & \ddots & \vdots \\ \alpha(K/2) & \alpha(K/2) & \cdots & \alpha(K/2) \end{bmatrix}}_{2Q+1 \text{ columns}} \right),$$

where $\alpha(k)$ is the importance of frequency bin k using the band importance function from Table 3 of [4], and

$$\boldsymbol{\beta}_m = \mathrm{vec}\left(\begin{bmatrix} \beta(0, m-Q) & \beta(0, m-Q+1) & \cdots & \beta(0, m+Q) \\ \beta(1, m-Q) & \beta(1, m-Q+1) & \cdots & \beta(1, m+Q) \\ \vdots & \vdots & \ddots & \vdots \\ \beta(K/2, m-Q) & \beta(K/2, m-Q+1) & \cdots & \beta(K/2, m+Q) \end{bmatrix} \right),$$

where $\beta(k, m)$ measures, for time-frequency bin (k, m), the expected reduction in the high resolution version of WSTOI that would result from applying the oracle target mask with the mask value in that bin inverted (i.e. the effect of a single error in the oracle mask at (k, m)), in the stochastic noise case. The oracle target mask with TF unit (a, b) inverted is defined as

$$B_T^{a,b}(k, m) = \begin{cases} 1 - B_T(k, m) & (k, m) = (a, b) \\ B_T(k, m) & \text{otherwise} \end{cases}.$$

We have

$$\beta\left(k,m\right)=\sum_{n=m}^{m+M-1} I_{k,n}\left(\left\langle d\left(\mathbf{x}_{k,n},\,\boldsymbol{b}_{k,n}\circ\mathbf{y}_{k,n}\right)\right\rangle-\left\langle d\left(\mathbf{x}_{k,n},\,\boldsymbol{b}_{k,n}^{k,m}\circ\mathbf{y}_{k,n}\right)\right\rangle\right),$$

where $\boldsymbol{b}_{k,m}$ is the modulation vector ending in (k,m), from the high resolution of WSTOI, formed from the mask B_T, i.e.

$$\boldsymbol{b}_{k,m}=\left[B_T\left(k,m-M+1\right),B_T\left(k,m-M+2\right),\,\ldots,\,B_T\left(k,m\right)\right]^{\mathrm{T}},$$

$\boldsymbol{b}_{k,n}^{k,m}$ is formed in the same way from $B_T^{k,m}\left(k,n\right)$, $\left\langle d\left(\cdot\right)\right\rangle$ is computed using (4.7), and $I_{k,m}$ is the WSTOI weight as defined in (3.1).

All features were scaled to have zero mean and unit variance when the LSTM or DNN was used. That is, the inputs to the neural networks were $\tilde{\boldsymbol{\kappa}}_j=\left[\tilde{\kappa}_{j,1},\,\ldots,\,\tilde{\kappa}_{j,N}\right]$ for $j=1,...,\mathit{\Delta}$, where

$$\tilde{\kappa}_{j,n}=\frac{\kappa_{j,n}-\bar{\kappa}_n}{\sqrt{\frac{1}{\mathit{\Delta}}\sum_{i=1}^{\mathit{\Delta}}\left(\kappa_{i,n}-\bar{\kappa}_n\right)^2}}\quad\text{for}\,j=1,...,\mathit{\Delta},\,n=1,...,N,$$

where

$$\bar{\kappa}_n=\frac{1}{\mathit{\Delta}}\sum_{i=1}^{\mathit{\Delta}}\kappa_{i,n}\quad\text{for}\,n=1,...,N.$$

5.3.5 Gaussian Mixture Model-based mask estimator

As an alternative to using a neural network as the mask estimation algorithm, we can use a Gaussian Mixture Model (GMM), with the Compact HSWOBM (CHSWOBM) as the target mask. One advantage of the GMM is that it typically uses fewer parameters than a neural network and may therefore be less prone to overfitting. Overfitting would damage the mask estimator's ability to generalise to new speakers and noises, which has been a well documented problem for mask estimators and in particular those based on neural networks [21, 20, 172]. However, in recent years more attention has been given to neural network-based mask estimators, which have been shown to improve speech intelligibility under certain conditions [65, 64, 21, 20]. GMMs have also been outperformed and largely superseded by neural networks in speech recognition [69, 58]. We will therefore compare mask estimators based on both algorithms.

5.3.5.1 GMM training

The Training dataset consists of the set of feature-label pairs $\Psi = \{(\boldsymbol{\kappa}_1, \varsigma_1), \ldots, (\boldsymbol{\kappa}_{\mathit{\Delta}}, \varsigma_{\mathit{\Delta}})\}$, where $\varsigma_j \in \{1, 2, 3, \ldots, D\}$ and D is the number of mask vectors in the library used to construct the CHSWOBM. The training data is grouped into D classes, $\lambda_1, \ldots, \lambda_D$, according to the labels, ς_j, so that $\Psi_i = \{(\boldsymbol{\kappa}_j, \varsigma_j) \in \Psi \,|\, \varsigma_j = i\}$ for $i = 1, \ldots, D$. This produces a set of $\mathit{\Delta}_i$ training vectors for each class, $\Psi_i = \left\{\boldsymbol{\kappa}_1^i, \ldots, \boldsymbol{\kappa}_{\mathit{\Delta}_i}^i\right\}$ for $i = 1, \ldots, D$. We then train a separate Gaussian mixture model on each set of training data, Ψ_i for $i = 1, \ldots, D$. The prior probability density of the classes, $p(\lambda_i)$, is computed as the fraction of the training data belonging to each class, i.e.

$$p(\lambda_i) = \frac{\mathit{\Delta}_i}{\sum\limits_{i=1}^{D} \mathit{\Delta}_i} = \frac{\mathit{\Delta}_i}{\mathit{\Delta}}.$$

Each set of training data, Ψ_i, is assumed to have been generated from a probability distribution of the form

$$p\left(\boldsymbol{\kappa} \,|\, \lambda_i\right) = \sum_{m=1}^{M} \phi_m^i \mathcal{N}\left(\boldsymbol{\kappa} \,|\, \boldsymbol{\mu}_m^i, \boldsymbol{\Sigma}_m^i\right),$$

where the parameters $\theta^i = \left(\phi_m^i,\, \boldsymbol{\mu}_m^i,\, \boldsymbol{\Sigma}_m^i\right)_{m=1,\,\dots,\,M}$ are unknown, where ϕ_m^i are the component weights and the Gaussian distributions, $\mathcal{N}\left(\boldsymbol{\kappa}^i \,|\, \boldsymbol{\mu}_m^i, \boldsymbol{\Sigma}_m^i\right)$, have means $\boldsymbol{\mu}_m^i \in \mathbb{R}^N$ and covariance matrices $\boldsymbol{\Sigma}_m^i \in \mathbb{R}^{N \times N}$. The number of components, M, is known. The component weights of each GMM sum to 1,

$$\sum_{m=1}^{M} \phi_m^i = 1.$$

ϕ_m^i represents the probability that a randomly selected sample generated by $p\left(\boldsymbol{\kappa} \,|\, \lambda_i\right)$ was generated by component m. We can write

$$\phi_m^i = p\left(z_m^i\right)$$

where $\boldsymbol{z}^i = \left[z_1^i,\, \dots,\, z_M^i\right]$ is a vector of binary random variables which are mutually exclusive and exhaustive, and $\boldsymbol{z}^i$ generates samples $\boldsymbol{z}_j^i = \left[z_{j,1}^i,\, \dots,\, z_{j,M}^i\right]$ corresponding to the training vectors $\boldsymbol{\kappa}_j^i$. To obtain estimates of the unknown parameters, $\theta^i = \left(\phi_m^i,\, \boldsymbol{\mu}_m^i,\, \boldsymbol{\Sigma}_m^i\right)$ for $m = 1,\, \dots,\, M,\ i = 1,\, \dots,\, D$, we use the Expectation-Maximisation algorithm [28]. EM is a numerical technique for performing maximum likelihood estimation that iteratively updates estimates of the model parameters by alternating between two steps: an expectation step and a maximisation step. We run the EM algorithm on the training data for the first class, $i = 1$, until convergence before starting on the second class, $i = 2$, and so on. In the expectation step, each datapoint, $\boldsymbol{\kappa}_j^i$, is assigned a "membership weighting" to each mixture component, reflecting the likelihood that the datapoint was generated by that component, given the current parameter estimates. The membership weight, $\omega_{j,m}^i$, of data point $\boldsymbol{\kappa}_j^i$ to cluster m is computed as

$$\omega_{j,m}^i = p\left(z_{j,m}^i = 1 \,|\, \boldsymbol{\kappa}_j^i,\, \theta^i\right) = \frac{\mathcal{N}\left(\boldsymbol{\kappa}_j^i \,|\, \boldsymbol{\mu}_m^i, \boldsymbol{\Sigma}_m^i\right)\phi_m^i}{\sum_{n=1}^M \mathcal{N}\left(\boldsymbol{\kappa} \,|\, \boldsymbol{\mu}_n^i, \boldsymbol{\Sigma}_n^i\right)\phi_n^i}, \quad m = 1, \ldots, M, \, j = 1, \ldots, \mathit{\Delta}_i,$$

which is obtained using Bayes rule.

In the maximisation step, the newly computed membership weightings, $\omega_{j,m}^i$, are used to update the parameter estimates. The component weights are estimated as the fraction of all membership weights assigned to that component,

$$\phi_m^i = \frac{1}{\mathit{\Delta}_i}\sum_{j=1}^{\mathit{\Delta}_i}\omega_{j,m}^i, \quad m = 1, \ldots, M.$$

The mixture means and covariances are estimated with weighted sample estimates using the newly computed membership weightings,

$$\boldsymbol{\mu}_m^i = \frac{1}{\mathit{\Delta}_i}\sum_{j=1}^{\mathit{\Delta}_i}\omega_{j,m}^i\boldsymbol{\kappa}_j^i, \quad m = 1, \ldots, M,$$

$$\boldsymbol{\Sigma}_m^i = \frac{1}{\mathit{\Delta}_i}\sum_{j=1}^{\mathit{\Delta}_i}\omega_{j,m}^i\left(\boldsymbol{\kappa}_j^i - \boldsymbol{\mu}_m^i\right)\left(\boldsymbol{\kappa}_j^i - \boldsymbol{\mu}_m^i\right)^{\mathrm{T}}, \quad m = 1, \ldots, M.$$

As the dimensionality of the data, N, increases, the use of full covariance matrices becomes more costly to compute and can result in a model which is too complex, leading to overfitting [115]. Therefore, in practice the covariance matrices, $\boldsymbol{\Sigma}_m^i$, are often restricted to be diagonal.

Before applying the EM algorithm the GMM parameters were initialised using the K-Harmonic Means clustering algorithm [183]. K-Harmonic Means is a clustering method which is similar to the K-Means algorithm but, when computing the performance function, instead of computing the average of the distances between the cluster centres and the data points we compute the harmonic average of the distances. This makes the algorithm less sensitive to the initialisation of the centres [183], which are randomly selected.

5.3.5.2 Mask estimation using the GMM

After training the GMM we can use it to estimate $B_T(k,m)$, the oracle target mask (CHSWOBM) in time-frequency bin (k,m), given feature vector $\boldsymbol{\kappa}_m$. The GMM provides the output probabilities

$$p\left(\boldsymbol{\kappa}_m \,|\, \lambda_i\right), \quad i = 1, \ldots, D.$$

Using Bayes theorem,

$$p\left(\lambda_i \,|\, \boldsymbol{\kappa}_m\right) = \frac{p\left(\boldsymbol{\kappa}_m \,|\, \lambda_i\right) p\left(\lambda_i\right)}{p\left(\boldsymbol{\kappa}_m\right)}.$$

If we define $\boldsymbol{r}_m = \left[B_T\left(0,m\right), \ldots, B_T\left(K/2,m\right)\right]$, then the estimate of $\boldsymbol{r}_m$, denoted $\hat{\boldsymbol{r}}_m$, is

$$\hat{\boldsymbol{r}}_m = \left\langle \boldsymbol{r}_m \,|\, \boldsymbol{\kappa}_m \right\rangle = \sum_{i=1}^{D} p\left(\lambda_i \,|\, \boldsymbol{\kappa}_m\right) \cdot \boldsymbol{v}_i$$

Dataset name	# utterances	Source	Noises
Training	3296	TIMIT training set	SS, babble
Validation	400	TIMIT training set	SS, babble
Test-2N	400	TIMIT test set	SS, babble
Test-8N	400	TIMIT test set	SS, babble, operations room, F16, Lynx, factory, Volvo, machine gun

Table 5.1: Summary of the datasets used for algorithm training and evaluation. The speech was taken from the TIMIT corpus [45], and the noises from the RSG.10 [139] database. The SNRs used are listed in Table 5.2.

where $\langle a \,|\, b \rangle$ is the expected value of a given b, and $\boldsymbol{v}_i \in \mathbb{R}^L$ is the i^{th} mask vector in the library used to construct the CHSWOBM. Therefore, the mask estimate is computed as

$$\hat{\boldsymbol{r}}_m = \frac{\sum_{i=1}^{D} p\left(\boldsymbol{\kappa}_m \,|\, \lambda_i\right) p\left(\lambda_i\right) \boldsymbol{v}_i}{p\left(\boldsymbol{\kappa}_m\right)}.$$

where

$$p\left(\boldsymbol{\kappa}_m\right) = \sum_{i=1}^{D} p\left(\boldsymbol{\kappa}_m \,|\, \lambda_i\right) p\left(\lambda_i\right).$$

5.4 Experimental Procedure

The proposed mask estimators were trained and tested on utterances from the TIMIT corpus [45] (excluding the diagnostic SA sentences), which was downsampled from the original sample rate of 16 kHz to 10 kHz, since frequencies above 5 kHz are considered by WSTOI to be irrelevant to intelligibility. The TIMIT training set was split randomly to give 3296 training utterances and 400 validation utterances, with the validation utterances

Noise type	SNRs for training and testing (dB)
SS	−6.89, −5.81, −4.85, −3.92, −2.92, −1.71
Babble	−5.53, −4.59, −3.75, −2.91, −2.01, −0.91
Lynx	−8.86, −7.80, −6.86, −5.95, −4.98, −3.80
Operations room	−7.19, −6.14, −5.21, −4.29, −3.31, −2.11
Machine gun	−24.5, −22.9, −21.5, −20.1, −18.6, −16.6
F16	−5.24, −4.28, −3.43, −2.61, −1.72, −0.64
Factory	−4.91, −3.96, −3.12, −2.29, −1.40, −0.31
Volvo	−30.5, −29.2, −28.0, −26.9, −25.7, −24.2

Table 5.2: The range of SNRs used for training and testing, for each noise type. The SNRs correspond to WSTOI values of {0.61, 0.63, 0.65, 0.67, 0.69, 0.72} which in turn correspond to predicted intelligibilities of {30, 40, 50, 60, 70, 80} % using the mapping between WSTOI and intelligibility, (3.3), from Chapter 3.

used for optimising hyperparameters. The algorithms were evaluated on 400 utterances selected from the TIMIT test set, ensuring there was no overlap in speakers or texts between training and testing. To generate the noisy utterances for training, validation and testing, the clean utterances were mixed with noises from the RSG.10 [139] database. In the Training and Validation datasets the utterances were mixed with speech shaped (SS) and babble noise. Two separate test datasets were formed from the test utterances. In one of these, denoted Test-2N, the test utterances were mixed with one of the 2 noise types used during training (SS and babble). In the other test dataset, denoted Test-8N, the test utterances were mixed with one of 8 noise types (SS, babble, operations room, F16, Lynx, factory, Volvo, machine gun), including the two used during training (SS, babble). The segments of SS and babble noise used to generate the Test-2N and Test-8N datasets were taken from different sections of the noise recordings than the segments that were used to generate the Training and Validation datasets, to ensure no overlap between the training and testing data. A summary of the four datasets is shown in Table 5.1. In all the datasets the noisy utterances had the average SNRs shown in Table 5.2, which for each noise type corresponds to WSTOI values of {0.61, 0.63, 0.65, 0.67, 0.69, 0.72}. These WSTOI values correspond to predicted intelligibilities of {30, 40, 50, 60, 70, 80} % using the mapping between WSTOI and intelligibility, (3.3), from Chapter 3.

The Test-8N dataset contains two particularly challenging noise types: Volvo car noise and machine gun noise. Volvo car noise is challenging since most of the signal energy is at very low frequencies (see the spectrogram in Fig. A.1 of the Appendix), so the SNRs of the test utterances had to be made extremely low (between -30 dB and -24 dB) in order that they would correspond to predicted unprocessed intelligibilities of {30, 40, 50, 60, 70, 80}

%. This range of SNRs is far lower than the range used during algorithm training. Machine gun noise is also extremely challenging as it is highly intermittent (see the spectrogram in Fig. A.1 in the Appendix). Since there is effectively no noise in the gaps between the machine gun bursts, the intelligibility of the unprocessed noisy speech remains fairly high, even at very low SNRs. This meant that the mean SNR of the test utterances also had to be extremely low (between -24 dB and -17 dB) to achieve the chosen predicted unprocessed intelligibilities. The SNR during the bursts is, of course, even lower still.

All three feature sets were computed in 25.6 ms frames centred at the centre times of the mask bins, i.e. at intervals of 12.8 ms. For the WSTOI and PESQ calculations, a gain floor of 0.1 was imposed on the estimated masks before they were applied, in order to improve the quality of the resulting speech as discussed in Sec. 2.2.5. The oracle HSWOBMs, SHSWOBMs and CHSWOBMs used to train the estimators were optimised for stochastic white Gaussian noise with a SNR of -5 dB. As discussed in Chapter 4, we believe that using a target mask optimised for a mismatched SNR and noise type may encourage the learning algorithm to focus more on the features present in the speech and less on the noise, and that this may result in our estimation algorithm performing better on new noise types that were not seen during training. An SNR of -5 dB was chosen as it is within the SNR range of the noisy test utterances, and in any case we saw in Chapter 4 that the value of STOI computed on noisy speech that had been processed with an oracle STOI-optimal mask was not very sensitive to the value of the SNR of the stochastic noise signal.

The centre frequencies of the lowest and highest triangular windows used to compute the proposed feature set were $F_l = 80$ Hz and $F_h = 5000$ Hz, respectively. The value

$F_l = 80$ Hz was chosen as this corresponds to the lower end of the range of typical male fundamental frequencies [156] and hence information below this is likely to be of little importance in terms of intelligibility. The value $F_h = 5000$ Hz was chosen as this corresponds to the centre frequency of the highest frequency bin in the proposed oracle masks and, since WSTOI ignores frequencies above 4.28 kHz, we assume them to be less critical for intelligibility and also omit them. The value $b_{min} = 600$ Hz was chosen to ensure that, for most voiced speech, the ERB bands for feature subset 3 will be at least as wide as 2 harmonics of the fundamental frequency of the speaker. This was done to ensure that if voiced speech was present at least 1 harmonic would be present within the ERB band for the voiced speech detector to detect. The cochleagram feature set was modified to have $\mathit{\Omega}$ frequency channels centred from 50 to F_h Hz instead of 64 frequency channels centred from 50 to 8000 Hz as in [21], to facilitate comparisons with the proposed feature set.

The DNNs and LSTMs were trained using the back-propagation and Truncated Back-Propagation Through Time (TBPTT) algorithms, respectively. The TBPTT algorithm used T time steps. Both algorithms used the Adam optimiser with the default parameters from [96] and a mini-batch size of B. For the DNNs and LSTMs, dropout regularisation [138] with probabilities of G and H was applied to the inputs of the first hidden layer and all following hidden layers, respectively. For the LSTM, dropout with a probability of ξ was also applied to the recurrent connections as recommended in [44]. During training, the learning rate parameter of the optimiser was reduced by a factor of 0.5 each time two epochs passed without an improvement in the validation error of at least 10^{-4}, which is similar to the procedure from [20], and was found to improve the validation error. The value 10^{-4} was found to be optimal in terms of the validation error among the candidates $\{10^{-3}, 10^{-4}, 10^{-5}\}$. The bias of the forget gate, $\boldsymbol{b}_f$, was initialised to 1 during training to enable gradient flow as is recommended in [48].

With all of the mask estimators tested in this chapter, the mask was applied in the conventional manner, by multiplying the noisy speech by the mask in the STFT-domain. In the next chapter, alternative ways of applying the masks will be discussed. To compute the HIT-FA rate, the estimated masks were converted to binary masks using a threshold of 0.5.

5.4.1 WSTOI mapping

Fig. A.2 in the Appendix contains plots of WSTOI against SNR for the 8 noise types used in this chapter. The results are plotted for 400 utterances selected randomly from the Training dataset. To generate the data in each plot, each of the utterances was mixed with each of the noise types at the following SNRs: $\{-60, -50, -40, -30, -20, -10, 0, 10, 20, 30, 60\}$ dB. The blue curve, which has the equation

$$y = \frac{1 + a\exp{(bx + c)}}{1 + \exp{(bx + c)}}, \tag{5.7}$$

with free parameters a, b, and c, was fitted to the data for each noise type using least squares optimisation. The inverse equation is given by

$$x = \frac{\ln\left(\frac{\mathrm{1-y}}{\mathrm{y-a}}\right) - \mathrm{c}}{b},$$

and is used to map the increase in WSTOI that results from applying the masks to an equivalent increase in SNR, denoted $\Delta\text{SNR}_{\text{WSTOI}}$. This is the hypothetical increase in the SNR of the noisy speech signal that would be required to result in the same increase in WSTOI that is provided by processing with the mask. The use of $\Delta\text{SNR}_{\text{WSTOI}}$ allows results to be presented in a way that is independent of the intelligibility metric that is used.

5.4.2 Hyperparameter optimisation

The mask estimators described in this chapter include a number of hyperparameters whose values must be selected. The optimal value for each hyperparameter was determined by performing a grid search to optimise performance on the Validation dataset. A summary of hyperparameters for each estimator, the grid values and the optimal values is shown in Table 5.3. Separate optimal hyperparameters were computed for the $Q = 0$ and $Q = 2$ cases for both the DNN and the LSTM, with the HSWOBM as the training target. The training target for the GMM was the CHSWOBM.

Fig. 5.8 shows the $\Delta\text{SNR}_{\text{WSTOI}}$ improvement resulting from applying the GMM-based enhancer to noisy speech, when each hyperparameter (the number of components, M, and the delta feature window length, Θ) was varied independently around its optimal value ($M = 5$, $\Theta = 2$), with the other hyperparameter held at its optimal value. The upper plot shows the effect of varying Θ while the lower plot shows the effect of varying M. Results were computed on the utterances from the Validation dataset described Sec. 5.4, with the mean computed across the 400 utterances. Each bar represents 8 points: the mean results of 8 identical experiments, each with a different random initialisation of the

K-harmonic means algorithm. It can be seen that the "optimal" hyperparameters ($M = 5$, $\Theta = 2$) correspond to a local maximum. The value $\Omega = 30$ was chosen as it optimised $\Delta\text{SNR}_{\text{WSTOI}}$ on the grid $\Omega = \{20, 30, 60\}$.

Figures 5.9 and 5.10 show the final value of the neural network loss function, J, from (5.6), after training the LSTM and DNN-based mask estimators respectively with $Q = 2$ on the Validation dataset from Sec. 5.4. In each plot, one of the hyperparameters is varied while the others are held at the optimal values given in Table 5.3. For the LSTM case, $W = 2000$ was the maximum tested value in the grid search due to limited working memory in the processor. Batch sizes below $B = 100$ were not tested systematically as they increased training times to an impractical length and gave only minor improvements in performance.

5.5 Results

5.5.1 Comparison of Direct Estimators

Fig. 5.11 compares the results obtained by using the LSTM and DNN, with output window sizes $Q = 0$ and $Q = 2$, to estimate the HSWOBM for the Validation dataset from Sec. 5.4 containing 400 utterances mixed with babble and SS noise at 6 SNRs. In all cases the Direct Estimator was used. The optimal values of the hyperparameters for these cases are listed in Table 5.3.

Although the differences are small, it can be seen from Fig. 5.11 that the neural network loss function, J, the WSTOI metric (i.e. $\Delta\text{SNR}_{\text{WSTOI}}$) and the HIT-FA rate all agree on the performance ranking of the four algorithms: the DNN always outperformed the LSTM

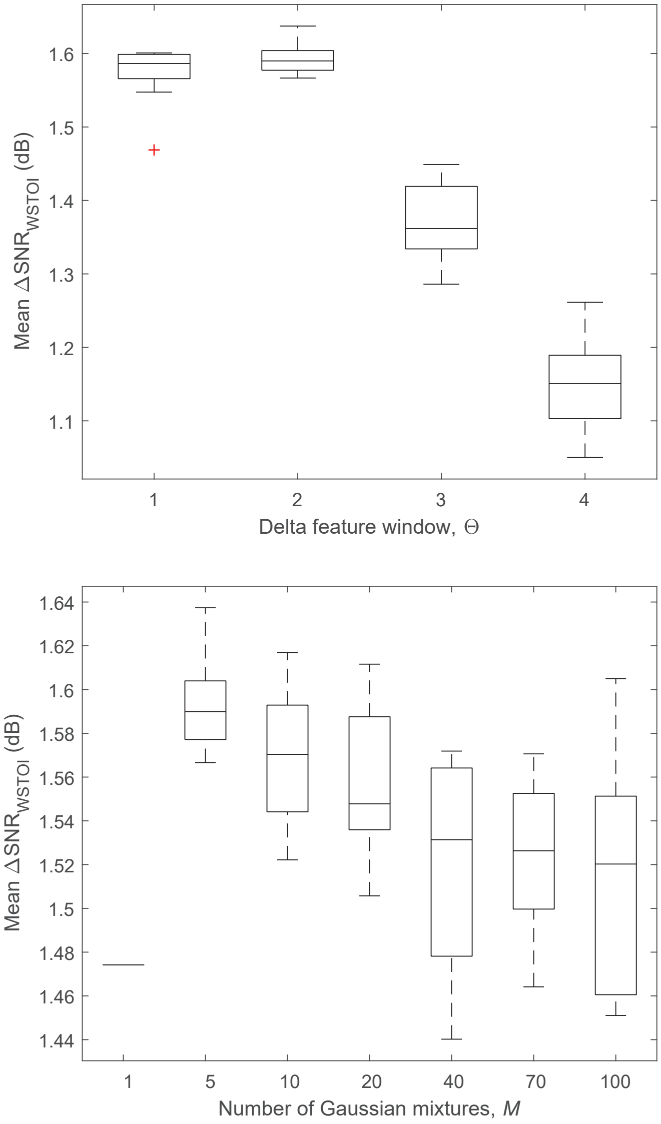


Figure 5.8: The mean $\Delta\mathrm{SNR}_{\mathrm{WSTOI}}$ resulting from applying the GMM-based enhancer to noisy speech, when each hyperparameter (M, Θ) is varied independently around its optimal value ($M = 5$, $\Theta = 2$), with the other hyperparameter held at its optimal value. Results were computed on the Validation dataset described in Sec. 5.4, with the mean computed across the 400 utterances. Each bar represents 8 points: the mean results of 8 identical experiments, each with a different random initialisation of the K-harmonic means algorithm.

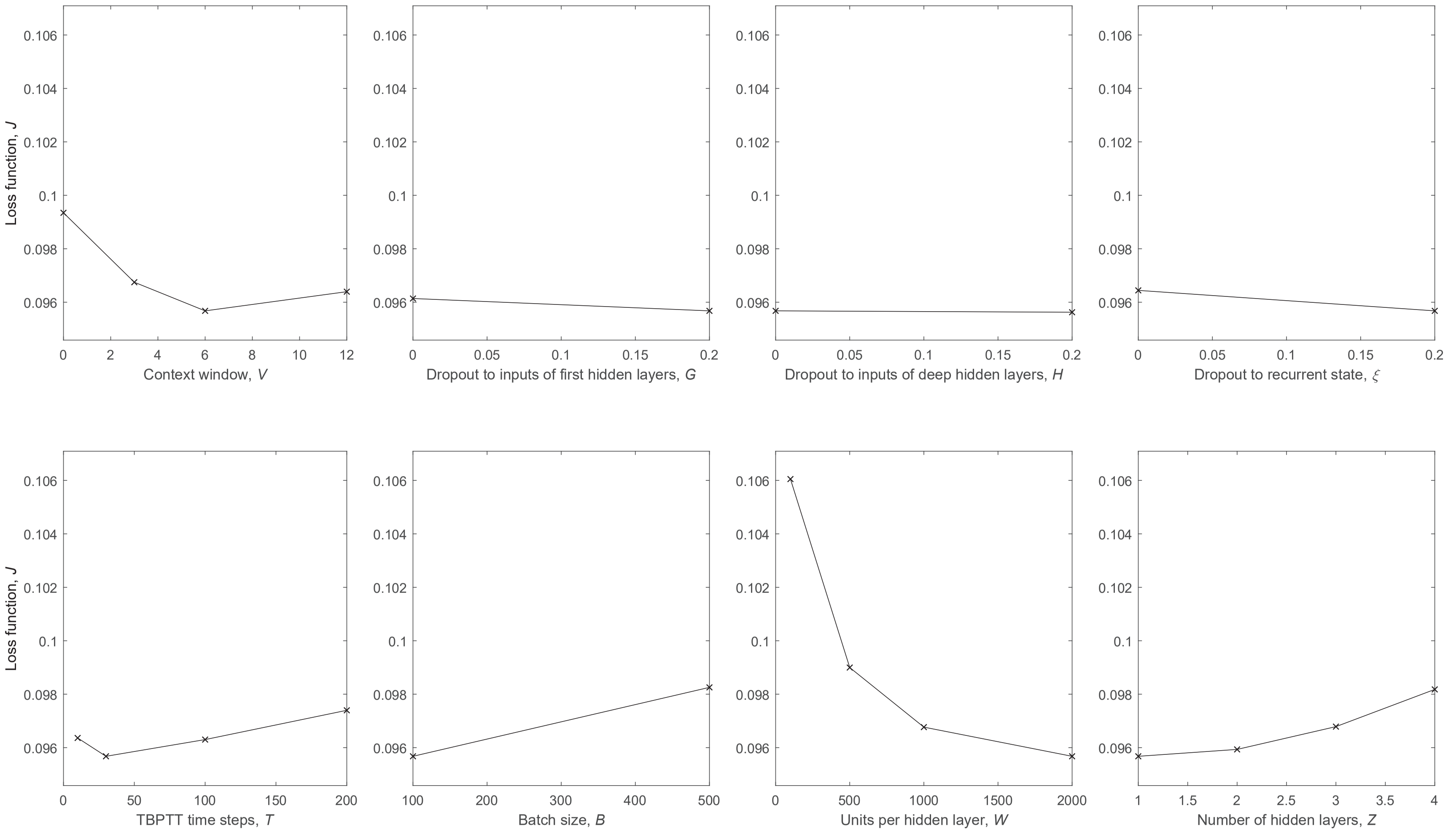


Figure 5.9: Plots of the neural network loss function, J, for the LSTM with $Q = 2$, when each hyperparameter is varied independently around its optimal value with the other hyperparameters held at their optimal values. Results were computed on the Validation dataset described Sec. 5.4.

