Temporal Processing of Speech in a Time-Feature Space

Carlos Avendaño
B.S., Instituto Tecnológico y de Estudios Superiores de Monterrey CEM, México, 1991
M.S., Oregon Graduate Institute of Science & Technology, 1993

A dissertation submitted to the faculty of the
Oregon Graduate Institute of Science & Technology
in partial fulfillment of the
requirements for the degree
Doctor of Philosophy
in
Electrical Engineering

April 1997

The dissertation "Temporal Processing of Speech in a Time-Feature Space" by Carlos Avendaño has been examined and approved by the following Examination Committee:

Hynek Hermansky
Associate Professor
Thesis Research Adviser

Misha Pavel
Associate Professor

Eric A. Wan
Assistant Professor

Yegnanarayana Bayya
Professor
Indian Institute of Technology, Madras

Man Mohan Sondhi
Distinguished Member of the Technical Staff
Bell Laboratories, Lucent Technologies

Dedication

A Ale

Acknowledgments

The work I present in this dissertation has been possible thanks to the collaboration and support that I received from all of the members of our lab. I am immensely grateful to Professor Hynek Hermansky for taking me as his student and guiding me throughout this quest for knowledge. In fact, many of the ideas behind this dissertation were stimulated by Hynek, and the contributions I present wouldn't have been possible without his involvement with my work.

I am indebted to Dr. Eric Wan for being my second advisor during the early stages of my research. Part of this dissertation was based on his input and original ideas. I would also like to thank the other members of my committee: Dr. Misha Pavel, Dr. Mohan Sondhi and Dr. B. Yegnanarayana, who kindly reviewed my thesis enriching it with their comments and suggestions.

My infinite gratitude to my wife Alejandra who shared with me this incredible experience, and whose love and support gave me the energy necessary to reach my goal. Two people who deserve a lot of the credit, as they were responsible for providing me with the tools to face any challenge in life, are my parents Pepina and Carlos. Special thanks to the two fellows who grew up with me, my brothers Mauricio and Leonardo, for supporting me and cheering me up in all my endeavors. Thank you all for your love!

I finally want to express my gratitude to my family and friends here and in México, to all the teachers I had during my life, the faculty and students at OGI, CIT, CSLU, and the organizations that provided the support for my graduate studies, CONACyT, OGI and USWEST.

Contents

Dedication iii

Acknowledgments iv

Abstract xi

1 Introduction 1
- 1.1 Speech Processing Applications 2
- 1.2 Relevant Background 2
- 1.3 Outline 4

2 Review of Short-Time Analysis of Signals 5
- 2.1 Time-Frequency Representation of Signals 5
  - 2.1.1 Relation to the Fourier Transform 8
  - 2.1.2 Discussion 9
  - 2.1.3 Filter Bank Interpretation of the STFT 9
- 2.2 Time-Feature Representations of Speech 10

3 Temporal Processing 12
- 3.1 Filtering of the Time Trajectories 12
- 3.2 CIT-MIF Modification of the Short-Time Spectrum 13
  - 3.2.1 Description of the CIT-MIF Modifications 13
  - 3.2.2 Synthesis from the STFT 14
  - 3.2.3 Time Domain Effects of CIT-MIF Modifications 15
  - 3.2.4 Filter Bank Interpretation 16
  - 3.2.5 Discussion 19
- 3.3 Summary 20

4 Temporal Processing in Non-Linear Domains 21
- 4.1 Temporal Processing of the STFTM 22
  - 4.1.1 Definitions of STFTM and STFTP 22
  - 4.1.2 CIT-MIF Modification of the STFTM 23

4.1.3 Phase Effects . . . 25
4.2 Temporal Processing in Other Non-Linear Domains . . . 28
4.2.1 Time Trajectory Filters . . . 29
4.2.2 Time-Domain Signal Resynthesis . . . 29
4.3 Summary . . . 29

5 Temporal Processing for Channel Normalization . . . 32
5.1 Background . . . 33
5.1.1 Cepstral Mean Subtraction . . . 34
5.1.2 RASTA Processing . . . 35
5.2 Convolutional Distortions . . . 37
5.2.1 Effects of the Channel on the STFT . . . 37
5.2.2 Discussion . . . 42
5.3 Summary . . . 42

6 Noise Reduction . . . 43
6.1 Background . . . 43
6.2 Motivation . . . 45
6.2.1 Previous Work . . . 45
6.3 RASTA-Like Noise Reduction Technique . . . 46
6.3.1 Filter Design . . . 47
6.3.2 Tests . . . 48
6.3.3 Parameter Values . . . 48
6.3.4 Evaluation . . . 50
6.3.5 Properties of RASTA-Like Filters . . . 50
6.3.6 Wiener-Like Behavior of RASTA-Like Filter Bank . . . 53
6.4 The Effect of Signal to Noise Ratio on the Properties of the RASTA-Like Filters . . . 55
6.4.1 Preliminary Studies . . . 55
6.4.2 SNR-dependent RASTA-like Filters . . . 56
6.5 Adaptive System Design . . . 58
6.5.1 SNR Estimation . . . 60
6.5.2 Filter Design . . . 60
6.5.3 Operation of the System . . . 61
6.6 Noise Reduction Results . . . 62
6.6.1 Known noise . . . 62
6.6.2 Unknown noise . . . 63
6.7 Summary . . . 65

7 Reverberation Reduction **66**
7.1 Background 66
7.1.1 The MTF and MI 67
7.1.2 Effects of Reverberation on Speech 68
7.2 Using the MI Concept for Reverberation Reduction 70
7.3 Preliminary Experiments 70
7.3.1 High-Pass Filtering of the STFT Power Spectrum 70
7.3.2 Inverting a Theoretical MTF 71
7.4 Technique 72
7.4.1 Filter Design 72
7.5 Experiments 73
7.5.1 Data-Derived Filters 73
7.5.2 Results 75
7.6 Summary 76

8 Data-Driven Filter Design for Channel Normalization in ASR **77**
8.1 Motivation 77
8.2 Filter Design by Constrained Optimization 78
8.2.1 Technique 80
8.2.2 Experimental Design 80
8.3 Results 83
8.3.1 Constraint effects 84
8.4 ASR Experiment 85
8.5 Summary 85

9 Multiresolution Channel Normalization for ASR in Reverberant Environments **86**
9.1 Introduction 86
9.1.1 Background 87
9.1.2 Problem 88
9.2 Multiresolution Concept 88
9.2.1 The Algorithm 90
9.3 Technique 91
9.3.1 Implementation 94
9.4 Experimental Results 96
9.4.1 Channel Independence 96
9.4.2 ASR Experiments 98
9.5 Summary 99

10 Conclusion and Future Directions **101**
10.1 Summary and Future Work 101
10.1.1 Noise Reduction for Speech Enhancement 102
10.1.2 Reverberation Reduction for Speech Enhancement 103
10.1.3 Data-Driven Design of Temporal Filters for Channel Normalization . 103
10.1.4 Multiresolution Channel Normalization for Reverberation Reduction in ASR 104

Bibliography **106**

A Derivation of (3.7) and (3.8) **112**

B Derivation of (4.12) **113**

C The Transformation Matrix A **115**

Biographical Note **118**

List of Figures

2.1 *Two-dimensional representation of a signal. As an example of a time-frequency representation, the short-time power spectrum is also depicted.* . . 6
2.2 *Filter bank interpretation of the STFT* 10

3.1 *(a) filter bank interpretation of temporal processing. (b) equivalent system* . 17
3.2 *(a) filter bank interpretation of temporal processing in the FBS method. (b) equivalent system* 18

4.1 *Block diagram of temporal processing on the STFTM* 31

5.1 *Effect of the channel on the STFT. (a) Filter bank interpretation. (b) Equivalent system.* 39

6.1 *Block diagram of noise reduction system. $x(n)$ is the noisy speech, and $\widehat{s}(n)$ the processed speech. The compression is $\gamma = 1.5$.* 49
6.2 *Frequency responses of RASTA-like filters* 51
6.3 *Frequency response of filters at different bands. The labels in this figure correspond to the regions with the same label in Fig. 6.2* 52
6.4 *Impulse responses of RASTA-like filters at (a) region A Fig. 6.2, (b) region B in Fig. 6.2, and (c) region C in Fig. 6.2. For comparison, the dark bar on the time axis corresponds to the length of the analysis window, i.e. 32 ms.* 53
6.5 *Wiener filter response and norm of RASTA-like filters.* 54
6.6 *Filter frequency responses (dotted lines) and mean response (solid lines) for several frequency-specific SNR levels* 57
6.7 *Block diagram of the adaptive system. $x(n)$ is the input corrupted speech, $\hat{s}(n)$ is the estimate of the clean speech ($\gamma = 1.5$)* 59
6.8 *Waveform and spectrogram of (a) original clean speech signal, (b) the noisy signal, and (c) the processed noisy signal.* 63
6.9 *(a) Noisy speech signal (above) and corresponding spectrogram (below). (b) time signal (above) and spectrogram (below) of the same noisy segment after processing.* 64

7.1 *Modulation index computation. After Houtgast and Steeneken (1985).* . . . 68
7.2 *Magnitude frequency response of a data-derived filter (at 1 kHz center frequency band) compared to the theoretical curve.* 74
7.3 *Modulation index at 1 kHz for clean speech, reverberant speech and processed speech.* 75

8.1 *Problem setup block diagram* 79
8.2 *Magnitude frequency response of COP and RASTA filters* 83
8.3 *Magnitude frequency response of COP filters for different critical bands* . . . 84

9.1 *Multiresolution Processing Concept.* 89
9.2 *Block diagram of the multiresolution normalization technique.* 93
9.3 *Channel independence results for multiresolution normalization. Critical band energy spectrograms of (a) clean and (b) the corresponding reverberant speech. Critical band spectrograms of (c) clean and (d) reverberant speech after multiresolution normalization.* 97

Abstract

Temporal Processing of Speech
in a
Time-Feature Space

Carlos Avendaño, Ph.D.
Oregon Graduate Institute of Science & Technology, 1997

Supervising Professor: Hynek Hermansky

The performance of speech communication systems often degrades under realistic environmental conditions. Adverse environmental factors include additive noise sources, room reverberation, and transmission channel distortions. This work studies the processing of speech in the temporal-feature or modulation spectrum domain, aiming for alleviation of the effects of such disturbances.

Speech reflects the geometry of the vocal organs, and the linguistically dominant component is in the shape of the vocal tract. At any given point in time, the shape of the vocal tract is reflected in the short-time spectral envelope of the speech signal. The rate of change of the vocal tract shape appears to be important for the identification of linguistic components. This rate of change, or the rate of change of the short-time spectral envelope can be described by the *modulation spectrum*, i.e. the spectrum of the time trajectories described by the short-time spectral envelope.

For a wide range of frequency bands, the modulation spectrum of speech exhibits a maximum at about 4 Hz, the average syllabic rate. Disturbances often have modulation

frequency components outside the speech range, and could in principle be attenuated without significantly affecting the range with relevant linguistic information.

Early efforts for exploiting the modulation spectrum domain (temporal processing), such as the dynamic cepstrum or the RASTA processing, used ad hoc designed processing and appear to be suboptimal. As a major contribution, in this dissertation we aim for a systematic data-driven design of temporal processing.

First we analytically derive and discuss some properties and merits of temporal processing for speech signals. We attempt to formalize the concept and provide a theoretical background which has been lacking in the field. In the experimental part we apply temporal processing to a number of problems including adaptive noise reduction in cellular telephone environments, reduction of reverberation for speech enhancement, and improvements on automatic recognition of speech degraded by linear distortions and reverberation.

Chapter 1

Introduction

Speech is one of the most complex means of human communication. It involves several stages, from the coding of an idea in the transmitter's brain, to its successful decoding by the receiver. In this mode of human communication, the acoustic signal at the output of the speech production system is the carrier of the message. The evolution of this signal has been influenced by the physical properties of both, the production system, and the perception apparatus in charge of decoding the message.

The signal carrying the message is often corrupted by environmental agents during its transmission. Such factors could be other sound sources (noise), wave reflections (reverberation, echoes), linear and non-linear distortions introduced by the transmission medium, etc. If the signal is further converted to an electrical signal and sent through a communication link, degradations may include electronic noise, electromagnetic interference, distortion and noise introduced by the signal processing, etc. All of these problems will in general degrade the message retrieving performance of the receiver.

The topic of this dissertation is the manipulation of the speech signal to reduce the adverse effects that the communication environment has on the ability of the receiver (human or machine) to successfully decode the message. The type of processing that we will study is intimately related to the nature of the speech signal. Our objective is to describe this processing accurately, and show a few applications for which it has provided good results and/or increased our understanding of the technique.

We begin by motivating the study of speech processing in general, and give some background on the main areas in which we are interested.

1.1 Speech Processing Applications

Since the initial development of voice telecommunication systems, there has been an interest in eliminating agents that impair remote human communication. The large amount of resources devoted to solve this problem by the telephone industry and the military, among others, has resulted in a rapid development of the speech signal processing area.

In our modern capitalist society, service quality is strongly related to the success of telecommunication companies who compete against each other in the market. Any improvement in delivering a cleaner signal will result in benefits for both, the customers and the service providers. To cite some other less money oriented applications, in the area of prosthetics, hearing impaired individuals would also greatly benefit from the development of signal processing algorithms for hearing aids that compensate for their hearing deficiencies. However, current hearing aids experience problems in the presence of room reverberation, background noise, and competing speakers.

The rapid advance of speech recognition technology has created needs for new speech processing algorithms. Machines, lacking human capabilities, are even more vulnerable to environmental factors (with the state-of-the-art speech recognition systems available). Thus any advance in making machines more reliable in real environments will greatly benefit many applications.

1.2 Relevant Background

Short-Time Analysis

Speech conveys the message in a sequential fashion. The frequency distribution of the speech signal changes in time rendering it a non-stationary signal. Given this non-stationarity, traditional speech analysis techniques segment the signal at time intervals over which it can be assumed to be stationary. In this way, powerful analysis and modeling procedures developed for stationary signals can be applied to these short intervals.

Modulation Spectrum

This particular segmentation in time produces a two dimensional signal, where each time segment is analyzed and/or modeled and is represented by a feature vector, for example a frequency representation [14]. Thus, each component of this feature vector varies in time, according to the changes of the speech signal, describing a time trajectory. The spectral components of a time trajectory constitute its *modulation spectrum.*

Effect of Adverse Environments

Adverse environmental agents, such as additive noise, may have different modulation spectrum properties than speech. Also, transmission media such as microphones, enclosures and communication channels in general modify the modulation spectrum properties in ways that may impair intelligibility for humans, or affect the performance of actual automatic speech recognizers. This suggests that processing time trajectories of degraded speech could reduce the detrimental effects of the adverse environments in human-human and human-computer communications applications.

Processing Strategy

The contribution of this work is the processing of the temporal dimension of the time-feature representation of the speech signal. The processing involves linear filtering of the time trajectories of speech features. We show that for different applications, the appropriate feature space is different, possibly involving non-linear transformations, thus effectively making the overall processing non-linear.

The originality and importance of this contribution is the fact that the time trajectory filters are designed from training data. As we show, this design procedure has its value not only in optimizing the parameters of a system, but has provided us with insights about the temporal properties of speech.

1.3 Outline

This dissertation is divided into two major sections. In the first one, composed of Chapter 2, Chapter 3, Chapter 4, and Chapter 5, we develop the theory necessary to understand and design speech processing algorithms based on temporal processing. The second part of the dissertation contains Chapter 6, Chapter 7, Chapter 8, and Chapter 9, which describe applications of temporal processing to different speech communication problems.

Chapter 2 contains a review of well known properties of the short-time analysis of signals. This first discussion will introduce the necessary notation and fundamental concepts of the short-time domain. In Chapter 3 and Chapter 4 we perform a detailed analysis of the temporal processing procedures which are the main topic of the dissertation. The analysis is based on the time domain formulation of the short-time transform, and requires only simple algebraic manipulations and well-known linear systems theory concepts. We mainly show that when temporal processing is applied to time trajectories that have been modified by a non-linear operation, an equivalent time-domain formulation does not exist. In Chapter 5 we present an analysis of the effects that a convolutional distortion has on the short-time transform of a signal. This will be useful when we discuss the channel normalization applications in the second part of the dissertation. We also describe the principles under which traditional channel normalization techniques work.

The second part of the work describes a series of applications of the data-driven temporal processing approach that we investigated. We demonstrate a data-driven technique for temporal filter design (Chapter 8), and a multiresolution normalization technique for reducing the effects of reverberation in automatic speech recognition (ASR) (Chapter 9). For speech enhancement we present a chapter (Chapter 6) on additive background noise reduction for cellular telephone communications, and one on reverberation reduction (Chapter 7). We conclude the dissertation with Chapter 10, where we discuss our contributions and possible research directions for the future.

Chapter 2

Review of Short-Time Analysis of Signals

In this chapter we review some basic concepts of the two-dimensional representation of signals and short-time analysis. First we introduce the computation of a two-dimensional signal representation. We look at the particular case where the representation is of the time-frequency type, specifically the short-time Fourier transform (STFT) and define the *time trajectory* concept. Then we briefly discuss the computation of other time-feature representations commonly used in speech processing and their relation to the STFT.

In the following analysis we refer particularly to speech signals, but it should be understood that the concepts are more general and can be applied to other signals.

2.1 Time-Frequency Representation of Signals

The acoustic speech waveform can be described as a sound pressure-versus-time signal. Given that the spectral properties of this signal vary with time, we wish to obtain shorter segments and analyze them separately to find what are the properties in each segment, and how they change from segment to segment. This segmentation operation can be described as looking at the signal through a sliding window as shown in Fig. 2.1. The segmented speech can be written as

$$s_w(n, m) = w(n - m)s(m). \tag{2.1}$$

In (2.1) $s(n)$ is the sampled speech signal and $w(n)$ is the window function, which has been assumed to be symmetric. The fixed observation time is n and the running time is

m. Throughout this dissertation we will use sampled signals and discrete-time/discrete-frequency signal processing for our experiments and implementations. Only for convenience is the following analysis carried out in the continuous frequency domain.

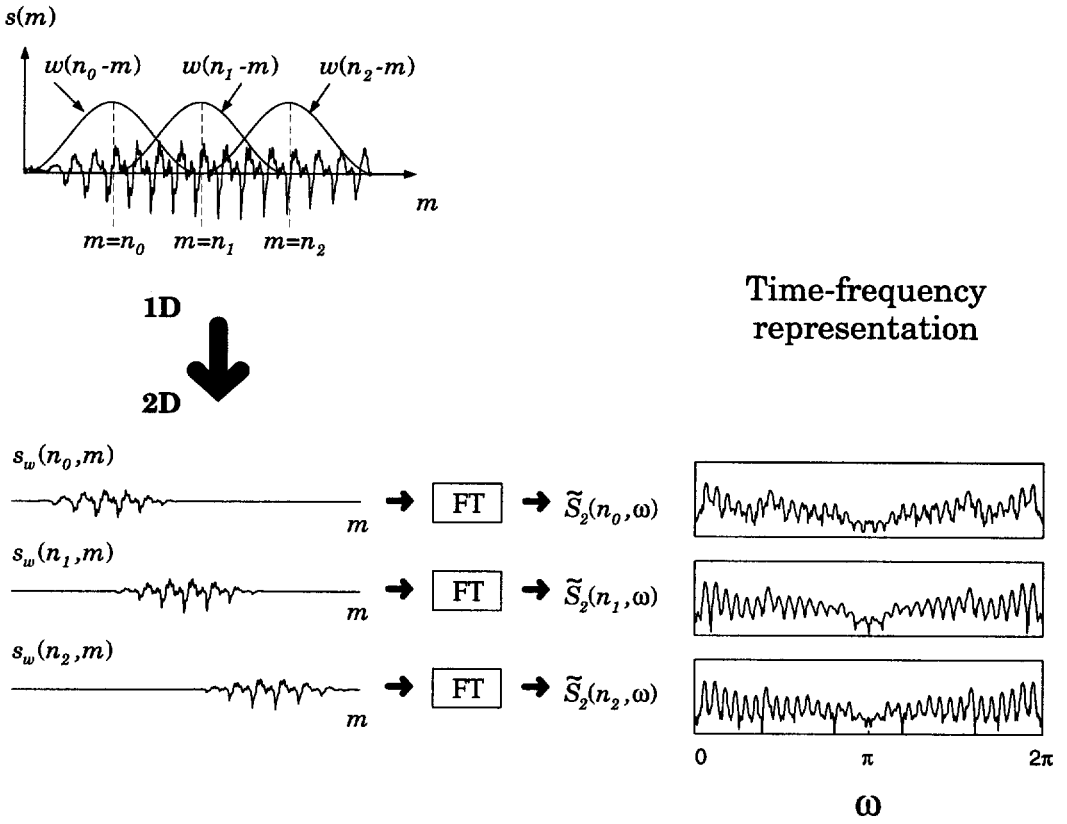


Figure 2.1: *Two-dimensional representation of a signal. As an example of a time-frequency representation, the short-time power spectrum is also depicted.*

If we describe $s(n)$ by a two-dimensional discrete-time sequence as in (2.1), we can obtain a frequency representation with respect to each of the time indices m and n. As in [49], applying the Fourier transform (FT) in each dimension (with respect to both time indices) we obtain the two-dimensional transform

$$\mathcal{S}(\theta,\omega) = \sum_{n=-\infty}^{\infty} \sum_{m=-\infty}^{\infty} s_w(n,m) e^{-j(\theta n+\omega m)} \tag{2.2}$$

where we assumed that the infinite summations converge. Applying the double inverse Fourier transform to (2.2) we obtain the inverse

$$s_w(n,m) = \frac{1}{(2\pi)^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \mathcal{S}(\theta,\omega) e^{j(\theta n + \omega m)} d\omega d\theta. \tag{2.3}$$

Throughout this dissertation we will be describing one-dimensional signals by two-dimensional representations. The following definitions formalize the treatment of such representations, and the interpretation of the equations will be given as we encounter them along our analysis.

Since the windowed signal (2.1) is two-dimensional, we can obtain its FT with respect to each time index. The FT of (2.1) with respect to the fixed time n can be written as

$$S_1(\theta,m) = \sum_{n=-\infty}^{\infty} s_w(n,m) e^{-j\theta n}, \tag{2.4}$$

with inverse

$$s_w(n,m) = \frac{1}{2\pi} \int_{-\pi}^{\pi} S_1(\theta,m) e^{j\theta n} d\theta. \tag{2.5}$$

In (2.4) the subindex 1 in $S_1(\theta,m)$ indicates that the transform was applied with respect to the first argument (i.e. time index n) of $s_w(n,m)$. By taking the Fourier transform of (2.1) with respect to the running time m, we obtain the frequency response of each time segment (indexed by fixed time n),

$$S_2(n,\omega) = \sum_{m=-\infty}^{\infty} s_w(n,m) e^{-j\omega m}, \tag{2.6}$$

with inverse

$$s_w(n,m) = \frac{1}{2\pi} \int_{-\pi}^{\pi} S_2(n,\omega) e^{j\omega m} d\omega, \tag{2.7}$$

where $S_2(n,\omega)$ is the one-dimensional transform with respect to the second argument of $s_w(n,m)$ (time index m). Given the previous definitions, the two-dimensional (or complete) transform can be obtained from the partial transforms (2.4) and (2.6) as

$$\mathcal{S}(\theta,\omega) = \sum_{n=-\infty}^{\infty} S_2(n,\omega) e^{-j\theta n} = \sum_{m=-\infty}^{\infty} S_1(\theta,m) e^{-j\omega m}. \tag{2.8}$$

The original signal $s(n)$ can be recovered from the complete or partial transforms. First, using the inverse transforms (2.3), (2.5), or (2.7) we can obtain the windowed signal

$s_w(n,m)$, and evaluating this two-dimensional signal at time $m = n$ we can recover $s(n)$ (within a scalar factor), i.e.

$$s_w(n,m)|_{m=n} = w(0)s(n) = s(n), \quad \text{for } w(0) = 1. \tag{2.9}$$

It is evident that in order to recover the original signal $s(n)$ from the two-dimensional representations we need to impose a constraint on the analysis window, namely $w(0) \neq 0$. Equation (2.9) is not the only way of recovering $s(n)$. The reader is referred to [49] for alternative inversion formulas.

2.1.1 Relation to the Fourier Transform

A relationship between the two-dimensional transform (2.2) with the Fourier transforms of the signal and window function, $S(\omega)$ and $W(\omega)$ respectively[1], can be obtained. Substituting $w(n-m)$ by its Fourier integral in the definition of $s_w(n,m)$ (equation (2.1)) we get

$$s_w(n,m) = \sum_{m=-\infty}^{\infty} \frac{1}{2\pi} \int_{-\pi}^{\pi} W(\theta) e^{j\theta(n-m)} s(m) d\theta, \tag{2.10}$$

and introducing this expanded form into (2.6), yields

$$S_2(n,\omega) = \frac{1}{2\pi} \int_{-\pi}^{\pi} S(\theta+\omega) W(\theta) e^{j\theta n} d\theta. \tag{2.11}$$

Recognizing the partial transform of (2.11) with respect to n, we obtain the relationship

$$\mathcal{S}(\theta,\omega) = S(\theta+\omega)W(\theta). \tag{2.12}$$

We observe from (2.12) the duality between the time domain sliding window concept, and a frequency domain sliding window interpretation, and both being inverse transforms of each other.

[1]Do not confuse the Fourier transforms with the short-time functions which are functions of two variables.

2.1.2 Discussion

Before continuing our analysis, an intuitive interpretation of (2.2) and the partial transforms, and their implications for speech processing will be given.

From the previous analysis we can immediately recognize a time-frequency representation (2.6) which has been extensively used in signal processing. The pair (2.6) and (2.7) describes the well-known short-time Fourier transform (STFT) ([53], [3]). The usefulness of this transform is mainly observed in the frequency analysis of signals with time-varying spectra [14], such as speech and most signals in nature. The time span over which the spectrum of a time-varying signal can be considered stationary will determine the time duration of the window $w(n)$ and consequently the frequency resolution of the representation. It is also well documented that the STFT is not the only time-frequency representation for speech. Depending on the specific requirements of the analysis, different time-frequency representations are available [14].

2.1.3 Filter Bank Interpretation of the STFT

The STFT can also be interpreted in terms of a filter bank [15]. This is clearly seen if, with aid of (2.1), we write (2.6) as the convolution sum

$$S_2(n,\omega) = \sum_{m=-\infty}^{\infty} w(n-m)s(m)e^{-j\omega m} = w(n) *_n s(n)e^{-j\omega n}, \tag{2.13}$$

where the $*_n$ operator is the linear convolution with respect to time index n.

If we visualize the continuous frequency domain ω as an infinite set of frequency bands, the output corresponding to each band describes a time sequence that is obtained by multiplying the signal $s(n)$ by a complex exponential function with frequency ω, and applying the low-pass filter with impulse response $w(n)$ to the product $s(n)e^{-j\omega n}$. In Fig. 2.2 we show the equivalent operation for an arbitrary frequency band.

We say that the time sequence at the output of the filter is the *time trajectory* at that particular frequency band (i.e. $S_2(n,\omega_k)$). The time trajectory is then obtained by evaluating the STFT at the desired frequency band. In this case the time trajectory is a complex sequence which describes the time evolution of the k^{th} spectral component.

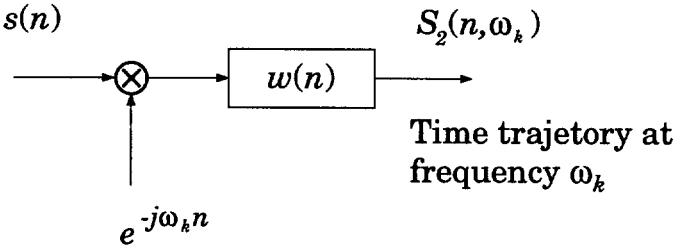


Figure 2.2: *Filter bank interpretation of the STFT*

The Modulation Frequency Concept

Now, if we keep in mind the filter bank point of view, The two-dimensional FT (2.2) can be interpreted as a frequency analysis on the outputs of the filter bank, i.e. the time trajectories. The frequency domain described by the variable θ is often referred as *modulation frequency*, and the power spectrum related to this domain as *modulation spectrum* [29]. In this dissertation we will use these terms whenever we refer to θ. As will be seen later, the modulation spectrum of speech has some particular properties which we will exploit for enhancing degraded speech in different applications.

2.2 Time-Feature Representations of Speech

In the previous section we described a particular time-feature representation of a signal. The feature in that case was the frequency spectrum, and the time-feature representation (i.e. the STFT) described how this feature varies with time. In the speech processing field, other features (described below) have been used for different applications [50].

As shown in [3], the STFT is a complete description of a time signal in the sense that the signal can be exactly recovered from its STFT by imposing only a few constraints during the analysis (e.g. $w(0) \neq 0$). However, for some applications one may be interested in only a few aspects of the speech signal. For example, in speech coding, where the goal is to describe a speech signal with as few parameters as possible, features like short-time spectral envelope (represented by e.g. linear predictive coding (LPC) coefficients), frame voicing, and frame pitch may be enough to describe speech in a useful way [6],

[5]. In other applications like automatic speech recognition (ASR), short-time parameters containing relevant linguistic information are required. Parameters commonly encountered in that field are the short-time LPC-cepstrum [5], mel-cepstrum [16], and perceptual linear prediction (PLP) coefficients [23].

Preprocessing of speech for noise reduction and/or channel normalization for ASR, like RelAtive SpectrAl (RASTA) processing [24] or cepstral mean subtraction (CMS) [51], involves applying linear filtering operations on some non-linear short-time feature domain. Examples of these features for ASR are the logarithm of the short-time spectrum, short-time cepstrum, mel cepstrum, PLP cepstrum, LPC cepstrum, etc. In speech enhancement, processing may be applied to the magnitude or some non-linear transformation of the magnitude of the STFT. For example in the spectral magnitude estimation for speech enhancement [35], [26].

Many of the short-time features previously mentioned can be derived from the STFT. For example, the critical band analyses involved in the mel cepstrum and PLP features consist of performing a weighted sum of the short-time power spectrum components. LPC parameters can also be efficiently computed by using the short-time power spectrum to estimate the short-time autocorrelation function [37].

In this dissertation we will be applying modifications to the time dimension of some of the time-feature representations discussed above. The features used will depend on the particular application.

Chapter 3

Temporal Processing

One way of modifying the modulation spectrum is filtering the *time trajectories* of speech features. In this chapter we present a formal treatment of temporal processing, i.e. processing of the time trajectories of a signal. This procedure will be described in detail and some of its properties will be derived.

Filtering of time trajectories has been applied in the past. However, to the best of our knowledge, a rigorous analysis of its properties does not exist. As one of the original contributions in this dissertation, we present a formal analysis of temporal processing, and show that filtering the time-trajectories in linear domains is a general case of other short-time modifications analyzed in the past [3], [49]. The results obtained will reveal some properties of this processing, and the existence of an equivalent time-domain linear filter.

3.1 Filtering of the Time Trajectories

Filtering the time trajectories of speech features is not a new concept. Blind deconvolution proposed by Stockham [57], and cepstral mean removal techniques in ASR have been quite successful [51]. These techniques are equivalent to filtering operations on the time trajectories of cepstral features. More recently, Hermansky and Morgan have applied bandpass filtering to the temporal dimension of logarithmic features [24] (A more detailed description of this technique will be given in Chapter 5). Hirsch has used used high-pass filtering in the trajectories of the short-time power spectrum to reduce reverberation [27].

In the area of speech enhancement, Langhans and Strube [33] applied temporal processing to additive noise and reverberation problems with limited success.

In this dissertation we will describe the filtering of time trajectories of different features depending on the particular application. In contrast with previous works (e.g. RASTA processing) that use ad-hoc designed filters, we use automatic data-driven filter design techniques. As will be discussed in later chapters, the optimization of the parameters of a system with the data-driven approach also provides insights about the speech signal properties under different adverse conditions.

3.2 CIT-MIF Modification of the Short-Time Spectrum

Modification of the short-time spectrum of speech has been previously studied in [1], [3], and [49]. In those contributions, fixed and time-varying multiplicative-in-frequency (MIF) modifications have been applied to the short-time spectrum. However, filtering of time trajectories has not been studied. In this section we derive the results for a convolutional-in-time and multiplicative-in-frequency (CIT-MIF) modification. The convolutional-in-time modification refers to the filtering along the time dimension of the short-time transform, while the multiplicative-in-frequency part indicates the general case in which different time trajectory filters can be applied at different frequency bands.

For simplicity, the analysis is initially performed on modifications to the short-time spectrum. A more relevant (to this work) case where the filtering is applied to other speech features will be discussed in Chapter 4.

3.2.1 Description of the CIT-MIF Modifications

The modification of the frequency and modulation frequency components of a signal (in the sense of weighting the components), can be described in terms of applying a multiplicative modification $\mathcal{F}(\theta,\omega)$ in the double transform domain, i.e.

$$\mathcal{Y}(\theta,\omega) = \mathcal{F}(\theta,\omega)\mathcal{S}(\theta,\omega) \tag{3.1}$$

The modification in (ref2Dmod:eq) can be written as a filtering operation (convolution)

in the fixed time domain n. The partial transform with respect to ω of (3.1) can be obtained by integrating with respect to θ and using the identity (2.8), thus obtaining

$$Y_2(n,\omega) = \sum_{r=-\infty}^{\infty} F_2(n-r,\omega)S_2(r,\omega) = F_2(n,\omega) *_n S_2(n,\omega). \tag{3.2}$$

Equation (3.2) represents the CIT-MIF modification of the short-time spectrum (observe that the time dimension of the short-time transforms is convolved, while the frequency dimension is multiplied). We adopted this terminology to indicate the specific operation upon the STFT, and not to indicate the effect that the modifications have on it. Both dimensions, time and frequency, are intimately related in the STFT, and modifications on one will result in modifications in the other.

We will also refer to the CIT-MIF modification as filtering of the time trajectories (or temporal filtering), and we will refer to $F_2(n,\omega)$ as the time trajectory filters. Whenever $F_2(n,\omega)$ becomes a function of time only, i.e. $F_2(n,\omega) = F_2(n)$, we will refer to it as a CIT-only modification.

3.2.2 Synthesis from the STFT

The time domain effects of STFT modifications will in general depend on the synthesis formula used to obtain a time domain signal [3]. A general synthesis formula which makes use of a synthesis window was derived by Portnoff in [49]. The two commonly used synthesis procedures, the overlap-add (OLA) and filter bank summation (FBS), are particular cases of Portnoff's formula. For the purposes of completeness we derive the time domain expressions for the general case and later show the particular results when the synthesis methods are the FBS and OLA.

Portnoff's time-invariant synthesis formula is written as

$$y(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi} \sum_{l=-\infty}^{\infty} q(n-l)Y_2(l,\omega)e^{j\omega n}d\omega, \tag{3.3}$$

where $q(n)$ is the synthesis window. In the FBS synthesis method the synthesis window is a unit sample (delta) function, $q(n) = \delta(n)$ and the synthesis equation becomes

$$y(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} Y_2(n,\omega) e^{j\omega n} d\omega. \tag{3.4}$$

For the OLA synthesis method, the synthesis window becomes $q(n) = \frac{1}{W(0)}$, where $W(0)$ is the dc response of the analysis window and (3.3) becomes

$$y(n) = \frac{1}{2\pi W(0)} \int_{-\pi}^{\pi} \sum_{l=-\infty}^{\infty} Y_2(l,\omega) e^{j\omega n} d\omega. \tag{3.5}$$

3.2.3 Time Domain Effects of CIT-MIF Modifications

To see the effect of the proposed CIT-MIF modification on the time domain, we resynthesize the signal after modifying the STFT. Introducing the modified STFT (3.2) into Portnoff's synthesis formula (3.3) we get

$$y(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \sum_{l=-\infty}^{\infty} q(n-l) \sum_{r=-\infty}^{\infty} F_2(l-r,\omega) S_2(r,\omega) e^{j\omega n} d\omega, \tag{3.6}$$

which can be simplified to (see appendix A for a derivation of this result)

$$y(n) = \sum_{m=-\infty}^{\infty} s(n-m)\tilde{f}(m) = s(n) * \tilde{f}(n) \tag{3.7}$$

where

$$\tilde{f}(n) = \sum_{r=-\infty}^{\infty} w(n-r) \sum_{l=-\infty}^{\infty} q(l) f(r-l,n), \tag{3.8}$$

and

$$f(n,m) = \frac{1}{2\pi} \int_{-\pi}^{\pi} F_2(n,\omega) e^{j\omega m} d\omega. \tag{3.9}$$

From (3.7) we see that the time domain equivalent of filtering the time trajectories is the convolution of the input sequence with a time-invariant filter. For an arbitrary modification $F_2(n,w)$ of the STFT, the time domain equivalent filter will be constrained by the analysis and synthesis windows used. This can be seen in (3.8), where both windows are convolved with the ISTFT (3.9) of the modification $F_2(n,\omega)$.

The result in (3.7) suggests that this method is equivalent to filtering the original signal in the time domain. However, depending on the synthesis method used, the constraints

on the time domain equivalent will be different and consequently the system design considerations will differ. Similar constraints for MIF-only modifications have been shown to exist in [49] and [3].

3.2.4 Filter Bank Interpretation

Even though (3.7) is the correct time domain formulation for CIT-MIF modifications of the STFT, an alternative and more intuitive explanation with respect to the *time trajectory filters* can be derived by using the filter bank interpretation (2.13) of the STFT.

To visualize the filter bank consider again an infinite number of frequency points ω_k indexed by k so that we can exchange the inverse FT integral for a summation over all k. In this way the modification $F_2(n,\omega)$ becomes $F_2(n,\omega_k)$ which can be interpreted as a set of time trajectory filters, each operating on a frequency band with center frequency ω_k.