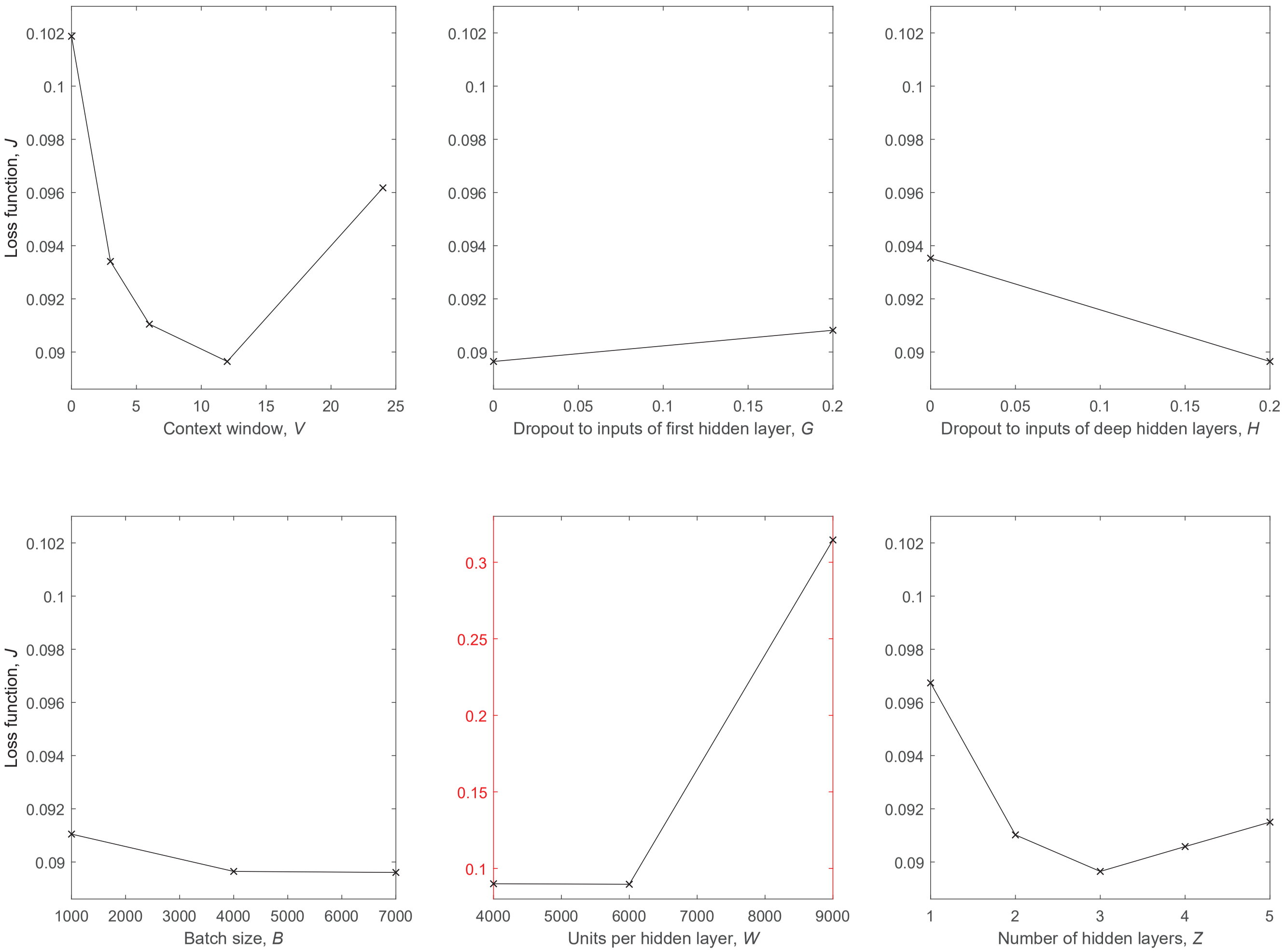


Figure 5.10: Plots of the neural network loss function, J, for the DNN with $Q = 2$, when each hyperparameter is varied independently around its optimal value with the other hyperparameters held at their optimal values. The y-axis for W is red to emphasise that its scale is different to the other plots. Results were computed on the Validation dataset described Sec. 5.4.

Algorithm	Parameter	Location in text (page #)	Tested values	Optimal values	
				$Q = 0$ case	$Q = 2$ case
DNN	V	138	0, 3, 6, 12, 24	12	12
	G	162	0, 0.2	0	0
	H	162	0, 0.2	0.2	0.2
	B	144	$\{1, 4, 7\} \times 10^3$	4000	4000
	W	141	$\{1, 2, 3, 4, 6, 9\} \times 10^3$	3000	6000
	Z	141	1, 2, 3, 4, 5	3	3
	Θ	137	1, 2, 3, 4	2	2
LSTM	V	138	0, 3, 6, 12, 24	0	6
	G	162	0, 0.2	0.2	0.2
	H	162	0, 0.2	0.2	0.2
	ξ	162	0, 0.2	0.2	0.2
	T	148	5, 10, 30, 100, 200	10	30
	B	144	100, 500	100	100
	W	141	100, 500, 1000, 2000	1000	2000
	Z	141	1, 2, 3, 4, 5	2	1
	Θ	137	1, 2, 3, 4	2	2
GMM	M	155	1, 5, 10, 20, 40, 70, 100	5	-
	$\varOmega$	128	20, 30, 60	30	-
	Θ	137	1, 2, 3, 4	2	-

Table 5.3: Summary of hyperparameters that were trained using a grid search, and their optimal values, when the training target was the HSWOBM. Hyperparameter optimisation was carried out on the Validation dataset described Sec. 5.4.

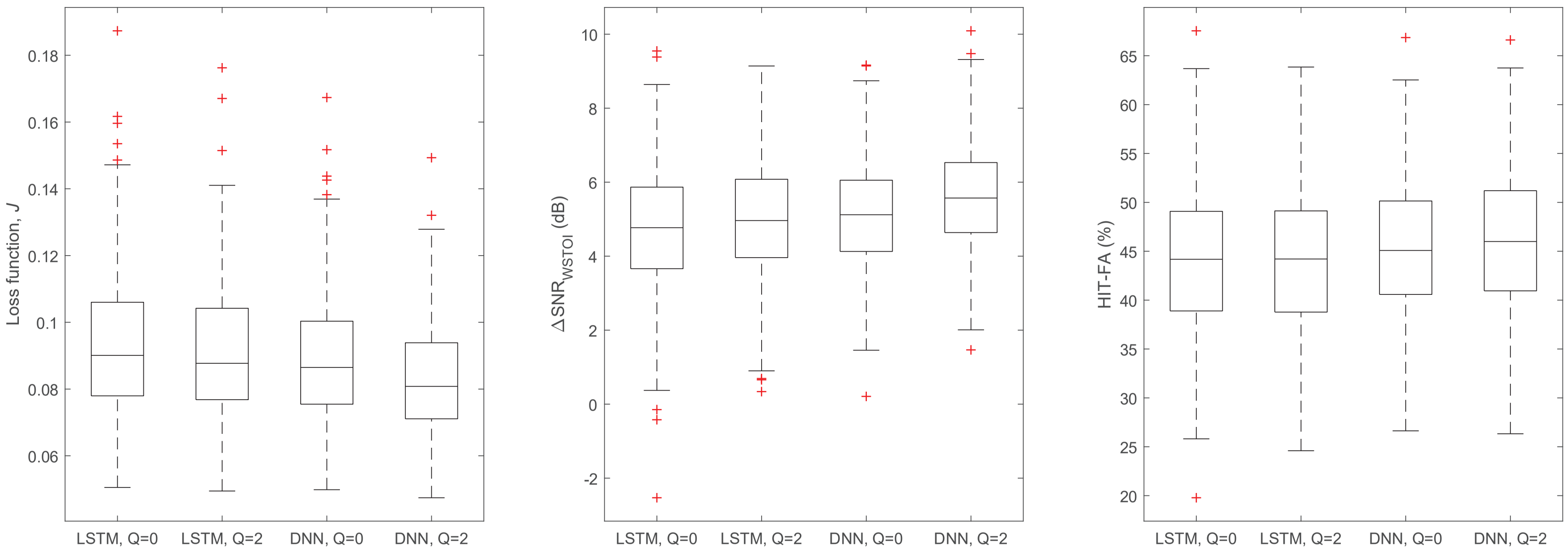


Figure 5.11: Boxplots of a) the neural network loss function, J, b) the $\Delta\text{SNR}_{\text{WSTOI}}$ and c) the HIT-FA rate for the DNN and LSTM-based mask estimators with $Q = 0$ and $Q = 2$, computed on the Validation dataset described in Sec. 5.4 .

for both values of Q, and $Q = 2$ always outperformed $Q = 0$.

Fig. 5.12 illustrates the effect of applying DNN-based mask estimators to an utterance of the phrase "you must explicitly delete files". Plots (a) and (b) show spectrograms of the clean and noisy speech mixed with babble noise at -7.8 dB SNR. Plot (c) shows the oracle HSWOBM mask while plots (d) and (e) show the masks estimated by the DNN-based mask estimators with $Q = 0$ and $Q = 2$ respectively. The noisy speech had a WSTOI of 0.61. This increased to WSTOIs of 0.72 and 0.75 when the masks with $Q = 0$ and $Q = 2$ respectively were applied, corresponding to $\Delta\mathrm{SNR}_{\mathrm{WSTOI}}$ values of 4.9 dB and 6.2 dB, respectively. Plot (f) shows the difference between the weighted intermediate WSTOI measure, $\rho_{j,m}$ (equation 3.2), computed on corresponding pairs of TF cells in the signals produced by applying the two masks to the noisy speech (positive values in blue indicate that $Q = 2$ outperforms $Q = 0$). The plots are aligned so that each plotted value of the difference in $\rho_{j,m}$ was computed with modulation vectors centred on the corresponding frame, m, in the spectrograms. It can be seen the two masks are quite similar, although as we might expect, in the $Q = 2$ case, where the mask was produced by averaging 5 mask estimates, the mask has less rapid transitions. In particular, we can look at the region highlighted in plot (f), where the mask with $Q = 2$ significantly outperforms mask with $Q = 0$. All of the time-frequency cells, j, m, contributing to $\rho_{j,m}$ in this region are highlighted in plots a-e. It can be seen that the mask for $Q = 2$ appears smoother and less noisy in this region.

Fig. 5.13 compares the effect of applying LSTM and DNN-based mask estimators to an utterance of the phrase "while waiting for Chipper she criss-crossed the square many times". Plots (a) and (b) show spectrograms of the clean and noisy speech mixed with babble noise at -12.6 dB SNR. Plot (c) shows the oracle HSWOBM mask while plots (d) and (e) respectively show the masks estimated by the LSTM-based and DNN-based estimators. The noisy speech had a WSTOI of 0.66. This increased to WSTOIs of 0.73 and 0.79 when the masks from the LSTM and DNN were applied, respectively, which corresponds to $\Delta\text{SNR}_{\text{WSTOI}}$ values of 3.1 dB and 5.8 dB, respectively. Although the performance of the masks is very similar in most TF cells, in the highlighted region the mask estimate produced by the DNN significantly outperformed the estimate from the LSTM. It can be seen that, in this region, the mask produced by the DNN more closely matches the amplitude modulation pattern of the clean speech than mask produced by the LSTM. In summary, the DNN with an output window of $Q = 2$ frames is the architecture that results in the best performance.

5.5.2 Comparison of Library Estimators

Fig. 5.14 compares the results obtained when the GMM, DNN and LSTM-based Library Estimators are used with the CHSWOBM as the target mask. For this experiment $V = 0$ and $Q = 0$. As discussed in Sec. 5.3.4.2, the Library Estimator constructs the estimated mask as a linear combination of the $D = 100$ mask vectors from the library used to construct the CHSWOBM. The results are computed for the Validation dataset which is outlined in Sec. 5.4 and contains babble and SS noise at 6 SNRs.

Fig. 5.14 shows that all three algorithms resulted in both an improvement in HIT-FA and in a positive $\Delta\text{SNR}_{\text{WSTOI}}$ for most of the utterances. Both metrics agree on the rank order and as with the Direct Estimator, the DNN performed best, followed quite closely by the LSTM and then the GMM.

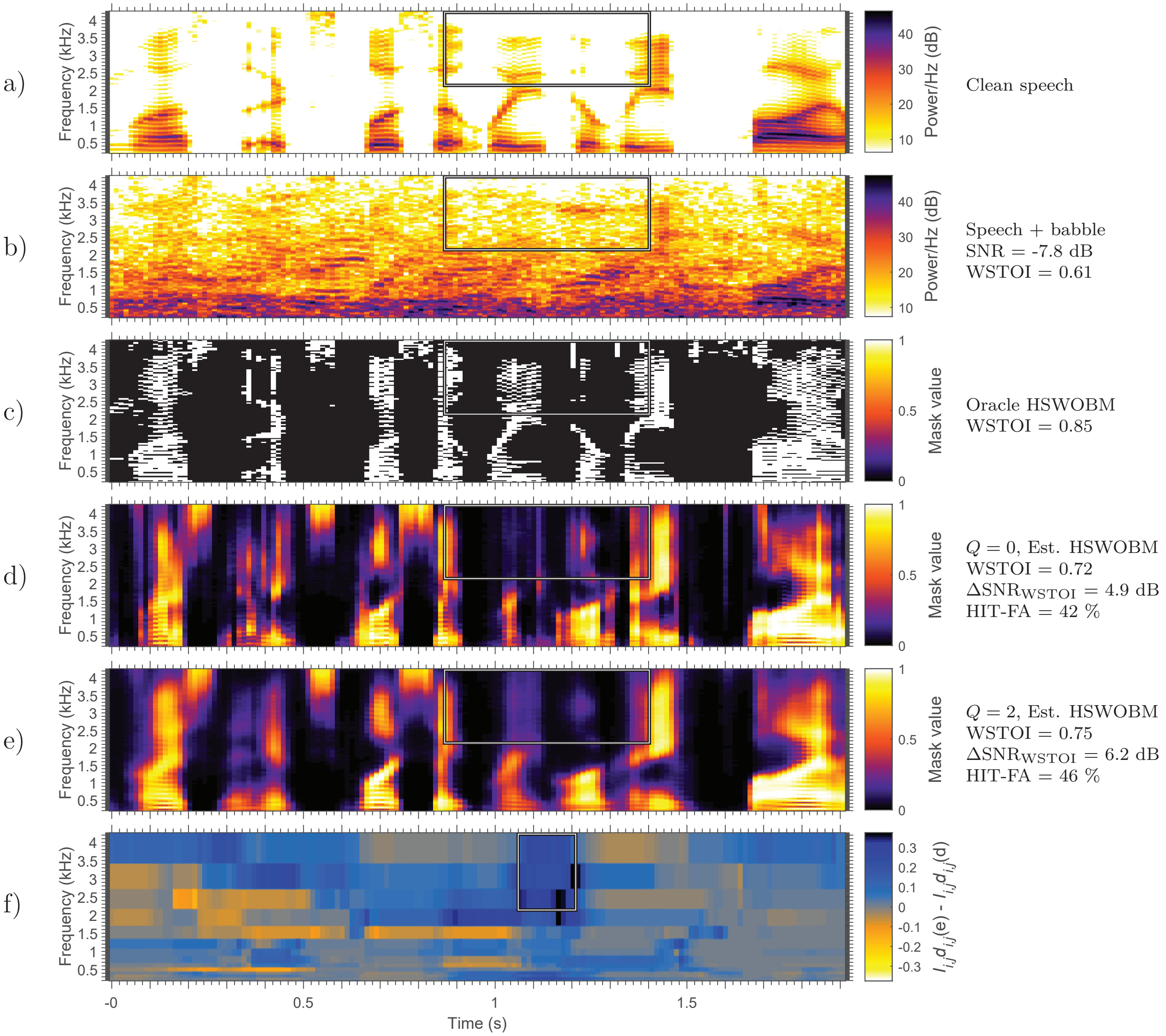


Figure 5.12: Spectrograms of a) part of an utterance of the phrase "you must explicitly delete files", b) the utterance after adding babble noise with an SNR of -7.8 dB, and c) the oracle CHSWOBM for stochastic white Gaussian noise with -5 dB SNR. Spectrograms of the masks produced by the DNN-based mask estimator with d) $Q = 0$, and e) $Q = 2$, and f) the difference between the weighted intermediate WSTOI measure, $\rho_{j,m}$, computed on corresponding pairs of TF cells in the signals produced by applying masks (e) and (d) to the noisy speech. A region is highlighted in plot f, and all TF cells, j, m, contributing to $\rho_{j,m}$ in this region are highlighted in plots a-e.

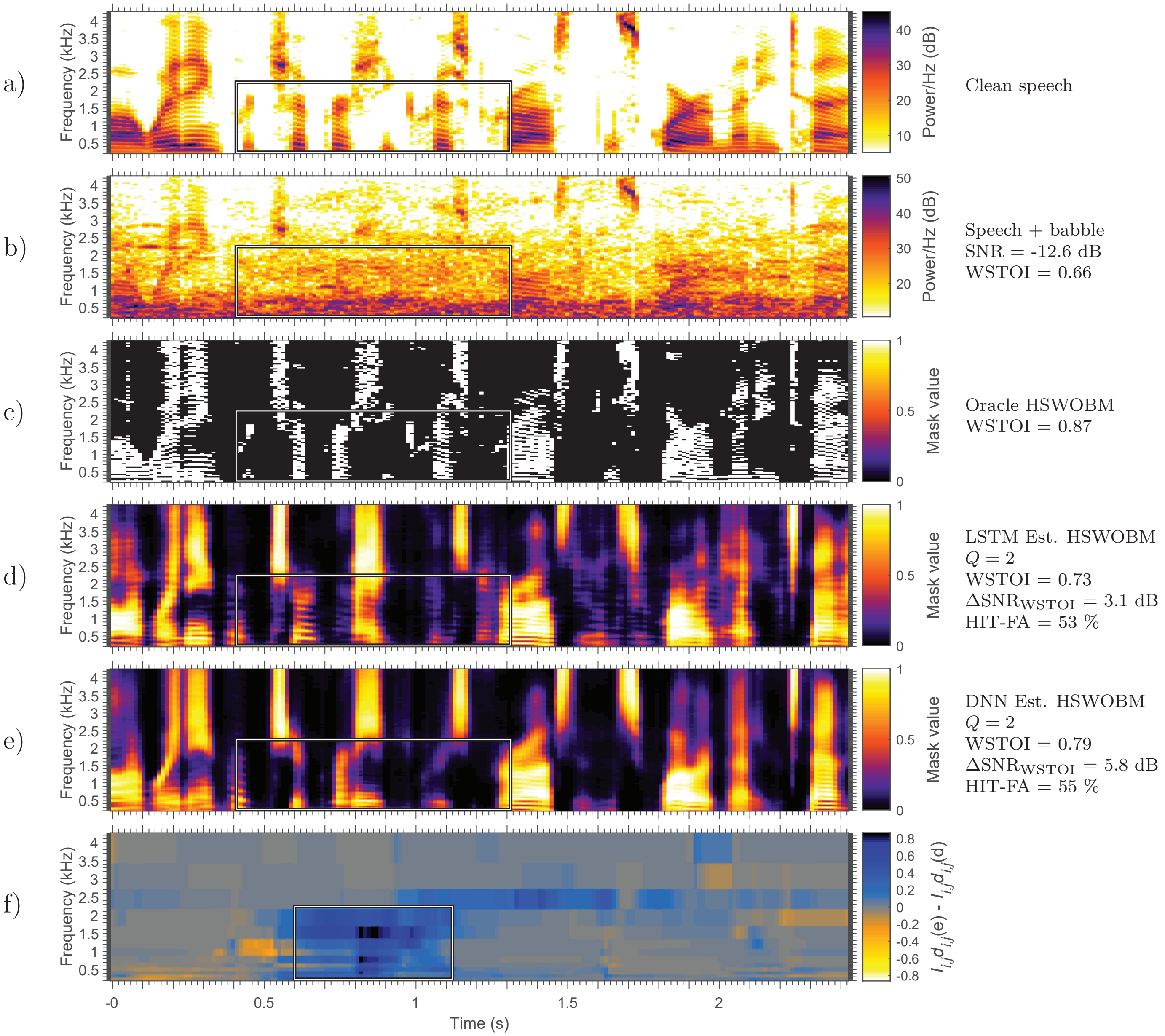


Figure 5.13: Spectrograms of a) part of an utterance of the phrase “while waiting for Chipper she criss-crossed the square many times”, b) the utterance after adding babble noise with an SNR of -12.6 dB, and c) the oracle CHSWOBM for stochastic white Gaussian noise with -5 dB SNR. Spectrograms of the masks produced by d) the LSTM-based mask estimator and e) the DNN-based mask estimator, both with $Q = 2$, and f) the difference between the weighted intermediate WSTOI measure, $\rho_{j,m}$, computed on corresponding pairs of TF cells in the signals produced by applying masks (e) and (d) to the noisy speech. A region is highlighted in plot f, and all TF cells, j, m, contributing to $\rho_{j,m}$ in this region are highlighted in plots a-e.

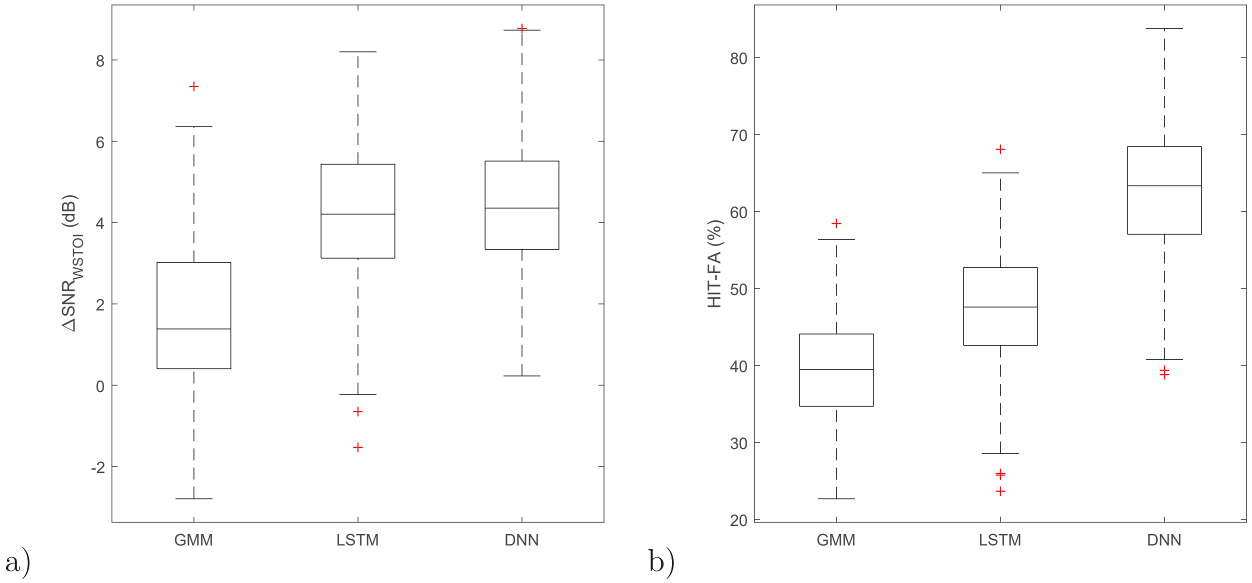


Figure 5.14: Boxplots of a) the $\Delta\mathrm{SNR}_{\mathrm{WSTOI}}$ and b) the HIT-FA rate for the GMM, DNN and LSTM-based Library Estimators, with the CHSWOBM as the target mask. For this experiment $V = 0$ and $Q = 0$. Results are computed on the Validation dataset described Sec. 5.4 .

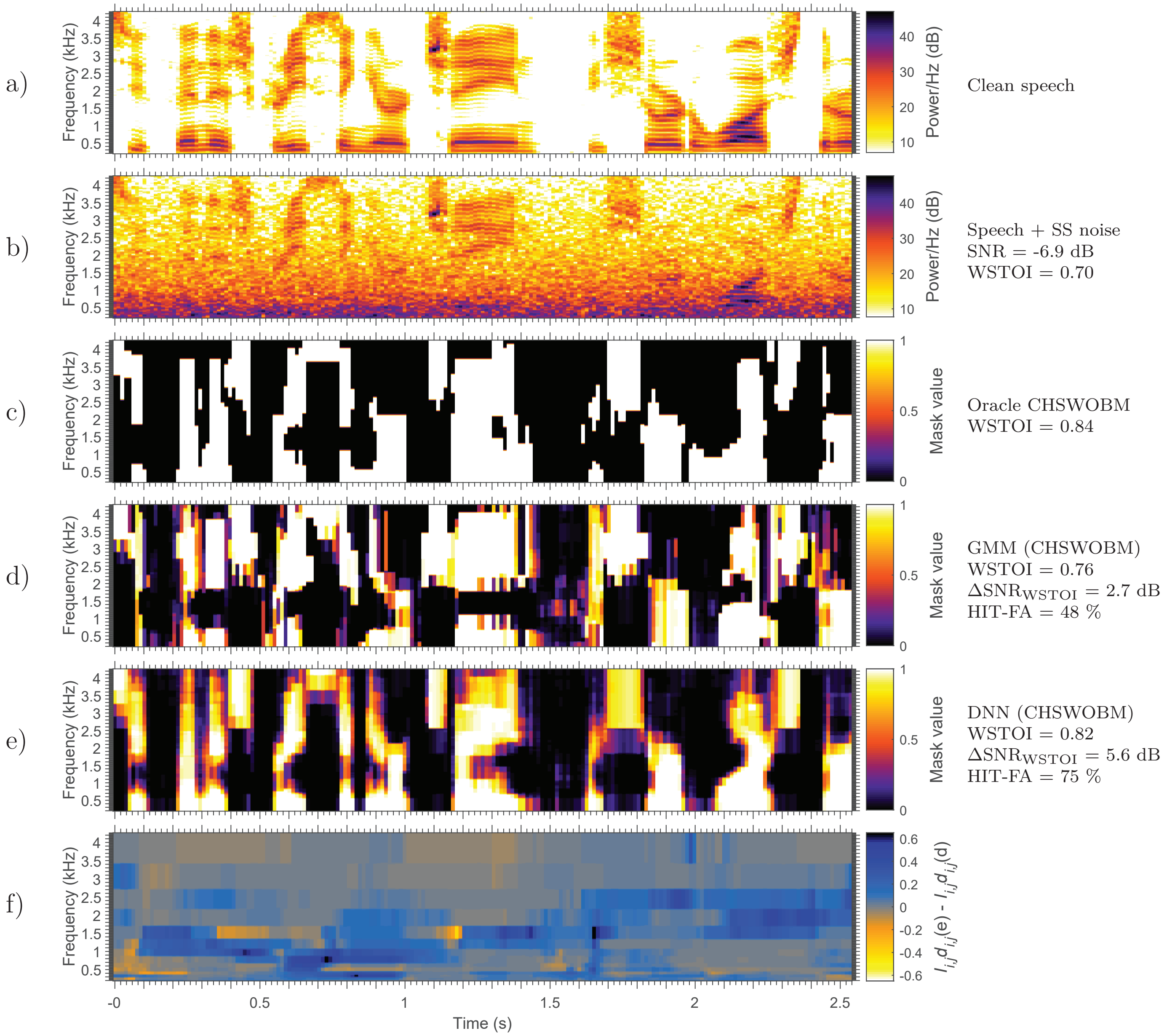


Figure 5.15: Spectrograms of a) part of an utterance of the phrase “once you’ve finished greasing your chain, be sure to wash thoroughly”, b) the utterance after adding speech shaped noise with an SNR of -6.9 dB, and c) the oracle CHSWOBM for stochastic white Gaussian noise with -5 dB SNR. Spectrograms of the CHSWOBM estimated using d) the GMM-based estimator, and e) the DNN-based mask estimator, and f) the difference between the weighted intermediate WSTOI measure, $\rho_{j,m}$, computed on corresponding pairs of TF cells in the signals produced by applying masks (e) and (d) to the noisy speech.

Fig 5.15 compares the effect of applying GMM and DNN-based library estimators to an utterance of the phrase "once you've finished greasing your chain, be sure to wash thoroughly". Plots (a) and (b) show spectrograms of the clean and noisy speech mixed with SS noise at -6.9 dB SNR. Plot (c) shows the oracle CHSWOBM mask while plots (d) and (e) respectively show the masks estimated by the GMM-based and DNN-based library estimators. The target mask for both estimators was the CHSWOBM optimised for stochastic white Gaussian noise with -5 dB SNR. The noisy speech had a WSTOI of 0.70 which increased to 0.76 ($\Delta\mathrm{SNR}_{\mathrm{WSTOI}} = 2.7$ dB) and 0.82 ($\Delta\mathrm{SNR}_{\mathrm{WSTOI}} = 5.6$ dB) respectively when the masks from the GMM and DNN were applied. The HIT-FA rate was also higher with the DNN (75 %) than with the GMM (48 %). Plot (f) shows the difference between the weighted intermediate WSTOI measure, $\rho_{j,m}$, when the DNN and GMM masks are applied to the noisy speech; the darker blue TF regions show where the DNN mask results in a higher WSTOI. Although it can be seen that both mask-estimators produce a similar overall pattern, the mask produced by the GMM, (d), appears to have more mask errors, varies less smoothly and includes more isolated peaks.

5.5.3 Comparison of mask estimation targets

Figure 5.16 compares the performance of the Library Estimator from Sec. 5.3.4.2 with that of the Direct Estimator from Sec. 5.3.4.1. Each plot shows three cases. In the first case, the Library Estimator was used to estimate the Compact HSWOBM (CHSWOBM). In the second and third cases, the Direct Estimator was used with the Smoothed HSWOBM (SHSWOBM) and High-resolution SWOBM (HSWOBM) as the target masks, respectively. All algorithms used the DNN with $Q = 0$, and the results are computed on the Validation

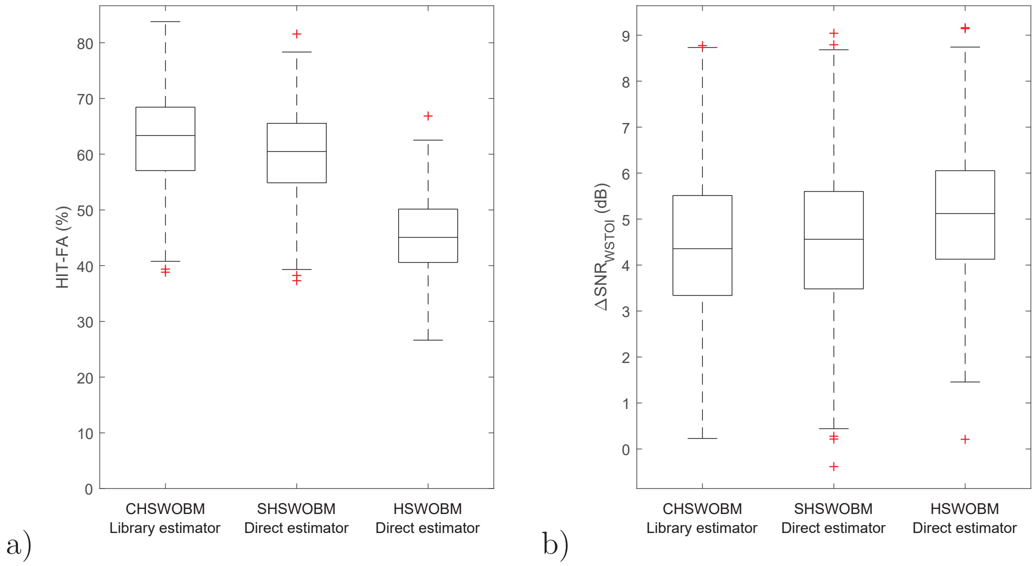


Figure 5.16: Comparison of the Library Estimator with the Direct Estimator. Boxplots of a) the HIT-FA rate and b) the $\Delta\mathrm{SNR}_{\mathrm{WSTOI}}$, for the DNN-based estimator with $Q = 0$. The target mask for the Library Estimator is the CHSWOBM. The target mask for the Direct Estimator is either the SHSWOBM or the HSWOBM. Results are computed on the Validation dataset described Sec. 5.4 .

dataset described Sec. 5.4.