With the above considerations the general synthesis equation (3.6) becomes

$$y(n) = \sum_k \sum_{l=-\infty}^{\infty} q(n-l) \sum_{r=-\infty}^{\infty} F_2(l-r,\omega_k) S_2(r,\omega_k) e^{j\omega_k n}, \tag{3.10}$$

and introducing the STFT definition (2.6) we can write

$$y(n) = \sum_k \sum_{l=-\infty}^{\infty} q(n-l) \sum_{r=-\infty}^{\infty} F_2(l-r,\omega_k) \sum_{m=-\infty}^{\infty} w(r-m) s(m) e^{j\omega_k (n-m)}, \tag{3.11}$$

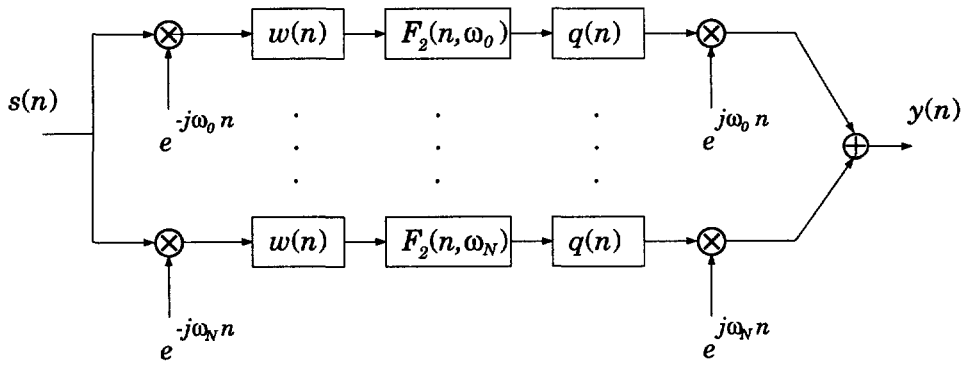
which after some manipulation can be rearranged into the form

$$y(n) = \sum_{m=-\infty}^{\infty} s(n-m) \sum_k \left[\sum_{l=-\infty}^{\infty} q(l) \sum_{r=-\infty}^{\infty} w(m-r-l) F_2(r,\omega_k) e^{j\omega_k m} \right] \tag{3.12}$$

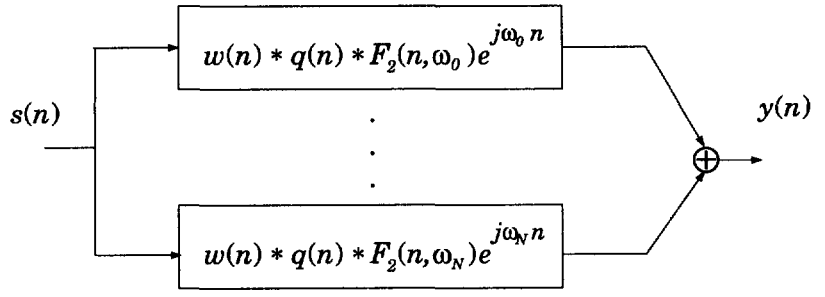
$$= s(n) *_n \left[\sum_k q(n) *_n w(n) *_n F_2(n,\omega_k) e^{j\omega_k n} \right].$$

In this form the effect of the synthesis in the time trajectory filters and on the time domain signal can be interpreted. In Fig. 3.1 we show a graphical description of the filter bank interpretation of (3.12).

As was seen in (3.7), the time domain effect of the CIT-MIF modification is an equivalent linear time-invariant filter $\tilde{f}(n)$. The analysis in (3.12) shows that this filter is the sum of bandpass filters whose base-band impulse response is given by "time-smeared" versions



(a)



(b)

Figure 3.1: *(a) filter bank interpretation of temporal processing. (b) equivalent system*

of the time trajectory filters $F_2(n,\omega_k)$ (see Fig. 3.1). Obviously the smearing depends on the analysis and synthesis windows used. For the FBS and OLA synthesis methods the effect is described as follows.

Modification Constraints in the FBS Synthesis Method

Recall that for the FBS method the synthesis window is a delta function and the synthesis equation is reduced to (3.4). If we let $q(n) = \delta(n)$ in (3.12) we obtain

$$y(n) = s(n) *_n \left[\sum_k w(n) *_n F_2(n,\omega_k) e^{j\omega_k n} \right]. \tag{3.13}$$

A simple block diagram interpretation of this result is shown in Fig. 3.2. For arbitrary

time trajectory filters $F_2(n, \omega_k)$, the modulation frequency range of the modifications will be determined by the analysis window bandwidth.

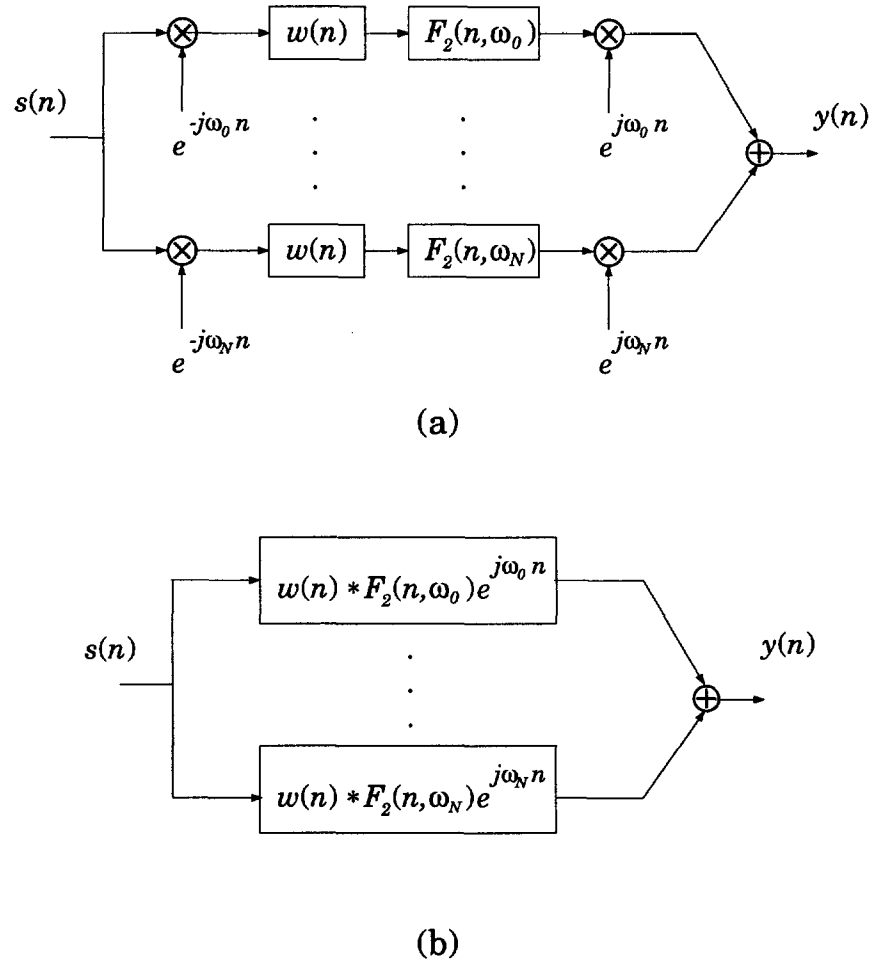


Figure 3.2: *(a) filter bank interpretation of temporal processing in the FBS method. (b) equivalent system*

The analysis window determines the bandwidth of each band of the STFT [3]. This means that the modulation frequency range over which the modifications can be performed is maximum for the FBS method.

Now we can see that the advantage of time trajectory filtering is that if the impulse response of the time trajectory filter is allowed to be longer than the analysis window length, additional modulation frequency resolution can be gained. This means that the

modulation frequency modifications can be made with any detail by just setting the appropriate filter length. The trade-off is of course bounded by Heisenberg's inequality [14] since obtaining higher modulation frequency resolution implies that more time information has to be accounted for in the time trajectory filtering operation, i.e. longer time trajectory filters.

In the case studied in this chapter, where the CIT-MIF modifications are applied directly to the STFT, the advantage of temporal processing over time-domain filtering is not obvious. The same modulation frequency modifications can be obtained by applying a long filter in the time domain (see equation (3.7)). However, in the next chapter, where we deal with non-linear transformations, we will show how temporal processing is indeed advantageous.

Modification Constraints in the OLA Synthesis Method

In the OLA synthesis case, the synthesis window is a constant (or rectangular window) as in (3.5), so (3.12) can be written as

$$y(n) = \frac{1}{W(0)} \sum_{m=-\infty}^{\infty} s(n-m) \left[\sum_{k} \sum_{l=-\infty}^{\infty} \sum_{r=-\infty}^{\infty} w(m-r-l) F_2(r, \omega_k) e^{j\omega_k m} \right], \qquad (3.14)$$

and we observe that there exists a smearing (given by the summation over r) due to the analysis filter as in the FBS method, but an additional smearing is introduced which depends on the properties of the analysis window. In practice, the analysis window has finite length so we can think about this additional smearing in terms of a rectangular synthesis window of the same length as $w(n)$. In this case the summation over l in (3.14) is finite and the additional time-smearing on the time trajectory filters will be solely determined by the analysis window length. This is in contrast with the FBS method, where the smearing depends on the bandwidth of $w(n)$.

3.2.5 Discussion

The range of modulation frequencies over which modifications can be made is thus reduced in the OLA case compared to the FBS method. In this sense we may be inclined to use

the FBS synthesis. On the other hand, the OLA method can be extended to the more general weighted overlap-add (WOLA) [15] where a synthesis window is multiplied with the reconstructed segments before overlap-adding. In this case, proper choice of the synthesis window, i.e. having a bandwidth comparable to that of the analysis window, will allow us to overcome the modulation bandwidth constraints imposed by OLA.

Moreover, the importance of using a synthesis window when STFT modifications have occurred has been pointed by Griffin and Lim [21]. They proposed that the synthesis window be the same as the analysis window, i.e. $q(n) = w(n)$, for which only some simple design constraints have to be imposed.

For implementation purposes, the OLA and WOLA method offer advantages over efficient FBS implementations, like helical interpolation [42], in terms of simplicity and storage requirements. Following the previous discussion the WOLA synthesis seems to be the appropriate method if full advantage of temporal processing is desired. Throughout the work leading to this dissertation we found that the OLA and WOLA methods seem to have similar performance for the speech processing applications that we explored. A reason for this will become apparent when we look at the properties of the time trajectory filters that we applied.

3.3 Summary

In this chapter we analyzed the particular case when the STFT is modified by applying a filtering operation to its time trajectories. We have called that operation a CIT-MIF modification given that the filters operate along the time dimension of the STFT in a convolutional way, and weight the frequency dimension in a multiplicative manner.

Time domain equivalents for filtering the time trajectories of the STFT have been found for different synthesis methods. We described how the synthesis method might constrain the properties of the resulting resynthesized signal. The results found are consistent with those obtained for other types of STFT modifications which can be considered to be special cases of the CIT-MIF (see [49] and [3]). In the next chapter we consider the case when temporal processing is applied to non-linear transformations of the STFT.

Chapter 4

Temporal Processing in Non-Linear Domains

In the previous chapter we described temporal processing in the STFT representation of signals. As we showed, filtering time trajectories in the short-time frequency domain has an interpretation in the time domain. Even when the action of the time-trajectory filters is restricted by the analysis/synthesis parameters, we can in principle implement the filtering scheme by proper design of an equivalent linear time-invariant filter $\tilde{f}(n)$ (see equation (3.7)).

As is common in many speech processing applications, modifications of the STFTM are often done in some non-linear domain. Short-time spectral estimators for speech enhancement have been successfully applied in non-linear functions of the spectrum such as square-root, logarithm and square law [48]. Homomorphic filtering or deconvolution techniques require non-linear domains such as the logarithmic power spectrum or the cepstrum [46], [57]. Other homomorphic deconvolution systems use power laws [34].

Continuing our contribution to the analysis of temporal processing, in this section we find that when the processing is applied to a non-linear transform of the STFT, the equivalent time domain filter is not easily found. In fact we show that the time domain equivalent operation is time-varying and STFT dependent, even for simple non-linear transforms like the STFT magnitude.

4.1 Temporal Processing of the STFTM

We begin our study by considering the common case where only the magnitude of the STFT (STFTM) is processed and the STFT phase (STFTP) is left unmodified. The motivation behind this restriction is that the relevant perceptual attributes of speech are considered to be included mainly in the STFTM rather than in the STFTP [35], [62]. Processing of the STFTM has been extensively applied in several areas of speech processing such as speech enhancement [13], time-scale modification of speech [52], and speech coding [18].

Another important reason for not modifying the STFTP is that it is not bounded if looked at as a time signal [18], and this behavior may not make it suitable for filtering or other time dependent modifications. Furthermore, STFTP modifications may result in destruction of the pitch structure of the resynthesized speech [52].

4.1.1 Definitions of STFTM and STFTP

We start by formalizing the definitions for the STFTM and STFTP. The STFT is a complex signal in its second argument and can also be written in terms of its real and imaginary parts [18]

$$S_2(n,\omega) = a(n,\omega) + jb(n,\omega), \tag{4.1}$$

and in terms of polar coordinates as

$$S_2(n,\omega) = |S_2(n,\omega)|e^{j\phi(n,\omega)}, \tag{4.2}$$

where

$$|S_2(n,\omega)| = \sqrt{a^2(n,\omega) + b^2(n,\omega)}, \tag{4.3}$$

and

$$\angle S_2(n,\omega) = \phi(n,\omega) = \tan^{-1}\left[\frac{b(n,\omega)}{a(n,\omega)}\right]. \tag{4.4}$$

The magnitude and phase just defined above are also two dimensional signals and their treatment should follow the rules that we formalized in Chapter 2 and Chapter 3.

4.1.2 CIT-MIF Modification of the STFTM

Now we begin investigating what is the time equivalent, if it exists, of applying a CIT-MIF modification to the STFTM. This is an important issue since it will help us to determine if the STFTM domain transformation of a signal is indeed necessary for implementing the desired CIT-MIF operation. The following analysis will also make evident some further complications that arise when we wish to process some non-linear transform of the STFTM, such as the short-time power spectrum or the logarithmic short-time spectrum (see section 4.2).

If the CIT-MIF modification is applied only to the time trajectories of $|S_2(n,\omega)|$, then the modified STFT $Y_2(n,\omega)$ can be written in terms of its magnitude and the original phase $\phi(n,\omega)$ as

$$Y_2(n,\omega) = |Y_2(n,\omega)|\, e^{j\phi(n,\omega)}. \tag{4.5}$$

with magnitude

$$|Y_2(n,\omega)| = \sum_{r=-\infty}^{\infty} F_2(n-r,\omega)\, |S_2(r,\omega)|, \tag{4.6}$$

In equation (4.6) we have assumed that the filtered STFTM is a valid magnitude, i.e. $|Y_2(n,\omega)| \geq 0$. In general there is no guarantee that negative numbers will not result from the time trajectory filtering operation. In practice it is common to set negative values to zero or take the absolute value of the right-hand side of (4.6) [35]. For purposes of simplifying our analysis we will assume that $|Y_2(n,\omega)|$ is a valid magnitude.

To resynthesize a signal from the filtered STFTM we apply a synthesis equation, e.g. (3.4), to (4.5) to obtain

$$y(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \sum_{r=-\infty}^{\infty} F_2(n-r,\omega)|S_2(r,\omega)|e^{j\phi(n,\omega)}e^{j\omega n} d\omega, \tag{4.7}$$

where we have again assumed that the filtered STFTM is a valid magnitude function.

The FBS synthesis method is used in (4.7) only for simplicity and to illustrate our point. A similar analysis can be carried out with OLA or WOLA methods yielding similar results and with the additional "smearing" effects that we have previously described.

Time Domain Equivalent

Since our aim is to find a time domain equivalent of filtering the STFTM, we would like to express (4.7) in terms of the input signal $s(n)$. To achieve this it is necessary to first obtain an expression in terms of the STFT $S_2(n,\omega)$. This can be accomplished by adding and subtracting a phase term $\phi(r,\omega)$ to (4.7), i.e.

$$y(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi}\sum_{r=-\infty}^{\infty} F_2(n-r,\omega)e^{j[\phi(n,w)-\phi(r,w)]}|S_2(r,\omega)|e^{j\phi(r,\omega)}e^{j\omega n}d\omega \tag{4.8}$$

$$= \frac{1}{2\pi}\int_{-\pi}^{\pi}\sum_{r=-\infty}^{\infty} F_2(n-r,\omega)e^{j[\phi(n,w)-\phi(r,w)]}S_2(r,\omega)e^{j\omega n}d\omega,$$

which expresses $y(n)$ in terms of the short-time transform $S_2(n,\omega)$.

In (4.8) we observe that the STFT is now being filtered by time-varying time trajectory filters which depend on the STFTP. We define these filters with the following notation:

$$f_\omega(n,r) = F_2(r,\omega)e^{j[\phi(n,\omega)-\phi(n-r,\omega)]}, \tag{4.9}$$

which can be interpreted as the time-varying filter response at time n to a unit sample applied r samples before [30]. With the introduction of this new notation (4.8) becomes

$$y(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi}\sum_{r=-\infty}^{\infty} f_\omega(n,n-r)S_2(r,\omega)e^{j\omega n}d\omega, \tag{4.10}$$

which is the time domain equivalent form that we were looking for.

Filter Bank Interpretation

As we did in section 3.2.4 we can use the filter bank interpretation to gain some more insight in terms of time trajectory filters. If we exchange the FT integral in (4.10) for a summation over k, then ω_k replaces ω and we to obtain

$$y(n) = \sum_k\sum_{r=-\infty}^{\infty} f_{\omega_k}(n,n-r)S_2(r,\omega_k)e^{j\omega_k n}. \tag{4.11}$$

It can be shown (see Appendix B for a derivation) that $y(n)$ can be expressed in terms of the convolution of the input signal $s(n)$ with the sum of a set of time-varying bandpass filters, i.e.

$$y(n) = \sum_{m=-\infty}^{\infty} s(n-m) \left[\sum_{k} g_{\omega_k}(n,m) e^{j\omega_k n} \right], \tag{4.12}$$

where the baseband time-varying filters are

$$g_{\omega_k}(n,m) = \sum_{r=-\infty}^{\infty} f_{\omega_k}(n,r) w(m-r). \tag{4.13}$$

Apparently we have arrived to a time domain equivalent of filtering the time trajectories of the STFTM. Note that even if we could afford the time-varying filtering operation described by (B.4) notice that the filters $g_{\omega_k}(n,m)$ are dependent on $\phi(n,\omega)$ (see (4.9)). The time-varying filtering operation implies that we know at least the phase of the STFT. So, for realizing the processing suggested in (4.12) we need to compute the STFT a priori. The result is not surprising given that we are constraining our processing to leave the STFTP unmodified. Furthermore, the previous analysis assumes that the STFTM remains a valid STFTM after it is filtered in its time dimension. If we were to assume otherwise, an extra absolute value, or rectification operation (procedures with no real justification) would have to be included which further complicate the time-domain equivalent analysis.

What the previous analysis clearly indicates is that when temporal processing in the STFT domain involves linear time-invariant filtering of the STFTM time trajectories, the time-domain equivalent implementation is not possible without prior knowledge of the original short-time transform and an implementation in the STFT domain is much simpler.

4.1.3 Phase Effects

We would also like to know what is the effect of a CIT-MIF modification of the STFTM on the phase of the resynthesized signal. Using the original the STFTP for resynthesis does not necessarily imply that undesired phase distortion will result in the time domain

signal. The following analysis is intended to give some insight into the problems that one may encounter in the design of time trajectory filters if a given time domain phase response is desired.

The Fourier transform $S(\omega)$ of a signal $s(n)$ can be obtained by evaluating its two-dimensional transform at modulation frequency $\theta = 0$, i.e.

$$\mathcal{S}(\theta,\omega)|_{\theta=0} = \mathcal{S}(0,\omega) = S(\omega)W(0), \tag{4.14}$$

thus

$$S(\omega) = \frac{1}{W(0)}\mathcal{S}(0,\omega), \tag{4.15}$$

which follow from (2.12). In terms of the STFT we can write

$$S(\omega) = \frac{1}{W(0)} \sum_{n=-\infty}^{\infty} S_2(n,\omega), \tag{4.16}$$

where we used the identity (2.8). The magnitude and phase of $S(\omega)$ are then

$$|S(\omega)| = \frac{1}{W(0)} \left| \sum_{n=-\infty}^{\infty} S_2(n,\omega) \right|, \tag{4.17}$$

and

$$\angle S(\omega) = \angle \left(\sum_{n=-\infty}^{\infty} S_2(n,\omega) \right). \tag{4.18}$$

Phase Effects in Resynthesis

Assuming that the modified short-time transform $Y_2(n,\omega)$ is valid in the sense that it has the properties of a STFT (see [49] and references therein for details) we can also obtain the FT of $y(n)$ as

$$Y(\omega) = \frac{Y(0,\omega)}{W(0)} = \frac{1}{W(0)} \sum_{n=-\infty}^{\infty} Y_2(n,\omega). \tag{4.19}$$

Recall from (4.5) that the modified STFT can be written in terms of the modified short-time magnitude and the original short-time phase. Since we are interested in looking at the phase of $Y(\omega)$, let us introduce (4.5) and (4.6) into (4.19) to obtain

$$Y(\omega) = \frac{1}{W(0)} \sum_{n=-\infty}^{\infty} \sum_{r=-\infty}^{\infty} F_2(n-r,\omega)\,|S_2(r,\omega)|\,e^{j\phi(n,\omega)}, \tag{4.20}$$

from which we can easily find the phase

$$\angle Y(\omega) = \angle \left(\sum_{n=-\infty}^{\infty} \left[\sum_{r=-\infty}^{\infty} F_2(n-r,\omega)\,|S_2(r,\omega)| \right] e^{j\phi(n,\omega)} \right). \tag{4.21}$$

Phase Distortion in the Time Domain

Now suppose that we would like to find out a constraint on the temporal processing such that no phase distortion will result in the resynthesized time domain signal. This constraint can be formally expressed as

$$\angle Y(\omega) = \angle S(\omega), \tag{4.22}$$

where $Y(\omega)$ is the FT of the resynthesized signal $y(n)$. Using the phase obtained in (4.21) and the phase term (4.18), the condition for no phase distortion in the time domain (4.22) can be expanded as

$$\angle \left(\sum_{n=-\infty}^{\infty} \left[\sum_{r=-\infty}^{\infty} F_2(n-r,\omega)\,|S_2(r,\omega)| \right] e^{j\phi(n,\omega)} \right) = \angle \left(\sum_{n=-\infty}^{\infty} S_2(n,\omega) \right). \tag{4.23}$$

One way of achieving this condition is to set the time trajectory filters to have the form $F_2(n,\omega) = \alpha_\omega \delta(n-\Delta)$, where α_ω is a real frequency weighting factor and Δ is a time delay factor. This form is just a scaling of the time trajectories (all-pass linear phase filters), and the modulation frequency modifications possible with this design are limited to a constant gain factor.

Other filter designs could meet condition (4.23), but the desired modulation frequency modifications would be harder to achieve with the introduction of the additional constraint. Moreover, we must know what the original STFTP is in order to design temporal filters that will introduce no phase distortion in the time domain. This is in accordance with the results previously obtained for the time domain equivalent of filtering the STFTM (see 4.12).

We might not always be interested in avoiding phase distortion in the time domain. An arbitrary phase response may be achieved by replacing the right-hand side of (4.23) with the desired phase. For this, the time trajectory filters would also have to be designed with prior knowledge of the STFTP of the signal.

In contrast, for time-domain filtering (e.g. Wiener filtering) the phase distortion restrictions are not as severe, i.e. the design does not depend on the signal. For example, FIR symmetric filters (generalized linear phase [45]) which introduce no phase distortion are easily designed and implemented. Also, arbitrary phase responses can be approximated by FIR or IIR time domain filters [55].

4.2 Temporal Processing in Other Non-Linear Domains

As we pointed out before, applying temporal processing in non-linear domains may have certain advantages. Recently, there has been an increasing interest in applying a type of homomorphic filtering technique known as RelAtive SpecTrAl (RASTA) processing to reduce channel effects in ASR [24]. RASTA does this by band-pass filtering time trajectories of parametric representations of speech in a domain in which the disturbing noisy components are additive (see Chapter 5). For convolutional noise, the representation is a logarithmic function of the short-time spectrum of the corrupted speech.

Several other examples exist that suggest that non-linear domains are advantageous in speech processing. This calls for some description of temporal processing in such domains.

Let us now formalize the temporal processing procedure by creating a more general notation. If temporal processing is applied to a non-linear function of the STFTM we can write the modified STFTM as

$$|Y_2(n,\omega)| = \mathcal{N}^{-1}\left\{\sum_{r=-\infty}^{\infty} F_2(n-r,\omega)\mathcal{N}\{|S_2(r,\omega)|\}\right\}, \tag{4.24}$$

where we have used the symbol $\mathcal{N}$ to denote a memoryless non-linear function (e.g. power-laws, logarithms, etc) of the STFTM, with inverse $\mathcal{N}^{-1}$. For many applications where resynthesis is not a concern, or only some features of the STFT need to be preserved, the STFTM term in (4.24) may be replaced by some other set of features like critical band

energies [12], LPC smoothed logarithmic spectrum [4], etc.

4.2.1 Time Trajectory Filters

In other works that apply filtering to the time trajectories in non-linear domains (e.g. [24], [28]) the filters $F_2(n,\omega)$ used were designed ad hoc, or heuristically. Moreover, the same filter was applied to all time trajectories. A major contribution of this dissertation is that the time trajectory filters will be derived systematically from training data. With this approach, the filters can be designed for each time trajectory independently.

4.2.2 Time-Domain Signal Resynthesis

If we wish to resynthesize a time-domain signal we can do it by using the original phase (delayed to compensate the group delay caused by the time trajectory filters), i.e.

$$y(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi} \mathcal{N}^{-1}\left\{\sum_{r=-\infty}^{\infty} F_2(n-r,\omega)\mathcal{N}\{|S_2(r,\omega)|\}\right\} e^{j\phi(n-\Delta,\omega)}e^{j\omega n}, \tag{4.25}$$

A block diagram representation of this operations is shown in Fig. 4.1. At the synthesis stage we put an inverse STFT (ISTFT) block which means that the synthesis method could be any of those described above. The block $z^{-\Delta}$ indicates a time delay in the discrete-time of Δ samples.

The memoryless non-linear operations and the STFT dependence make this scheme a processing domain which can not be directly related to any time domain processing. Many possibilities exist in this domain and the following chapters of this dissertation will present some of which have proven useful in speech processing applications.

4.3 Summary

When the CIT-MIF modification is applied to the STFTM only, and the original short-time phase is used for resynthesis, the time domain equivalent filter is shown to be time varying and STFTP dependent. Thus there is no simpler way of implementing the desired operation than the straightforward STFT domain filtering. Further complications arise

when the time trajectory filtering is applied to a non-linear function of the STFTM since there is no equivalent linear filter in the time domain that can accomplish such operation. The previous arguments confirm that non-linear STFT domains are an open field to explore new temporal processing applications.

In summary, up to this point we have set the theoretical background necessary to develop useful applications of temporal processing of speech. We can now proceed to show some implementations of this technique towards the solution of real speech processing problems.

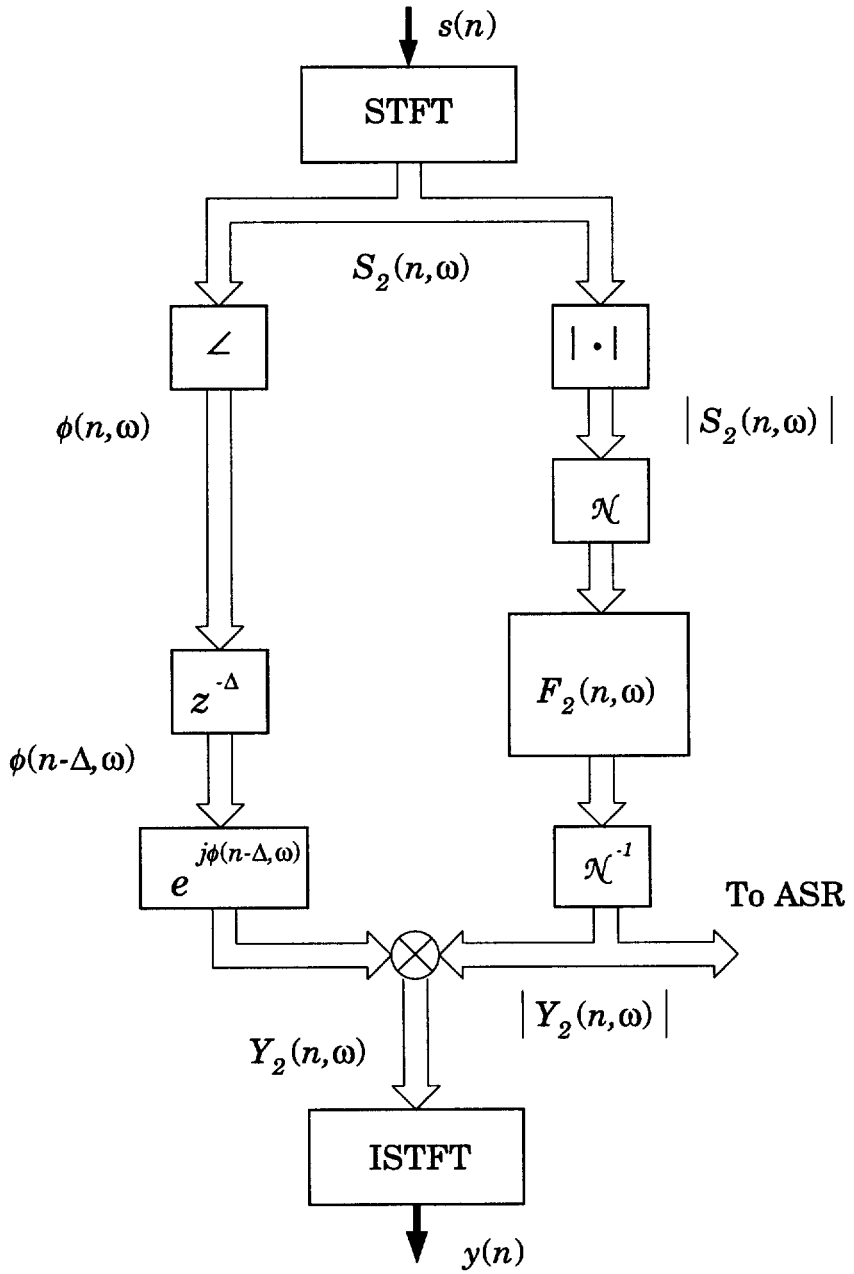


Figure 4.1: *Block diagram of temporal processing on the STFTM*

Chapter 5

Temporal Processing for Channel Normalization

Another known application of temporal processing is in the area of automatic speech recognition (ASR). Temporal processing has been used to reduce the effects of convolutional distortions introduced by the communication channel [1] [24], [41]. In this chapter we review the basic concepts involved in temporal processing for channel normalization. We perform an analysis to show the effect of the channel in the short-time representation of speech. Our goal is to gain a clear understanding of the true effects of the convolutional distortion on the STFT and the validity of the approximations used in practical implementations.

We will review in more detail the RASTA processing technique that has previously been cited. Also, the motivation behind cepstral mean subtraction (CMS) and its relation to temporal processing will be briefly mentioned. We show a simple analysis to point out how the ratio between the length of the channel impulse response and the short-time analysis parameters determine the limitations on the approximations made by channel normalization techniques.

The chapter is intended to give a theoretical background needed to present the last two applications of temporal processing that we discuss in this dissertation (Chapter 8 and Chapter 9).

[1] The term "channel" will be used to refer to the transmission medium.

5.1 Background

The idea of channel normalization is based on the homomorphic filtering theory developed by Oppenheim [46]. For convolved signals, separation by subtraction is possible in some transform domain where the signals become additive, i.e. logarithmic spectrum and cepstrum. This is true provided that we know the cepstrum or logarithmic spectrum of one of the components. When no explicit knowledge about any of the components is available, blind deconvolution methods may be applied. In these methods any a priori knowledge about the signals involved can be used to improve the results.

The pioneering work on blind deconvolution presented in [57] inspired some of the most popular channel normalization techniques used nowadays. In that work, the goal was to restore old audio recordings by removing the convolutional distortions introduced by primitive recording systems. Since no explicit knowledge of the signal or recording apparatus were available, the authors had to make some assumptions about the original signal. For this they used a recently recorded version of the recording that they wanted to restore ("Vesti la Guibba" by Enrico Caruso). The recent version was recorded with modern equipment and was sung by a tenor with voice characteristics similar to those of Caruso. The average spectrum of this new version was used as an estimate of Caruso's original performance. Dividing the average spectrum of the old recording by this new average, they obtained an estimate of the transfer function of the old recorder. The assumption in this case was that the transfer function of the modern equipment would approximate a flat response. From the estimate of the old recorder they designed an inverse filter. Processing the old recording with this filter restored Caruso's original voice.

In the context of ASR, the channel involved can be the microphone used to capture the speech signal, the handset, the telephone channel, etc. These channels have relatively short impulse responses. Additionally, the speech can be further distorted when produced inside enclosures by the impulse response between the speaker and the transducer. Room impulse responses are generally longer and present other difficulties (in Chapter 9 we develop a technique to deal with this situation).

Speech communication between humans suffer little degradation under channel distortions when the impulse responses involved are short (assuming that the bandwidth of the channel is large enough to preserve intelligibility) [63]. ASR systems, on the other hand, perform poorly if there are channel discrepancies between training and testing data [51].

We will now review two common channel normalization techniques which have a relation with homomorphic blind deconvolution, CMS and RASTA processing. Assumptions made by these techniques as well as their temporal processing properties will be briefly discussed.

5.1.1 Cepstral Mean Subtraction

Following the reasoning in [57], suppose that we segment a corrupted speech signal $x(n)$ into D (possibly overlapping) segments. The corrupted signal can be written as

$$x(n) = s(n) * h(n), \tag{5.1}$$

and the segments as

$$x_i(n) = s_i(n) * h(n), \qquad i = 1, 2, ..., D. \tag{5.2}$$

In (5.1) $s(n)$ is the speech signal and $h(n)$ is the transmission medium. The segments in (5.2) are not necessarily non-overlapping. Also, in (5.2) we have assumed that there are no truncation effects, i.e. the impulse response of the channel is much shorter than the segment length. This approximation will introduce an error but we will discuss its relevance later in the chapter. We have also assumed that the segments are longer than the impulse response of the channel and shorter compared to the length of the speech sample, so that the available number of frames D is large.

Taking the FT and the logarithm of each segment we obtain

$$\log[X_i(\omega)] = \log[S_i(\omega)] + \log[H(\omega)], \tag{5.3}$$

where $X_i(\omega)$, $S_i(\omega)$ and $H(\omega)$ are the Fourier transforms of $x_i(n)$, $s_i(n)$ and $h(n)$ respectively. Performing an inverse Fourier transform (IFT) on (5.3) we obtain the cepstrum

$$\widehat{x}_i(n) = \widehat{s}_i(n) + \widehat{h}(n). \tag{5.4}$$

The main idea behind CMS is that averaging (5.4) over several segments will yield an estimate of the channel cepstrum, i. e.

$$\bar{x}(n) = \frac{1}{D}\sum_{i=1}^{D}\hat{x}_i(n) = \frac{1}{D}\sum_{i=1}^{D}\hat{s}_i(n) + \hat{h}(n) \simeq \hat{h}(n), \tag{5.5}$$

assuming that the average speech signal cepstrum vanishes [5]. Thus CMS consists of subtracting the average in (5.5) from each segment of the distorted signal.

The last assumption is not valid in general for speech signals, so CMS removes not only the effect of the channel but also anything in the speech signal that is constant and common to all segments [5]. If we were to use the average $\bar{x}(n)$ as an estimate of the channel for cepstral deconvolution, the resulting reconstructed signal would suffer unwanted distortion. This is the reason why Stockham, et. al opted for using an estimate of Caruso's speech for the blind channel estimation in [57].

For some applications in speech recognition, the technique is successful because it achieves channel independence, i.e. regardless of what the channel is (as long as it is much shorter than the analysis window length), or what the long-term average properties of speech are, the recognizer is consistently presented with features whose time average value has been removed.

From the temporal processing point of view, the mean subtraction can be seen as a non-causal FIR filter (see [24] for more details). The properties of the implied filter depend on the available number of segments over which the averaging is performed, but in general such a moving average will have a high-pass magnitude frequency response. The segmentation and Fourier transformation in (5.2) and (5.3) can be viewed as an STFT. Then, mean subtraction can be viewed as a CIT-only modification, given that all the implied filters have the same impulse response for all frequency or quefrency bands.

5.1.2 RASTA Processing

For the past several years, speech recognition researchers have been working on the incorporation of temporal auditory masking into speech processing [24]. As an engineering simulation of this powerful auditory constraint they proposed to filter out slow and fast

changes in the trajectories of the logarithmic short-time spectrum of speech. This operation can be written using equation (6.1), and the short-time feature after RASTA processing becomes

$$Y(n,\omega_k) = \exp\left\{ \sum_{r=-\infty}^{\infty} R(n-r)\log[A(r,\omega_k)] \right\}, \tag{5.6}$$

where $A(n,\omega_k)$ could be the STFTM or a critical band integrated spectrum [16], [23]. Sometimes RASTA is applied to short-time cepstrum trajectories [41],[51].

Notice that we have dropped the subindex "2" from the short-time notation in (5.6). Throughout the rest of this dissertation we will drop this subindex and it should be understood that all short-time functions are frequency transforms with respect to the second argument (as in (2.6)).

The RASTA filter is a band-pass filter which has a spectral zero at zero modulation frequency and a pass-band approximately from 1 Hz to 16 Hz (see Fig. 8.2). It is implemented as an IIR filter and the same filter is used for all frequency bands, i.e. a CIT-only modification. Additionally, the impulse response of RASTA is causal (except for two taps that look 20 ms into the future).

We see that RASTA follows the mean subtraction idea by removing the dc component of the trajectory. It also attenuates low and high modulation frequencies. In contrast with the CMS filters (which may vary depending on the averaging time used), the transition band of RASTA is fixed. Another property is the low-pass characteristic that attenuates higher modulation frequencies. While attenuation of such frequencies seems to bring advantages, it is the flat pass-band between 1 and 16 Hz that preserves the perceptually relevant modulation frequency range [12], [4].

The parameters that control the response of the RASTA filter were experimentally obtained in a series of speech recognition experiments. In Chapter 8 we will use real speech data to derive RASTA-like filters, and we will gain understanding on the properties of the modulation spectrum of the data and the resulting filters, which validate many ideas behind RASTA processing.

5.2 Convolutional Distortions

As we stated in the previous section, channel normalization techniques that involve temporal processing make a series of assumptions about the effects of the channel on the short-time representation of speech. It is usually assumed that the short-time spectrum of the corrupted signal is equivalent to the short-time spectrum of the original signal multiplied by the Fourier transform of the corrupting channel [44], [51], [5]. It is our objective in this section to analyze in detail the true effects and understand to what extent can channel normalization techniques can be effective.

We also set the necessary background that will help us to develop the last application of this dissertation, which deals with normalization of channels with long impulse responses, e.g. room reverberation.

5.2.1 Effects of the Channel on the STFT

Let the corrupted speech signal be defined as in (5.1), which can be explicitly written as

$$x(n) = \sum_{r=-\infty}^{\infty} h(r)s(n-r) = \sum_{r=-\infty}^{\infty} s(r)h(n-r). \tag{5.7}$$

We would like to know if there is any relation to describe the STFT of $x(n)$ in terms of the STFT of the speech signal $s(n)$. Taking the STFT of (5.7) we obtain

$$X(n,\omega) = \sum_{m=-\infty}^{\infty} w(n-m)x(m)e^{-j\omega m} \tag{5.8}$$

$$= \sum_{m=-\infty}^{\infty} w(n-m) \sum_{r=-\infty}^{\infty} h(r)s(m-r)e^{-j\omega m},$$

or equivalently

$$X(n,\omega) = \sum_{m=-\infty}^{\infty} w(n-m) \sum_{r=-\infty}^{\infty} s(r)h(m-r)e^{-j\omega m}. \tag{5.9}$$

By making the change of variables $m' = m - r$ and interchanging the order of summation we arrive at the following expression:

$$X(n,\omega) = \sum_{m'=-\infty}^{\infty} \sum_{r=-\infty}^{\infty} s(r)w(n-m'-r)h(m')e^{-j\omega(m'+r)}, \tag{5.10}$$

letting $m = m'$ and rearranging the terms we get

$$X(n,\omega) = \sum_{m=-\infty}^{\infty} h(m)e^{-j\omega m} \sum_{r=-\infty}^{\infty} w(n-m-r)s(r)e^{-j\omega r}, \tag{5.11}$$

which can finally be written in terms of the STFT of $s(n)$ as

$$X(n,\omega) = \sum_{m=-\infty}^{\infty} h(m)e^{-j\omega m}S(n-m,\omega) = h_\omega(n) *_n S(n,\omega). \tag{5.12}$$

with

$$h_\omega(n) = h(n)e^{-j\omega n}.$$

Filter Bank Interpretation

The result in (5.12) can be explained in a more intuitive way by using the filter bank interpretation discussed in section 2.1.3. In Fig. 5.1 we show the block diagram filter bank interpretation of the original problem, i.e. equation (5.8), and the equivalent system described by (5.12). Simple block diagram operations can lead to the result in (5.12).