In all three cases there was an improvement in WSTOI (i.e. a positive $\Delta\text{SNR}_{\text{WSTOI}}$) after processing with the estimated masks. In terms of the HIT-FA rate, which measures the accuracy of the mask estimate, the order of performance, from worst to best, was the HSWOBM, the SHSWOBM, then the CHSWOBM. This order reflects the degree of compression in the target mask; increasing the compression increased the classification accuracy. However, the order of performance was reversed for the WSTOI metric; increasing the compression resulted in less improvement in WSTOI.

Fig. 5.17 illustrates the effect of applying an estimated CHSWOBM and an estimated HSWOBM to a brief extract from a speech utterance containing babble noise at –0.64 dB SNR. Plots (a) and (b) show the clean and noisy speech. Plots (c), (d) and (e) show respectively the oracle CHSWOBM mask, the estimated mask and the classification results of the estimated mask. Plots (f), (g) and (h) show the corresponding plots for the HSWOBM mask instead. Consistently with Fig. 5.16, the HSWOBM results in a lower HIT-FA rate than the CHSWOBM, but a higher WSTOI: 33% versus 61% and 0.76 versus 0.70 respectively. Most of the difference in the HIT-FA rate arises from the difference in HIT rates (68 % for the CHSWOBM, 41 % for the HSWOBM). This is partly due to the oracle HSWOBM being more sparse; the percentage of ones in the oracle HSWOBM is 24 %, compared with 42 % in the oracle CHSWOBM. This means that a smaller difference in the number of absolute errors (1160 misses compared with 1319 for the HSWOBM) translates into a large difference in the HIT rate. The difference in HIT-FA may be due to the compressed masks being an easier target for the estimation algorithm; compressing the information in the HSWOBM that is important for speech intelligibility into a more

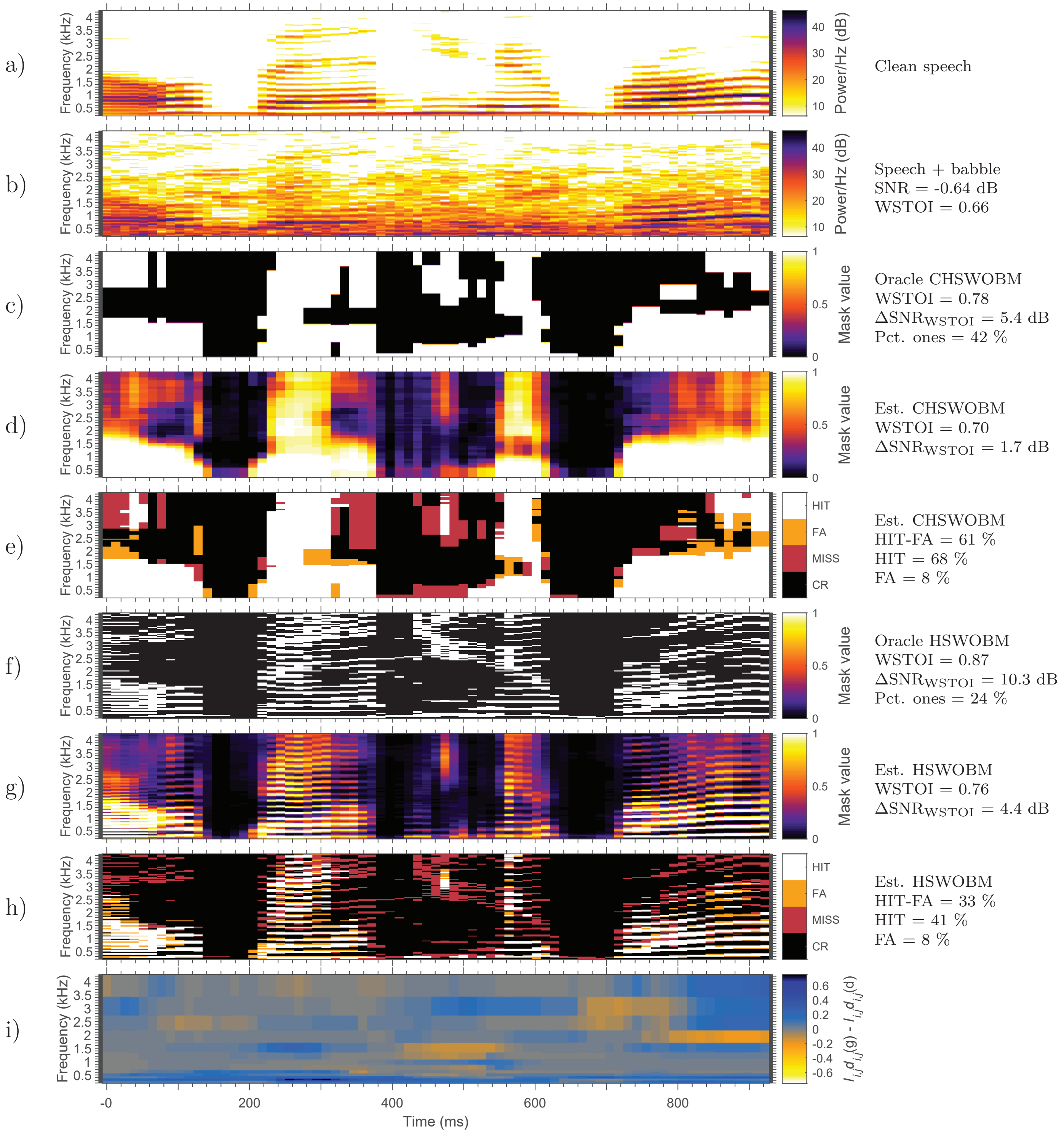


Figure 5.17: Spectrograms of a) part of an utterance of the phrase “well then who brought it?”, b) the utterance after adding babble noise with an SNR of -0.64 dB. Spectrograms of

the masks produced by c) the oracle CHSWOBM for stochastic white Gaussian noise with -5 dB SNR, d) an estimated CHSWOBM, e) errors in the estimated CHSWOBM after binarising (HITs, False Alarms, MISSes and Correct Rejections), f) the oracle HSWOBM, g) an estimated HSWOBM, h) errors in the above mask after binarising and i) the difference between the weighted intermediate WSTOI measure, $I_{j,m}d_{j,m}$, computed on corresponding pairs of TF cells in the signals produced by applying masks g) and d) to the noisy speech. The estimated masks were obtained using DNN-based estimators with $Q = 1$.

compact form may make it easier for the mask estimator to learn the mapping between features and mask.

The reduction in WSTOI that resulted from using compressed target masks may be due to the fact that the compression process destroyed some information that contributed to speech intelligibility. This is evidenced by the fact that the compressed oracle masks gave slightly lower WSTOIs when they were applied directly; the improvement in WSTOI for these samples decreases from 0.21 for the oracle HSWOBM to 0.18 and 0.17 for the oracle SHSWOBM and CHSWOBM, respectively. The reduction in WSTOI that resulted from this loss of information seems to have outweighed any positive effect of the compression on WSTOI that might have arisen through improving the mask estimation accuracy. It can be seen that, despite having a worse HIT-FA rate, the estimated HSWOBM still captures much of the fine detail in the speech such as the harmonics of the fundamental frequency of the speaker, which cannot be captured by the estimated CHSWOBM as this information is lost during the target mask compression.

5.5.4 Comparison of feature sets

Fig. 5.18 compares the $\Delta\text{SNR}_{\text{WSTOI}}$ for a range of noise types when the estimator used either the proposed feature set from Sec. 5.2, or the alternative cochleagram feature set from Sec. 5.2.4. The DNN-based estimator was used with $Q = 2$, and the HSWOBM as the target. The algorithm was trained on utterances from the Training dataset described in Sec. 5.4 mixed with speech shaped noise and babble noise, and results were computed on utterances from the Validation dataset mixed with 8 noise types from the RSG.10 [139] database, including the 2 that were used during training. The 8 noise types were Speech Shaped (SS), babble, operations room, F16, Lynx, factory, Volvo and machine gun.

When the noise type matched the noises used in training (SS and babble noise) the two feature sets showed very similar performance. However, in 4 out of the remaining 6 noise types (operations room, Lynx, F16, machine gun) the proposed feature set clearly outperformed the cochleagram feature set. In factory noise, the performance of the two feature sets was virtually identical, and in Volvo noise the performance of the proposed feature set was worse than the alternative at low SNRs but better at high SNRs. Both feature sets performed poorly on the two most challenging noise types (machine gun noise and Volvo noise).

5.5.5 Comparisons with other mask estimators

The proposed mask estimator was compared with two existing methods of estimating a binary mask: the algorithm from [95], denoted here as AMS-GMM, and the algorithm from [21], denoted here as Cochleagram-DNN or Cg-DNN.

The AMS-GMM algorithm [95] uses Amplitude Modulation Spectrogram (AMS) features and GMMs to estimate the IBM. To compute the AMS features, the noisy signal is first band-passed filtered into 25 channels which are equally spaced on the Mel frequency

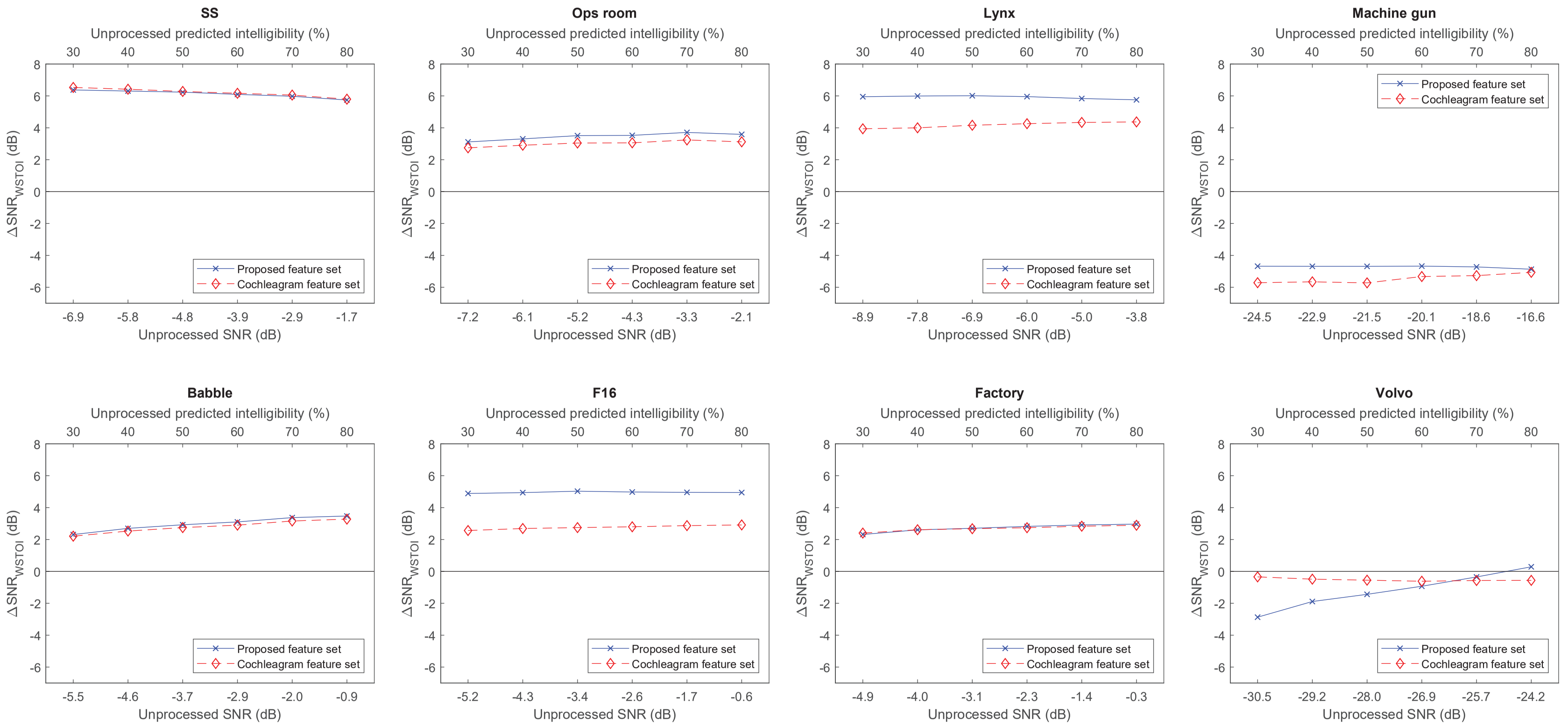


Figure 5.18: $\Delta\mathrm{SNR}_{\mathrm{WSTOI}}$ versus the SNR of the unprocessed noisy speech when the estimator used the proposed feature set from Sec. 5.2, and the alternative feature set from Sec. 5.2.4. The DNN-based estimator was used with $Q = 2$, and the HSWOBM as the target. The algorithms were trained on utterances from the Training dataset described in Sec. 5.4, which contains speech and babble noise, and results were computed on utterances from the Test-8N dataset.

scale [142]. Signal envelopes are then extracted by full-wave rectification, the envelopes are segmented into overlapping segments, the segments are Hanning windowed, and an FFT is performed on each windowed segment. The FFT coefficients are then summed within 15 triangular shaped windows spaced uniformly between 15.6 - 400 Hz, to give 15 modulation amplitudes. Delta features are computed across time (15 features) and frequency (15 features), to give $15 \times 3 = 45$ features in total per TF unit. The target of the classifier is a modified IBM which has two β parameters, one for the 15 lowest frequency bands, β_l, and another for the 10 higher frequency bands, β_h. In the algorithm training phase, the TF units are first divided into two groups (corresponding to mask zeros and ones) by comparing the local SNR in that TF unit to either β_l or β_h. Within each group, the TF units are then divided into two subgroups by comparing their SNR to another threshold. Each of these subgroups corresponds to a classifier class, so that there are four classes in total: two classes corresponding to mask zeros and two classes to mask ones. One GMM is trained per class, on the features extracted from the TF units belonging to that class. In the test phase, TF units are classified as 0 or 1 according to which of the four GMMs provides the highest posterior probability, the probability of that class given the features. In the experiments below, the AMS features were computed using the code from [112]. The parameters β_l and β_h were set equal to the average SNR in the training utterances in the lowest 15 and highest 10 frequency bands, respectively. The SNR thresholds used to form the subgroups were then chosen to ensure an equal number of TF units in each subgroup.

The other algorithm included for comparison, the Cg-DNN algorithm [21], uses a cochleagram feature set similar to the one outlined in Sec. 5.2.4, and a DNN, to estimate the

Ideal Ratio Mask (IRM). Apart from the target mask, the estimator is similar to the Direct Estimator with $Q = 2$, but with a few differences, such as the size of the frames (20-ms instead of 25.6 ms), and the resolution of the cochleagram (64 frequency channels centred from 50 to 8000 Hz instead of $\Omega = 90$ frequency channels centred from 50 to 5000 Hz). The target IRMs are defined by comparing the cochleagram representations of the speech and noise.

In the first experiment, the AMS-GMM algorithm was trained and tested at a high SNR in order to verify its operation: the algorithm was trained on a modified version of the Training dataset described in Sec. 5.4, modified so that all the utterances had an SNR of +10 dB which is much higher than the SNR values in the original Training dataset, which were $\{-6.89, -5.81, -4.85, -3.92, -2.92, -1.71\}$ dB for SS noise and $\{-5.53, -4.59, -3.75, -2.91, -2.01, -0.91\}$ dB for babble noise. The results, shown in Fig. 5.19, were computed on a modified version of the Validation dataset where the SNR was also +10 dB for all utterances. The results show quite a high HIT-FA rate, which indicates that the algorithm functions correctly. The average $\Delta\text{SNR}_{\text{WSTOI}}$ was +0.2 dB, which is very small, but this is not surprising since the WSTOI of the speech before enhancement (at +10 dB SNR) was very high and was therefore difficult to improve upon. Fig. 5.20 shows spectrograms of one of the noisy utterances from this experiment alongside the oracle IBM for the utterance and the IBM estimate produced by the AMS-GMM algorithm. The estimated IBM is similar to the oracle IBM in most TF regions but contains a large number of isolated peaks.

Fig. 5.21 and Fig. 5.22 compare the Cg-DNN and AMS-GMM algorithms with three variations of the proposed Direct Estimator, all using $Q = 2$ and the proposed feature set.

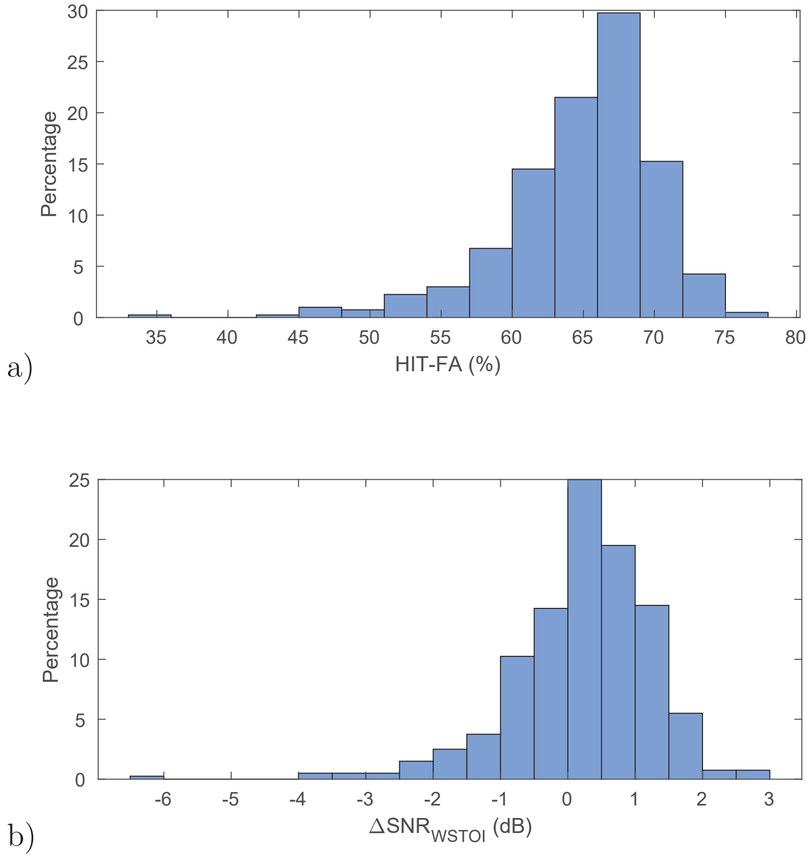


Figure 5.19: Histograms of a) the HIT-FA rate and b) $\Delta\text{SNR}_{\text{WSTOI}}$ for IBMs estimated using the AMS-GMM algorithm from [95]. The algorithm was trained and tested on modified versions of the Training and Validation datasets described in Sec. 5.4, modified so that all the utterances had much higher SNRs of +10 dB.

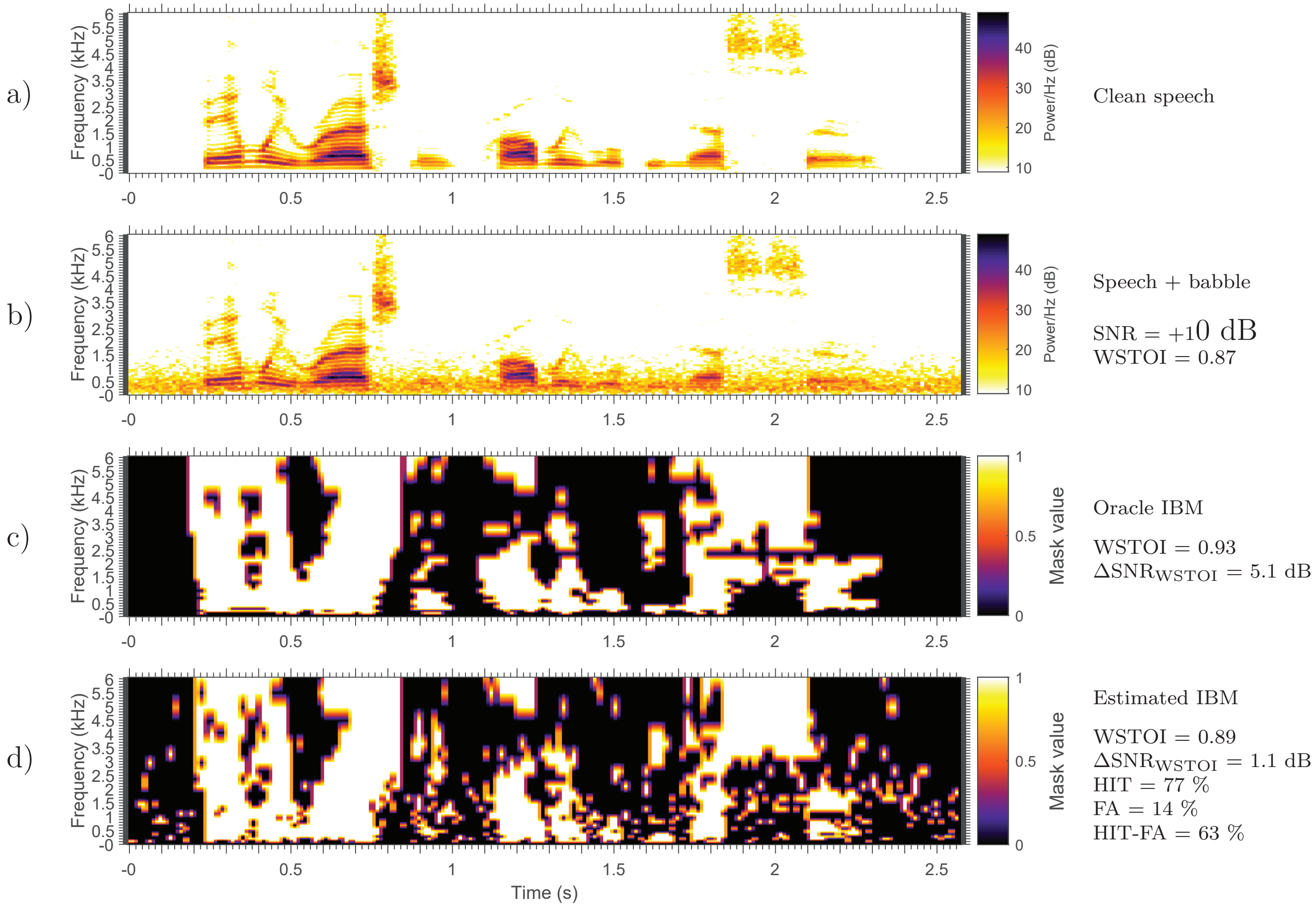


Figure 5.20: Spectrograms of a) part of an utterance of the phrase “family rationing probably will be necessary”, b) the utterance after adding speech shaped with an SNR of +10 dB, c) an oracle IBM and d) the IBM estimated using the AMS-GMM algorithm from [95]. The algorithm was trained and tested on modified versions of the Training and Validation datasets, modified so that all the utterances had SNRs of +10 dB.

Of the Direct Estimators, one used the DNN with an HSWOBM target, another used the DNN with an IBM target, and the third used the LSTM with an HSWOBM target. The algorithm parameters were trained on the Training dataset, algorithm hyperparameters were trained on the Validation dataset, and the results were computed on the Test-8N dataset. The results for six of the noise types are shown on separate plots in Fig. 5.21; those for the two very challenging noise types, Volvo and machine gun noise, are plotted separately in Fig. 5.22.

For the six noise types shown in Fig. 5.21, all the algorithms improved WSTOI apart from the AMS-GMM algorithm which damaged WSTOI for all noise types. Of the Direct Estimators which estimated the HSWOBM, the DNN algorithm outperformed the LSTM algorithm in all of these noises. Of the Direct Estimators which were based on the DNN, the IBM estimator performed similarly to the HSWOBM estimator, apart from in F16 noise, where the HSWOBM estimator outperformed the IBM estimator. All of the Direct Estimators outperformed the existing algorithms (Cg-DNN and AMS-GMM) in all the noise types.

With machine gun noise, shown in the left plot of Fig. 5.22, all of the algorithms damaged the WSTOI scores of the noisy utterances. In Volvo noise, shown in the right plot of Fig. 5.22, only the Cg-DNN algorithm resulted in a substantial improvement in WSTOI. Fig. 5.23 compares the Cg-DNN algorithm with the proposed DNN-based Direct Estimator, with $Q = 2$ and a HSWOBM target. Plots (a) and (b) respectively show spectrograms of the clean speech and of noisy speech with Volvo noise at -30.4 dB SNR. As it can be seen from plot (b), the SNR is so low that the speech is barely visible in the spectrogram of the noisy speech. Plots (c) and (d) show the oracle IRM and its estimate using the Cg-DNN algorithm while plots (e) and (f) show the oracle HSWOBM and its estimate using the proposed DNN algorithm. Plot (g) shows the difference in the intermediate WSTOI resulting from applying the two masks where positive values (blue) indicate that the Cg-DNN algorithm results in higher intelligibility. The small plot to the right of plot (g) shows the temporal mean of the plot; it can be seen that most of the difference in terms of WSTOI between the methods is due to differences in the high frequency bands. In the highlighted region in particular, the Cg-DNN algorithm significantly outperformed the proposed algorithm. In this region, the Cg-DNN algorithm applies a very low valued mask which varies smoothly, whereas the proposed algorithm imposes a more severe mask with sudden transitions and a modulation pattern which is less similar to the clean speech.

5.6 Summary

This chapter has presented a number of procedures for estimating a binary mask from noisy speech. It began by defining a feature set to use as the input to the estimation algorithm. The feature set is based on the TF gains estimated by a classical speech enhancement algorithm, and an estimate of the local VSNNR in different TF regions, obtained using a pitch estimator. It was found that a DNN-based estimator outperformed estimators based on an LSTM and a GMM. It also demonstrated that estimators trained on each of the three proposed target masks were all able to increase the WSTOI of noisy speech. Of these estimators, the Direct Estimator trained using the HSWOBM as the target provided the greatest improvement in WSTOI. It was also found that the proposed feature set

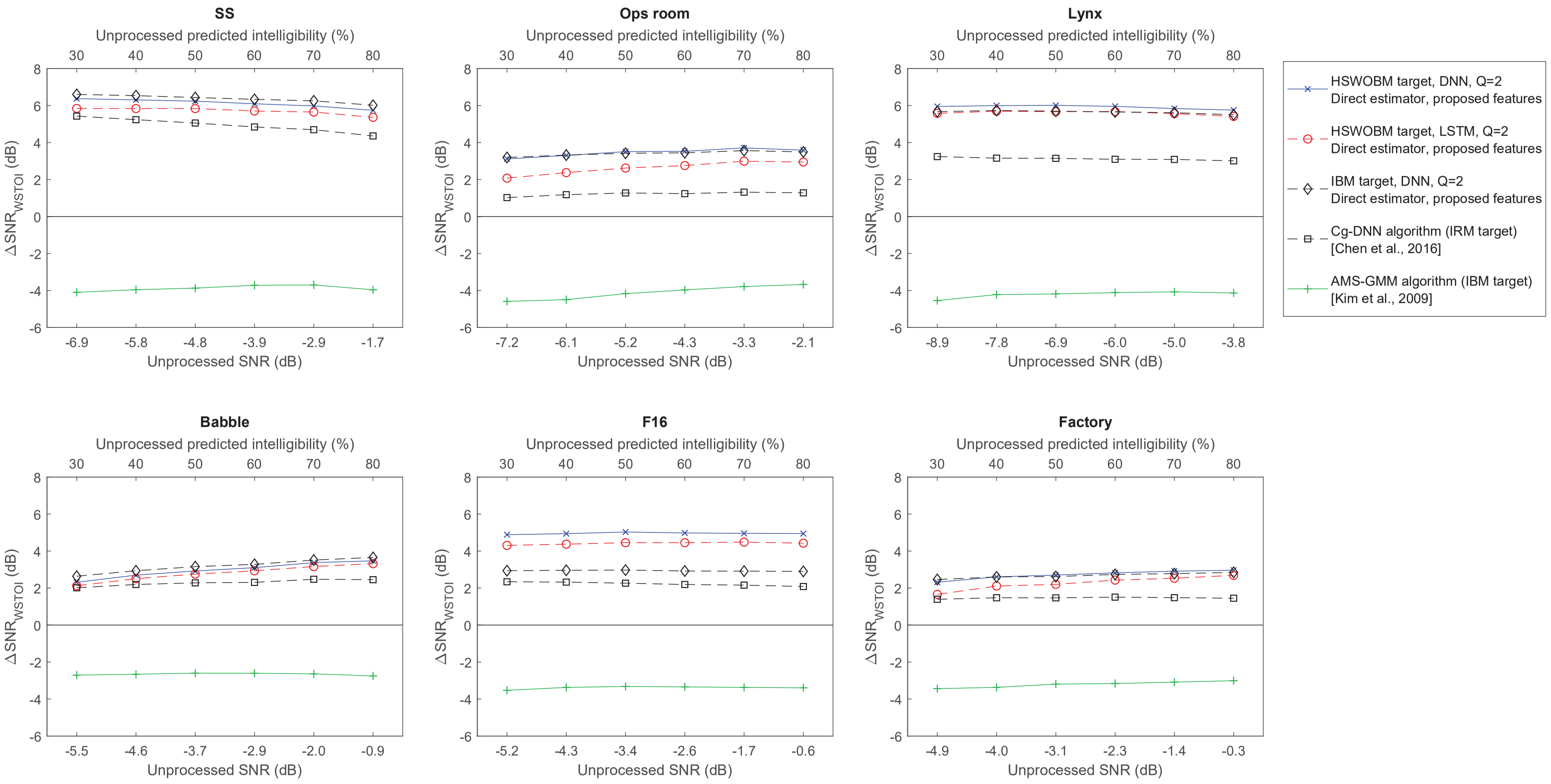


Figure 5.21: $\Delta \mathrm{SNR}_{\mathrm{WSTOI}}$ against the SNR of the unprocessed noisy speech, for three variations of the Direct Estimator, all using $Q = 2$ and the proposed feature set, and for the Cg-DNN algorithm [21] and the AMS-GMM algorithm [95]. The algorithms were trained on the Training and Validation datasets, and the results were computed on the Test-8N dataset.

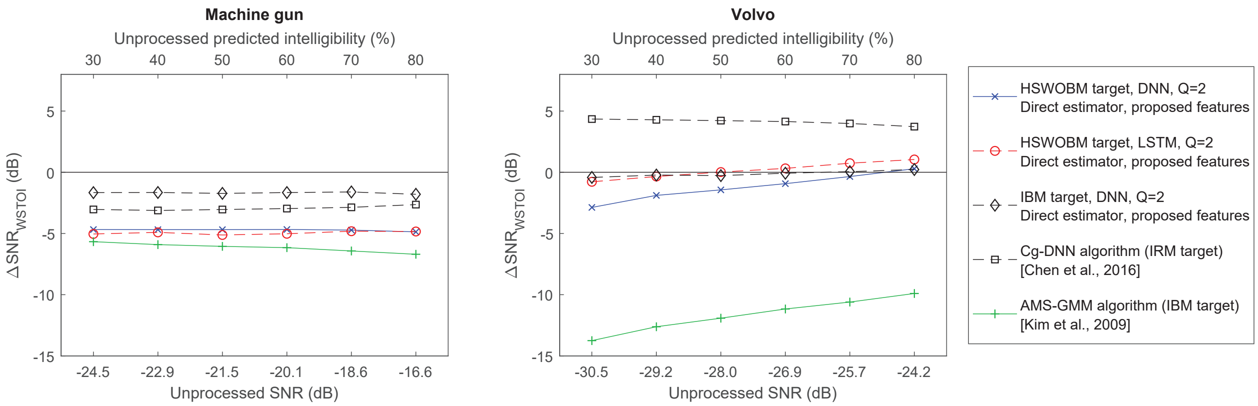


Figure 5.22: $\Delta \mathrm{SNR}_{\mathrm{WSTOI}}$ against the SNR of the unprocessed noisy speech, for three variations of the Direct Estimator, all using $Q = 2$ and the proposed feature set, and for the Cg-DNN algorithm [21] and the AMS-GMM algorithm [95]. The algorithms were trained on the Training and Validation datasets, and the results were computed on the Test-8N dataset. Results are displayed for the two most challenging noise types.

matched or outperformed the cochleagram feature set from [21] in 7 of the 8 tested noise types, including 5 of the 6 noise types that were not seen by the algorithm during training. Finally, we observed that the proposed mask estimation algorithm outperformed the Cg-DNN mask estimator from [21] in 6 of the 8 tested noise types.