What the previous analysis shows is that the effect of the channel is not multiplicative in the short-time frequency domain. It remains as a convolutional distortion and the channel normalization techniques rely on the approximation (5.3). So we can ask why is it that channel normalization has been successful? Somehow the multiplicative property must show in some cases.

Intuitively we can see that depending on the properties of the window $w(n)$ and the channel $h(n)$, the convolution can approximate a multiplication. For example, if the window has very narrow bandwidth then when it is convolved with the modulated channel response it will serve as a frequency analyzer to the channel. Equivalently, if the window is long compared to the channel, the convolution can be turned into multiplication. In the next section we formalize this intuition.

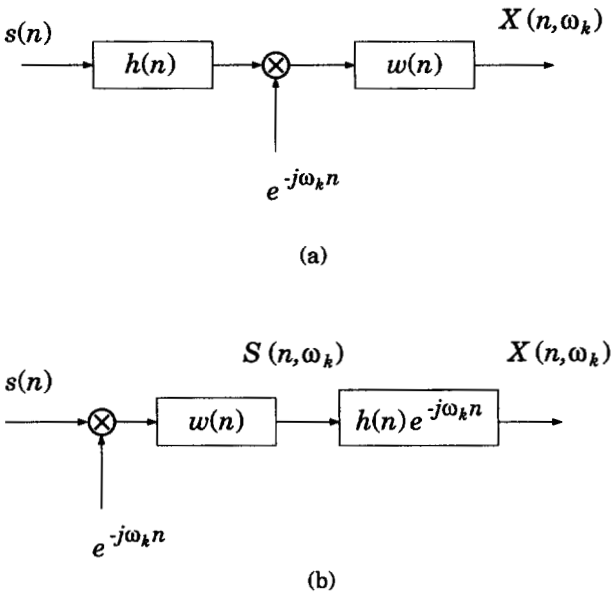


Figure 5.1: *Effect of the channel on the STFT. (a) Filter bank interpretation. (b) Equivalent system.*

Approximate Effects of the Channel on the STFT

First, let us interchange the summation order in equation (5.10) and rewrite it as

$$X(n,\omega) = \sum_{r=-\infty}^{\infty} s(r)e^{-j\omega r} \sum_{m=-\infty}^{\infty} w(n-m-r)h(m)e^{-j\omega m}. \tag{5.13}$$

Suppose that the window function $w(n)$ is long compared to the length of the impulse response $h(n)$, so that $w(n)$ is approximately constant over the duration of $h(n)$. Then

$$h(m)w(n-m) \simeq h(m)w(n), \tag{5.14}$$

and (5.13) becomes

$$X(n,\omega) \simeq \sum_{r=-\infty}^{\infty} s(r)e^{-j\omega r} \sum_{m=-\infty}^{\infty} w(n-r)h(m)e^{-j\omega m} \tag{5.15}$$

$$= \sum_{r=-\infty}^{\infty} w(n-r)s(r)e^{-j\omega r} \sum_{m=-\infty}^{\infty} h(m)e^{-j\omega m},$$

and recognizing the summation over r as the STFT of $s(n)$, and the summation over m as the FT of $h(n)$, i.e.

$$H(\omega) = \sum_{m=-\infty}^{\infty} h(m)e^{-j\omega m} \tag{5.16}$$

we finally arrive to an equation in terms of the STFT of the clean signal

$$X(n,\omega) \simeq S(n,\omega)H(\omega). \tag{5.17}$$

Equation (5.17) represents the desired condition for channel normalization, i.e. the channel shows as a multiplicative (rather than convolutional) factor in the short-time representation of the signal.

Frequency Domain Interpretation

The approximation for the channel effects on the short-time spectrum obtained in (5.17) represents the desired condition for channel normalization. The assumption (5.14) used to obtain this approximation can also be interpreted from the frequency domain point of view.

Taking the FT with respect to time n of the left hand side of (5.14) we get

$$\sum_{m=-\infty}^{\infty} h(m)w(n-m)e^{-j\omega m} = H(\omega) *_{\omega} [W(-\omega)e^{-j\omega n}], \tag{5.18}$$

where $W(\omega)$ is the Fourier transform of the window function, and the operator $*_{\omega}$ denotes convolution with respect to the frequency variable. The FT of the right hand side of (5.14) is

$$\sum_{m=-\infty}^{\infty} h(m)w(n)e^{-j\omega m} = w(n)H(\omega) = \alpha_n H(\omega). \tag{5.19}$$

where $\alpha_n = w(n)$ is a constant for the fixed time transform. We observe that for (5.18) and (5.19) to be similar we require that the window frequency response approximate a delta (unit sample) function of frequency, i.e. a filter with very narrow passband

$$W(-\omega)e^{-j\omega n} = \alpha_n \delta(\omega). \tag{5.20}$$

In the limit, for an infinitely long constant amplitude window the short-time transform would approximate the FT of the signal, and the channel effect would be exactly multiplicative. This result is not surprising since it just states the convolution theorem for the Fourier transform [45].

Discrete Time Implementation Considerations

From the previous section we conclude that the channel effect is never really multiplicative in the STFT domain. However, we want to get as close to this approximation as possible in order to be able to use homomorphic filtering-based normalization. We now discuss some considerations for the design and implementation of useful systems.

Practical implementations use finite length analysis windows. The length of the window determines the minimum of sampling points required in the frequency domain (Nyquist theorem). Thus, assuming that the channel has finite length, the minimum length of the analysis window must be at least that of the channel. This minimum requirement is not expected to yield good results since the approximation (5.17) requires the window to be constant over the duration of the channel for all n, and a finite length rectangular window will not fulfill this requirement at the end points.

A long and smooth window which tappers down the end points is then desirable. Commonly used windows with this property (Hamming, Hanning, Kaiser) have wider bandwidths. To fulfill the frequency domain requirement (narrow bandwidth) they also need to be longer.

The type of the window will determine the bandwidth and the amount of frequency aliasing introduced [45]. Aliasing will introduce errors which we have not considered in our analysis, and it is also desired to minimize these effects. Hamming and Hanning windows (just to mention two commonly used) have lower side-lobes and meet this requirement.

Based on the above considerations, it is reasonable to use a window with low aliasing, smooth edges, and long enough to satisfy the frequency and time domain conditions required. It is our experience that a Hamming window with a length at least 4 times the channel length provides a good approximation to (5.17) [9].

5.2.2 Discussion

Fortunately, some commonly encountered channels, like telephone channels, have short impulse responses compared to the analysis window lengths needed for speech analysis. As speech is non-stationary, windows longer than 32 ms are rarely used. We have observed that for the TIMIT and NTIMIT[2] databases, the approximation in (5.17) is reasonable.

However, if the channel is related to the impulse response of a room, such approximation is not valid, unless long windows are used. Such long windows limit the time resolution needed for ASR feature computation. In Chapter 9 we will develop a multiresolution technique to overcome these limitations.

5.3 Summary

In this chapter we have described two common channel normalization procedures based on temporal processing, and briefly discussed their motivation. We have shown that the condition assumed by these techniques is only approximately met. The channel shows up as a multiplicative term on the STFT only when the analysis window length is greater than the impulse response of the channel. In the frequency domain, the pass-band of the window should be as narrow as possible.

With this background we can now proceed to present our work on temporal processing for ASR channel normalization applications.

[2]NTIMIT consists of the same speech data in TIMIT passed through a telephone channel.

Chapter 6

Noise Reduction

In this chapter we present an application of temporal processing to the reduction of additive background noise in telephone communications. As is often the case with speech enhancement systems based on short-term spectral estimation [13], [35], our noise suppression algorithm is based on modifying the STFTM and resynthesizing a processed signal by using the STPTP of the original noisy speech.

In contrast with the mentioned methods, we perform the modification by filtering the trajectories (CIT-MIF modification) of the STFTM. The design of appropriate time trajectory filters is a major contribution that we present in this chapter. We illustrate how these filters can be designed based on training data, and then analyze their properties. We conclude the chapter by demonstrating an adaptive noise reduction algorithm [10]. The adaptive technique is based on selecting a set of pre-computed filters to process the STFTM trajectories of noisy speech. The responses of the pre-computed filters depend solely on the signal to noise ratio of the time trajectories, and does not depend on the center frequency of the band. This allows for a compact design in which the estimate of the signal to noise ratio at each frequency band is used as a filter bank design criterion.

6.1 Background

The enhancement of noisy speech is of great importance in voice communication systems that are to be used in real acoustic environments. For telecommunications applications, the need for speech enhancement systems increases with the spread of mobile telephony. Calls may originate from noisy environments such as moving cars or crowded public places.

The objective of a speech enhancement algorithm for a human-human communication is the improvement of the perceptual aspects of the speech signal, such as quality and intelligibility [17]. Quality is a subjective measure which indicates how pleasant or disturbing the signal is to the listener, while intelligibility is an objective measure of the amount of information in the signal that can be retrieved by the listener. These two perceptual aspects are not necessarily equivalent.

In applications such as hearing aids and transcription of forensic material, intelligibility improvement may be paramount. For other applications it may suffice to reduce the noise level to a degree at which listeners prefer the processed speech to the original noisy signal, even if intelligibility is not increased. Moreover, in most known cases, quality improvement can be achieved only at the expense of intelligibility [17], [35].

Depending on the particular application, the noise can be introduced to the communication link at different points. It could be introduced by the communication channel at the transmission point before the signal is transmitted, at the receiver side, or at several processing stages. One of the main problems of reducing noise in telephone communications is that the speech signal is corrupted before entering the system and there is no possibility to process it beforehand. Except for a very few cases, we do not have a noise pick-up microphone which could provide a reference signal to the noise and make noise cancelling strategies feasible. As pointed out by Ephraim [17], this situation constitutes one of the most difficult problems for speech enhancement.

To show the usefulness of temporal processing of speech we will devote this chapter of the dissertation to the enhancement of noisy speech in cellular telephone communications. Throughout the rest of the chapter we will refer to the speech enhancement problem as *noise reduction* to stress the fact that our goal is the enhancement of speech by reducing the background noise level, in contrast to enhancement systems that deal with other disturbances like reverberation [8], missing frequency extrapolation techniques [11], or noise cancelling algorithms. Let us first motivate the use of temporal processing to tackle this problem.

6.2 Motivation

The corrupting background noise encountered in mobile telephony can be stationary, or at least changes rather slowly compared to the rate of change of speech. Relevant modulation frequency components of speech are mainly concentrated between 1 and 16 Hz [4], with higher energies around 3-5 Hz [29]. Slowly-varying or fast-varying noises will have components outside the speech range. For example, steady tones will only have modulation frequency components at dc, i.e. ($\theta = 0$ Hz).

A system capable of modifying the modulation frequency content of a noisy speech signal in a controllable way could be useful for speech enhancement. As we discussed in Chapter 3, temporal processing can achieve such modifications. Based on prior work in the area (to be discussed next), we decided to apply time trajectory filtering to the noise reduction problem.

6.2.1 Previous Work

RASTA processing, a technique which modifies the modulation spectra by filtering the time trajectories of a time-feature representation of speech, has been successfully applied in channel normalization for ASR. A more detailed description of RASTA in the context of channel normalization was given in Chapter 5. It has also been recently applied to enhancement of noisy speech [25]. In that case, RASTA filtering was applied to the magnitude (or to the cubic-root compressed power spectrum) of the STFT of noisy speech, while keeping the phase of the original signal.

The spectral modifications caused by RASTA are of the CIT-only type because the same time trajectory filter is used for all frequency bands. The modified STFTM is

$$|Y_2(n,\omega)| = \mathcal{N}^{-1}\left\{\sum_{r=-\infty}^{\infty} R(n-r)\mathcal{N}\{|S_2(r,\omega)|\}\right\}. \tag{6.1}$$

The RASTA filter $R(n)$ is implemented as an autoregressive-moving average (ARMA) infinite impulse response (IIR) band-pass filter, with a magnitude frequency response which suppresses modulation frequencies below 1 Hz and above 16 Hz. Applying rather

aggressive [1] fixed RASTA filters (designed for suppression of convolutional distortions in ASR) for additive noise reduction yields results similar to spectral subtraction [13], i.e. enhanced speech often contains musical noise and the technique typically degrades clean speech.

As in spectral subtraction, the appearance of musical noise is also related to the fact that for (6.1) to be a valid STFTM, negative values resulting from band-pass filtering the time trajectories (RASTA suppresses the dc component) have to be removed prior to resynthesis.

The RASTA filter was optimized to improve the performance of speech recognizers in the presence of convolutional distortions. Under those optimal parameters there may not necessarily be a performance improvement if our application is different, such as noise reduction. Furthermore, applying the same filter to all frequency bands may be justified for the original problem [12], but there is no reason to impose such a constraint in a system that deals with a different situation.

6.3 RASTA-Like Noise Reduction Technique

We consider a scheme where the CIT-MIF modifications replace the RASTA filter. The operations involved in the technique are conceptually described by equation (4.25). In practice, the CIT-MIF modification is implemented with a finite number of time trajectory filters. We call this time trajectory filters, RASTA-like filters. We let the filters be non-causal finite impulse response (FIR) filters. For the nonlinear function in (4.25), we use a power-law. Under these requirements (4.25) becomes

$$y(n) = \sum_{k=0}^{K-1} \left\{ \sum_{r=-L}^{L} F_2(r,\omega_k)|S_2(n-r,\omega_k)|^{1/\gamma} \right\}^{\gamma} e^{j\phi(n-\Delta,\omega_k)} e^{j\omega_k n}, \qquad (6.2)$$

where the filters $F_2(n,\omega_k)$, as well as the parameters γ and Δ, are still to be determined. The number of frequency bands is K. Since speech is a real signal, its STFTM is a

[1] We use this term to indicate that the filter attenuation at dc in modulation spectrum is strong, e.g. > 20 dB.

symmetric function of frequency, and only $\frac{K}{2}+1$ (if K is even as is often the case in efficient FFT implementations) filters need to be specified.

Without loss of generality we assume that all filters have the same length $M = 2L+1$. The frequency sampling is set according to the Nyquist criterion, i.e. $\omega_k = \frac{2\pi k}{K}$, where $k = 0, ..., K-1$ and $K \geq N$, with N being the STFT analysis window length. In (6.2) we have used the FBS synthesis formula only to illustrate the technique. OLA synthesis was used for all our simulations with no substantial difference in the results.

6.3.1 Filter Design

To design the RASTA-like filters we generated a database from a pair of parallel recordings of clean and noisy speech. This database consisted of approximately 2 minutes of speech of one male talker recorded over a public analog cellular line from a relatively quiet laboratory. The speech was artificially corrupted by additive noise recorded over a second cellular channel from:

- a car driving on a freeway with the windows closed,
- a car driving on a freeway with one window open, and
- a busy shopping mall.

The STFTM of each recording was estimated using the parameters shown in Table 6.1. Each filter $F_2(n, \omega_k)$ was designed to optimally (in the least squares sense) map a time window (corresponding to the length of the filter M) of the noisy speech time trajectory at frequency ω_k, to a single point of the corresponding time trajectory of the clean speech.

If the STFTs of the clean speech and noisy speech are denoted by $S_2(n, \omega_k)$ and $X_2(n, \omega_k)$ respectively then we estimate the time trajectory of the clean speech $|\widehat{S}_2(n, \omega_k)|$ by

$$|\widehat{S}_2(n, \omega_k)|^{1/\gamma} = \sum_{r=-L}^{L} F_2(r, \omega_k) |X_2(n-r, \omega_k)|^{1/\gamma}. \tag{6.3}$$

The RASTA-like filter coefficients are found such that $|\widehat{S}_2(n, \omega_k)|^{1/\gamma}$ is the least squares estimate of the compressed time trajectory $|S_2(n, \omega_k)|^{1/\gamma}$ for each frequency band ω_k. This

Table 6.1: Noise Reduction Parameter Values

Parameter	Value	Number of Samples at 8 kHz
STFT window type	Hamming	
STFT window length N	32 ms	256
STFT window overlap	24 ms	192
STFT DFT length K	32 ms	256
Filter length M	264 ms	33
$\Delta = L$	128 ms	16
γ	1.5	

procedure is just a Wiener filter design on the compressed time trajectories of clean and noisy speech.

6.3.2 Tests

In a series of informal listening tests involving processed samples under different parameter settings, we determined the best setting based on the quality improvement of the processed sample. The parameter values for which we obtained the best quality results are shown in Table 6.1. The third column in the table shows the value of the parameters in number of samples for an 8 kHz sampling rate.

6.3.3 Parameter Values

For the analysis we used a STFT window with the same length as the discrete Fourier transform (DFT). It is well known that if modifications of the STFT are to be performed, the length of the DFT (K) should be larger than the length of the analysis window to avoid time aliasing during resynthesis [1]. We found no difference between the results obtained by oversampling the DFT in the STFT (i.e. $K > N$) and setting its length equal to the analysis window length ($K = N$). This may be explained by the fact (as discussed next) that the STFTM modifications were relatively mild so that their influence may not last more than N time samples. We did not extensively test these parameters, but rather fixed the analysis with values commonly used in traditional short-time analyses of speech.

To set the length of the time trajectory filters we systematically increased the number

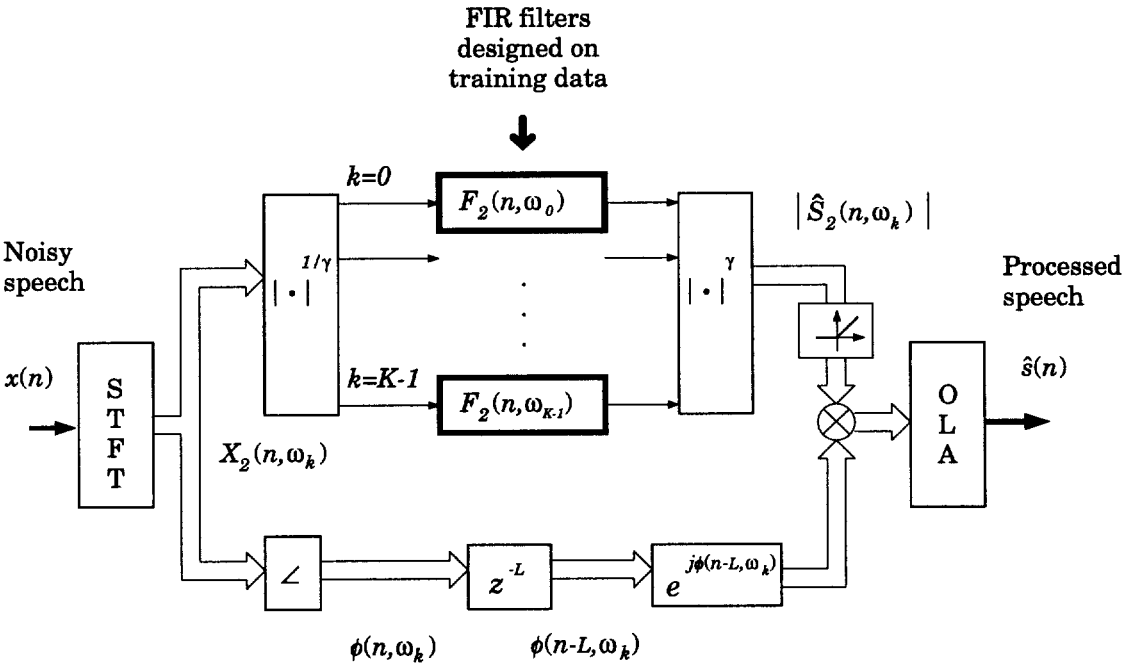


Figure 6.1: *Block diagram of noise reduction system.* $x(n)$ *is the noisy speech, and* $\hat{s}(n)$ *the processed speech. The compression is* $\gamma = 1.5$.

of taps from $M = 5$ to $M = 97$ in 4 tap steps. As we increased M we observed a better performance in the mean squared error sense as expected. The perceptual quality improved as M increased, but beyond $M = 33$ it did not improve significantly. For lengths greater than $M = 65$ the perceptual quality started to degrade (due to the appearance of echoes).