Although the procedures proposed in this chapter result in an increase in the predicted intelligibility of noisy speech, it will be shown, in the following chapter, that the predicted quality of the speech resulting from these procedures is quite poor. The focus of the next chapter will therefore be an alternative way of applying the estimated binary mask that results in a higher predicted quality than the approach used in this chapter whilst retaining the intelligibility gains provided by this approach.

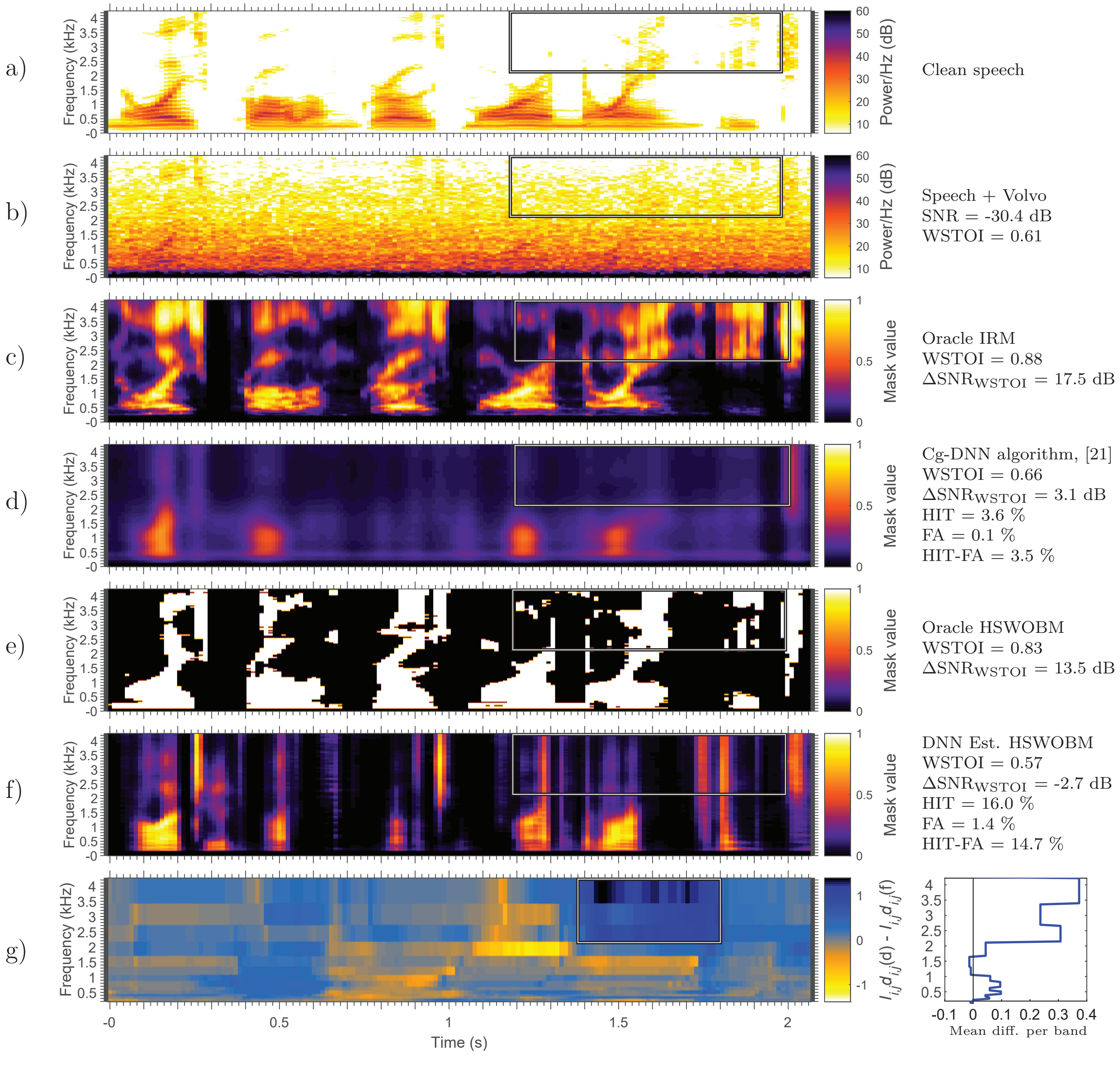


Figure 5.23: Spectrograms of a) part of an utterance of the phrase “that noise problem grows more annoying each day”, b) the utterance after adding Volvo car noise with an SNR of -30.4 dB, c) the oracle IRM, d) the mask produced by the Cg-DNN algorithm from [21], e) the oracle HSWOBM for stochastic white Gaussian noise with -5 dB SNR, f) the estimated HSWOBM produced by the proposed DNN-based enhancer, and g) the difference between the weighted intermediate WSTOI measure, $I_{j,m}d_{j,m}$, computed on corresponding pairs of TF cells in the signals produced by applying masks d) and f) to the noisy speech.

Chapter 6

Optimal mask application

6.1 Introduction

The conventional way to apply a binary mask to noisy speech is to multiply speech by the mask in the STFT-domain and then convert the resulting signal back into the time-domain. However, although applying a binary mask in this manner can improve the intelligibility of noisy speech, the enhanced speech often has very poor perceptual quality. This may be partly due to the fact that the gain changes instantaneously between TF units in neighbouring frames with different mask values. This makes the speech and noise switch on and off abruptly and synchronously, giving a harsh and unnatural quality to the speech. The mask may also contain isolated peaks which give rise to musical noise and classification errors which can introduce distortion artefacts into the speech.

In this chapter we present an alternative approach to applying a binary mask that preserves the intelligibility gains of conventional binary masking whilst addressing the issue of poor speech quality. This approach encompasses the “Apply mask” module of the mask-based enhancer shown in Fig. 5.1. We are motivated by the observation from [98], described in Sec. 2.2.1, that the intelligibility gains of binary masked speech arise because the mask identifies the TF cells containing significant speech energy. Accordingly, in our proposed approach we do not use the mask directly as a TF gain but instead use it to supply prior information about the Speech Presence Probability (SPP) to a classical speech enhancer [23] that minimises the expected squared error in the Log Spectral Amplitude (LSA). The enhancer from [23] was chosen as it modifies the popular speech enhancer from [37] to apply the optimal gain under the conditions of a signal presence uncertainty, and was shown to improve the performance. To evaluate this approach we have used an oracle HSWOBM. We have also used an oracle IBM in order to demonstrate that the proposed approach to mask estimation works well with other binary masks.

6.2 Signal presence and absence

The proposed approach and the algorithm from [23] both assume that the noisy speech STFT coefficients, $Y(k,m)$, can be modelled as arising from one of two probability distributions, according to whether or not speech is “present” or “absent” in TF bin (k,m). Fig. 6.1 shows histograms of the magnitudes of the STFT coefficients of the speech, X, noise, N, and noisy speech, Y, for frequency bins between 1.5 kHz and 2.3 kHz, in speech utterances containing speech shaped noise at an SNR of 0 dB. Two histograms are plotted for each signal, with each STFT coefficient assigned to one of the two histograms according to whether speech was determined to be present or absent in that bin. Speech was

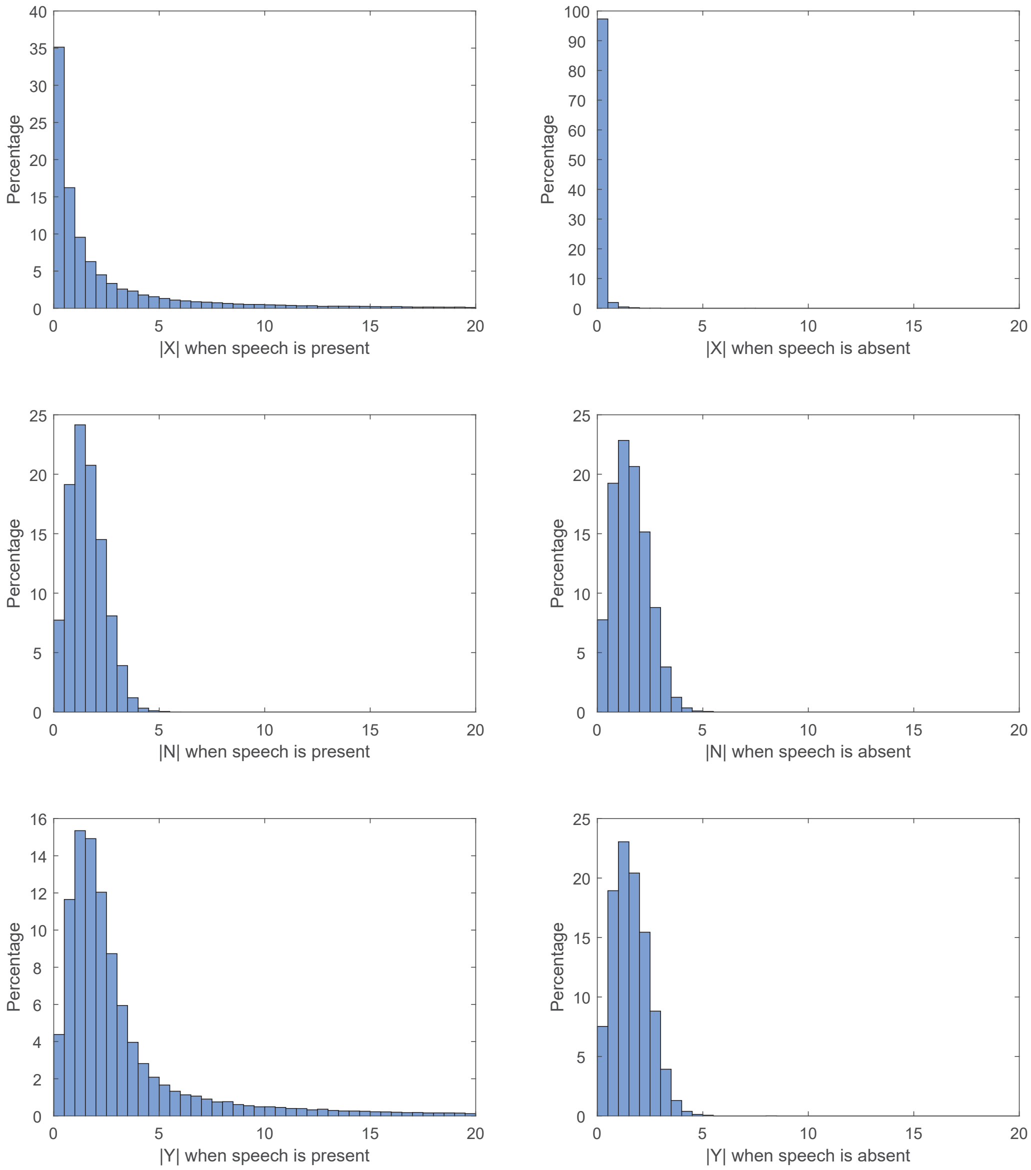


Figure 6.1: Histograms of the magnitude of the STFT coefficients of the speech, X, noise, N, and noisy speech, Y, for frequency bins between 1.5 kHz and 2.3 kHz in speech utterances containing speech shaped noise at an SNR of 0 dB. Two histograms are plotted for each signal, with STFT coefficients assigned to one of the histograms according to whether speech was determined to be present or absent in that bin. Speech was determined to be present or absent in each frame, m, by applying the voice activity detection algorithm from [137] to the clean speech.

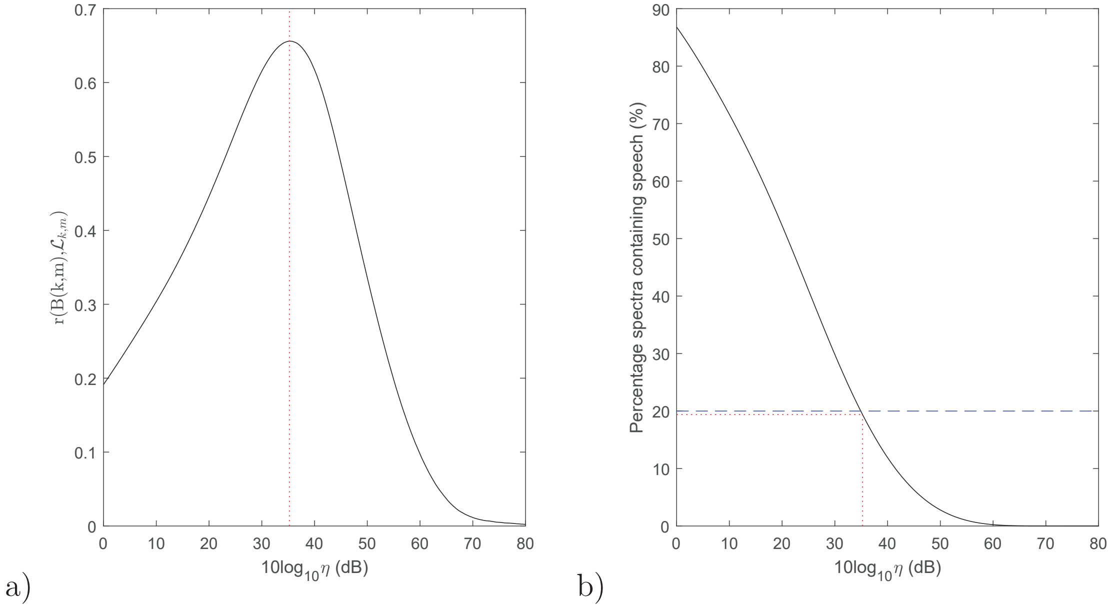


Figure 6.2: a) Plot of the Pearson correlation coefficient, $r(\cdot)$, computed between the signals $B(k, m)$ and $\mathcal{L}_{k,m}$ over all bins (k, m), against η, where $B(k, m)$ were oracle HSWOBMs for speech utterances containing stochastic white Gaussian noise signals noise at -5 dB SNR, and b) the percentage of TF units in which speech is considered (the sparsity of the speech) to be activate against η. In (a) the value of $10\log_{10}\eta$ corresponding to the maximum correlation (35 dB) is highlighted with the red dotted line. In (b) the sparsity of the speech corresponding to the maximum correlation (19.4 %) is highlighted with the red dotted line. The percentage of ones in the oracle HSWOBMs (20.0 %) is marked with the blue dashed line.

determined to be present or absent in each frame, m, by applying the voice activity detection algorithm from [137] to the clean speech. The plot was formed using 400 utterances from the training set of the TIMIT corpus [45], mixed with noise from the RSG.10 [139] database. From the plot it can be seen that the distribution of the noisy speech STFT coefficients, Y, depends on whether the speech signal is “present” or “absent”; when the speech is present, the distribution is broader.

A further assumption of the proposed approach is that the estimated binary mask is able to provide information about the SPP in each TF bin. If this is true, we would expect the oracle binary mask to be a good estimator of speech presence. To test this hypothesis we define a simple speech presence detector $\mathcal{L}_{k,m}$, where

$$\mathcal{L}_{k,m} = \begin{cases} 1 & \left|X(k,\, m)\right|^2 > \eta \cdot E\left[\left|D(k,\, m)\right|^2\right] \\ 0 & \text{otherwise} \end{cases} \qquad \forall (k,\, m)$$

where $D(k,\, m)$ is an internal ear noise which models the absolute threshold of human hearing, and values of $E\left[\left|D(k,\, m)\right|^2\right]$ were obtained by scaling the reference internal noise spectrum levels from Table 3 of [4] for each utterance so that the mean speech-to-internal-noise power ratio of the utterance during active speech periods matched the ratio of the speech and noise spectrum levels for a "normal" vocal effort. Active periods were identified using the procedure in [82]. The factor η is included to prevent time-frequency bins with extremely low signal energies from being labelled as having speech present, on the grounds that the quantity of speech energy present is likely to be insignificant in terms of quality or intelligibility. When $\eta = 1$, all TF regions in which the speech signal power exceeds the power of the internal ear noise are considered to contain speech. Fig. 6.2a shows the Pearson correlation coefficient, $r\left(\cdot\right)$, computed between the signals $B(k,\, m)$ and $\mathcal{L}_{k,m}$ over all bins, $(k,\, m)$, where $B(k,\, m)$ were oracle HSWOBMs computed for speech utterances containing stochastic white Gaussian noise at -5 dB SNR. The plot was formed using 400 utterances from the training set of the TIMIT corpus [45]. The value of $10\log_{10}\eta$ corresponding to the maximum correlation (35 dB) is highlighted with the red dotted line. It can be seen that, over all reasonable values of η, i.e. $0 \leq 10\log_{10}\eta < 80$ dB, the oracle HSWOBM was positively correlated with speech presence, and the correlation was strongest when $10\log_{10}\eta = 35$ dB. Fig. 6.2b shows the percentage of TF units in which speech is considered to be activate (i.e. the sparsity of the speech) against η. The sparsity of the speech corresponding to the maximum correlation (19.4 %) is highlighted with the red dotted line. The percentage of ones in the oracle HSWOBMs (20.0 %) is marked with the blue dashed line. The speech detector which gives the maximum correlation coefficient has approximately the same sparsity as the oracle mask.

6.3 Optimally-modified log-spectral estimator

Here we present a brief overview of the Log Spectral Amplitude (LSA) algorithm from [37], and the Optimally-Modified Log-Spectral Amplitude (OM-LSA) algorithm from [23] (shown in block diagram form in the upper plot of Fig. 6.3) on which our approach is based. The algorithm from [37] applies a gain to each STFT cell that minimises the mean-square error between the log-spectral amplitudes of the clean and processed speech signals under the assumption that the STFT coefficients of the speech and noise are statistically independent zero-mean complex Gaussian random variables. The algorithm from [23] extends this model to take account of signal presence uncertainty.

In both algorithms the noisy speech is first converted into the STFT-domain using overlapping Hamming analysis windows. Recall that $X(k, m)$, $N(k, m)$ and $Y(k, m)$ denote the zero-mean complex STFT coefficients of the clean speech, the noise and noisy speech respectively in frequency bin k of frame m. The variances of $X(k, m)$ and $N(k, m)$

are denoted by

$$\lambda_x\left(k,m\right) \triangleq E\left[\left|X(k,\, m)\right|^2\right],$$

$$\lambda_n\left(k,m\right) \triangleq E\left[\left|N(k,\, m)\right|^2\right].$$

A gain function, $G(k,\, m)$, is applied to frequency bin k of frame m, which satisfies

$$\log\left(G(k,\, m)\left|Y(k,\, m)\right|\right) = E\left[\log|X(k,\, m)| \mid Y(k,\, m)\right] \tag{6.1}$$

where $E\left[\cdot\right]$ is the expectation operator. In [37] the gain function is shown to be equal to

$$G(k,\, m) = \frac{\xi(k,\, m)}{1+\xi(k,\, m)}\exp\left(\frac{1}{2}\int_{v(k,\, m)}^{\infty}\frac{e^{-t}}{t}dt\right), \tag{6.2}$$

where

$$\xi(k,\, m) \triangleq \frac{\lambda_x\left(k,m\right)}{\lambda_n\left(k,m\right)} \tag{6.3}$$

is the *a priori* SNR, and

$$v(k,\, m) \triangleq \gamma(k,\, m)\xi(k,\, m)/\left(1+\xi(k,\, m)\right),$$

where

$$\gamma(k,\, m) \triangleq \frac{\left|Y(k,\, m)\right|^2}{\lambda_n\left(k,m\right)}$$

is the *a posteriori* SNR. An estimate of $\lambda_n\left(k,m\right)$ is usually obtained using a separate noise estimation algorithm, and is then used to estimate $\xi(k,\, m)$. A common approach to estimating $\xi(k,\, m)$ is the iterative "decision-directed" approach. Since

$$\xi(k,\, m) = E\left[\gamma(k,\, m) - 1\right], \tag{6.4}$$

we can combine (6.3) and (6.4) and write

$$\xi(k,\, m) = \alpha \frac{\lambda_x\left(k, m\right)}{\lambda_n\left(k, m\right)} + (1-\alpha)\, E\left[\gamma(k,\, m) - 1\right],$$

for $0 \leq \alpha < 1$. From this equation the authors of [37] deduce the estimator

$$\begin{aligned}\hat{\xi}(k,\, m) = \alpha G^2(k, m-1)\gamma(k, m-1)& \\ + (1-\alpha) \max\left\{\gamma(k,\, m) - 1, 0\right\}.&\end{aligned} \tag{6.5}$$

The approach is "decision-directed" since the gain of the current frame, $G(k,\, m)$, depends on the gain calculated in the previous iteration for the previous frame, $G(k, m-1)$.

In [37] the authors experimented with modifying the LSA algorithm to include the SPP, as they had previously done with a similar algorithm in [36]. Speech is now considered to be present in STFT bin $(k,\, m)$ under the hypothesis $H_1(k,\, m)$ and absent under the hypothesis $H_0(k,\, m)$, and the noisy coefficients $Y(k,\, m)$ are assumed to arise from one of two distributions according to which hypothesis is true. The distributions are

$$p\left(Y(k,\, m) | H_0(k,\, m)\right) = \frac{1}{\pi \lambda_n\left(k, m\right)} \exp\left\{-\frac{\left|Y(k,\, m)\right|^2}{\lambda_n\left(k, m\right)}\right\} \tag{6.6}$$

$$p\left(Y(k,\, m)|H_1(k,\, m)\right) = \frac{1}{\pi\left(\lambda_x\left(k,m\right)+\lambda_n\left(k,m\right)\right)}\exp\left\{-\frac{\left|Y(k,\, m)\right|^2}{\lambda_x\left(k,m\right)+\lambda_n\left(k,m\right)}\right\}. \tag{6.7}$$

We now have

$$\begin{aligned}\log\left(G(k,\, m)\left|Y(k,\, m)\right|\right) &= E\left[\log|X(k,\, m)| \mid Y(k,\, m),\, H_1(k,\, m)\right] P\left(H_1(k,\, m) \mid Y(k,\, m)\right)\\ &\quad + E\left[\log|X(k,\, m)| \mid Y(k,\, m),\, H_0(k,\, m)\right] P\left(H_0(k,\, m) \mid Y(k,\, m)\right).\end{aligned} \tag{6.8}$$

We define the gain functions under hypotheses H_1 and H_0 as $G_{H_1}(k,\, m)$ and $G_{H_0}(k,\, m)$ respectively, where

$$\log\left(G_{H_1}(k,\, m)\left|Y(k,\, m)\right|\right) \triangleq E\left[\log|X(k,\, m)| \mid Y(k,\, m),\, H_1(k,\, m)\right], \tag{6.9}$$

$$\log\left(G_{H_0}(k,\, m)\left|Y(k,\, m)\right|\right) \triangleq E\left[\log|X(k,\, m)| \mid Y(k,\, m),\, H_0(k,\, m)\right]. \tag{6.10}$$

hence

$$G_{H_1}(k,\, m) = \frac{\xi(k,\, m)}{1+\xi(k,\, m)}\exp\left(\frac{1}{2}\int_{v(k,\, m)}^{\infty}\frac{e^{-t}}{t}dt\right).$$

Since (6.10) is equal to negative infinity then (6.8) must also be equal to negative infinity, and the gain $G(k,\, m)$ is zero. To resolve this issue, in [23] the authors impose a minimum gain, G_{min}. during speech absence, so that

$$\begin{aligned}\log\left(G(k,\, m)\left|Y(k,\, m)\right|\right) &= \log\left(G_{H_1}(k,\, m)\left|Y(k,\, m)\right|\right)p(k,\, m)\\ &\quad + \log\left(G_{min}\left|Y(k,\, m)\right|\right)\left(1-p(k,\, m)\right),\end{aligned} \tag{6.11}$$

where the conditional speech presence probability,

$$p(k,\, m) \triangleq P\left(H_1(k,\, m) \mid Y(k,\, m)\right),$$

is computed as

$$p(k,\, m) = \left\{1 + \frac{1-\rho(k,\, m)}{\rho(k,\, m)}\left(1+\xi(k,\, m)\right)\exp\left(-v(k,l)\right)\right\}^{-1},$$

where $\rho(k,\, m) \triangleq P\left(H_1(k,\, m)\right)$ is the *a priori* SPP. From (6.11) we can obtain

$$G(k,\, m) = \left\{G_{H_1}(k,\, m)\right\}^{p(k,\, m)} G_{min}^{1-p(k,\, m)}.$$

The introduction of G_{min} has the effect of making the gain, $G(k,\, m)$, depend multiplicatively on the speech presence probability, and was shown to improvement performance. An estimate, $\hat{\xi}(k,\, m)$, of $\xi(k,\, m)$ is obtained using a modified version of (6.5) which takes account of the speech presence uncertainty,

$$\hat{\xi}(k,\, m) = \alpha G_{H_1}^2(k, m-1)\gamma(k, m-1)$$

$$+(1-\alpha)\max\left\{\gamma(k,\, m)-1, 0\right\}.$$

6.4 Speech presence probability prior

In [23] an estimator $\hat{\rho}(k, m)$ was used to obtain the SPP from $\hat{\xi}(k, m)$. We propose to instead obtain $\hat{\rho}(k, m)$ from a binary mask, $B(k, m)$, by setting

$$\hat{\rho}(k, m) = \begin{cases} \phi^1 & B(k, m) = 1 \\ \phi^0 & B(k, m) = 0 \end{cases} \tag{6.12}$$

where ϕ^1 and ϕ^0 are free parameters. Similarly, the value of G_{min} is set to

$$G_{min} = \begin{cases} G^1 & B(k, m) = 1 \\ G^0 & B(k, m) = 0 \end{cases} \tag{6.13}$$

where G^1 and G^0 are free parameters. A diagram of the proposed method is shown in Fig. 6.3, with the differences from the method of [23] highlighted in red. This method is denoted Minimum Mean Squared Error Mask Application (MMSE-MA).

By using the value of the binary mask to control the probability of speech presence in this way, the algorithm softly imposes on the enhanced speech the spectro-temporal modulations that are encapsulated in the mask and that are important for speech intelligibility [98, 151]. At the same time, the algorithm improves the SNR and the perceived quality of the speech by applying an SNR-dependent time-frequency gain, $G(k, m)$.

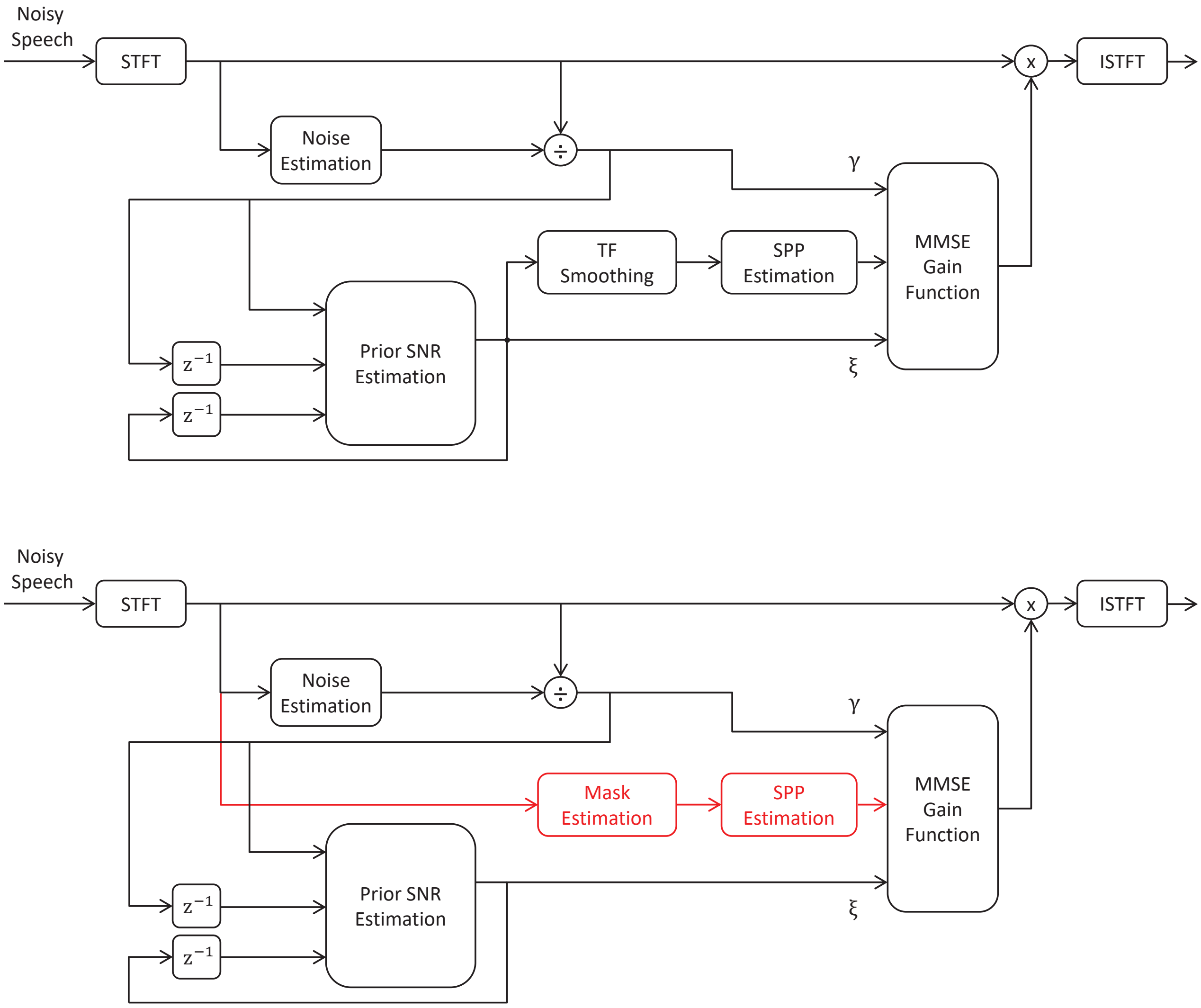


Figure 6.3: Top) A diagram of the algorithm from [23]. Bottom) A diagram of the proposed method of applying the binary mask, with the differences with the method from [23] highlighted in red.

6.5 Existing methods for comparison

For comparison we will test the effect of applying the binary mask in several other ways. Firstly, we evaluate applying the mask in the conventional way. We apply a gain, $G(k,\, m)$, such that

$$G(k,\, m) = \begin{cases} 1 & B(k,\, m) = 1 \\ 0 & B(k,\, m) = 0 \end{cases}.$$

This is denoted as Conventional Mask Application (CMA). We also test the effect of imposing a minimum gain on the mask, so that

$$G(k,\, m) = \begin{cases} 1 & B(k,\, m) = 1 \\ U & B(k,\, m) = 0 \end{cases}$$

where U is the minimum gain. This has been shown to improve perceptual quality in listening tests [114], and is denoted as Conventional Mask Application with a Minimum Gain (CMA-MG).

The final method of mask application used for comparison involves applying temporal smoothing to the cepstrum of the mask [114] before applying it, and is denoted as Conventional Mask Application with Cepstral Smoothing (CMA-CS). We first compute

$$D(k,\, m) = \begin{cases} 1 & B(k,\, m) = 1 \\ U & B(k,\, m) = 0 \end{cases},$$

followed by its cepstrum,

$$D^{\mathrm{cepst}}(l,m) = \mathrm{DFT}^{-1}\left\{\ln\left(D(k,\, m)\mid_{k=0,\ldots,K-1}\right)\right\},$$

where l is the quefrency bin index, $\mathrm{DFT}\{\cdot\}$ represents the discrete Fourier transform operator, and the coefficients $D(k,\, m)$ for $k = K/2+1, ..., K-1$ are obtained using the symmetry of the DFT. We then apply smoothing across time to the cepstrum,

$$\overline{D}^{\mathrm{cepst}}(l,m) = \beta_l \overline{D}^{\mathrm{cepst}}(l,m-1) + (1-\beta_l)\left(D^{\mathrm{cepst}}(l,m)\right),$$

where smoothing constants β_l are chosen separately for different quefrequency bins l according to:

$$\beta_l = \begin{cases} \beta_{\mathrm{env}} & \text{if } l \in \{0, ..., l_{\mathrm{env}}\} \\ \beta_{\mathrm{pitch}} & \text{if } l = l_{\mathrm{pitch}} \\ \beta_{\mathrm{peak}} & \text{if } l \in \{(l_{\mathrm{env}}+1), ..., K/2\} \setminus \{l_{\mathrm{pitch}}\}. \end{cases}$$

For the lower bins, $l \in \{0, ..., l_{\mathrm{env}}\}$, the values of $D^{\mathrm{cepst}}(l,m)$ contain information about the spectral envelope of the speech and β_{env} should therefore have a very low value to prevent introducing distortion to the speech. Likewise, since $l = l_{\mathrm{pitch}}$ is the quefrency bin that represents the regular structure of the pitch harmonics in $D^{\mathrm{cepst}}(l,m)$ we also desire a relatively low value of β_{pitch}. The quefrency bins $l \in \{(l_{\mathrm{env}}+1), ..., K/2\} \setminus \{l_{\mathrm{pitch}}\}$ represent the fine structure of $D^{\mathrm{cepst}}(l,m)$ that is not related to the speech, such as isolated random peaks with a very short duration that cause musical noise, hence we desire strong smoothing with $\beta_{\text{peak}} > \beta_{\text{pitch}}$. The value of l_{pitch} is selected as

$$l_{\text{pitch}} = \underset{l}{\operatorname{argmax}} \left\{ D^{\text{cepst}}(l, m) \mid l_{\text{low}} \leq l \leq l_{\text{high}} \right\},$$

where $\{l_{\text{low}}, l_{\text{high}}\}$ determines the range over which to search. For bins $l > K/2$ we have $\overline{D}^{\text{cepst}}(l, m) = \overline{D}^{\text{cepst}}(K - l, m)$ due to the symmetry of the DFT.

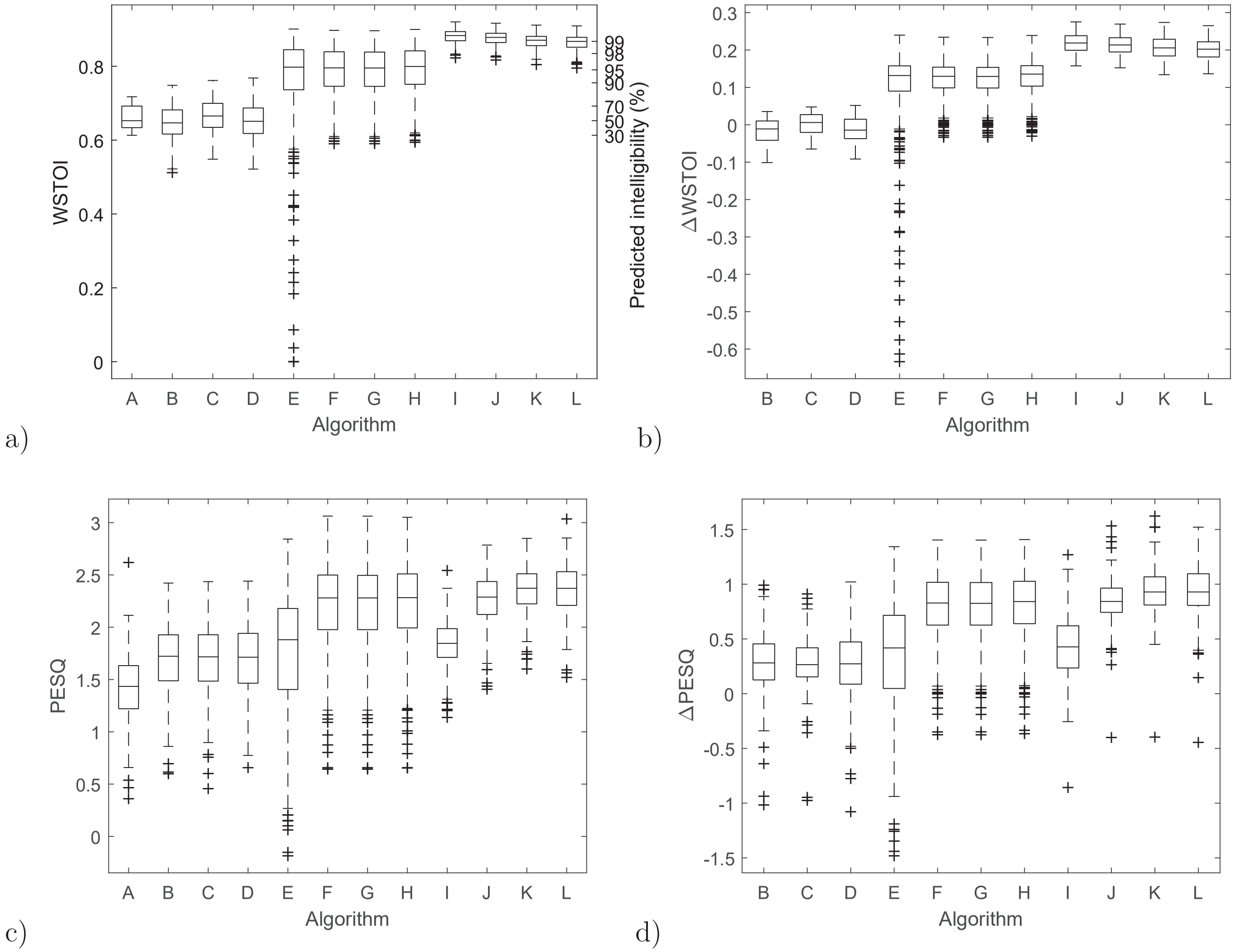
Finally, we apply a gain, $G(k,\, m)$, where

$$G(k,\, m) = \exp\left(\text{DFT}\left\{\overline{D}^{\text{cepst}}(l, m) \mid_{l=0,\ldots,K-1}\right\}\right).$$

It can be seen that if we select $\beta_{\text{env}} = \beta_{\text{pitch}} = \beta_{\text{peak}} = 0$ then $\overline{D}^{\text{cepst}}(l, m) = D^{\text{cepst}}(l, m)$ and $G(k,\, m) = D(k,\, m)$. In this case we are simply imposing a minimum gain.

6.6 Experimental procedures

The algorithm parameters are listed in Table 6.1 and were trained on 80 TIMIT utterances selected randomly from the Training dataset from Chapter 5. The algorithms were then tested on the Validation dataset from Chapter 5, comprising 400 TIMIT utterances containing babble and SS noise, with SNRs of $\{-6.89,\, -5.81,\, -4.85,\, -3.92,\, -2.92,\, -1.71\}$ dB for SS noise and $\{-5.53,\, -4.59,\, -3.75,\, -2.91,\, -2.01,\, -0.91\}$ dB for babble noise. The selection procedure for the training utterances was constrained to give an almost identical number of samples for each noise type (babble and SS) and each noise level. All signals were resampled to 10 kHz. The STFT used to compute the IBM used 50% overlapping Hanning analysis windows of length 25.6 ms.



	Algorithm Key
A	Noisy speech
B	LSA [37]
C	OM-LSA [23], max-WSTOI
D	OM-LSA [23], max-PESQ
E	IBM with CMA
F	IBM with CMA-MG [144]
G	IBM with CMA-CS [114]
H	IBM with MMSE-MA
I	HSWOBM with CMA
J	HSWOBM with CMA-MG
K	HSWOBM with CMA-CS
L	HSWOBM with MMSE-MA

Figure 6.4: Boxplots of a) WSTOI, b) PESQ and c) increase in PESQ after processing for noisy speech utterances after applying different enhancement algorithms. Methods E-H use an oracle IBM and methods I-L use an oracle HSWOBM.

A total of 11 enhancement methods were evaluated (labelled B through L in Fig. 6.4). The proposed method for mask application was first tested with oracle masks, which are computed using knowledge of the clean speech: both an oracle IBM (H) and an oracle HSWOBM (L) were tested. These were compared with two classical enhancement algorithms: the LSA estimator (B) [37], the OM-LSA estimator (C, D) [23], in addition to the IBM [164] and HSWOBM with different methods of mask application: CMA (E, I), CMA-CS (G, K) [114], and CMA-MG (F, J) [144].

The parameters of methods F, G, J and K were chosen to maximise the PESQ objective quality metric. The parameters of method H and L were chosen to maximise the sum of a normalised predicted intelligibility score and a normalised predicted MOS, where each PESQ score was mapped to a predicted MOS using the mapping from [85] and each WSTOI score was mapped to a predicted intelligibility using the mapping from Chapter 3. For the methods that used an oracle IBM (F, G, H), the optimal value of the LC, β, was obtained for each utterance using the mapping given by (2.2). The OM-LSA algorithm parameter was chosen to optimise either WSTOI (method C) or PESQ (method D). For all algorithm parameters other than those listed in Table 6.1, the default values from [15, 114, 23] were used. The l_{env}, l_{low} and l_{high} cepstral smoothing parameters in [114] were adjusted to account for the 10 kHz sample rate instead of the 8 kHz used in the study. The original values of $l_{\text{low}} = 16$ and $l_{\text{high}} = 120$ corresponded to a pitch search window of $\{8000/120 = 67\,\text{Hz}, 8000/16 = 500\,\text{Hz}\}$. This was adjusted to $l_{\text{low}} = 20$ and $l_{\text{high}} = 150$ which corresponds to the same pitch search window,. i.e. $\{10000/150 = 67\,\text{Hz}, 10000/20 = 500\,\text{Hz}\}$. The value of l_{env} was similarly adjusted from $l_{\text{env}} = 8$ to $l_{\text{env}} = 10$. We set $\beta_{\text{peak}} = 0$ as in [114]. The LSA (B), the OM-LSA (C, D) and

Algorithm	Parameter	Optimal value	
		IBM (F, G, H)	HSWOBM (J, K, L)
(F, J): Binary mask, gain floor	U	0.07	0.06
(G, K): Binary mask, cepst. smoothing	U	0.07	0.05
	β_{pitch}	0.11	0
	β_{peak}	0	0.465
(H, L): Proposed method	G^1	-20 dB	-1 dB
	G^0	-43 dB	-31 dB
	ϕ^1	0	0.415
	ϕ^0	0	0
C: OM-LSA max-WSTOI	G_{min}	-12 dB	
D: OM-LSA max-PESQ	G_{min}	-19 dB	

Table 6.1: Summary of trained parameters and their optimal values.

the proposed method (H, L) used the noise estimator from [47, 15].

6.7 Results

Figures 6.4 (a-d) show the WSTOI, ΔWSTOI, PESQ and ΔPESQ scores, respectively, for the noisy speech utterances after processing with different enhancement methods. The LSA (B) and OM-LSA (C, D) algorithms resulted in an average improvement in PESQ of about 0.3 compared with the unprocessed noisy speech. However, on average these

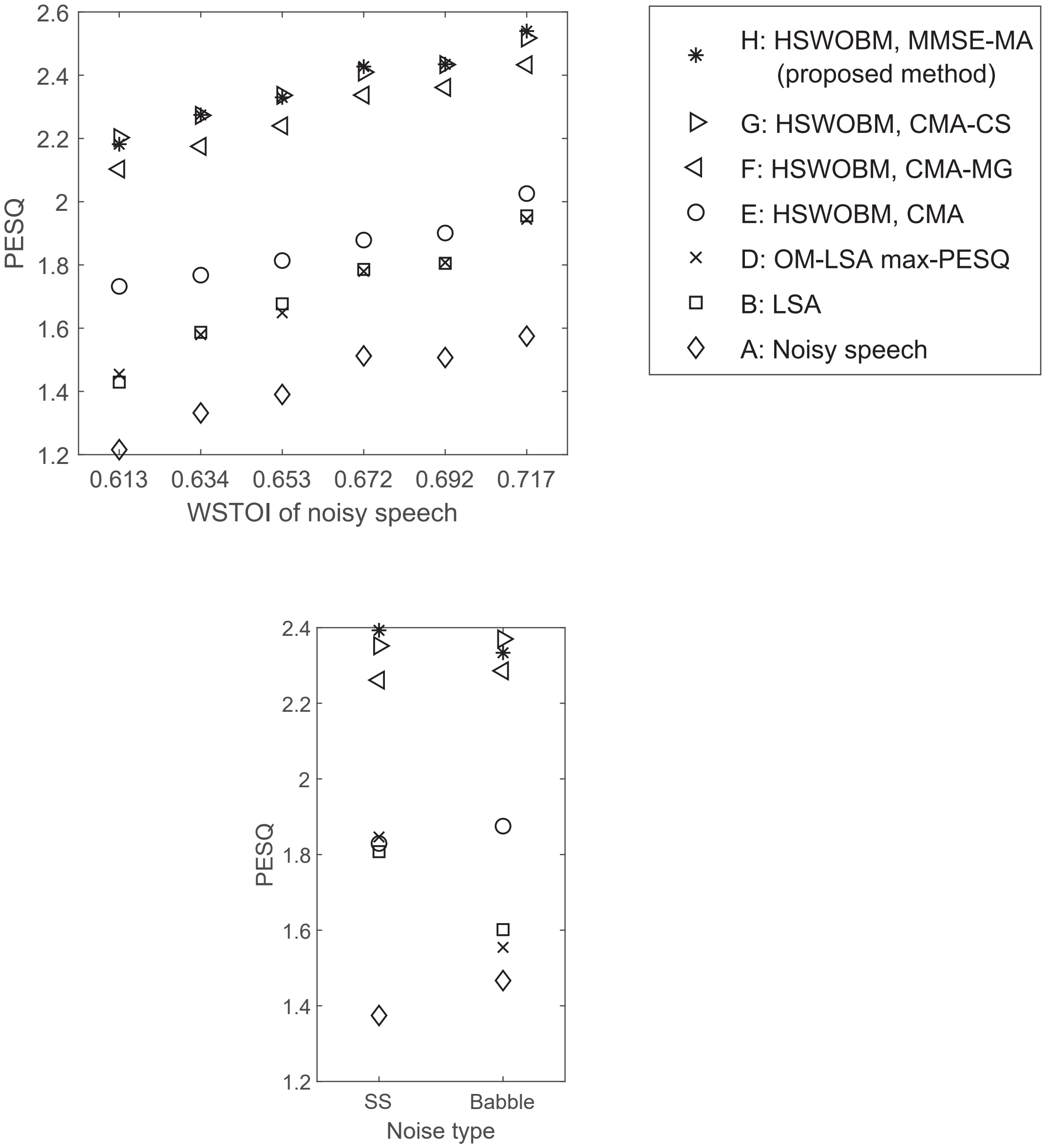


Figure 6.5: PESQ scores of the noisy speech and the noisy speech after processing with different methods. PESQ against (upper) the WSTOI score of the unprocessed noisy speech, and (lower) the noise type.

two algorithms gave no improvement in WSTOI and both of the algorithms damaged the WSTOI scores of some utterances as can be seen from Fig. 6.4b.

The oracle IBM with CMA (E) improved the WSTOI score of the majority of the utterances, with the processed utterances having a median predicted intelligibility of 96%, as indicated by the scale on the rightmost edge of Fig. 6.4a. However, the WSTOI scores of several utterances were severely reduced, as shown by the outliers. In these utterances, the oracle IBM was very sparse, which resulted in the CMA deleting large portions of the speech. By contrast, the oracle HSWOBM with CMA (I) resulted in near full predicted intelligibility for every utterance as can be see from the vertical scale on the right side of Fig. 6.4a. In terms of PESQ, shown in Fig 6.4(c-d), the two approaches were more similarly matched in performance; although the oracle IBM with CMA (E) and the oracle HSWOBM with CMA (I) resulted in an improvement in PESQ for most utterances compared with the noisy speech, the resulting PESQ scores were still quite poor, and both approaches damaged the PESQ scores of some utterances.

With both the oracle IBM (E-H) and the oracle HSWOBM (I-L), the three alternative methods of applying the mask (CMA-MG, CMA-CS and MMSE-MA) all resulted in a much larger ΔPESQ than CMA, whilst achieving almost as high a predicted intelligibility. With the oracle IBM (E-H), the utterances whose intelligibility was severely damaged by CMA (E) did not have their intelligibility substantially damaged by any of the three alternative methods of applying the mask (F-H). With the oracle HSWOBM, MMSE-MA (L) and CMA-CS (K) performed best and approximately equally in terms of PESQ, with CMA-MG (J) closely behind. As we observed in Sec. 6.5, applying a minimum gain is a special case of cepstral smoothing where the smoothing parameters are set to zero. Since the optimal minimum gain parameter, U, for the CMA-MG and CMA-CS methods was almost the same (0.06 and 0.05 respectively), and the difference in PESQ between CMA-CS (K) and CMA-MG (J) was much smaller than the difference in PESQ between CMA-CS (K) and CMA (I), most of the improvement in PESQ given by CMA-CS over CMA must have been due to the use of the minimum gain parameter, U. With the oracle IBM, the three alternative methods of applying the mask (F-H) performed approximately equally in terms of PESQ. The oracle IBM with CMA-MG (F) and CMA-CS (G) gave approximately the same results because the optimal minimum gain, U, was identical for both methods, and very little smoothing was applied in CMA-CS (the optimal $\beta_{\text{pitch}} = 0.11$ and $\beta_{\text{peak}} = 0$), meaning that, with the optimal parameters, the two methods were almost equivalent. These two methods gave approximately the same results as MMSE-MA (H) because the optimal $\phi^1 = 0$ and $\phi^0 = 0$, so that $p(k, m) = 0$ for all (k, m) and $G(k, m) = G_{min}$, where G_{min} is given by (6.13). Applying one of two gains according to the estimated mask in this way is equivalent to applying the mask with a minimum gain. With these optimal parameters, MMSE-MA is therefore equivalent to CMA-MG, and all three alternative methods of applying the mask are almost equivalent. We emphasise that methods E-L all make use of a binary mask which was computed using oracle knowledge of the SNR in each time-frequency bin.

Fig. 6.5a plots the mean PESQ scores of the noisy speech utterances after processing with different enhancement methods against the mean WSTOI scores of the unprocessed noisy speech, in the case of the oracle HSWOBM. The improvement in PESQ resulting from applying the HSWOBM with MMSE-MA was largely independent of the WSTOI of the unprocessed noisy speech. Fig. 6.5b shows the PESQ scores of the different methods with the two noise types plotted separately. MMSE-MA performed best in SS noise and CMA-CS performed best in babble noise.

6.8 MMSE mask application with estimated masks

In order to combine the proposed approach to mask application with the proposed mask estimator from Chapter 5, a modification to the MMSE-MA approach was made to account for the fact that the estimated mask, $B(k,m)$, is continuous-valued. Equations (6.12) and (6.13) were replaced with

$$\hat{\rho}(k,\, m) = \phi^0 + \left(\phi^1 - \phi^0\right) B(k,m)$$

and

$$G_{min} = G^0 + \left(G^1 - G^0\right) B(k,m)$$

so that these quantities vary linearly with the value of the estimated mask. An additional modification was made to suppress musical noise during periods where no speech was detected. Instead of applying $G(k,m)$ we apply a gain $G'(k,m)$ where

$$G'(k,m) = \begin{cases} \Omega & B(k,m) < \Gamma \\ G(k,m) & B(k,m) \geq \Gamma \end{cases}.$$

A final modification was made to allow the quantities (G^1, G^0, ϕ^1, ϕ^0, Ω, Γ) to vary linearly with ERB frequency. Each of these 6 parameters was replaced with 2 parameters: one parameter which determined the value of the quantity at 0 Hz, and another parameter which determined the rate of change of the quantity with ERB frequency. These quantities are now denoted as $(G^1(k),\, G^0(k),\, \phi^1(k),\, \phi^0(k),\, \Omega(k),\, \Gamma(k))$ for $k = 0,\, \ldots,\, K/2$.

We first compute the quantities $(G^1(k),\, G^0(k),\, \Omega(k),\, \Gamma(k))$ and the intermediate quantities $(\hat{\phi}^1(k),\, \hat{\phi}^0(k))$. Each of these, denoted $\beta(k)$, are computed as

$$\beta(k) = \beta_0 \left(1 + \frac{(\Delta_\beta - 1)\, \mathit{\Phi}(\frac{kf_s}{K})}{\mathit{\Phi}(\frac{f_s}{2})} \right) \quad \text{for}\, k = 0,\, \ldots,\, K/2. \tag{6.14}$$

$\beta(k)$ varies linearly with ERB frequency between some value β_0 (at 0 Hz) and $\Delta_\beta \cdot \beta_0$ (at $0.5 f_s$) where Δ_β is a parameter which determines how $\beta(k)$ changes with ERB frequency, and f_s is the sample frequency. The mapping, $\mathit{\Phi}(f)$, between Hz and ERBs can be approximated as (4.10) from Chapter 4. In the next step, $\hat{\phi}^1(k)$ and $\hat{\phi}^0(k)$ are clipped to produce $\phi^1(k)$ and $\phi^0(k)$, so that $0 \leq \phi^n(k) \leq 1$ for $n = 1,\, 2$,

$$\phi^n(k) = \min\left(\max\left(\hat{\phi}^{\mathrm{n}}(\mathrm{k}),\, 1\right),\, 0\right) \ \text{for}\, n = 1,\, 2,\;\; k = 0,\, \ldots,\, K/2,$$

to produce the final set of quantities, $(G^1(k),\, G^0(k),\, \phi^1(k),\, \phi^0(k),\, \Omega(k),\, \Gamma(k))$.

6.8.1 Experimental Procedure

The proposed mask application method (MMSE-MA) was combined with the best performing of the proposed mask estimators from Chapter 5, i.e. the Direct Estimator based on the DNN with $Q = 2$ and the HSWOBM as the target mask. The Cg-DNN algorithm from [21], which estimates the IRM, was included in the experiments for comparison. The mask estimation algorithms were trained on the Training dataset from Chapter 5, with hyperparameters optimised on the Validation dataset. The free parameters of the mask

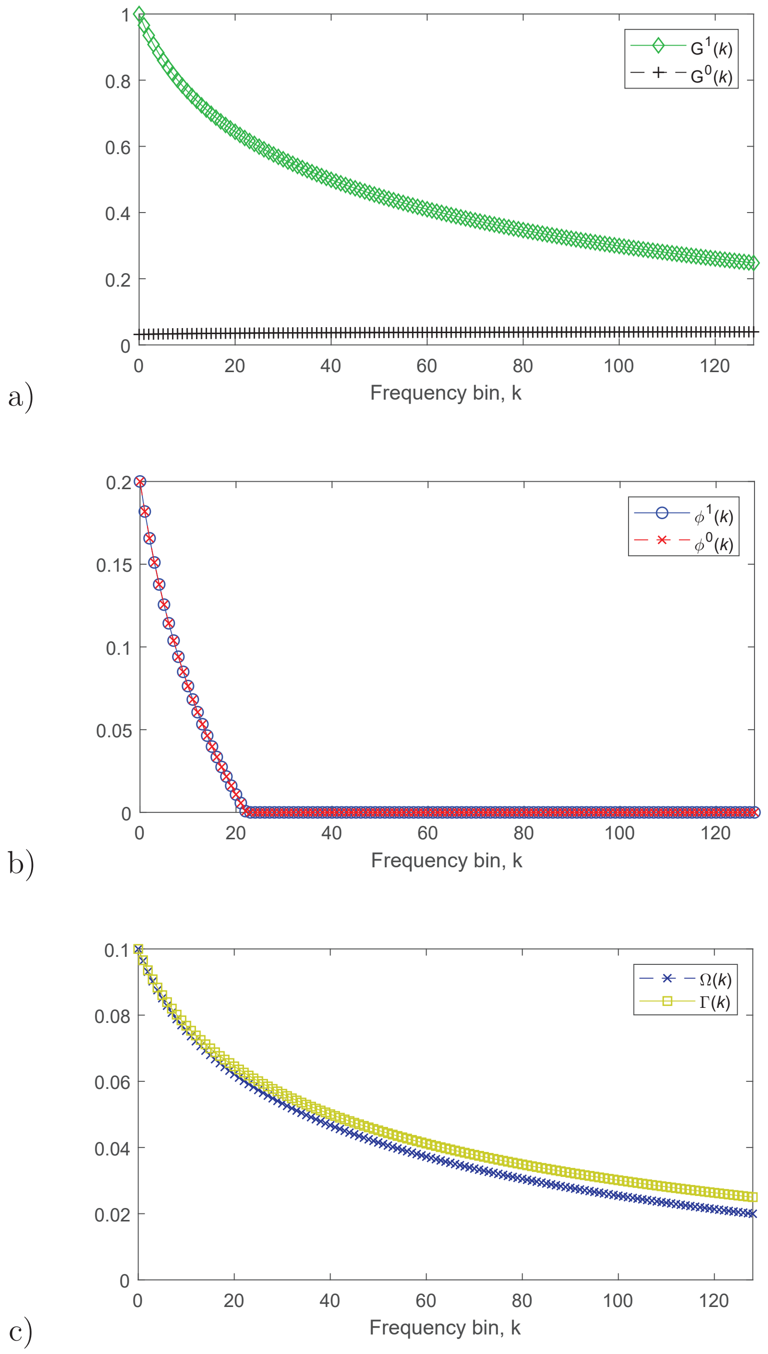


Figure 6.6: The quantities $(G^1(k), G^0(k), \phi^1(k), \phi^0(k), \Omega(k), \Gamma(k))$, produced by the optimal set of parameters, $(G_0^1, G_0^0, \phi_0^1, \phi_0^0, \Omega_0, \Gamma_0, \Delta_{G^1}, \Delta_{G^0}, \Delta_{\phi^1}, \Delta_{\phi^0}, \Delta_\Omega, \Delta_\Gamma)$, for the MMSE-MA algorithm, plotted against k.

Quantity	β_0	Δ_β
$G^1(k)$	1	0.25
$G^0(k)$	0.03	1.25
$\phi^1(k)$	0.2	-1
$\phi^0(k)$	0.2	-1
$\Omega(k)$	0.1	0.2
$\Gamma(k)$	0.1	0.25

Table 6.2: Summary of optimal parameters of the MMSE-MA algorithm, obtained using the Validation dataset.

application methods were optimised using a grid search on a subset of 48 utterances from the Validation dataset, selected semi-randomly to give an equal number of utterances with each noise type and noise level. The optimal values of the free parameters for the MMSE-MA algorithm are listed in Table 6.2 and the corresponding optimal values of the quantities ($G^1(k)$, $G^0(k)$, $\phi^1(k)$, $\phi^0(k)$, $\Omega(k)$, $\Gamma(k)$) are plotted in Fig. 6.6.

6.8.2 Results

Fig. 6.7 and 6.8 show the values of $\Delta\mathrm{SNR}_{\mathrm{WSTOI}}$ and ΔPESQ respectively obtained by applying the proposed DNN-based Direct Estimator (with $Q = 2$ and the HSWOBM as the target mask) combined with different approaches for applying the estimated mask. The results for the two very challenging noise types (Volvo and machine gun) are plotted separately in Fig. 6.9. Excluding these noises, all of the algorithms resulted in an improvement in predicted intelligibility compared with the noisy speech. The proposed estimator

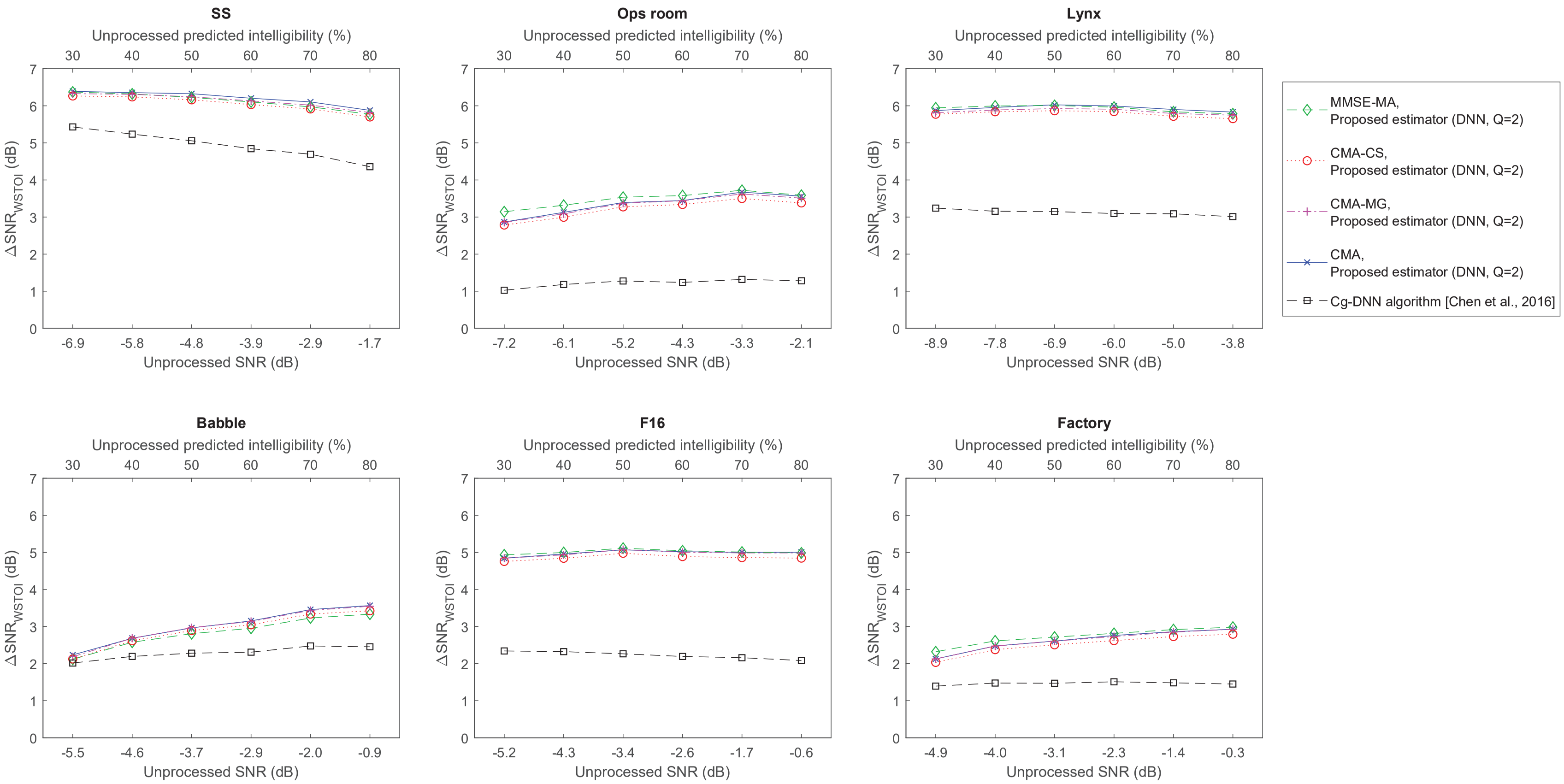


Figure 6.7: $\Delta\mathrm{SNR}_{\mathrm{WSTOI}}$ against the SNR of the unprocessed noisy speech, for the proposed DNN-based Direct Estimator (with $Q = 2$ and the HSWOBM as the target mask) combined with different approaches for applying the estimated mask. Results are also plotted for the Cg-DNN algorithm from [21]. The algorithms were trained on the Training and Validation datasets, and the results were computed on the Test-8N dataset.