The best γ was found by varying its value from $\gamma = 0.5$ to $\gamma = 5$ in 0.5 steps. A very noticeable quality improvement was found when we tested $\gamma = 1.5$ compared to lower values. As the value increased beyond $\gamma = 1.5$, the improvement was not perceptually significant. Interestingly, this value of γ seems to correspond to the exponent of the loudness-intensity power law relationship encountered in psychoacoustics [39].

A block diagram of the noise reduction system with these parameters is depicted in Fig. 6.1. Notice that we have shown a filtering operation for each of the K time trajectories. However, it should be understood that given the symmetry of the STFTM, we only need to filter the first $\frac{K}{2} + 1$ trajectories and copy the result to the remaining trajectories so as to achieve a symmetric processed STFTM. We also show a rectification stage before

resynthesis that is needed to maintain positive magnitude values.

6.3.4 Evaluation

Once the best parameters were set we conducted an evaluation to compare the performance of the new system to spectral subtraction [13]. The evaluation consisted on processing a noisy speech sample with the two techniques, and compute a perceptually-based distance between the processed samples and the original noise-free speech. The noise was artificially added and it was car noise similar to the noise used to train the RASTA-like filters. The distance measure we used was the normalized averaged mean squared error between logarithmic critical band energies. The critical band energies were simulated by a weighted summation over the STFTM frequency components of the signals. The weighting coefficients (critical band shapes) used were the same as those described in [23].

In Table 6.2 we show the results obtained for a noisy speech sample with a signal to noise ratio of 5 dB. The spectral subtraction algorithm used is described in [31].

Table 6.2: Averaged Mean Squared Error

	Noisy	Spectral subtraction	RASTA-like filtering	Clean RASTA-like
Clean Speech	3.84	5.71	3.11	0.97

In the last column of Table 6.2 we report the result obtained when clean speech was processed by the RASTA-like filters. This result indicates that the system introduces a small amount of distortion.

6.3.5 Properties of RASTA-Like Filters

Frequency Responses

Filter magnitude frequency responses are shown in Fig. 6.2 (darker shades represent larger values). Filters for different frequency channels differ. The whole frequency band between

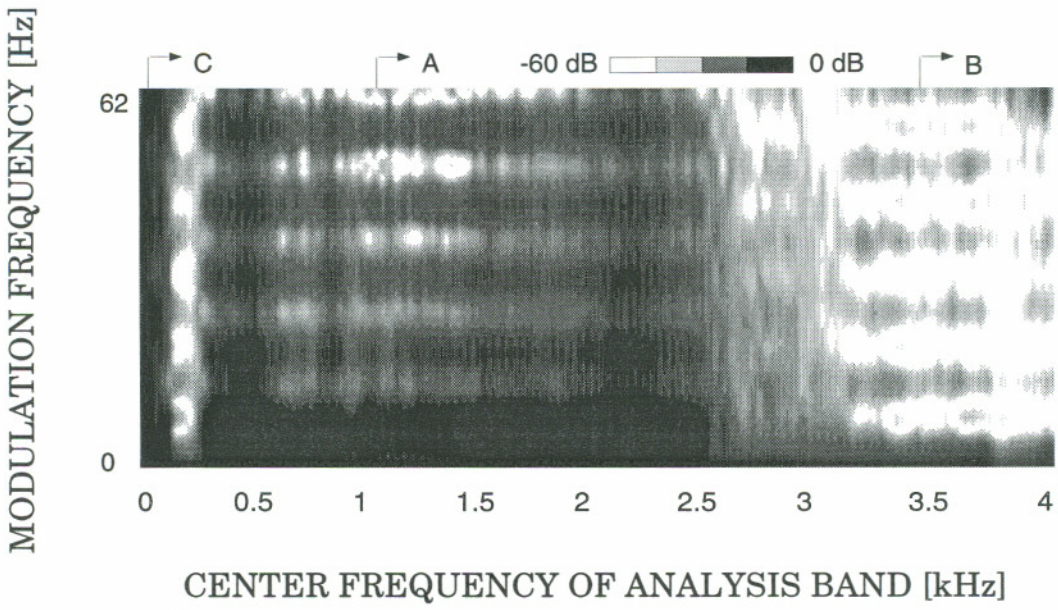


Figure 6.2: *Frequency responses of RASTA-like filters*

0 and 4 kHz appears to be sub-divided into several regions, each characterized by its own RASTA-like processing.

The highest gain RASTA-like filters are applied in the frequency bands between 300 Hz and 2300 Hz. Frequency responses of typical filters in this region (slice A in Fig. 6.3) are shown in Fig. 6.3(A). Typically, filters in this frequency band have a band-pass character, emphasizing modulation frequencies around 4-5 Hz. Comparing to the original ad hoc designed RASTA filter, the low frequency band-stop is much milder, being only at most 10 dB down from the maximum.

Filters for very low frequencies (0-100 Hz) are high-gain filters with a rather flat frequency response (slice C in Fig. 6.2 and Fig. 6.3(C)) [2]. Filters in the 150-250 Hz and the 2700-4000 Hz regions are low-gain low-pass filters (slice B in Fig. 6.2 and Fig. 6.3) with at least 10 dB attenuation for modulation frequencies above 2 Hz. The low frequency pass-band of these filters are typically below the pass-band of the high-gain band-pass filters of Fig. 6.3(A).

[2]The reason for this behavior will be explained later in section 6.4.2.

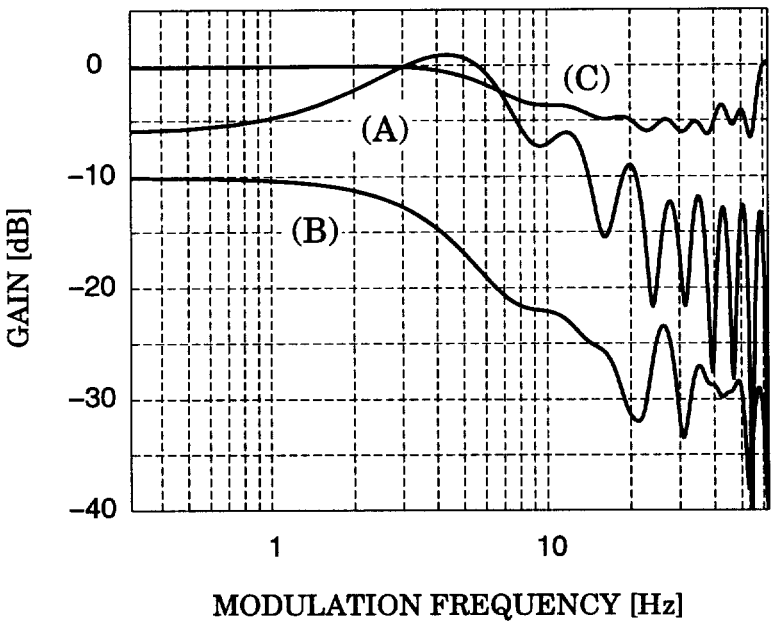


Figure 6.3: *Frequency response of filters at different bands. The labels in this figure correspond to the regions with the same label in Fig. 6.2*

Impulse Responses

The impulse responses of the filters are approximately symmetric with near linear phase and consequently approximately constant group delay (see Fig. 6.4). That is the reason why the phase delay term Δ was chosen to be half the filter length, i.e. the position of the tap with highest magnitude, and center of symmetry. This delay was our choice since we designed the filters to be non-causal and looking the same number of frames into the past as in the future.

Discussion

We mentioned in section 6.3.1 that oversampling the STFT in frequency did not change the results. This can be explained by looking at the dc attenuation of the resulting RASTA-like filters, which is low at high energy frequency bands. Assuming that the time trajectory filters were single tap filters ($M = 1$), the equivalent time domain impulse response of this MIF modification would not require a large number of taps thus making the time domain aliasing almost negligible for practical purposes. In the following section we discuss our

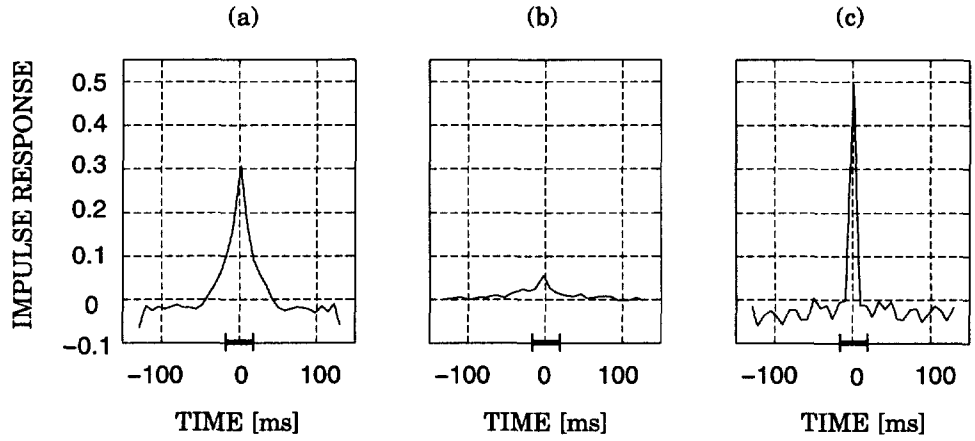


Figure 6.4: *Impulse responses of RASTA-like filters at (a) region A Fig. 6.2, (b) region B in Fig. 6.2, and (c) region C in Fig. 6.2. For comparison, the dark bar on the time axis corresponds to the length of the analysis window, i.e.* 32 *ms.*

results from the Wiener filter point of view.

6.3.6 Wiener-Like Behavior of RASTA-Like Filter Bank

To test the advantage of the CIT-MIF modification over an MIF-only modification, we designed a time domain Wiener filter on the same noisy and clean data [22]. As was shown in [3], a time-invariant MIF-only modification of the STFT is equivalent to a time domain time-invariant linear filter. As we discussed in Chapter 3, the CIT-MIF modification of the STFT can also be implemented as a linear time-invariant filter in the time domain. In this sense the advantage of increased modulation frequency resolution (due to the temporal filter) is tested. However, the proposed noise reduction operates on the STFTM for which there is no time domain equivalent, and the advantage of this fact over time domain processing is also tested.

The length of the Wiener filter was set to K taps to achieve the same frequency resolution as the STFT. The magnitude frequency response of the filter is shown by the solid line in Fig. 6.5. For comparison, the norm of the RASTA-like filters, designed on magnitude spectrum (i.e. $\gamma = 1$), are shown in the figure by the dashed line.

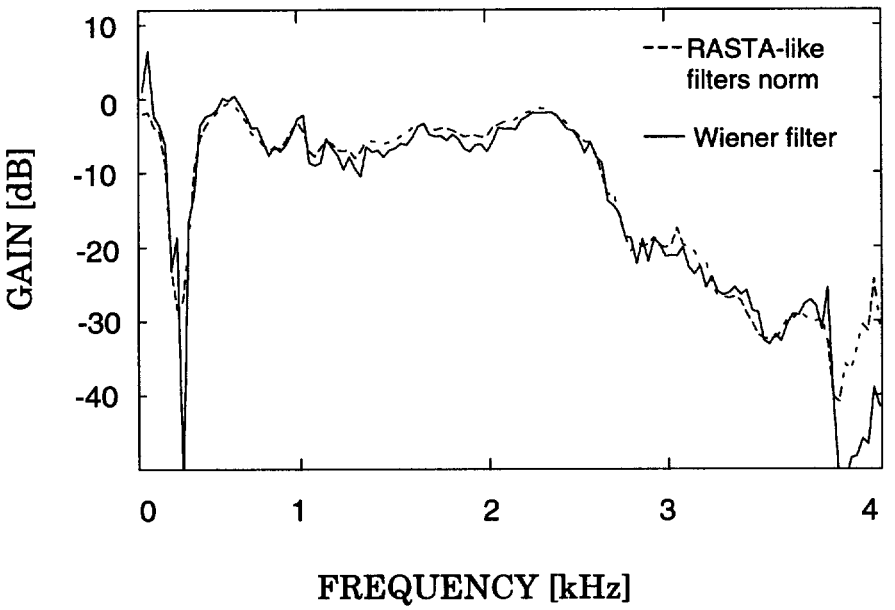


Figure 6.5: *Wiener filter response and norm of RASTA-like filters.*

Listening Tests

Informal listening comparisons between the quality of speech processed by the time-domain Wiener filter and these RASTA-like filters indicate that processing the STFTM and the additional modulation frequency resolution gained with the CIT-MIF perform better (in the perceptual sense) than the time domain Wiener filtering.

However, for these tests we did not consider the effect that the residual noise may have on the listeners. It was obvious that the reduction of the noise level was perceptually greater with the RASTA-like filters, but the quality of the residual noise was different compared to the Wiener filter results. While the residual from Wiener filtering appears to be louder, it does not have level fluctuations present in the residual obtained from the temporal processing.

Compared to the spectral subtraction results (see section 6.3.4), both Wiener and RASTA-like filters produce a less annoying residual.

6.4 The Effect of Signal to Noise Ratio on the Properties of the RASTA-Like Filters

Two important aspects of the RASTA-like filters of the previous section were observed:

1. The magnitude frequency response of filters corresponding to frequency regions of high speech energy showed suppression of low ($\theta < 2$ Hz) and high ($\theta > 8$ Hz) modulation frequencies, while enhancing modulations around 5 Hz. Filters at regions of low spectral energy were low-pass or flat.

2. The dc gain of the filters was high at high signal to noise ratio (SNR) time trajectories and low at low SNR time trajectories, thus following the Wiener principle of optimal noise suppression.

The observations above suggest two possibilities. One is that the filter characteristics may depend on the energy of the speech signal relative to the noise level at each time trajectory. Thus a filter bank could be designed based on these local SNR levels (frequency-specific SNR levels) rather than on specific noisy training data.

The second possibility is that the shape of the RASTA-like filters might depend on the center frequency of the trajectory, while the gain might depend on the frequency-specific SNR. To find out what are the factors that determine the frequency response of the filters we performed the following experiments.

6.4.1 Preliminary Studies

The first question that we formulated based on our observations was whether the filter responses depend only on the local SNR or if they also depend on the center frequency ω_k of the time trajectory for which they are designed.

To answer this question we constructed a database by corrupting a sample of clean speech (approximately 180 s in length, taken from the TIMIT[3] database) with additive

[3]TIMIT is a speech database recorded by Texas Instruments (TI) and the Massachusetts Institute of Technology (MIT).

white Gaussian noise (AWGN) at different overall SNR levels (20, 15, 10, 7, 5, 3, 0, −3, −5, −7, −10, −15, −20 dB). These SNR levels were computed as

$$\mathrm{SNR}_{\mathrm{dB}} = 10\log_{10}\left\{\frac{\sum_n s^2(n)}{\sum_n v^2(n)}\right\}, \tag{6.4}$$

where $v(n)$ is the noise sequence, and the summation over n goes for the time length of the speech signal.

Given that the speech signal energy is not uniformly distributed in frequency, by computing the STFTM of a database set we can identify the SNR for each particular frequency band. This follows from the fact that we know both the clean signal and the noise sample, and we can individually compute their STFTs. This frequency-specific SNR can be computed as the ratio of the total power of the time trajectories of the STFTMs of speech and noise signals at the given frequency band, i.e.

$$\mathrm{SNR}_{\mathrm{dB}}(\omega_k) = 10\log_{10}\left\{\frac{\sum_n |S_2(n,\omega_k)|^2}{\sum_n |V_2(n,\omega_k)|^2}\right\}, \tag{6.5}$$

where the summation over n goes for the length of the STFT, and $V_2(n,\omega_k)$ is the STFT of $v(n)$.

For each database set we designed a RASTA-like filter bank following the procedure described in section 6.3.1, with the parameters shown in Table 6.1. Thus, a total of 1677 RASTA-like filters were designed.

6.4.2 SNR-dependent RASTA-like Filters

Fig. 6.6 shows the filter characteristics for different SNR levels. Each plot in the figure shows the magnitude frequency responses of filters derived at a given SNR for several frequency bands (dotted lines), together with the mean response (solid line) of the filters. We computed the frequency response of the filters for a given frequency-specific SNR only at some representative ω_k s covering the frequency range of interest (0 Hz to 4 kHz in this case). The representative ω_k s were selected by sampling an SNR versus frequency plane. This plane was constructed by computing the frequency specific SNR levels at 129 equally spaced frequency bands for each of the 13 databases of section 6.4.1. Thus, the

plane consisted of 1677 points, each corresponding to the SNR condition under which each of the 1677 filters was designed. For a given SNR we found all points on the plane which lied close (±0.01 dB) to that value.

Even when this procedure did not yield an equally spaced frequency sampling, the selected ω_k s covered the whole frequency range with less than 10 bands (or about 300 Hz) separation between them.

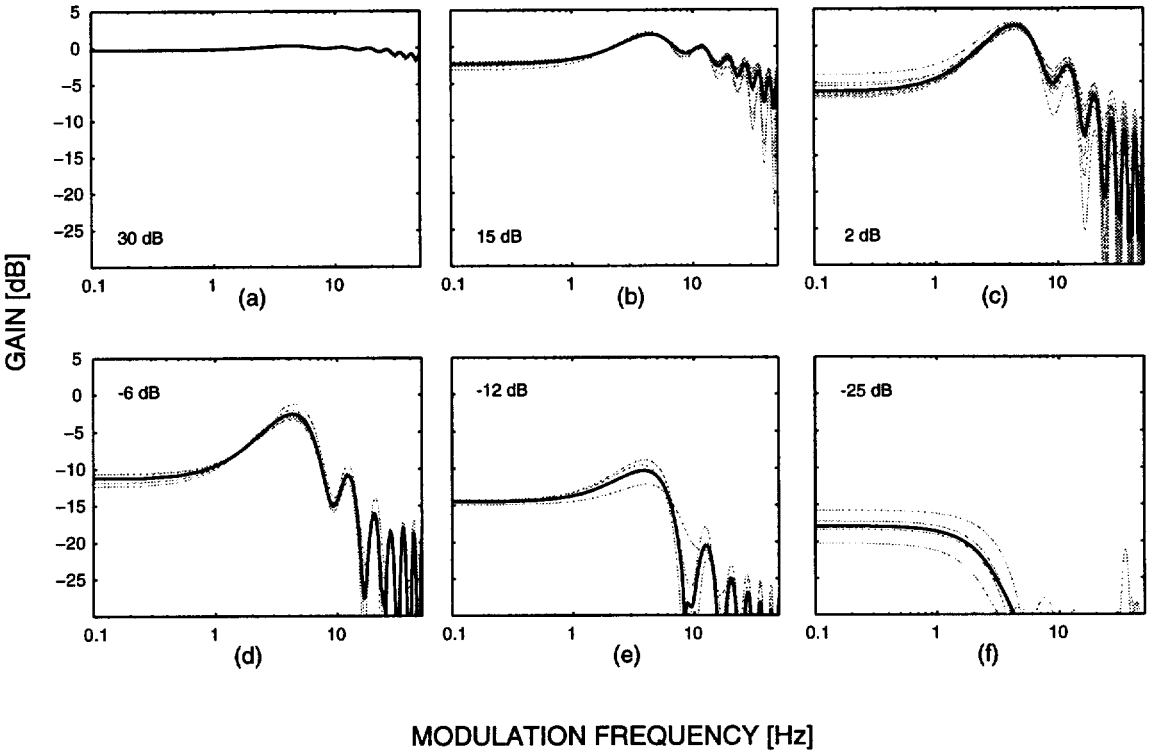


Figure 6.6: *Filter frequency responses (dotted lines) and mean response (solid lines) for several frequency-specific SNR levels*

Observations

For a wide range of SNR values we made the following observations. As the frequency-specific SNR decreases, the magnitude frequency response of the filters changes from

- a flat response (i.e. no filtering, see Fig. 6.6(a)), through

- a strong band-pass response enhancing modulation frequencies around 5 Hz (i.e. speech enhancement, see Fig. 6.6(c) and Fig. 6.6(d)), to
- a low gain, low cut-off frequency low-pass response (i.e. suppression of the given component, Fig. 6.6(f)).