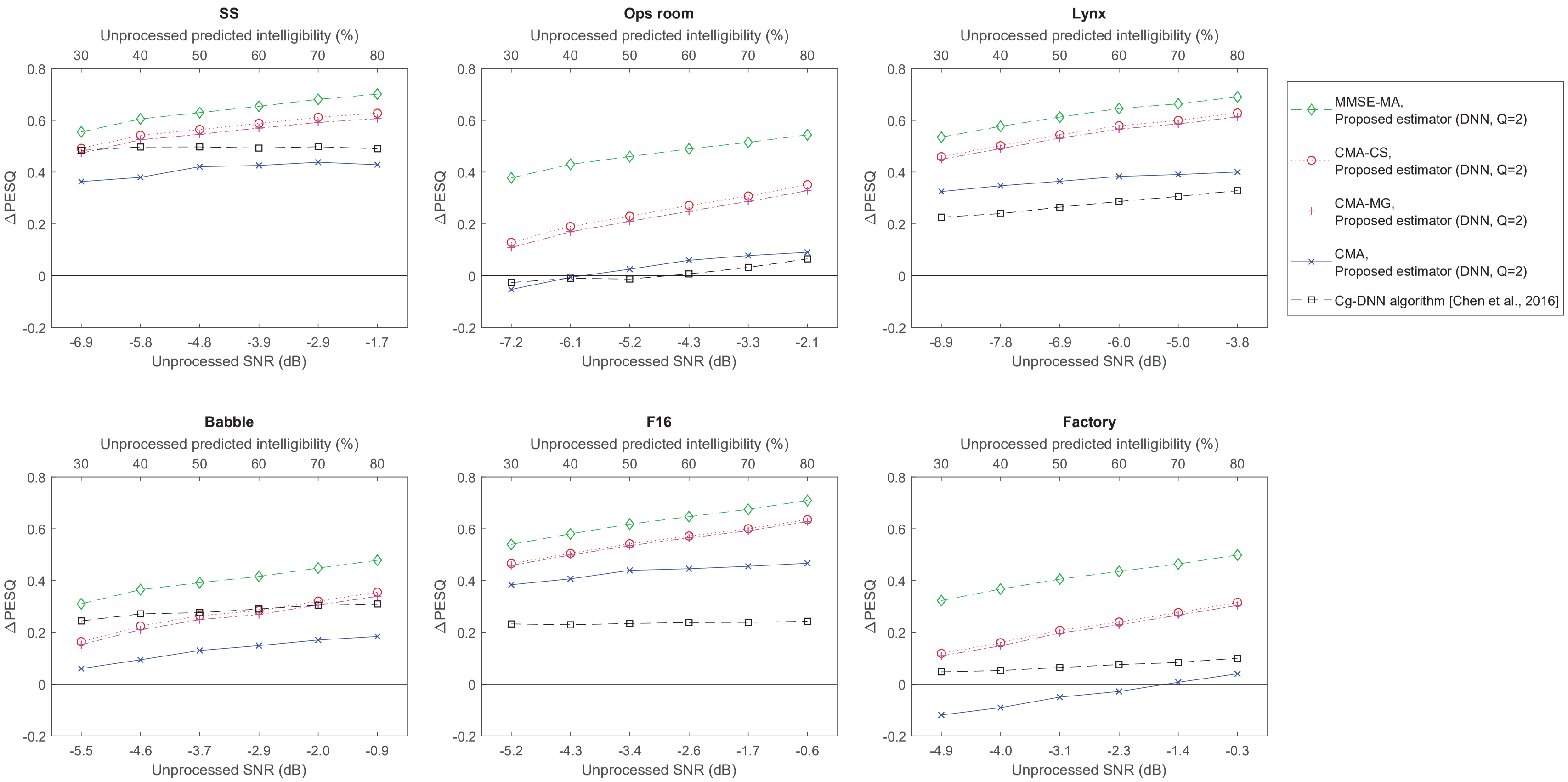


Figure 6.8: ΔPESQ against the SNR of the unprocessed noisy speech, for the proposed DNN-based Direct Estimator (with $Q = 2$ and the HSWOBM as the target mask) combined with different approaches for applying the estimated mask. Results are also plotted for the Cg-DNN algorithm from [21]. The algorithms were trained on the Training and Validation datasets, and the results were computed on the Test-8N dataset.

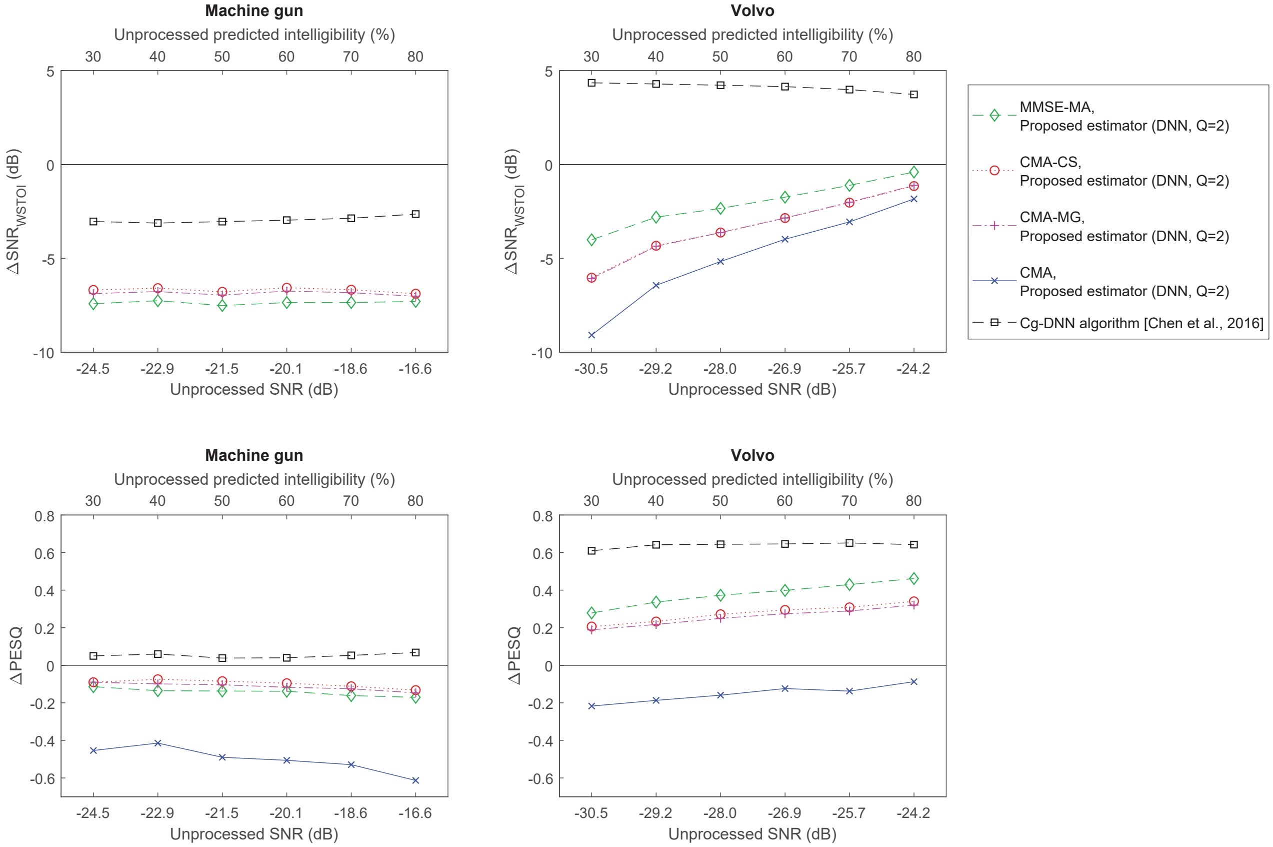


Figure 6.9: $\Delta\mathrm{SNR}_{\mathrm{WSTOI}}$ and ΔPESQ against the SNR of the unprocessed noisy speech, for the proposed DNN-based Direct Estimator (with $Q = 2$ and the HSWOBM as the target mask) combined with different approaches for applying the estimated mask. Results are also plotted for the Cg-DNN algorithm from [21]. The algorithms were trained on the Training and Validation datasets, and the results were computed on the Test-8N dataset.

gave a larger $\Delta\mathrm{SNR}_{\mathrm{WSTOI}}$ than the Cg-DNN algorithm from [21] in each of the 6 noise types. The methods of applying the estimated mask (MMSE-MA, CMA-CS, CMA-MG and CMA) all resulted in very similar values of $\Delta\mathrm{SNR}_{\mathrm{WSTOI}}$. However, the proposed estimator with MMSE-MA gave the largest ΔPESQ for all of the 6 noise types. As with the oracle masks, CMA-MG and CMA-CS produced very similar results.

With machine gun noise, shown in the left column of Fig. 6.9, all of the algorithms resulted in a decrease in WSTOI, with the Cg-DNN algorithm being the least harmful. All algorithms apart from the Cg-DNN algorithm resulted in a reduction in PESQ. With the Volvo noise, shown in the right column of Fig. 6.9, all of the algorithms resulted in a decrease in WSTOI apart from the IRM, which increased WSTOI. All but one algorithm (proposed estimator with CMA) increased PESQ, with the Cg-DNN algorithm resulting in the largest increase.

Fig. 6.10 compares the MMSE-MA algorithm with the CMA and CMA-MG methods of applying the mask from the DNN-based Direct Estimator (with $Q = 2$ and the HSWOBM as the target). Plots (a) and (b) respectively show spectrograms of the clean speech and the noisy speech with babble noise at -4.1 dB SNR. Plots (d) and (e) show the results of applying the mask with the CMA and CMA-MG methods; plot (f) shows the gain of the MMSE-MA method and plot (g) the spectrogram of the resultant enhanced speech. Recall from Chapter 2 that PESQ is a linear combination of the average disturbance value, D_{av}, and the average asymmetrical disturbance value, DA_{av}, where DA_{av} measures only degradations that result in an increase in signal energy, whilst D_{av} measures both degradations that result in an increase and those that result in a decrease in signal energy. Table 6.3 shows the value of different metrics for the signals shown in the spectrograms.

Applying the mask with CMA-MG (Fig. 6.10e) resulted in a higher PESQ than applying the mask with CMA (d), and approximately the same WSTOI score, which is consistent with the results from Fig. 6.7 and Fig. 6.8. The higher PESQ with CMA-MG compared with CMA is due to a lower value of D_{av}, and occurs despite the fact that DA_{av} is higher with this method. Since DA_{av} measures distortions which increase spectral energy, it is therefore not surprising that it is higher with CMA-MG, since this method applies a higher gain in regions where there is no detected speech and which are therefore likely to contain only noise. The combination of a lower D_{av} and a higher DA_{av} indicates a reduction in distortions which decrease spectral energy. This is also not surprising, since imposing a minimum gain will limit the attentuation of speech components which were not detected by the mask estimation algorithm. In summary, PESQ may have determined that the increased noise due to use of a minimum gain is less detrimental to quality than the deletion of undetected speech components that occurs when there is no minimum gain. Using the proposed MMSE-MA algorithm (g) resulted in a higher PESQ than the CMA and CMA-MG algorithms, and approximately the same WSTOI score, which is consistent with the results from Fig. 6.7 and 6.8. MMSE-MA (g) has a similar DA_{av} to CMA, but a lower D_{av} than both CMA and CMA-MG. The lower PESQ score than these methods must therefore be mostly due to having less distortions that decrease signal energy. This could occur, for example, when the estimated mask has a low value (i.e. the mask estimator predicts a low probability of speech presence) but the MMSE-MA algorithm, due to the readings from its noise estimator, determines nonetheless that speech is present and applies a high gain.

Signal	D_{av}	DA_{av}	PESQ	WSTOI	$\Delta\text{SNR}_{\text{WSTOI}}$
Speech + babble -4.1 dB SNR $Y(k,\, m)$	23.0	44.0	0.84	0.64	-
CMA (Estimated HSWOBM) $Y(k,\, m)B(k,\, m)$	21.2	42.8	1.06	0.68	1.8 dB
CMA-MG (Estimated HSWOBM) $Y(k,\, m) \cdot (\max\{B(k,\, m),\, U\})$	19.4	43.4	1.22	0.68	1.8 dB
MMSE-MA (Estimated HSWOBM) $Y(k,\, m)G(k,\, m)$	17.5	42.7	1.43	0.68	1.8 dB

Table 6.3: The average disturbance values, D_{av}, average asymmetrical disturbance values, DA_{av}, PESQ scores, WSTOI scores and $\Delta\text{SNR}_{\text{WSTOI}}$ scores for the signals shown in the spectrograms in Fig. 6.10. PESQ $= 4.5 - 0.1 \cdot D_{av} - 0.0309 \cdot DA_{av}$.

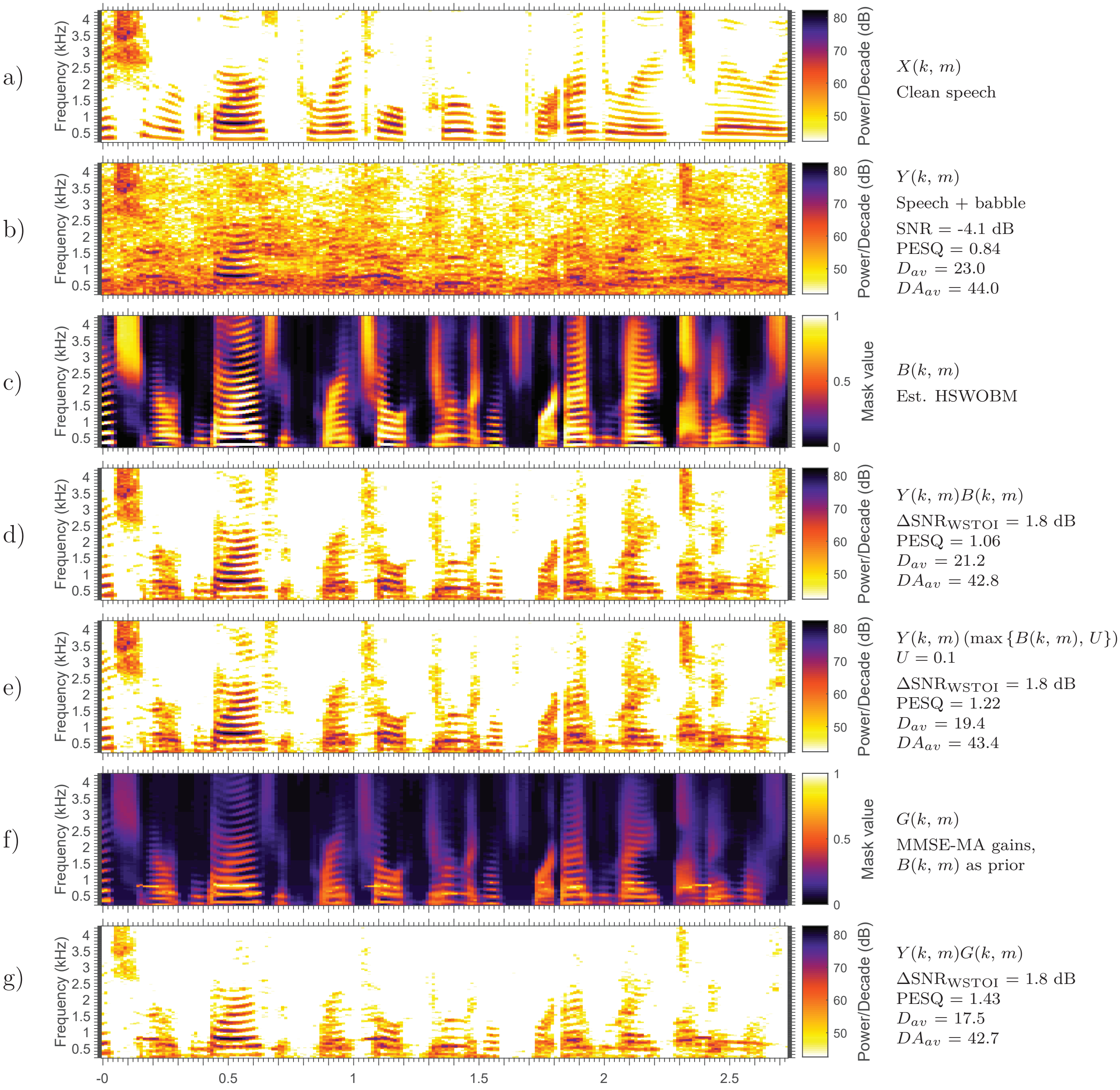


Figure 6.10: Spectrograms of a) part of an utterance of the phrase “tradition requires parental approval for underage marriage”, b) the utterance after adding babble noise with an SNR of -4.1 dB, c) the estimated HSWOBM for stochastic white Gaussian noise with -5 dB SNR, estimated using the DNN-based mask estimator with $Q = 2$, d) the noisy speech spectra after applying the estimated mask using CMA, e) the noisy speech spectra after applying the estimated mask using CMA-MG with $U = 0.1$, f) the gains produced by the MMSE-MA algorithm using the estimated HSWOBM as the prior information, g) the noisy speech spectra after applying the MMSE-MA gains.

6.9 No algorithmic delay

The proposed approach to mask estimation, outlined in Chapter 5, uses a context windows of $2V+1$ feature frames and applies smoothing to the estimated mask with a window of $2Q+1$ frames, with both windows centred on the frame being estimated. Since each frame occupies 25.6 ms and there is a 12.8 ms overlap between frames, this introduces an algorithmic delay equal to $12.8 \times \max(Q, V)$ ms, in addition to the 25.6 ms delay of the STFT. For real-time applications such as hearing aids, telephony and video conferencing it is not viable to have an algorithmic delay of multiple frames. We therefore propose here a modified algorithm with no additional algorithmic delay beyond the 1-frame delay of the STFT. The context window of $V = 12$ frames is now positioned so that it looks only into past frames, and the estimated mask is not smoothed, i.e. $Q = 0$.

Fig. 6.11 and Fig. 6.12 respectively show the $\Delta\mathrm{SNR}_{\mathrm{WSTOI}}$ and ΔPESQ obtained when the modified mask estimation algorithm with no algorithmic delay was combined with the proposed mask application method, MMSE-MA. The algorithm chosen for modification was the best performing of the proposed mask estimators from Chapter 5 that had $Q = 0$, i.e. the Direct Estimator based on the DNN with the HSWOBM as the target mask. This is compared with the best-performing of the proposed mask estimators from Chapter 5, which was the DNN-based Direct Estimator with $Q = 2$ and the HSWOBM as the target mask. The algorithms were trained on the Training and Validation datasets, and the results were computed on the Test-8N dataset.

The enhancer which had no additional algorithmic delay performed worse, in terms of both WSTOI and PESQ, than the enhancer which had algorithmic delay. However, the enhancer with no delay still improved WSTOI in 6 of 8 noise types and PESQ in 7 of 8 noise types.

6.10 Summary

This chapter has presented an alternative approach to applying a binary mask, Minimum Mean Squared Error Mask Application (MMSE-MA), that preserves the intelligibility gains given by conventional binary masking but also incorporates a speech enhancer's ability to improve perceptual quality. In the proposed method the mask is used to supply prior information about the probability of speech presence to a classical speech enhancer that minimises the expected squared error in the LSAs. When MMSE-MA was tested with the masks produced by a DNN-based mask estimator that had been trained with HSWOBM target masks, it resulted in a larger improvement in PESQ than other methods of applying the masks, whilst preserving the improvements in predicted intelligibility given by these methods. The proposed end-to-end enhancer (the DNN-based HSWOBM estimator with MMSE-MA) outperformed the Cg-DNN algorithm from [21] in terms of both WSTOI and PESQ in 6 of 8 tested noise types. When this enhancer was modified to have no algorithmic delay, it still improved WSTOI in 6 of 8 noise types and PESQ in 7 of 8 noise types.

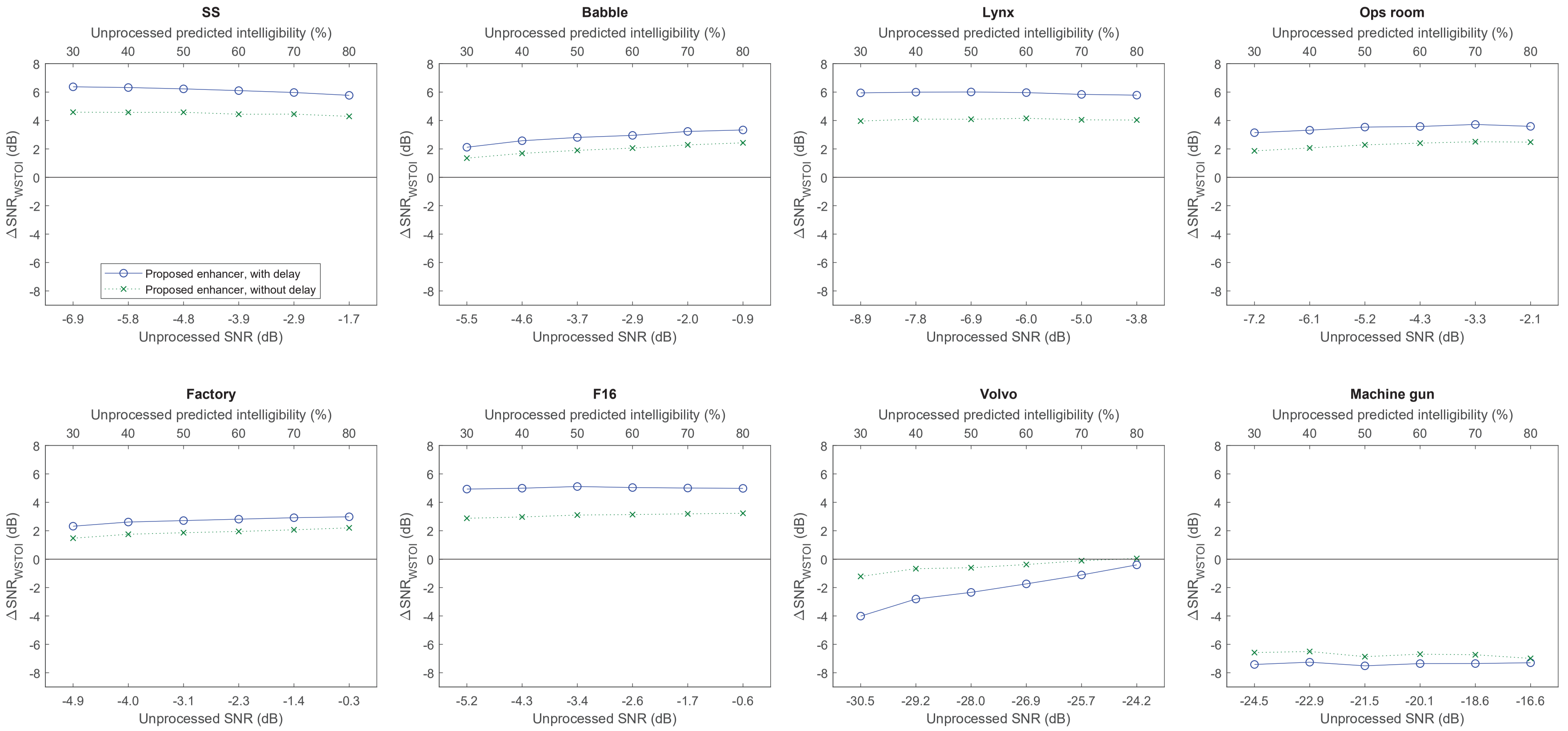


Figure 6.11: $\Delta\text{SNR}_{\text{WSTOI}}$ against the SNR of the unprocessed noisy speech, for the proposed enhancer and the modified version of the enhancer with no algorithmic delay. The proposed enhancer (with algorithmic delay) is the DNN-based Direct Estimator with MMSE-MA, $Q = 2$ and a feature context window that contains past and future frames. The modified enhancer without an algorithmic delay is the DNN-based Direct Estimator with MMSE-MA, $Q = 0$ and a feature context window that contains only past and current frames. Both algorithms had $V = 12$ and the HSWOBM as the target mask. The algorithms were trained on the Training and Validation datasets, and the results were computed on the Test-8N dataset.

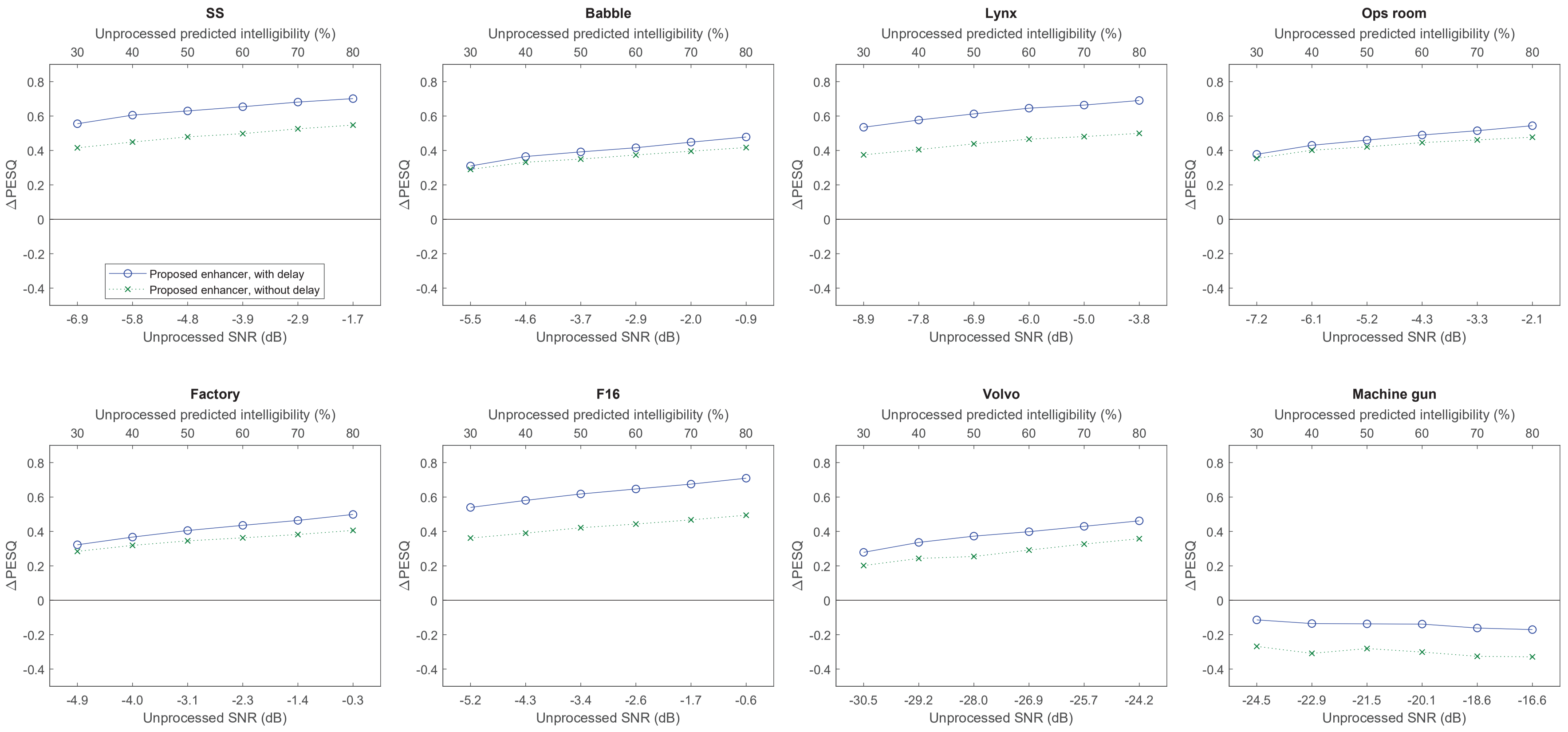


Figure 6.12: ΔPESQ against the SNR of the unprocessed noisy speech, for the proposed enhancer and the modified version of the enhancer with no algorithmic delay. The proposed enhancer with an algorithmic delay is the DNN-based Direct Estimator with MMSE-MA, $Q = 2$ and a feature context window that contains past and future frames. The proposed enhancer without an algorithmic delay is the DNN-based Direct Estimator with MMSE-MA, $Q = 0$ and a feature context window that contains only past frames. Both algorithms had $V = 12$ and the HSWOBM as the target mask. The algorithms were trained on the Training and Validation datasets, and the results were computed on the Test-8N dataset.

Chapter 7

Conclusions

7.1 Thesis summary

Although conventional speech enhancement algorithms can improve perceptual quality, they typically have either very little effect, or a detrimental effect on intelligibility [13, 157, 7, 76, 111, 113]. The motivation for the work described in this thesis was the finding in numerous studies [5, 110, 17, 167, 98] that the intelligibility of noisy speech can be improved dramatically by applying a binary-valued time-frequency mask to the signal. The thesis has addressed the following aspects of the use of binary masks for speech enhancement namely

(a) what is the "best" binary mask to use?

(b) how should the mask be estimated from noisy speech?

(c) how should the mask best be used to enhance the speech?

7.1.1 The WSTOI intelligibility metric

In order to provide a measure of a binary mask's effectiveness, Chapter 3 presented the Weighted-STOI (WSTOI) intelligibility metric. WSTOI is a modified version of STOI in which the contribution of each time frame to the metric is weighted by its estimated contribution to intelligibility. This estimated contribution is given by the mutual information between two versions of a hypothetical signal at either end of a simplified model of human communication. The modification improves STOI by better accounting for the variation in information content of a speech signal in time and frequency. An advantage of WSTOI is that, since "silent" frames contain little or no information, they are automatically downweighted and it is no longer necessary to detect and delete these frames explicitly as in the STOI metric. The result is a more physiologically motivated way of handling silences which, unlike STOI, does not require a hard decision on whether a frame is active or silent. Evaluation showed that the modification improved the prediction accuracy of STOI at all performance levels on both long and short utterances. An improvement was observed across all tested noise types and suppression algorithms.

7.1.2 STOI-optimal binary masks

Existing oracle masks, such as the IBM, TBM, UTBM and IRM have been shown to be capable of improving the intelligibility of noisy speech. However, there is evidence that the intelligibility of speech depends not only on the instantaneous spectrum but also on its temporal modulation [8, 32]. The intelligibility of the mask-processed speech will not therefore be maximised if the classifier training target uses a mask such as the IBM, TBM, UTBM or IRM since these depend only on the instantaneous power spectra of the speech and noise. In Chapter 4 new oracle binary masks are presented, the STOI-Optimal Binary Mask (SOBM) and the WSTOI-Optimal Binary Mask (WOBM), that explicitly maximise intelligibility metrics, STOI and WSTOI, which take account of spectral modulation. The SOBM is derived for two cases: for a deterministic noise signal (DSOBM) and for stochastic noise with a known power spectrum (SSOBM). For deterministic additive noise, the DSOBM always results in a higher STOI value than other oracle masks. By assuming a stochastic noise signal, the SSOBM achieves a performance close to the DSOBM for a wide range of SNRs and noise types, even when the noises used for mask optimisation and testing are mismatched. A further motivation for a stochastic mask was the suggestion in [98] that a mask estimation algorithm is likely to generalise better to new noise conditions if it is trained with a target mask that is independent of the noise, since the estimation algorithm is then more likely to focus on modelling features present in the speech rather than the noise. This may lead to better generalisation since the statistics of noise encountered in a real environment may differ significantly from those in the training set, whereas the features in the speech are likely to be more consistent between the training and testing data sets. Analogously to the SSOBM, the SWOBM optimises the WSTOI intelligibility metric for stochastic noise signals. An extension to the SWOBM is the HSWOBM which has an increased frequency resolution and results in speech with a higher predicted quality. The SHSWOBM is a smoothed version of the HSWOBM in which the pitch information, which is difficult for a mask estimation algorithm to reliably estimate from noisy speech, has been largely removed. The CHSWOBM is a modified version of the SHSWOBM in which the information that is important for speech intelligibility is compressed into a more compact form. These two modified versions of the HSWOBM largely preserve its intelligibility benefits but, in oracle form, result in significantly lower quality. However, it was anticipated that they might be easier to estimate from noisy speech.

7.1.3 Optimal mask estimation

This chapter presented a number of procedures for estimating a binary mask from noisy speech. We first outlined a feature set to use as the input to the estimation algorithm. The feature set is based on the TF gains estimated by a classical speech enhancement algorithm, and an estimate of the local VSNNR in different TF regions, obtained using a pitch estimator. We observed that a DNN-based estimator outperformed estimators based on an LSTM or a GMM. We then saw that estimators trained on each of the three proposed target masks were all able to increase the WSTOI of noisy speech. Of these estimators, the Direct Estimator (which did not use a library of mask patterns) trained using the HSWOBM as the target provided the greatest improvement in WSTOI. We also observed that the proposed feature set matched or outperformed the cochleagram feature set from [21] in 7 of the 8 tested noise types, including 5 of the 6 noise types that were not seen by the algorithm during training. Finally, we observed that the proposed mask estimation algorithm outperformed the Cg-DNN mask estimator from [21] in the majority of the tested noise types.

7.1.4 Optimal mask application

The conventional way to apply a binary mask to noisy speech is to multiply speech by the mask in the STFT-domain and then convert the resulting signal into the time-domain. However, although applying a binary mask in this manner can improve the intelligibility of noisy speech, the resulting speech often has very poor perceptual quality. This chapter presented an alternative approach to applying a binary mask, Minimum Mean Squared Error Mask Application (MMSE-MA), that preserves the intelligibility gains given by conventional binary masking but also incorporates a speech enhancer's ability to improve perceptual quality. In the proposed method the mask is used to supply prior information about the Speech Presence Probability (SPP) to a classical speech enhancer that minimises the expected squared error in the LSAs. When MMSE-MA was tested with the masks produced by a DNN-based mask estimator that had been trained with HSWOBM target masks, it resulted in a larger improvement in PESQ than other methods of applying the masks, whilst preserving the improvements in predicted intelligibility given by these methods. The proposed end-to-end enhancer (the DNN-based HSWOBM estimator with MMSE-MA) outperformed the Cg-DNN algorithm from [21] in terms of both WSTOI and PESQ in 6 of 8 tested noise types. When this enhancer was modified to have no algorithmic delay, it still improved WSTOI in 6 of 8 noise types and PESQ in 7 of 8 noise types.

7.2 Future work

This section identifies a number of ways in which the work described in this thesis could be taken further.

7.2.1 Intelligibility metrics

The WSTOI and STOI metrics estimate intelligibility from the temporal correlation between the spectral envelopes of clean and degraded speech in windows of length 384 ms. However, it has been observed that the STOI metric underestimates the intelligibility of speech that has been corrupted by fluctuating of intermittent noise and it has been suggested that this is because listeners are able to "glimpse" the target speech in short intervals when the noise is low [163]. It is possible that the accuracy of WSTOI could be improved when fluctuating noise is present by modifying it to take account of this phenomenon using a similar approach to that proposed in [87].

7.2.2 Oracle WSTOI-optimal masks

The primary reason that HSWOBM, outlined in 4, was chosen to be binary-valued was in order that its computation be tractable. A continuous-valued version of the oracle HSWOBM would result in either greater or approximately equal WSTOI scores, and may serve as a better target for a mask estimation algorithm. One way to obtain this continuous-valued mask might be a gradient-descent based algorithm, initialised with the HSWOBM.

7.2.3 Estimation algorithms

The mask estimation approach outlined in Chapter 5 uses a loss function which minimises the mean-squared error between the the estimated mask and an explicit target mask, weighted by a measure of the sensitivity of WSTOI to each mask value. Rather than defining an explicit mask target, an alternative approach to mask estimation could involve using a loss function based on the WSTOI and/or PESQ scores, of the enhanced speech at the output of the mask application module. This approach might involve jointly optimising the parameters of the mask estimation and mask application modules. In order to decouple the estimated masks from the noise in the training data, the loss function of this alternative approach could be based on the expected value of WSTOI in the case of a stochastic noise signal, which would be conceptually similar to using the HSWOBM instead of the equivalent deterministic mask, as we discussed in Chapter 5. As with the approaches in [174, 40], this would not require a target mask to be explicitly defined. This approach has several potential advantages over the approach presented in Chapters 5 and 6.1. Jointly optimising the parameters of the mask estimation and mask application modules has the potential to find a more optimal solution than optimising them independently, as in the method outlined in Chapter 6. Also, the target mask used in Chapter 6 (the HSWOBM) is WSTOI-optimal when applied using Conventional Mask Application (CMA), but not when applied using Minimum Mean Squared Error Mask Application (MMSE-MA). The alternative approach may therefore result in a higher value of WSTOI. Compared with the approach outlined in Chapter 5, it also avoids the loss of information that occurs by constraining the oracle mask to be binary. Finally, the alternative approach would remove the need for the WSTOI-sensitivity weighting in the loss function, which introduces inaccuracy by assuming that errors in the HSWOBM occur in isolation from one another. A possible drawback of this approach is the increased number of computations that would be required to optimise WSTOI and PESQ directly compared to using the weighted mean squared error loss function.

7.2.4 Mask application

The approach to mask application outlined in Chapter 6 involves using the mask to estimate a speech presence probability, and incorporating this as prior information into a classical speech enhancer. Within the enhancer, the speech presence probability is used to calculate a prior distribution for the speech spectral amplitudes. An alternative approach would be to avoid calculating the speech presence probability explicitly, but instead to estimate directly the prior distribution of the speech spectral amplitude conditional on the mask value and the estimated SNR. This distribution could, for example, be described by a Gaussian Mixture Model (GMM) whose parameters are determined empirically from training data.

Bibliography

[1] G. A. Miller and J. C. R. Licklider. The intelligibility of interrupted speech. *J. Acoust. Soc. Am.*, 22(2):167–173, Mar. 1950.

[2] A. H. Andersen, J. M. de Haan, Z. H. Tan, and J. Jensen. Predicting the intelligibility of noisy and nonlinearly processed binaural speech. *IEEE Trans. Audio, Speech, Lang. Process.*, 24(11):1908–1920, Nov. 2016.

[3] ANSI. Methods for the calculation of the articulation index. ANSI Standard S3.5–1969, American National Standards Institute, 1969.

[4] ANSI. Methods for the calculation of the speech intelligibility index. ANSI Standard S3.5–1997 (R2007), American National Standards Institute, 1997.

[5] M. Anzalone, L. Calandruccio, K. Doherty, and L. Carney. Determination of the potential benefit of time-frequency gain manipulation. *Ear & Hearing*, 27(5):480–492, Oct. 2006.

[6] T. Arai, M. Pavel, H. Hermansky, and C. Avendano. Syllable intelligibility for temporally filtered LPC cepstral trajectories. *J. Acoust. Soc. Am.*, 105(5):2783–2791, May 1999.

[7] K. Arehart, J. Hansen, S. Gallant, and L. Kalstein. Evaluation of an auditory masked threshold noise suppression algorithm in normal-hearing and hearing-impaired listeners. *Speech Communication*, 40(4):572–592, June 2003.

[8] L. Atlas and S. A. Shamma. Joint acoustic and modulation frequency. *EURASIP Journal on Applied Signal Processing*, 7:668–675, June 2003.

[9] R. J. Baken. *Clinical Measurement of Speech and Voice*. Taylor & Francis Ltd., London, UK, 1987.

[10] Y. Bengio, P. Simard, and P. Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE Trans. on Neural Networks*, 5(2):157–166, Mar. 1994.

[11] M. Berouti, R. Schwartz, and J. Makhoul. Enhancement of speech corrupted by acoustic noise. In *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pages 208–211, 1979.

[12] J. Blauert and J. Braasch, editors. *The technology of binaural understanding*. Springer, 2020.

[13] S. F. Boll. Suppression of acoustic noise in speech using spectral subtraction. *IEEE Trans. Acoust., Speech, Signal Process.*, 27(2):113–120, Apr. 1979.

[14] A. S. Bregman. *Auditory Scene Analysis: The Perceptual Organization of Sound*. MIT Press, 1990.

[15] D. M. Brookes. VOICEBOX: A speech processing toolbox for MATLAB. `http://www.ee.ic.ac.uk/hp/staff/dmb/voicebox/voicebox.html`, 1997–2019.

[16] M. Brookes and N. D. Gaubitch. *Image, Video Processing and Analysis, Hardware, Audio, Acoustic and Speech Processing*, chapter 35 Enhancement, pages 1019–1056. Elsevier Ltd Academic Press, 2014.

[17] D. S. Brungart, P. S. Chang, B. D. Simpson, and D. Wang. Isolating the energetic component of speech-on-speech masking with ideal time-frequency segregation. *J. Acoust. Soc. Am.*, 120(6):4007–4018, Jan. 2006.

[18] D. Byrne, H. Dillon, K. Tran, S. Arlinger, K. Wilbraham, R. Cox, B. Hayerman, R. Hetu, J. Kei, C. Lui, J. Kiessling, M. N. Kotby, N. H. A. Nasser, W. A. H. E. Kholy, Y. Nakanishi, H. Oyer, R. Powell, D. Stephens, T. Sirimanna, G. Tavartkiladze, G. I. Frolenkov, S. Westerman, and C. Ludvigsen. An international comparison of long-term average speech spectra. *J. Acoust. Soc. Am.*, 96(4):2108–2120, Oct. 1994.

[19] F. Chen and P. C. Loizou. Contributions of cochlea-scaled entropy and consonant-vowel boundaries to prediction of speech intelligibility in noise. *J. Acoust. Soc. Am.*, 131(5):4104–4113, May 2012.

[20] J. Chen and D. Wang. Long short-term memory for speaker generalization in supervised speech separation. *J. Acoust. Soc. Am.*, 141(6):4705–4714, June 2017.

[21] J. Chen, Y. Wang, S. E. Yoho, D. Wang, and E. W. Healy. Large-scale training to increase speech intelligibility for hearing-impaired listeners in novel noises. *J. Acoust. Soc. Am.*, 139(5):2604–2612, May 2016.

[22] S. F. Chen and J. Goodman. An empirical study of smoothing techniques for language modeling. *Computer Speech and Language*, 13(4):359–393, Oct. 1999.

[23] I. Cohen. Optimal speech enhancement under signal presence uncertainty using log-spectral amplitude estimator. *IEEE Signal Process. Lett.*, 9(4):113–116, Apr. 2002.

[24] R. A. Cole, Y. Yan, B. Mak, M. Fanty, and T. Bailey. The contribution of consonants versus vowels to word recognition in fluent speech. In *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pages 853–856, 1996.

[25] T. M. Cover and J. A. Thomas. *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing)*. Wiley-Interscience, New York, NY, USA, 2006.

[26] S. B. Davis and P. Mermelstein. Comparison of parametric representations for monosyllabic word recognition in continously spoken sentences. *IEEE Trans. Acoust., Speech, Signal Process.*, 28(4):357–366, Aug. 1980.

[27] M. Delfarah and D. Wang. Features for masking-based monaural speech separation in reverberant conditions. *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, 25(5):1085–1094, May 2017.

[28] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal Royal Statistical Society, Series B*, 39(1):1–38, 1977.

[29] L. Deng, J. Droppo, and A. Acero. Estimating cepstrum of speech under the presence of noise using a joint prior of static and dynamic features. *IEEE Trans. Speech Audio Process.*, 12(3):218–233, May 2004.

[30] R. Drullman et al. Temporal envelope and fine structure cues for speech intelligibility. *J. Acoust. Soc. Am.*, 97(1):585–592, Jan. 1995.

[31] R. Drullman, J. M. Festen, and R. Plomp. Effect of reducing slow temporal modulations on speech reception. *J. Acoust. Soc. Am.*, 95(5):2670–2680, May 1994.

[32] R. Drullman, J. M. Festen, and R. Plomp. Effect of temporal envelope smearing on speech reception. *J. Acoust. Soc. Am.*, 95(5):1053–1064, June 1994.

[33] F. Dubbelboer and T. Houtgast. A detailed study on the effects of noise on speech intelligibility. *J. Acoust. Soc. Am.*, 122(5):2865–2871, Nov. 2007.

[34] F. Dubbelboer and T. Houtgast. The concept of signal-to-noise ratio in the modulation domain and speech intelligibility. *J. Acoust. Soc. Am.*, 124(6):3937–3946, Dec. 2008.

[35] A. Duquesnoy. Effect of a single interfering noise or speech source on the binaural sentence intelligibility of aged persons. *J. Acoust. Soc. Am.*, 74(3):739–743, Oct. 1983.

[36] Y. Ephraim and D. Malah. Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator. *IEEE Trans. Acoust., Speech, Signal Process.*, 32(6):1109–1121, Dec. 1984.

[37] Y. Ephraim and D. Malah. Speech enhancement using a minimum mean-square error log-spectral amplitude estimator. *IEEE Trans. Acoust., Speech, Signal Process.*, 33(2):443–445, Apr. 1985.

[38] Y. Ephraim, D. Malah, and B.-H. Juang. On the application of hidden Markov models for enhancing noisy speech. *IEEE Trans. Acoust., Speech, Signal Process.*, 37(12):1846–1856, Dec. 1989.

[39] Y. Ephraim and H. L. Van Trees. A signal subspace approach for speech enhancement. *IEEE Trans. Speech Audio Process.*, 3(4):251–266, July 1995.

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

[41] D. Fogerty and D. Kewley-Port. Perceptual contributions of the consonant-vowel boundary to sentence intelligibility. *J. Acoust. Soc. Am.*, 126(2):847–857, Aug. 2009.

[42] N. R. French and J. C. Steinberg. Factors governing the intelligibility of speech sounds. *J. Acoust. Soc. Am.*, 19(1):90–119, 1947.

[43] S. Furui. Speaker-independent isolated word recognition using dynamic features of speech spectrum. *IEEE Trans. Acoust., Speech, Signal Process.*, 34(1):52–59, Feb. 1986.

[44] Y. Gal and Z. Ghahramani. A theoretically grounded application of dropout in recurrent neural networks. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, pages 1027–1035, 2016.

[45] J. S. Garofolo. Getting started with the DARPA TIMIT CD-ROM: An acoustic phonetic continuous speech database. Technical report, National Institute of Standards and Technology (NIST), Gaithersburg, Maryland, Dec. 1988.

[46] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallett, N. L. Dahlgren, and V. Zue. TIMIT acoustic-phonetic continuous speech corpus. Corpus LDC93S1, Linguistic Data Consortium, Philadelphia, 1993.

[47] T. Gerkmann and R. C. Hendriks. Unbiased MMSE-based noise power estimation with low complexity and low tracking delay. *IEEE Trans. Audio, Speech, Lang. Process.*, 20(4):1383–1393, May 2012.

[48] F. A. Gers, J. Schmidhuber, and F. Cummins. Learning to forget: Continual prediction with LSTM. *Neural Computation*, 12(10):2451–2471, Oct. 2000.

[49] J. D. Gibson, B. Koo, and S. D. Gray. Filtering of colored noise for speech enhancement and coding. *IEEE Trans. Signal Process.*, 39(8):1732–1742, Aug. 1991.

[50] Z. Goh, K.-C. Tan, and B. T. G. Tan. Kalman-filtering speech enhancement method based on a voiced-unvoiced speech model. *IEEE Trans. Speech Audio Process.*, 7(5):510–524, Sept. 1999.

[51] R. L. Goldsworthy and J. E. Greenberg. Analysis of speech-based speech transmission index methods with implications for nonlinear operations. *J. Acoust. Soc. Am.*, 116(6):3679–3689, Dec. 2004.

[52] A. M. Gomez, B. Schwerin, and K. Paliwal. Objective intelligibility prediction of speech by combining correlation and distortion based techniques. In *Proc. Interspeech Conf.*, pages 1225–1228, 2011.

[53] S. Gonzalez. *Analysis of Very Low Quality Speech for Mask-Based Enhancement*. PhD thesis, Imperial College London, 2013.

[54] S. Gonzalez and M. Brookes. A pitch estimation filter robust to high levels of noise (PEFAC). In *Proc. European Signal Processing Conf. (EUSIPCO)*, pages 451–455, 2011.

[55] S. Gonzalez and M. Brookes. Sibilant speech detection in noise. In *Proc. Interspeech Conf.*, pages 1488–1491, 2012.

[56] S. Gonzalez and M. Brookes. Mask-based enhancement for very low quality speech. In *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7029–7033, 2014.

[57] S. Gonzalez and M. Brookes. PEFAC - a pitch estimation algorithm robust to high levels of noise. *IEEE Trans. Audio, Speech, Lang. Process.*, 22(2):518–530, Feb. 2014.

[58] A. Graves, A.-r. Mohamed, and G. Hinton. Speech recognition with deep recurrent neural networks. In *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6645–6649, 2013.

[59] K. Greff, R. K. Srivastava, J. Koutník, B. Steunebrink, and J. Schmidhuber. LSTM: A search space odyssey. *IEEE Transactions on Neural Networks and Learning Systems*, 28(10):2222–2232, Oct. 2017.

[60] K. Han and D. Wang. An SVM based classification approach to speech separation. In *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4632–4635, 2011.

[61] K. Han and D. Wang. A classification based approach to speech segregation. *J. Acoust. Soc. Am.*, 132(5):3475–3483, Nov. 2012.

[62] K. Han and D. Wang. Towards generalizing classification based speech separation. *IEEE Trans. Audio, Speech, Lang. Process.*, 21(1):168–177, Jan. 2013.

[63] B. A. Hanson and T. H. Applebaum. Robust speaker-independent word recognition using static, dynamic and acceleration features: experiments with lombard and noisy speech. In *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pages 857–860, 1990.

[64] E. W. Healy, S. E. Yoho, J. Chen, Y. Wang, and D. Wang. An algorithm to increase speech intelligibility for hearing-impaired listeners in novel segments of the same noise type. *J. Acoust. Soc. Am.*, 138(3):1660–1669, Sept. 2015.

[65] E. W. Healy, S. E. Yoho, Y. Wang, and D. Wang. An algorithm to improve speech recognition in noise for hearing-impaired listeners. *J. Acoust. Soc. Am.*, 134(4):3029–3038, Oct. 2013.

[66] H. Hermansky. Perceptual linear predictive (PLP) analysis of speech. *J. Acoust. Soc. Am.*, 87(4):1738–1752, Apr. 1990.

[67] H. Hermansky and N. Morgan. RASTA processing of speech. *IEEE Trans. Speech Audio Process.*, 2(4):578–589, Oct. 1994.

[68] G. Hilkhuysen, N. Gaubitch, M. Brookes, and M. Huckvale. Effects of noise suppression on intelligibility: dependency on signal-to-noise ratios. *J. Acoust. Soc. Am.*, 131(1):531–539, Jan. 2012.

[69] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, and B. Kingsbury. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine*, 29(6):82–97, Nov 2012.

[70] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, Nov. 1997.

[71] V. Hohmann. Frequency analysis and synthesis using a gammatone filterbank. *Acta Acustica united with Acustica*, 88(3):433–442, May 2002.

[72] R. A. Horn and C. R. Johnson, editors. *Matrix Analysis.* Cambridge University Press, New York, NY, USA, second edition, 2013.

[73] P. Howard-Jones and S. Rosen. The perception of speech in fluctuating noise. *Acta Acustica united with Acustica*, 78(5):258–272, Jan. 1993.

[74] P. Howard-Jones and S. Rosen. Uncomodulated glimpsing in "checkerboard" noise. *J. Acoust. Soc. Am.*, 93(5):2915–22, June 1993.

[75] Y. Hu and P. C. Loizou. A generalized subspace approach for enhancing speech corrupted by colored noise. *IEEE Trans. Speech Audio Process.*, 11(4):334–341, July 2003.

[76] Y. Hu and P. C. Loizou. A comparative intelligibility study of single-microphone noise reduction algorithms. *J. Acoust. Soc. Am.*, 122(3):1777–1786, Sept. 2007.

[77] Y. Hu and P. C. Loizou. Evaluation of objective quality measures for speech enhancement. *Trans. Audio, Speech and Lang. Proc.*, 16(1):229–238, Jan. 2008.

[78] Y. Hu and P. C. Loizou. Environment-specific noise suppression for improved speech intelligibility by cochlear implant users. *J. Acoust. Soc. Am.*, 127(6):3689–3695, June 2010.

[79] P.-S. Huang, M. Kim, M. Hasegawa-Johnson, and P. Smaragdis. Joint optimization of masks and deep recurrent neural networks for monaural source separation. 23(12):1–12, Dec. 2015.

[80] IEC. Objective rating of speech intelligibility by speech transmission index. EU Standard EN60268-16, International Electrotechnical Commission, 2003.

[81] ISO/TC43. Acoustics – normal equal-loudness-level contours. Standard ISO 226:2003, 2003.

[82] ITU-T. Objective measurement of active speech level. Recommendation P.56, International Telecommunications Union (ITU-T), 1993.

[83] ITU-T. Methods for subjective determination of transmission quality. Recommendation P.800, International Telecommunications Union (ITU-T), 1996.

[84] ITU-T. Perceptual evaluation of speech quality (PESQ), an objective method for end-to-end speech quality assessment of narrowband telephone networks and speech codecs. Recommendation P.862, International Telecommunications Union (ITU-T), 2001.

[85] ITU-T. Mapping function for transforming P.862 raw result scores to MOS-LQO. Recommendation P.862.1, International Telecommunications Union (ITU-T), 2003.

[86] ITU-T. Subjective test methodology for evaluating speech communication systems that include noise suppression algorithms. Recommendation P.835, International Telecommunications Union (ITU-T), 2003.

[87] J. Jensen and C. Taal. 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, Nov. 2016.

[88] J. Jensen and C. H. Taal. Speech intelligibility prediction based on mutual information. *IEEE Trans. Audio, Speech, Lang. Process.*, 22(2):430–440, Feb. 2014.

[89] Z. Jin and D. Wang. A multipitch tracking algorithm for noisy and reverberant speech. In *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4218–4221, 2010.

[90] Z. Jin and D. L. Wang. HMM-based multipitch tracking for noisy and reverberant speech. *IEEE Trans. Audio, Speech, Lang. Process.*, 19(5):1091–1102, July 2011.

[91] S. Jørgensen and T. Dau. Predicting speech intelligibility based on the signal-to-noise envelope power ratio after modulation-frequency selective processing. *J. Acoust. Soc. Am.*, 130(3):1475–1487, Sept. 2011.

[92] T. Kailath. An innovations approach to least-squares estimation–part I: Linear filtering in additive white noise. *IEEE Trans. Autom. Control*, 13(6):646–655, 1968.

[93] D. Kewley-Port, T. Z. Burkle, and J. H. Lee. Contribution of consonant versus vowel information to sentence intelligibility for young normal-hearing and elderly hearing-impaired listeners. *J. Acoust. Soc. Am.*, 122(4):2365–2375, Oct. 2007.

[94] G. Kim and P. Loizou. Improving speech intelligibility in noise using environment-optimized algorithms. *IEEE Trans. Audio, Speech, Lang. Process.*, 18(8):2080–2090, Nov. 2010.

[95] G. Kim, Y. Lu, Y. Hu, and P. C. Loizou. An algorithm that improves speech intelligibility in noise for normal-hearing listeners. *J. Acoust. Soc. Am.*, 126(3):1486–1494, Sept. 2009.

[96] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *3rd International Conference for Learning Representations*, 2015.

[97] B. E. D. Kingsbury, N. Morgan, and S. Greenberg. Robust speech recognition using the modulation spectrogram. *Speech communication*, 25(1):117–132, Aug. 1998.

[98] U. Kjems, J. B. Boldt, M. S. Pedersen, T. Lunner, and D. Wang. Role of mask pattern in intelligibility of ideal binary-masked noisy speech. *J. Acoust. Soc. Am.*, 126(3):1415–1426, Sept. 2009.

[99] U. Kjems, M. S. Pedersen, J. B. Boldt, T. Lunner, and D. Wang. Speech intelligibility of ideal binary masked mixtures. In *Proc. European Signal Processing Conf. (EUSIPCO)*, pages 1909–1913, 2010.

[100] B. W. Kleijn and R. C. Hendriks. A simple model of speech communication and its application to intelligibility enhancement. *IEEE Signal Process. Lett.*, 22(3):303–307, Mar. 2015.

[101] K. R. Kluender, J. A. Coady, and M. Kiefte. Sensitivity to change in perception of speech. *Speech communication*, 41(1):59–69, Aug. 2003.

[102] R. Kneser and H. Ney. Improved backing-off for m-gram language modeling. In *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pages 181–184, 1995.

[103] B. Kollmeier and R. Koch. Speech enhancement based on physiological and psychoacoustical models of modulation perception and binaural interaction. *J. Acoust. Soc. Am.*, 95(3):1593–1602, Mar. 1994.

[104] T. H. Koornwinder, R. Wong, R. Koekoek, and R. F. Swarttouw. Orthogonal polynomials. In Olver et al. [123], chapter 18, pages 436–484.

[105] T. Kristjansson and J. Hershey. High resolution signal reconstruction. In *Proc. IEEE Workshop on Automatic Speech Recognition and Understanding*, pages 291–296, Dec. 2003.

[106] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Proc. of the 25th International Conference on Neural Information Processing Systems*, pages 1097–1105, 2012.

[107] M. Lavandier and V. Best. Modeling binaural speech understanding in complex situations. In Blauert and Braasch [12], chapter 19.

[108] C.-H. Lee, F. K. Soong, and K. K. Paliwal, editors. *Automatic Speech and Speaker Recognition: Advanced Topics*. Kluwer Academic Publishers, Norwell, MA, USA, 1999.

[109] K.-F. Lee and H.-W. Hon. Speaker-independent phone recognition using hidden markov models. *IEEE Trans. Acoust., Speech, Signal Process.*, 37(11):1641–1648, Nov. 1989.

[110] N. Li and P. C. Loizou. Factors influencing intelligibility of ideal binary-masked speech: Implications for noise reduction. *J. Acoust. Soc. Am.*, 123(3):1673–1682, Mar. 2008.

[111] P. C. Loizou. *Speech Enhancement Theory and Practice*. Taylor & Francis, 2007.

[112] Y. Lu and G. Kim. AMS features. MATLAB code, University of Texas at Dallas, 2009.

[113] C. Ludvigsen, C. Elberling, and G. Keidser. Evaluation of noise reduction method: comparison between observed scores and scores predicted from STI. *Scandinavian audiology. Supplementum.*, 38:50–55, 1993.

[114] N. Madhu, C. Breithaupt, and R. Martin. Temporal smoothing of spectral masks in the cepstral domain for speech separation. In *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pages 45–48, 2008.

[115] M. Magdon-Ismail and J. Purnell. Approximating the covariance matrix of GMMs with low-rank perturbations. *International Journal of Data Mining, Modelling and Management*, 4:300–307, Oct. 2010.

[116] F. Mayer, D. Williamson, P. Mowlaee, and D. Wang. Impact of phase estimation on single-channel speech separation based on time-frequency masking. *J. Acoust. Soc. Am.*, 141(6):4668–4679, June 2017.

[117] G. A. Miller. The masking of speech. *Psychological bulletin*, 44(2):105–129, March 1947.

[118] D. Mishkin, N. Sergievskiy, and J. Matas. Systematic evaluation of convolution neural network advances on the imagenet. *Computer Vision and Image Understanding*, 161:11–19, Aug. 2017.

[119] B. C. J. Moore and B. R. Glasberg. Suggested formulae for calculating auditory-filter bandwidths and excitation patterns. *J. Acoust. Soc. Am.*, 74(3):750–753, Sept. 1983.

[120] N. Moritz, J. Anemüller, and B. Kollmeier. Amplitude modulation spectrogram based features for robust speech recognition in noisy and reverberant environments. In *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5492–5495, 2011.

[121] M. Mozer. A focused backpropagation algorithm for temporal pattern recognition. *Complex Systems*, 3:349–381, 1989.

[122] V. Nair and Hinton. Rectified linear units improve restricted Boltzmann machines. In *Proc. of the 27th International Conference on Machine Learning*, pages 807–814, 2010.

[123] F. W. J. Olver, D. W. Lozier, R. F. Boisvert, and C. W. Clark, editors. *NIST Handbook of Mathematical Functions*. CUP, 2010.

[124] J. H. Park. Moments of the generalized Rayleigh distribution. *Quarterly of Applied Mathematics*, 19:45–49, 1961.

[125] R. Pascanu, T. Mikolov, and Y. Bengio. On the difficulty of training recurrent neural networks. *Proceedings of the 30th International Conference on Machine Learning*, 28:1310–1318, June 2013.

[126] R. Patterson, I. Nimmo-Smith, J. Holdsworth, and P. Rice. An efficient auditory filterbank based on the gammatone function. Technical report, MRC Applied Physiology Unit, Cambridge, Dec. 1987.

[127] S. Pirhosseinloo and J. S. Brumberg. A new feature set for masking-based monaural speech separation. In *2018 52nd Asilomar Conference on Signals, Systems, and Computers*, pages 828–832, 2018.

[128] L. R. Rabiner and R. W. Schafer. *Digital Processing of Speech Signals*. Prentice-Hall, Englewood Cliffs, New Jersey, USA, 1978.

[129] D. A. Reynolds and R. C. Rose. Robust text-independent speaker identification using Gaussian mixture speaker models. *IEEE Trans. Speech Audio Process.*, 3:72–83, Jan. 1995.

[130] K. S. Rhebergen and N. J. Versfeld. A speech intelligibility index-based approach to predict the speech reception threshold for sentences in fluctuating noise for normal-hearing listeners. *J. Acoust. Soc. Am.*, 117(4):2181–2192, Apr. 2005.

[131] A. Rix, J. Beerends, M. Hollier, and A. Hekstra. Perceptual evaluation of speech quality (PESQ) - a new method for speech quality assessment of telephone networks and codecs. In *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pages 749–752, 2001.

[132] T. Rohdenburg, V. Hohmann, and B. Kollmeier. Objective perceptual quality measures for the evaluation of noise reduction schemes. In *Proc. Intl. Workshop Acoust. Echo Noise Control (IWAENC)*, pages 169–172, 2009.

[133] E. H. Rothauser, W. D. Chapman, N. Guttman, M. H. L. Hecker, K. S. Nordby, H. R. Silbiger, G. E. Urbanek, and M. Weinstock. IEEE recommended practice

for speech quality measurements. *IEEE Trans. Audio Electroacoust.*, 17(3):225–246, 1969.

[134] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning representations by back-propagating errors. *Nature*, 323:533–536, Oct. 1986.

[135] M. Seltzer, B. Raj, and R. Stern. A Bayesian classifier for spectrographic mask estimation for missing feature speech recognition. *Speech Communication*, 43(4):379–393, Sept. 2004.

[136] S. So and K. K. Paliwal. Modulation-domain Kalman filtering for single-channel speech enhancement. *Speech Communication*, 53(6):818–829, July 2011.

[137] J. Sohn, N. S. Kim, and W. Sung. A statistical model-based voice activity detection. *IEEE Signal Process. Lett.*, 6(1):1–3, Jan. 1999.

[138] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, 15(1):1929–1958, Jan. 2014.

[139] H. J. M. Steeneken and F. W. M. Geurtsen. Description of the RSG.10 noise database. Technical Report IZF 1988–3, TNO Institute for perception, 1988.