Notice that the attenuation of the dc component (i.e. $\theta = 0$) increases with decreasing frequency-specific SNR. The results obtained in this section confirm the idea that the RASTA-like filters are strongly dependent on the SNR of the time trajectory and relatively independent of the center frequency ω_k. Similar behavior was observed when the filters were designed from a smaller database [4] created by artificially adding car noise to the clean speech.

Discussion

With the conclusions drawn from this preliminary study we can interpret the results obtained in section 6.3.5. Filter responses in region A of Fig. 6.2 can easily be interpreted given that speech energy and noise energy are comparable in the 500-2500 Hz range and the filters tend to enhance speech modulation frequencies as in Fig. 6.6(b-d).

Higher frequency regions (above 3000 Hz) have low speech energy and consequently low SNR, and the filters tend to suppress those bands (see Fig. 6.6(e-f)). For the low frequency bands in region C of Fig 6.2, the filter responses appear to be all-pass. An explanation to this is that for deriving the filter bank in section 6.3.1, we generated our database by artificially adding noise to a clean cellular telephone speech sample. Analysis revealed that the additive noise lacked components at those low frequencies and the filters just tried to map the uncorrupted speech sample to itself, yielding the all-pass characteristic.

6.5 Adaptive System Design

Based on our observations about the nature of RASTA-like filtering for noise reduction, in this section we describe an adaptive RASTA-like noise reduction technique intended for

[4] Only −20, −10, −3, 0, 3, 10 and 20 dB overall SNR levels were used.

applications in services such as voice mail where the noisy speech recording is available for non-real time processing. With some modifications, the system is in principle also suitable for real-time processing.

As mentioned before, one of the problems that needs to be considered in mobile telephone communications is that, in general, background noise has different characteristics from one call to the next. A successful noise suppression system needs to use some strategy to deal with this factor.

The observations described in the previous sections allow us to design a noise reduction system which adapts to a specific noise condition. This extension makes the system applicable in realistic situations with noises and speech of unknown variance and coloration.

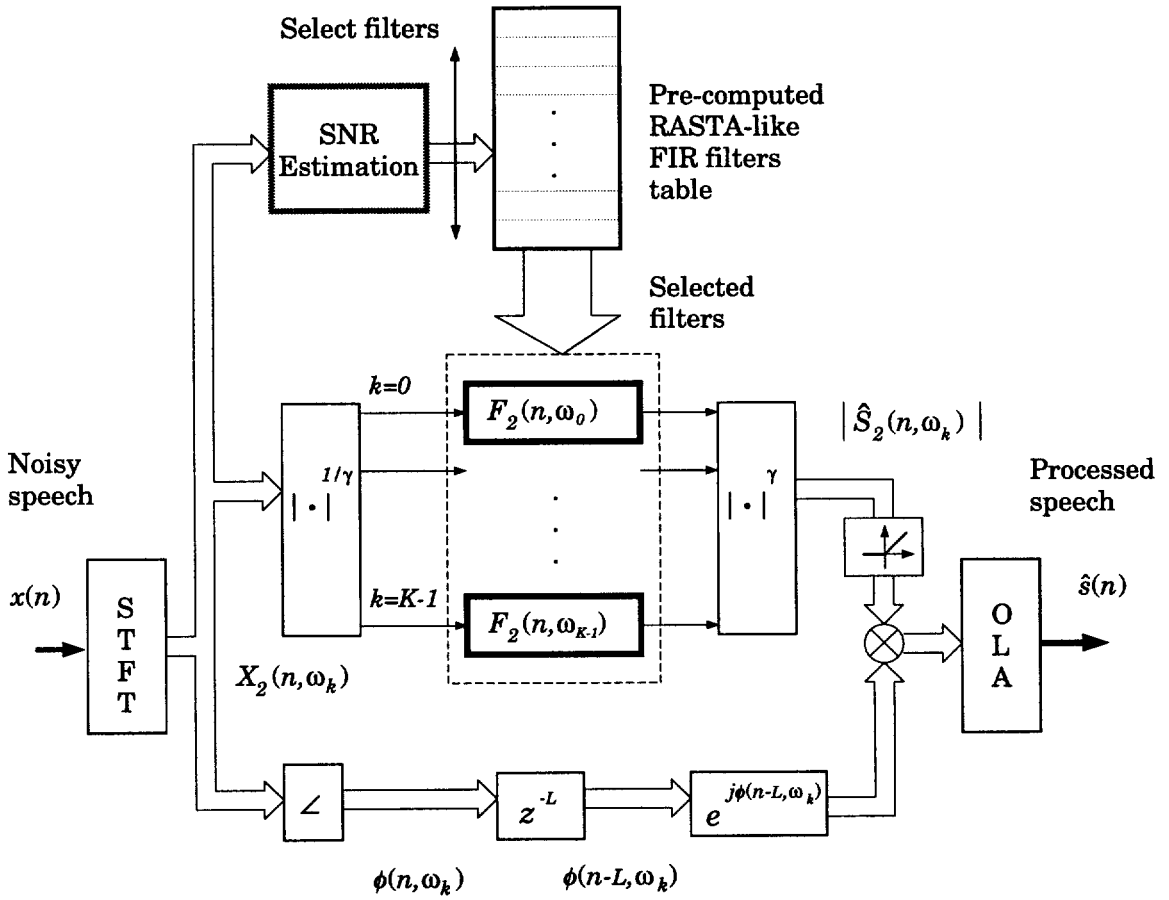


Figure 6.7: *Block diagram of the adaptive system.* $x(n)$ *is the input corrupted speech,* $\hat{s}(n)$ *is the estimate of the clean speech (* $\gamma = 1.5$*)*

The system configuration is shown in Fig. 6.7. To assemble the appropriate filter bank for a particular noisy speech recording we compute the frequency-specific SNR for each STFTM time trajectory over the whole recording, and select a RASTA-like filter from a basis set of a few pre-computed basic filter shapes. After all filters for all time trajectories are selected, we proceed to filter the compressed STFTM trajectories, expand and resynthesize using the OLA synthesis technique.

6.5.1 SNR Estimation

In practical situations we do not know the frequency-specific SNR levels so an estimation procedure is required. We are primarily interested in the internal consistency of the estimate (rather than in the accuracy of the actual SNR estimate) as a measure of its usefulness for selecting a set of filters.

For this purpose we apply a noise estimation procedure proposed by Hirsch [28], in which the noise power at each magnitude STFT trajectory is estimated by computing a histogram of its amplitude. The peak of the smoothed histogram is chosen as the noise amplitude estimate. Since we do not know the power of the clean speech signal, we use the power of the available noisy signal, thus obtaining an estimate of the noisy signal to noise ratio. For our purpose, the performance of this estimator was found to be reasonable.

6.5.2 Filter Design

Pre-computation of the Filters

To design the set of basic RASTA-like filters we used the same clean and noisy data as reported in section 6.4.1 above. For this, we assumed that the additive noise sources of interest have Gaussian distributions. This constitutes the most general case where the noise has components at all modulation frequencies. In the case of colored noise sources, if the correlation time of the source is shorter than the analysis window length, the coloration of the noise is irrelevant. This is because individually, the subband noise components from a colored Gaussian noise signal behave in the same way as if they were derived from a white source (in terms of its time trajectory distribution, regardless of its variance which is determined by the subband energy).

To derive a set of SNR-specific filters we averaged the magnitude frequency responses of filters computed at a given SNR, and designed a non-causal linear phase FIR filter from the averaged response. We excluded filters with center frequencies below 100 Hz from the average because their responses were found to deviate slightly from the average (mainly in the dc gain factor). The linear phase assumption is justified from the observation that all the filters computed for the preliminary study section above are approximately linear phase. A total of 25 filters, each corresponding to a frequency-specific SNR in 1 dB steps, was found to perform reasonably well.

Construction of the Filter Table

In order to calibrate the SNR estimator which is used during processing (i.e. to find a mapping between the estimated and actual frequency-specific SNR levels) the SNR levels corresponding to each filter were estimated using the histogram technique. The 25 filters were stored in a table along with their corresponding estimated frequency-specific SNR levels.

6.5.3 Operation of the System

During the operation of the adaptive noise reduction system on data with unknown noise (e.g. real telephone calls), the SNR is estimated for each time trajectory and a proper filter bank is built by selecting the appropriate filters from the table.

Although originally designed for the off-line applications in enhancement of noisy voice-mail recordings [10], the technique is not constrained to non-real time processing. We did not yet extensively experiment with the real-time processing, but the frequency-specific SNR estimation procedure can be done in real time if a first estimate is computed during the first few seconds of a conversation and updated periodically over the length of the sample. As such, this adaptive update has the ability to adapt to time-varying conditions.

6.6 Noise Reduction Results

To evaluate the performance of the system under different conditions we conducted the following set of tests:

6.6.1 Known noise

To test the system independently of the SNR estimator, we artificially corrupted the clean speech (with colored Gaussian noise) and applied the processing with prior exact knowledge of the frequency-specific SNR. The result indicated a strong suppression of background noise while preserving the speech signal with very minor audible distortions. The residual noise has a very different character than the original disturbance. While the noise is not as annoying as the musical noise in spectral subtraction, it presents periodic level fluctuations. These fluctuations are related to the enhancement of certain modulation frequencies imposed by the filters in the medium SNR range (see Fig. 6.6). The modulation frequencies of the residual noise around 5 Hz are also enhanced and can be heard as a periodic disturbance. The distortion of speech is minimal compared to the distortion introduced by spectral subtraction.

Example

In Fig. 6.8 we show an example of the performance of the adaptive RASTA-like filtering for the case in which the SNR is known. Part (a) shows the waveform and spectrogram of the original clean speech. The noisy waveform and spectrogram are shown in part (b) of the figure. The noise was additive colored Gaussian noise, artificially added to produce an overall SNR of 10 dB. In part (c) of the figure we show the waveform and spectrogram of the noisy speech after processing.

Since the frequency-specific SNR levels are available, this example shows the best performance that we can obtain with the adaptive algorithm. Even when the level of perceived noise is considerably reduced and the quality improved, the SNR after processing was only 11.62 dB. This shows that there may be some distortion of the speech signal introduced by the processing, and that the SNR improvement on the time signals is not a

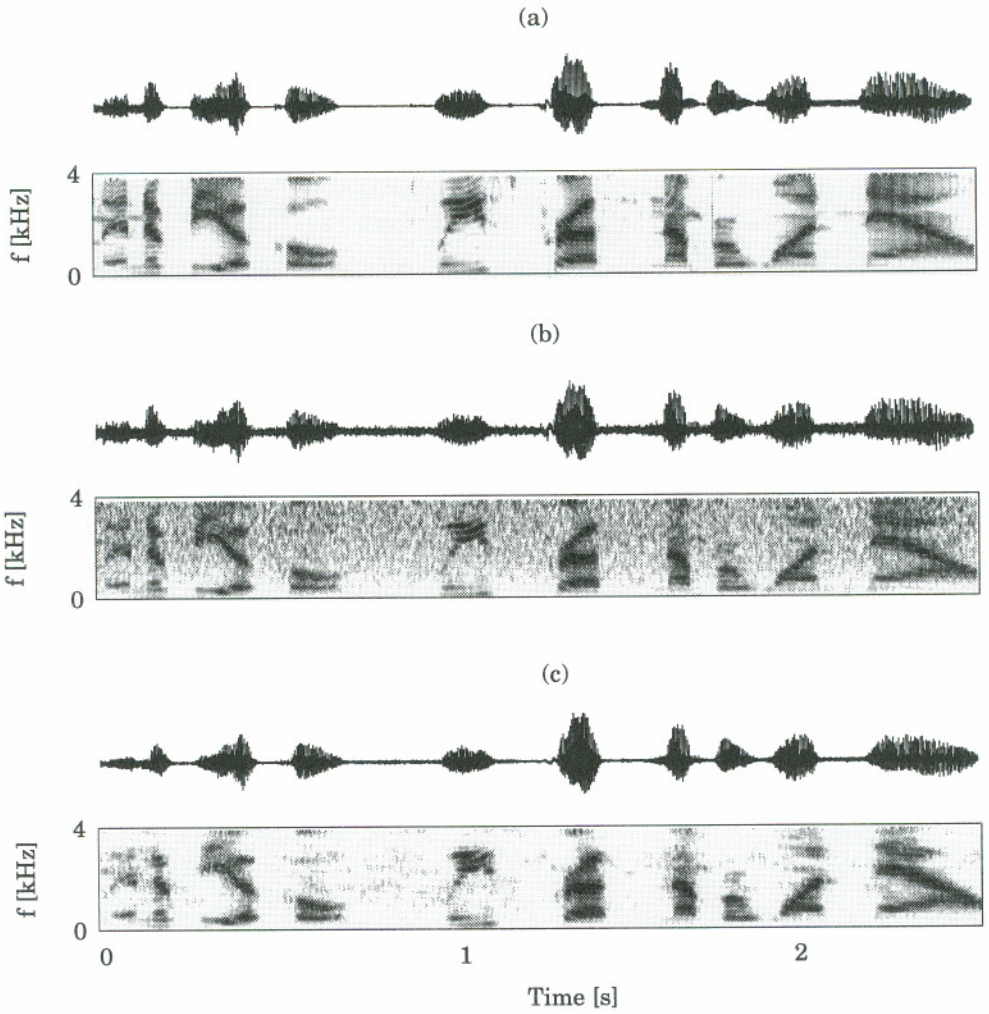


Figure 6.8: *Waveform and spectrogram of (a) original clean speech signal, (b) the noisy signal, and (c) the processed noisy signal.*

good indicator of performance.

6.6.2 Unknown noise

Applying the algorithm based on the frequency-specific SNR estimates, we found very similar results. However, the noise level was underestimated and the suppression was slightly milder. Tuning the estimated to real SNR map, or biasing the SNR estimator itself might be helpful, but a better and more robust solution to the SNR estimation problem needs to be found if we want to take full advantage of the adaptive structure.

For a wide range of noise types and levels present in real cellular telephone calls we found a noticeable suppression of the perceived noise. In several informal preference tests we presented the subjects with 6 pairs of representative (real telephone calls) noisy and processed samples, and asked them which of the samples did they prefer. We found that more than 50% of the time subjects preferred the processed samples.

Other researchers have evaluated this adaptive system by comparing its performance with spectral subtraction and a novel dual Kalman filtering approach [43]. The evaluation criteria was the SNR improvement in the time-domain signals. They found that for a sample of speech corrupted by colored noise, the SNR improvements were 4.87 dB, 5.71 dB, and 5.27 dB for spectral subtraction, dual Kalman filtering, and adaptive RASTA-like filtering respectively.

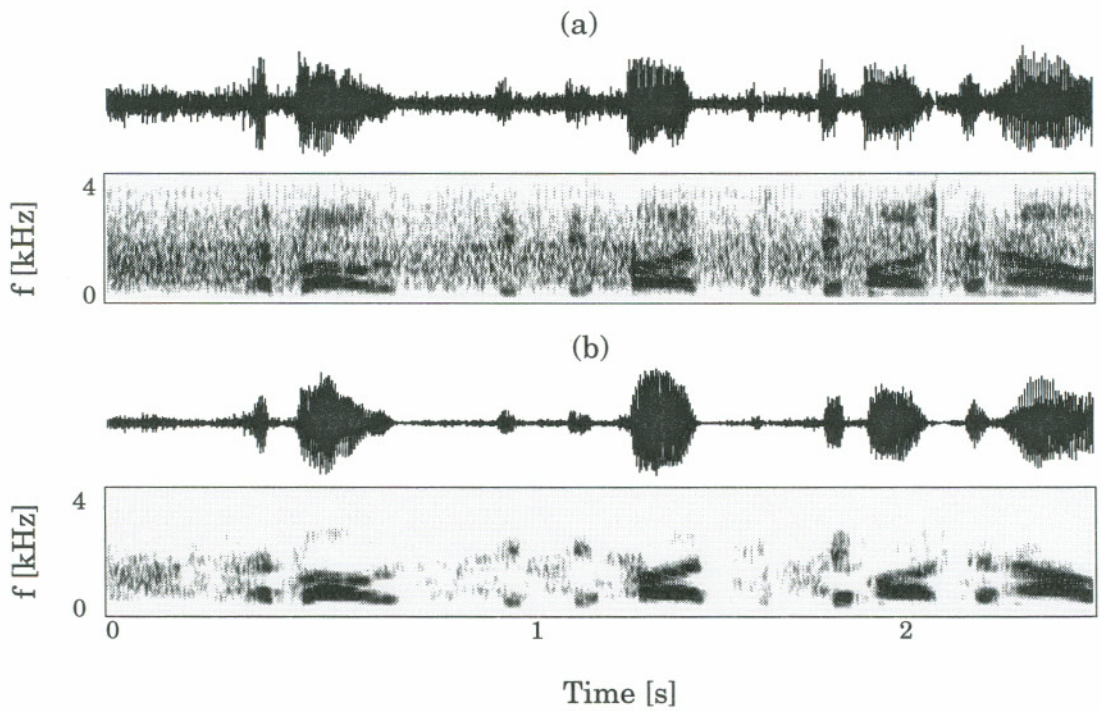


Figure 6.9: *(a) Noisy speech signal (above) and corresponding spectrogram (below). (b) time signal (above) and spectrogram (below) of the same noisy segment after processing.*

Example

Here we show an example of noise suppression for the real case where the noise is unknown, Fig. 6.9 shows the waveform and corresponding spectrogram of a real noisy cellular telephone call and the time signal and spectrogram of the same sample after processing.

In situations similar to this example we observed that there is some distortion of the speech signal which can be attributed to the inaccuracy of the SNR estimator. However the quality improvement after applying the processing is considerable.

6.7 Summary

In this chapter we presented an application of temporal processing to the reduction of noise in telephone communications. The CIT-MIF modifications were applied to the compressed STFTM and implemented as FIR RASTA-like filters. Our novel approach of designing the time trajectory filters from training data proved to be powerful. A few observations on the properties of the filters designed from a particular training set lead to a study of the factors which affect their characteristics and ultimately to the design of an adaptive noise reduction technique.

Our evaluations indicate that the algorithm generalizes quite well over different types and levels of noises. An important feature is that the optimization of the system based on speech allows for a reduced audible speech distortion compared to other methods. While the adaptive system was originally designed to process voice-mail recordings it can be modified to operate on real time situations.