[140] H. J. M. Steeneken and T. Houtgast. A physical method for measuring speech-transmission quality. *J. Acoust. Soc. Am.*, 67(1):318–326, Jan. 1980.

[141] S. S. Stevens. A scale for the measurement of a psychological magnitude: Loudness. *Psychological Review*, 43(5):405–416, 1936.

[142] S. S. Stevens, J. Volkman, and E. B. Newman. A scale for the measurement of the psychological magnitude of pitch. *J. Acoust. Soc. Am.*, 8:185–19, 1937.

[143] C. E. Stilp and K. R. Kluender. Cochlea-scaled entropy, not consonants, vowels, or time, best predicts speech intelligibility. *Proc. National Academy of Sciences*, 107(27):12387–12392, July 2010.

[144] T. Stokes, C. Hummersone, and T. Brookes. Reducing binary masking artefacts in blind audio source separation. In *Audio Engineering Society Convention 134*, pages 243–250, 2013.

[145] I. Stuijt and R. Drullman. Effect of reducing temporal intensity modulations on sentence intelligibility. *J. Acoust. Soc. Am.*, 101(1):498–502, Feb. 1997.

[146] M. Sundermeyer, R. Schlüter, and H. Ney. LSTM neural networks for language modeling. In *Proc. Interspeech Conf.*, pages 194–197, 2012.

[147] I. Sutskever, O. Vinyals, and Q. Le. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems*, pages 3104–3112, 2014.

[148] C. Taal, R. C. Hendriks, H. Richard, J. Jensen, and U. Kjems. An evaluation of objective quality measures for speech intelligibility prediction. In *Proc. Interspeech Conf.*, pages 1947–1950, 2009.

[149] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen. On predicting the difference in intelligibility before and after single-channel noise reduction. In *Proc. Intl. Workshop Acoust. Echo Noise Control (IWAENC)*, 2010.

[150] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen. A short-time objective intelligibility measure for time-frequency weighted noisy speech. In *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4214–4217, 2010.

[151] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen. An algorithm for intelligibility prediction of time-frequency weighted noisy speech. *IEEE Trans. Audio, Speech, Lang. Process.*, 19(7):2125–2136, Sept. 2011.

[152] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen. An evaluation of objective measures for intelligibility prediction of time-frequency weighted noisy speech. *J. Acoust. Soc. Am.*, 130(5):3013–3027, Nov. 2011.

[153] J. Taghia and R. Martin. Objective intelligibility measures based on mutual information for speech subjected to speech enhancement processing. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 22(1):6–16, Jan 2014.

[154] J. Taghia, R. Martin, and R. C. Hendriks. On mutual information as a measure of speech intelligibility. In *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pages 65–68, 2012.

[155] D. Talkin. A robust algorithm for pitch tracking (RAPT). In W. B. Kleijn and K. K. Paliwal, editors, *Speech Coding and Synthesis*, pages 495–518. Elsevier, Amsterdam, 1995.

[156] I. R. Titze. *Principles of Voice Production.* Prentice Hall, 1994.

[157] D. E. Tsoukalas, J. N. Mourjopoulos, and G. Kokkinakis. Speech enhancement based on audible noise suppression. *IEEE Trans. Speech Audio Process.*, 5(6):497–514, Nov. 1997.

[158] G. Van den Brink. Detection of tone pulse of various durations in noise of various bandwidths. *J. Acoust. Soc. Am.*, 36(6):1206–1211, June 1964.

[159] S. Van Kuyk, W. B. Kleijn, and R. C. Hendriks. An intelligibility metric based on a simple model of speech communication. In *Proc. Intl. Workshop Acoust. Signal Enhancement (IWAENC)*, pages 1–5, 2016.

[160] S. Van Kuyk, W. B. Kleijn, and R. C. Hendriks. An instrumental intelligibility metric based on information theory. *IEEE Signal Processing Letters*, 25(1):115–119, Jan 2018.

[161] A. Varga and H. J. M. Steeneken. Assessment for automatic speech recognition II: NOISEX-92: a database and an experiment to study the effect of additive noise on speech recognition systems. *Speech Communication*, 3(3):247–251, July 1993.

[162] N. Virag. Single channel speech enhancement based on masking properties of the human auditory system. *IEEE Trans. Speech Audio Process.*, 7(2):126–137, Mar. 1999.

[163] R. W. Peters, B. Moore, and T. Baer. Speech reception thresholds in noise with and without spectral and temporal dips for hearing-impaired and normally hearing people. *J. Acoust. Soc. Am.*, 103(1):577–587, Feb. 1998.

[164] D. Wang. On ideal binary mask as the computational goal of auditory scene analysis. In P. Divenyi, editor, *Speech Separation by Humans and Machines*, pages 181–197. Kluwer Academic, 2005.

[165] D. Wang. Cochleagram feature extraction. MATLAB code, Ohio State, 2008.

[166] D. Wang and G. Brown, editors. *Computational Auditory Scene Analysis: Principles, Algorithms, and Applications*. Wiley, 2006.

[167] D. Wang, U. Kjems, M. S. Pedersen, J. B. Boldt, and T. Lunner. Speech intelligibility in background noise with ideal binary time-frequency masking. *J. Acoust. Soc. Am.*, 125(4):2336–2347, Apr. 2009.

[168] D. Wang and J. Lim. The unimportance of phase in speech enhancement. *IEEE Trans. Acoust., Speech, Signal Process.*, 30(4):679–681, Aug. 1982.

[169] Y. Wang and M. Brookes. Model-based speech enhancement in the modulation domain. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 26(3):580–594, March 2018.

[170] Y. Wang, K. Han, and D. Wang. Exploring monaural features for classification-based speech segregation. *IEEE Trans. Audio, Speech, Lang. Process.*, 21(2):270–279, Feb. 2013.

[171] Y. Wang, A. Narayanan, and D. Wang. On training targets for supervised speech separation. *IEEE Trans. Audio, Speech, Lang. Process.*, 22(12):1849–1858, Dec. 2014.

[172] Y. Wang and D. Wang. Towards scaling up classification-based speech separation. *IEEE Trans. Audio, Speech, Lang. Process.*, 21(7):1381–1390, July 2013.

[173] Z.-Q. Wang, J. Le Roux, D. Wang, and J. Hershey. End-to-end speech separation with unfolded iterative phase reconstruction. In *Proc. Interspeech Conf.*, pages 2708–2712, 2018.

[174] F. Weninger, J. R. Hershey, J. Le Roux, and B. Schuller. Discriminatively trained recurrent neural networks for single-channel speech separation. In *Proc. IEEE Global Conf. Signal and Information Processing (GlobalSIP)*, pages 577–581, 2014.

[175] R. J. Williams and J. Peng. An efficient gradient-based algorithm for on-line training of recurrent network trajectories. *Neural Computation*, 2:490–501, 1990.

[176] D. S. Williamson and D. Wang. Speech dereverberation and denoising using complex ratio masks. In *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5590–5594, 2017.

[177] D. S. Williamson, Y. Wang, and D. Wang. Reconstruction techniques for improving the perceptual quality of binary masked speech. *J. Acoust. Soc. Am.*, 136(2):892–902, Aug. 2014.

[178] D. S. Williamson, Y. Wang, and D. Wang. Complex ratio masking for monaural speech separation. *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, 24(3):483–492, Mar. 2016.

[179] K. Wilson, M. Chinen, J. Thorpe, B. Patton, J. Hershey, R. Saurous, J. Skoglund, and R. Lyon. Exploring tradeoffs in models for low-latency speech enhancement. In *Proc. Intl. Workshop Acoust. Signal Enhancement (IWAENC)*, pages 366–370, 2018.

[180] S. Wisdom, J. R. Hershey, K. Wilson, J. Thorpe, M. Chinen, B. Patton, and R. A. Saurous. Differentiable consistency constraints for improved deep speech enhancement. In *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pages 900–904, 2019.

[181] A. Yasmin, P. Fieguth, and L. Deng. Speech enhancement using voice source models. In *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, pages 797–800, 1999.

[182] M. Yuan and Y. Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):49–67, Dec. 2006.

[183] B. Zhang, M. Hsu, U. Dayal, and M. Data. K-harmonic means - a data clustering algorithm. *Hewlett Packard Research Laboratory Technical Report*, Dec. 1999.

[184] Y. Zhao, D. Wang, E. M. Johnson, and E. W. Healy. A deep learning based segregation algorithm to increase speech intelligibility for hearing-impaired listeners in reverberant-noisy conditions. *J. Acoust. Soc. Am.*, 144(3):1627–1637, Sept. 2018.

[185] E. Zwicker. Subdivision of audible frequency range into critical bands. *J. Acoust. Soc. Am.*, 33(2):248, Feb. 1961.

Appendix A

Noise Databases

The eight acoustic noise signals used for evaluating algorithms in this thesis were taken from the RSG.10 [139] database. Figure A.1 shows a spectrogram of a 3-second extract of each of the eight noise types. The first two, speech-shaped-noise (SS) and multi-talker-babble (babble), have the same long-term spectrum as speech. It can be seen that the last two noise types are substantially different from the others: machine-gun noise is highly non-stationary while Volvo noise (recorded inside a moving car) is strongly concentrated at low frequencies. Figure A.2 plots the value of the WSTOI intelligibility metric versus SNR for each of the noise types. It can be seen that, for most noise types, 0 dB SNR corresponds to a WSTOI value of about 0.7, which corresponds to an intelligibility of approximately 73.4 % using the mapping between WSTOI and predicted intelligibility, (3.3), from Chapter 3. In contrast, the atypical noise types, machine gun and Volvo, result in WSTOI=0.7 at SNRs of -18.0 and -25.2 dB respectively, meaning that they require much higher noise levels to obtain the same reduction in predicted intelligibility as the other noises.

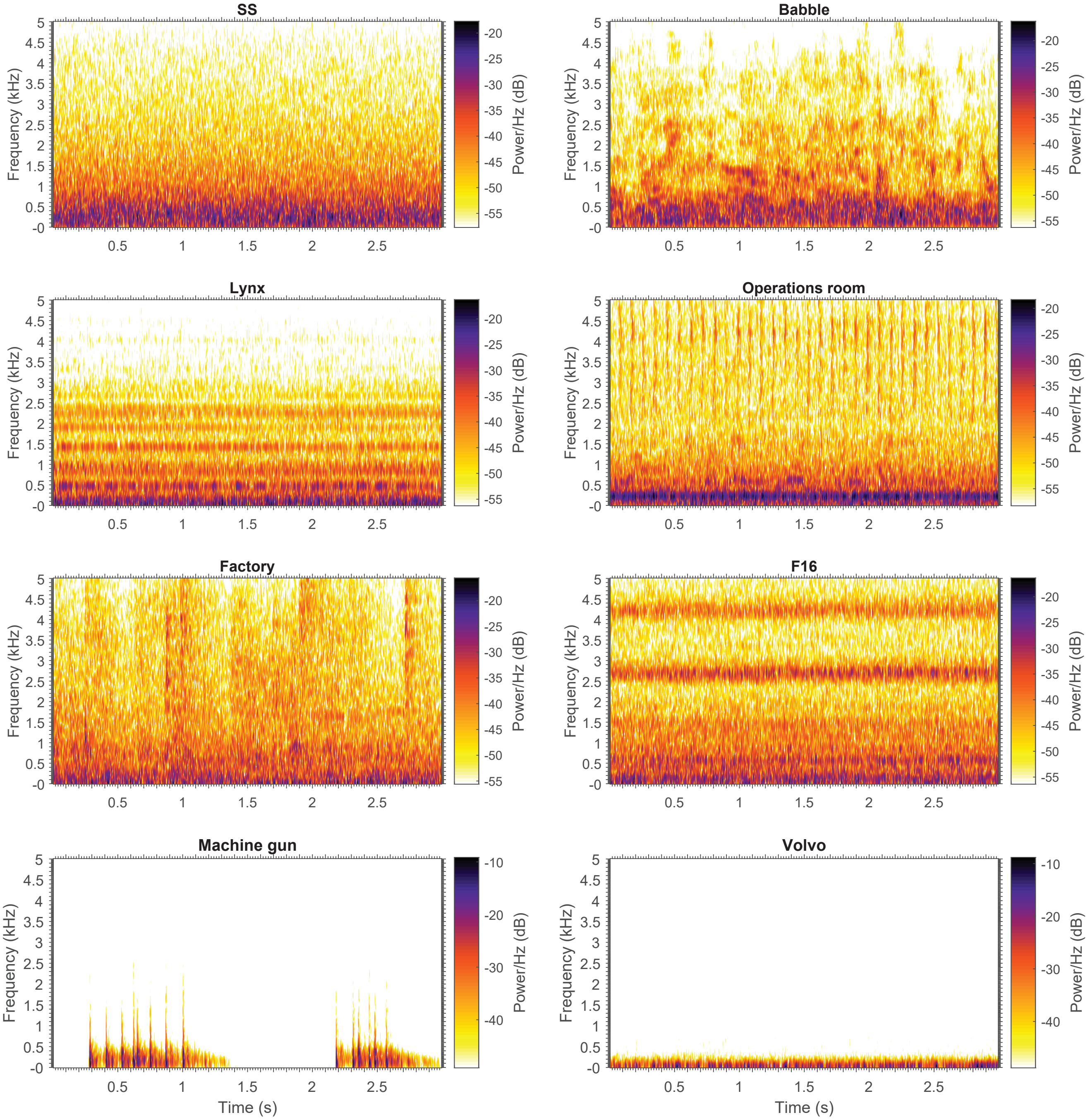


Figure A.1: Spectrograms of 3 second extracts of 8 noise types taken from the RSG.10 [139] database. The noises were downsampled from the original sample rate of 16 kHz to 10 kHz.

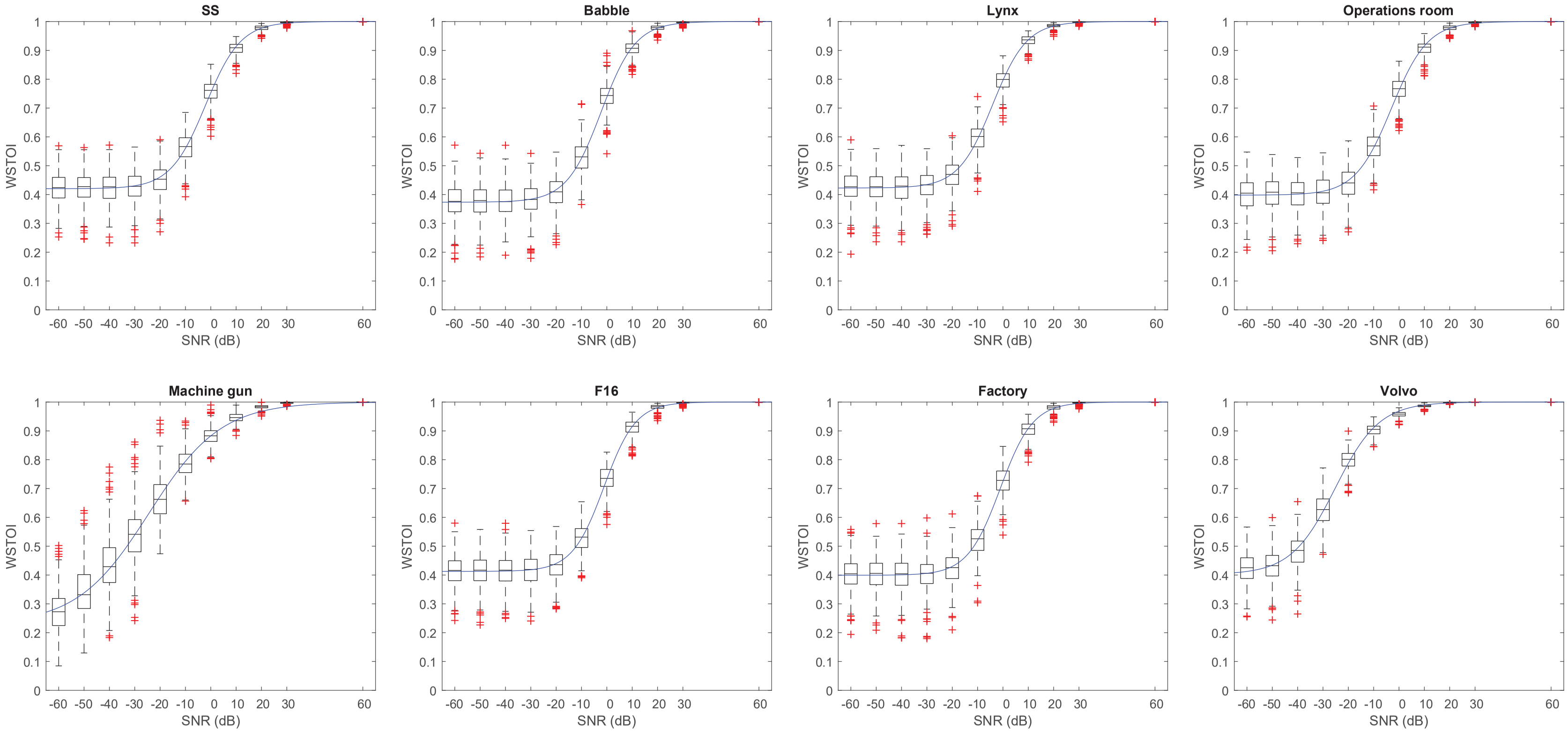


Figure A.2: Boxplots of WSTOI against SNR for the 8 noise types used for algorithm testing. The noise types are from the RSG.10 [139] database. 400 utterances were randomly selected from the Training dataset outlined in Sec. 5.4, which is formed from the training set of the TIMIT corpus [45]. To generate the data in each plot, each of the 400 utterances was mixed with each noise at each of the SNRs displayed in the plot. The blue curve is equation 5.7, fitted to the data in each plot using least squares optimisation.

Appendix B

SSOBM moments

Section 4.3 presented SSOBM, the binary mask that maximises STOI for a stochastic noise signal with a known power spectrum. This appendix derives expressions for the mean and variance of the masked modulation vectors arising in this case.

B.1 Distribution of a single frequency bin

If $\left\langle |N|^2 \right\rangle = \sigma^2$ and it is assumed that the real and imaginary parts of N are Gaussian with the same variance, we have $\Re\left(N\right) \sim \mathcal{N}(0,\, 0.5\sigma^2)$. Note that we have omitted the time and frequency-bin indices m and k. Normalising to unit variance gives $\Re\left(\sqrt{2}\sigma^{-1}N\right) \sim \mathcal{N}(0, 1)$. Therefore

$$\left|\sqrt{2}\sigma^{-1}N\right|^2 = \Re^2(\sqrt{2}\sigma^{-1}N) + \Im^2(\sqrt{2}\sigma^{-1}N) \sim \chi_2^2$$

has a χ^2 distribution with 2 degrees of freedom. The normalised noisy speech coefficient is

$$\begin{aligned}\sqrt{2}\sigma^{-1}Y &= \sqrt{2}\sigma^{-1}X + \sqrt{2}\sigma^{-1}N \\ &= \Re\left(\sqrt{2}\sigma^{-1}X + \sqrt{2}\sigma^{-1}N\right) + j\Im\left(\sqrt{2}\sigma^{-1}X + \sqrt{2}\sigma^{-1}N\right)\end{aligned}$$

where

$$\begin{aligned}\Re\left(\sqrt{2}\sigma^{-1}X + \sqrt{2}\sigma^{-1}N\right) &\sim \mathcal{N}\left(\Re\left(\sqrt{2}\sigma^{-1}X\right), 1\right) \\ \Im\left(\sqrt{2}\sigma^{-1}X + \sqrt{2}\sigma^{-1}N\right) &\sim \mathcal{N}\left(\Im\left(\sqrt{2}\sigma^{-1}X\right), 1\right).\end{aligned}$$

It follows that the distribution of $\left|\sqrt{2}\sigma^{-1}Y\right|^2$ is a non-central χ_2^2 distribution whose non-centrality parameter, R, is given by

$$\begin{aligned}R &= \Re^2\left(\sqrt{2}\sigma^{-1}X\right) + \Im^2\left(\sqrt{2}\sigma^{-1}X\right) \\ &= 2\sigma^{-2}\left|X\right|^2.\end{aligned}$$

B.2 Distribution of multiple bins

We now consider a single third-octave band given by

$$Y_j = \sqrt{\sum_{k=K_j}^{K_{j+1}-1} \left|Y(k)\right|^2}$$

as in (2.5). Note that, unlike $Y(k)$, Y_j is real-valued and positive. Generalising the previous discussion for a single frequency bin, the distribution of $w_j^2 = 2\sigma_j^{-2}Y_j^2$ is a non-central $\chi^2_{\nu_j}$distribution with $\nu_j \triangleq 2\left(K_{j+1} - K_j\right)$ degrees of freedom and a non-centrality parameter

$$R_j = 2\sigma_j^{-2} \sum_{k=K_j}^{K_{j+1}-1} \left|X(k)\right|^2 .$$

It follows that, from (1.2) of [124] (with a corrected sign), the distribution of $w_j = \sqrt{2}\sigma^{-1}Y_j$ is a non-central χ_{ν_j} distribution with PDF

$$\chi_{\nu_j}(w_j;\, R_j) = R_j^{0.5-0.25\nu_j} w_j^{0.5\nu_j} \exp\left(-0.5\left(w_j^2 + R_j\right)\right) I_{0.5\nu_j-1}\left(R_j^{0.5} w_j\right)$$

where $I_\alpha\left(\cdots\right)$ is a modified Bessel function of the first kind. From (1.6) of [124], the mean of this distribution may be expressed in terms of either the confluent hypergeometric function, ${}_1F_1(a,\, b,\, z) \equiv M(a,\, b,\, z)$, also called Kummer's M-function, or the generalised Laguerre polynomial, $L_n^{(\alpha)}(z)$ as

$$\begin{aligned}\left\langle \sqrt{2}\sigma_j^{-1}Y_j \right\rangle \equiv \langle w_j \rangle &= 2^{0.5} \exp\left(-0.5R_j\right) \frac{\Gamma\left(0.5\left(\nu_j+1\right)\right)}{\Gamma\left(0.5\nu_j\right)} M\left(0.5\left(\nu_j+1\right), 0.5\nu_j, 0.5R_j\right) \\ &= 2^{0.5} \frac{\Gamma\left(0.5\left(\nu_j+1\right)\right)}{\Gamma\left(0.5\nu_j\right)} M\left(-0.5, 0.5\nu_j, -0.5R_j\right) \\ &= 2^{0.5}\Gamma\left(1.5\right) L_{0.5}^{0.5\nu_j-1}\left(-0.5R_j\right) \\ &= 2^{-0.5}\pi^{0.5} L_{0.5}^{0.5\nu_j-1}\left(-0.5R_j\right).\end{aligned}$$

where the second line uses (13.2.39), the third line uses (13.6.19) and (5.2.5) and the last line uses (5.4.6) and (5.5.1) all from [123] . The second raw moment, $\left\langle w_j^2 \right\rangle$, is the mean of the corresponding $\chi^2_{\nu_j}$distribution and equals $\nu_j + R_j$. Removing the normalisation gives equations (4.3) and (4.4) from Section 4.3 as

$$\left\langle Y_j \right\rangle = 2^{-0.5}\sigma_j \left\langle w_j \right\rangle = 0.5\pi^{0.5}\sigma_j L_{0.5}^{0.5\nu_j - 1}\left(-0.5R_j\right) \tag{B.1}$$

$$\left\langle Y_j^2 \right\rangle = 0.5\sigma_j^2 \left\langle w_j^2 \right\rangle = 0.5\sigma_j^2 \left(\nu_j + R_j\right). \tag{B.2}$$

B.3 Statistics of masked noisy speech

Analogous to (2.6), we now define the length-M masked-speech and non-centrality vectors, $\mathbf{z}_j = \mathbf{b}_j \circ \mathbf{y}_j$ and $\mathbf{r}_j$, where $\mathbf{b}_j$ is the mask vector and $\circ$ denotes element-wise multiplication. Analogous to (2.8), we define the mean of the elements of $\mathbf{z}_j$ to be $\bar{z}_j = \frac{1}{M}\mathbf{1}^T\mathbf{z}_j$ where $\mathbf{1}$ denotes a vector of ones. Dropping the j subscript for clarity, we can now write

$$\begin{aligned}
\left\langle \left\|\mathbf{z} - \mathbf{1}\bar{z}\right\|^2 \right\rangle &= \left\langle \left(\mathbf{z} - \frac{1}{M}\mathbf{1}\bar{z}\right)^T \left(\mathbf{z} - \frac{1}{M}\mathbf{1}\bar{z}\right) \right\rangle \\
&= \operatorname{tr}\left(\left\langle \left(\mathbf{z} - \frac{1}{M}\mathbf{1}\mathbf{1}^T\mathbf{z}\right)\left(\mathbf{z} - \frac{1}{M}\mathbf{1}\mathbf{1}^T\mathbf{z}\right)^T \right\rangle\right) \\
&= \operatorname{tr}\left(\left\langle \mathbf{z}\mathbf{z}^T \right\rangle\right) - \frac{2}{M}tr\left(\left\langle \mathbf{1}\mathbf{1}^T\mathbf{z}\mathbf{z}^T \right\rangle\right) + \frac{1}{M^2}tr\left(\left\langle \mathbf{1}\mathbf{1}^T\mathbf{z}\mathbf{z}^T\mathbf{1}\mathbf{1}^T \right\rangle\right) \\
&= \operatorname{tr}\left(\left\langle \mathbf{z}\mathbf{z}^T \right\rangle\right) - \frac{2}{M}\mathbf{1}^T\left\langle \mathbf{z}\mathbf{z}^T \right\rangle\mathbf{1} + \frac{\mathbf{1}^T\mathbf{1}}{M^2}\mathbf{1}^T\left\langle \mathbf{z}\mathbf{z}^T \right\rangle\mathbf{1} \\
&= \operatorname{tr}\left(\left\langle \mathbf{z}\mathbf{z}^T \right\rangle\right) - \frac{1}{M}\mathbf{1}^T\left\langle \mathbf{z}\mathbf{z}^T \right\rangle\mathbf{1}.
\end{aligned}$$

We now assume that the elements of $\mathbf{z}$ are uncorrelated, so that

$$\begin{aligned}
\left\langle \mathbf{z}\mathbf{z}^T \right\rangle &= \left\langle \mathbf{z} \right\rangle \left\langle \mathbf{z} \right\rangle^T + \operatorname{diag}\left(\left\langle \mathbf{z}\circ\mathbf{z} \right\rangle - \left\langle \mathbf{z} \right\rangle \circ \left\langle \mathbf{z} \right\rangle\right) \\
&= \left\langle \mathbf{z} \right\rangle \left\langle \mathbf{z} \right\rangle^T + \operatorname{diag}\left(\mathbf{b}\circ\left\langle \mathbf{y}\circ\mathbf{y} \right\rangle\right) - \operatorname{diag}\left(\left\langle \mathbf{z} \right\rangle \circ \left\langle \mathbf{z} \right\rangle\right)
\end{aligned}$$

where the second term in the first line is a diagonal covariance matrix. Using the matrix identities $\mathrm{tr}\left(\mathbf{p}\mathbf{q}^T\right) = \mathbf{p}^T\mathbf{q}$ and $\mathrm{tr}\left(\mathrm{diag}\left(\mathbf{p} \circ \mathbf{q}\right)\right) = \mathbf{p}^T\mathbf{q}$, it follows that

$$\mathrm{tr}\left(\left\langle \mathbf{z}\mathbf{z}^T \right\rangle\right) = \left\langle \mathbf{z} \right\rangle^T \left\langle \mathbf{z} \right\rangle + \left\langle \mathbf{b} \right\rangle^T \left\langle \mathbf{y} \circ \mathbf{y} \right\rangle - \left\langle \mathbf{z} \right\rangle^T \left\langle \mathbf{z} \right\rangle = \left\langle \mathbf{b} \right\rangle^T \left\langle \mathbf{y} \circ \mathbf{y} \right\rangle .$$

Thus we can write

$$\begin{aligned}\left\langle \|\mathbf{z} - \mathbf{1}\bar{z}\|^2 \right\rangle &= \mathrm{tr}\left(\left\langle \mathbf{z}\mathbf{z}^T \right\rangle\right) - \frac{1}{M}\mathbf{1}^T \left\langle \mathbf{z}\mathbf{z}^T \right\rangle \mathbf{1} \\ &= \left\langle \mathbf{b} \right\rangle^T \left\langle \mathbf{y} \circ \mathbf{y} \right\rangle - \frac{1}{M}\mathbf{1}^T \left\langle \mathbf{z} \right\rangle \left\langle \mathbf{z} \right\rangle^T \mathbf{1} - \frac{1}{M}\mathbf{1}^T \mathrm{diag}\left(\mathbf{b} \circ \left\langle \mathbf{y} \circ \mathbf{y} \right\rangle\right) \mathbf{1} \\ &\qquad + \frac{1}{M}\mathbf{1}^T \mathrm{diag}\left(\left\langle \mathbf{z} \right\rangle \circ \left\langle \mathbf{z} \right\rangle\right) \mathbf{1} \\ &= \left\langle \mathbf{b} \right\rangle^T \left\langle \mathbf{y} \circ \mathbf{y} \right\rangle - \frac{1}{M}\left(\mathbf{1}^T \left\langle \mathbf{z} \right\rangle\right)^2 - \frac{1}{M}\mathbf{1}^T \mathrm{diag}\left(\mathbf{b} \circ \left\langle \mathbf{y} \circ \mathbf{y} \right\rangle\right) \mathbf{1} \\ &\qquad + \frac{1}{M}\mathbf{1}^T \mathrm{diag}\left(\left\langle \mathbf{z} \right\rangle \circ \left\langle \mathbf{z} \right\rangle\right) \mathbf{1}.\end{aligned}$$

Noting that $\mathbf{1}^T\mathrm{diag}(\mathbf{p} \circ \mathbf{q})\mathbf{1} = \mathbf{1}^T\left(\mathbf{p} \circ \mathbf{q}\right) = \mathbf{p}^T\mathbf{q}$, this simplifies to

$$\begin{aligned}\left\langle \|\mathbf{z} - \mathbf{1}\bar{z}\|^2 \right\rangle &= \left\langle \mathbf{b} \right\rangle^T \left\langle \mathbf{y} \circ \mathbf{y} \right\rangle - \frac{1}{M}\left(\mathbf{1}^T \left\langle \mathbf{z} \right\rangle\right)^2 - \frac{1}{M}\mathbf{b}^T \left\langle \mathbf{y} \circ \mathbf{y} \right\rangle + \frac{1}{M}\left\langle \mathbf{b} \circ \mathbf{y} \right\rangle^T \left\langle \mathbf{b} \circ \mathbf{y} \right\rangle \\ &= \frac{M-1}{M}\left\langle \mathbf{b} \right\rangle^T \left\langle \mathbf{y} \circ \mathbf{y} \right\rangle - \frac{1}{M}\left(\mathbf{b}^T \left\langle \mathbf{y} \right\rangle\right)^2 + \frac{1}{M}\left\|\mathbf{b} \circ \left\langle \mathbf{y} \right\rangle\right\|^2 \\ &= 0.5\sigma^2\frac{M-1}{M}\left\langle \mathbf{b} \right\rangle^T \left(\nu\mathbf{1} + \mathbf{r}\right) - \frac{\pi\sigma_j^2}{4M}\left(\mathbf{b}^T L_{0.5}^{0.5\nu_j - 1}\left(-0.5\mathbf{r}\right)\right)^2 \\ &\qquad + \frac{\pi\sigma_j^2}{4M}\left\|\mathbf{b} \circ L_{0.5}^{0.5\nu_j - 1}\left(-0.5\mathbf{r}\right)\right\|^2\end{aligned}$$

which is (4.6) in Section 4.3.