Chapter 7

Reverberation Reduction

In this chapter we explore speech reverberation reduction using the CIT-MIF modification of the short-time spectrum. The principle is the recovery of the average envelope modulation spectrum of the original (anechoic) speech. Previous work using these principle has been reported in [33] and [27], where high-pass filtering and inverse modulation transfer functions respectively, have been used. Based on our previous experience with modulation spectrum modifications for additive noise reduction (Chapter 6), we apply the data-driven RASTA-like filter design technique to the reverberant speech. Comparing our results with other works we discuss the effectiveness and limitations of this type of approaches [26].

7.1 Background

Recent advances in teleconferencing, multi-media and mobile hands-free telephony have spurred an interest in reducing the effect of room reverberation in speech communications. In the past, the problem has been approached from several different perspectives depending on the particular application. While several viable solutions exist in situations where more than one channel is available (multi-microphone systems) [2], [47], single channel systems still pose a formidable challenge.

For single channel systems two main approaches have been taken. One approach consists of estimating some properties of the room from the corrupted data and applying deconvolution techniques to recover the speech. The main drawback in such cases is that some simplifying assumptions about the speech and channel have to be made, thus yielding only suboptimal solutions [60],[56].

In a different approach, an attempt is made to recover the energy envelope of the original (anechoic) speech by applying a theoretically derived inverse modulation transfer function (defined below), or ad hoc high-pass filtering [27]. Such approaches were motivated by studies on the the effect of reverberation on the modulation index (also defined below) of speech and the reduction of intelligibility in reverberant environments [29].

7.1.1 The MTF and MI

In this section we briefly discuss two concepts that will be necessary to motivate our signal processing procedures.

The Modulation Transfer Function

The modulation transfer function (MTF) was first introduced as part of a procedure to assess the performance of optical systems [29]. In sound transmission in rooms, the MTF refers to the transfer function that characterizes a system in terms of the changes in the modulation depth of a temporally sine-wave modulated test signal (e.g. a white noise sequence). The modulation depth reduction can differ for different modulation frequencies, thus the modulation depth reduction as a function of modulation frequency constitutes the MTF [29].

To determine if the MTF of a system will affect the transmission of a signal it is necessary to determine which modulation frequencies are present in the signal, which are more important, and how they will become affected.

The Modulation Index

The modulation index (MI) is a measure of the energy distribution in modulation frequency domain. As in the MTF case, the MI can also vary between analysis frequency bands. An example of MI computation is depicted in Fig. 7.1. The squared magnitude of a time trajectory (originally Langhans and Strube used $\frac{1}{3}$ octave frequency bands) is analyzed in its modulation frequency components. Then the modulation magnitude frequency response is normalized by the mean energy I, computed from the squared magnitude of the trajectory.

For example, for a temporally sine-wave modulated white noise sequence the MI for all frequency bands will be identical and will consist of a unit pulse located at the frequency of the sine wave.

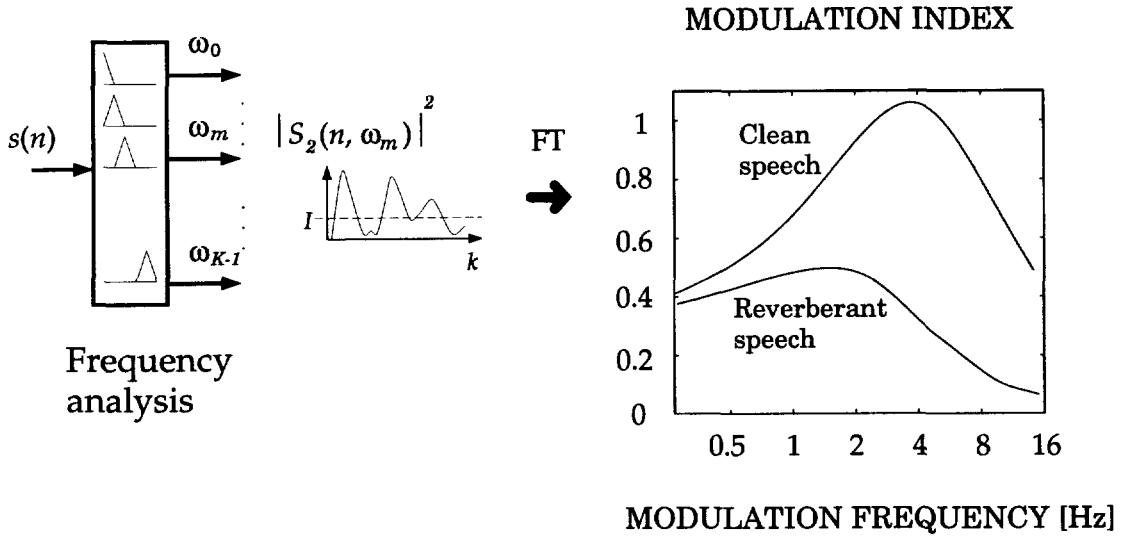


Figure 7.1: *Modulation index computation. After Houtgast and Steeneken (1985).*

7.1.2 Effects of Reverberation on Speech

Linear System

Reverberation can be formally described in terms of a linear system characterized by the impulse response between source and receiver in the room. Reverberant speech is then modeled as the convolution of speech with this impulse response. The effective length (the T_{60} or reverberation time [54]) of this impulse response can be very long. In fact, most of the times it is longer than the interval over which the speech signal can be considered as stationary (i.e. around 20-40 ms).

Deconvolution techniques take advantage of this model and are effective when some knowledge about the room response can be obtained. In situations where several channels and/or a reference signal are available, an estimate of the room's characteristics can be derived, and inverse filtering can be applied to recover the original speech ([47], [59]). In real applications sending a reference signal is not feasible, and current blind identification

techniques to estimate the impulse response from several observations yield poor estimates in the presence of noise [7].

When the impulse response is available, the inverse filtering approaches have had as main problem the non-minimum phase nature of the reverberation process [59], thus making its inversion not a feasible procedure (if the application makes prohibitive the introduction of delay to stabilize the inverse filter). With the availability of multiple microphones (at least 2) exact inverse filtering has been achieved in [38] where the Multi-INput/output Theorem (MINT) was derived. It is rather unfortunate that room impulse responses are hardly available, making the inversion methods impractical.

Envelope Smearing

As speech is produced inside an enclosure, the finer details of its time-intensity distribution are blurred before reaching the listener. This modification results from the superposition of the reflected sound waves with different delays and intensities to the original (direct path) waveform [29].

In the absence of discrete echoes, the effect of such a superposition results in reverberation tails on the energy envelope of the signal. These tails have an approximately exponentially decaying envelope with a time constant determined by the room's dimensions, wall reflectivities and the positions of the source and receiver.

Tails produced by past acoustic events fill in low energy regions between consecutive sounds reducing the modulation depth of the original envelope, and thus modifying its MI [29]. The MTF of the room can be derived from it's impulse response and thus the effect of the room on the MTF of speech is predictable [54]. This motivates the application of inverse MTF's to recover the original modulations present in the original (anechoic) speech. Notice that in this case the phase modifications in the fine structure are not considered.

7.2 Using the MI Concept for Reverberation Reduction

We have achieved a considerable reduction of additive noise by filtering compressed STFTM time trajectories of noisy speech (see Chapter 6). The magnitude frequency response of the data-derived filters showed that, in the presence of additive noise, modulation frequencies characteristic of clean speech (around 4-5 Hz) need to be enhanced, and other frequencies outside this range attenuated. This gave us some indication that the data-derived filters were partially compensating for the deteriorating effects of some disturbances on the MI of speech.

From Fig. 7.1 we observe that the MI of speech is considerably modified in the presence of room reverberation. Our previous experience with the data-derived filters, which are capable of modifying the modulation spectrum, indicates that the data-driven approach can also be used to design a system to reduce reverberation.

7.3 Preliminary Experiments

In this section we describe the preliminary experiments that we performed to understand the effects of two different envelope modification techniques proposed in the past. The modification of the modulation frequency components (or modification of the MI) to reduce the effect of reverberation was the prime motivation behind both techniques presented in [33] and [27]. Those techniques used time trajectory filters designed heuristically [27], or based on analytic forms for the inversion of the MTF of simple reverberation models [33].

7.3.1 High-Pass Filtering of the STFT Power Spectrum

Hirsch reported an improvement on the computer recognition of reverberant speech by using high-pass filtering of the STFT power spectrum trajectories [27]. Improvement of the subjective quality of the reconstructed speech after filtering was also reported and this constitutes the object of our investigation.

We hypothesized that the main effect of the high-pass filtering was due to the fact that, after filtering, a considerable number of points in the resulting power spectra were negative and thus eliminated during rectification. The filter in [27] was a high-pass filter with a real

zero at dc, thus removing the mean of the time trajectories makes an important percentage (depending on the filter characteristics) of the power spectrum energies negative. The necessary rectification step in Fig. 6.1 sets all these values to zero, thus effectively removing them. After filtering, a high percentage of negative values correspond to low energy regions likely to contain reverberation tails. While removal of low spectral energy values reduces the reverberation effects considerably, it may also cause a loss of useful speech information and distortion of the perceived speech signal.

We performed two simple experiments to test our hypothesis. In the first experiment we applied full-wave rather than half-wave rectification (thus negative values were not effectively removed) and found *no* reduction of the reverberation.

In the second experiment we center clipped the STFT power spectrum of reverberant speech below a certain threshold (20% of the maximum value). The result was very similar to that obtained with high-pass filtering.

Although some improvement of the MI is evident after the processing using Hirsch's filter, it appears that the main effect of the high-pass filtering technique is in removing the low-energy spectral values, rather than achieving a restoration of the MI.

7.3.2 Inverting a Theoretical MTF

Langhans and Strube applied a theoretically derived inverse MTF to reduce reverberation [33]. In their method the inverse MTF (IMTF) was applied in critical bands simulated by a weighted sum of the STFT power spectrum trajectories. The IMTF used was the inverse of a first order low-pass characteristic with a cut-off frequency proportional to the reverberation time (T_{60}) considered (this is analytically derived for artificial reverberation [54]). The modulation frequencies above 10 Hz were not allowed to exceed in amplitude above a certain threshold to avoid strong fast fluctuations, and those above 40 Hz were further attenuated.

The results obtained with this technique were not reported to be very satisfactory. We decided to investigate the matter comparing the theoretical curve to the transfer function of the filters obtained from our data-driven approach.

7.4 Technique

In this section we briefly review the RASTA-like filter bank technique and its design from clean and reverberant speech data.

The technique used in Chapter 6 consists of the following steps. First the STFTM of the degraded speech is computed using the STFT. After application of a fixed zero-memory non-linearity, each time trajectory of this new representation is filtered by a data designed RASTA-like filter. After filtering, the result is transformed back to the STFTM domain by applying the inverse non-linearity, and combined with the original short-time phase to yield a resynthesized time domain signal.

Since in the particular case of MI recovery we are after compensation of the short-term power spectrum, we use $\gamma = 0.5$ in Fig. 6.1, i.e. we perform the linear filtering on the short-time power spectrum,

$$|\widehat{S}_2(n,\omega_k)|^2 = \sum_{r=-L}^{L} F_2(r,\omega_k)|X_2(n-r,\omega_k)|^2, \tag{7.1}$$

where $X_2(n,\omega_k)$ corresponds to the STFT of the reverberant data. To compare to other techniques we applied the temporal filtering to the outputs of critical bands simulated by a weighted summation of STFTM components.

7.4.1 Filter Design

As in Chapter 6, the temporal filters are FIR non-causal filters derived by solving the Wiener-Hopf equation for the minimization of the Euclidean distance between the filtered power spectrum time trajectories of the corrupted speech and the corresponding desired trajectories of clean speech [22].

A filter is designed for each frequency channel. For the compensation of the effects of additive noise in Chapter 6, filter lengths were typically chosen to be the average duration of a syllable, i.e. about 200 ms (section 6.3.1). For reducing reverberation we use filters whose length is greater than the reverberation time T_{60} of the impulse response used to corrupt the data.

For the experiments described in the next section, the data used were generated by

convolving clean speech (sampled at 8kHz) with artificial room impulse responses. For these impulse responses, reverberation tails were produced using Schroeder's model (i.e. a decaying exponential envelope modulated by a white noise sequence [54] and early echoes were simulated by randomly spaced non-zero taps with random values [40]. The reason for using this simple model was that its MTF can be analytically derived.

7.5 Experiments

Using the data-driven approach we found the optimal time trajectory filters and compared them with the theoretical transfer function used in [33].

7.5.1 Data-Derived Filters

A set of filters was obtained using artificially reverberated speech in the way described in section 7.4.1. In this experiment we used simulated critical band energies of corrupted and clean speech. These energies were produced by a weighted sum over the STFT power spectrum time trajectories in one third octave rectangular windows. This smoothing was necessary only to compare our results to those described in [33]. The reverberation time used to obtain the results reported here was $T_{60} = 0.75$ s but we got similar results under other test conditions. Filter lengths M were chosen to be at least twice as long as the reverberation time considered. The time trajectory sampling rate in this experiment was set to 250 Hz.

Resulting Filters

Fig. 7.2 shows the filter for the critical band with center frequency at 1 kHz. The filters for other critical bands have very similar shapes (since the MTF produced by the artificial reverberation is the same for all frequency bands). At low modulation frequencies we can see a close correspondence between the data-derived and theoretical frequency responses. However, at higher modulation frequencies the filter characteristics differ significantly: the data-derived filters exhibit a strong low-pass character, suppressing modulation frequencies above 10 Hz.

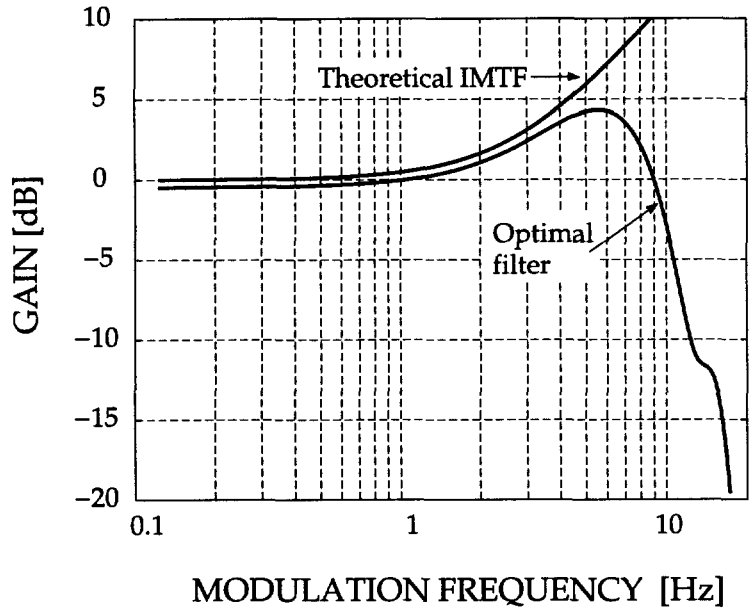


Figure 7.2: *Magnitude frequency response of a data-derived filter (at 1 kHz center frequency band) compared to the theoretical curve.*

We have observed such suppression of higher modulation frequencies in many of the filters which we have designed for noisy and linearly distorted speech [26], [12]. It appears that high modulation frequencies of the short-time spectrum of speech are highly corruptible by many kinds of distortions, and the data-derived filters always tend to alleviate this. The frequency response at high modulation frequency can be explained by the lower signal energy at those particular modulation frequencies in the speech signal. (Notice that Langhans and Strube elected at least not to enhance such higher modulation frequencies in their attempt for the reverberation reduction in spite of the fact that their theoretically derived compensation filter suggested to do so [33]).

The impulse responses of the filters obtained were not symmetric in this case. In contrast to the case of additive noise reduction, the time trajectory filter obtained for this data is acting as an inverse filter since the disturbance shows as a convolution in the time trajectories. This fact explains the limited success obtained when applying the filters to test data with different reverberation parameters.

For subsequent tests we decided to design linear phase filters from the magnitude frequency response obtained with the data. In this way we could test the effectiveness of modulation frequency modification, regardless of the phase distortion introduced by the original filters.

7.5.2 Results

To test the system we reverberated a speech sample with artificial impulse responses. The responses used were different than the impulse response used for training. The reverberation time for this testing responses was set to be similar to the reverberation time of the training data. Our observations on the data-derived filters indicated a strong dependence on the reverberation time, so we did not expect any improvements if the reverberation time of the training and testing data differed significantly.

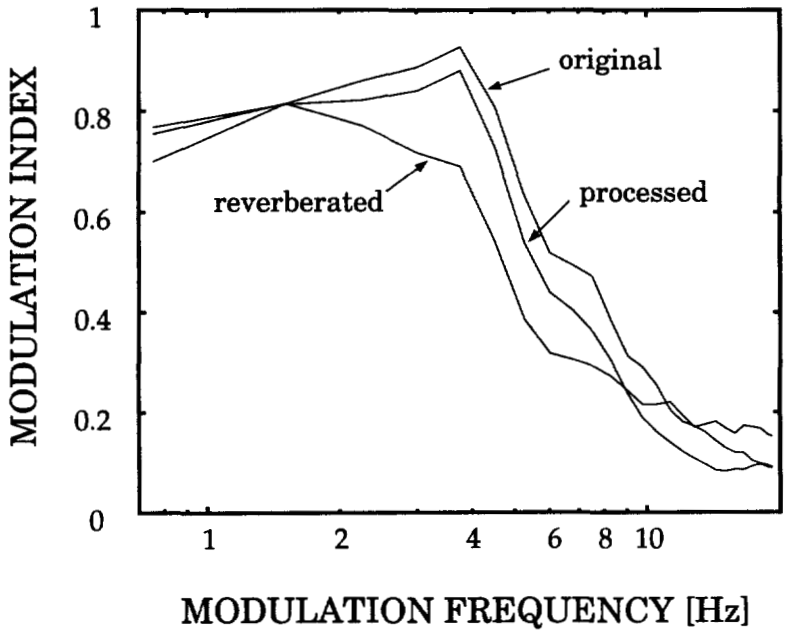


Figure 7.3: *Modulation index at 1 kHz for clean speech, reverberant speech and processed speech.*

After processing the smoothed power spectrum trajectories with the data-derived filters we resynthesized the signal by applying the envelope modification to the original STFT (see [33] and [8] for details). We observed that a reduction of the reverberation was

audible. However, no improvement over using the theoretical IMTF was apparent.

A reasonable question to ask in this case is if the filtering really compensated for the reduction of the modulation index of the corrupted speech. Fig. 7.3 shows the modulation index of the 1kHz centered bands of original speech, reverberant speech, and reconstructed speech after temporal processing with the data-derived filters. Restoration of the modulation frequencies which were suppressed by reverberation is significant in the reconstructed speech and was consistently found also at other frequency bands.

Unfortunately, we must conclude that even when the desired restoration of suppressed modulation frequencies is achieved, the recovery of the modulations alone does not guarantee good quality speech. In this scheme the resynthesis procedure makes use of the original corrupted phase which undoubtedly contributes to the perceived artifacts.

7.6 Summary

We have obtained a filter-bank from training data for processing the simulated critical bands of reverberant speech. Results show that these filters approximate some of the characteristics of the theoretical transfer functions used in the past. Listening tests indicate that an audible reduction of reverberation is achieved but artifacts in the processed speech signal are somewhat severe. Although no formal intelligibility tests were carried out, it is likely that the restoration of the MI might not improve intelligibility in our system. This result indicates that the MI is not necessarily a good indicator of quality or intelligibility of the impaired speech.

Another conclusion we can draw from our results is that the average modulation spectrum, as is representative of the MI, is not as important perceptually as the short-time spectra.

Chapter 8

Data-Driven Filter Design for Channel Normalization in ASR

In this chapter we design a system that will reduce convolutional distortions on the basis of training data. As mentioned above, previous techniques have used ad hoc designed filters for this purpose. For example, the initial ad hoc form of the RASTA filters was optimized on a relatively small series of ASR experiments with noisy telephone digits. Optimizations using ASR experiments are costly and there is no guarantee that the solutions obtained will not be specific to a given ASR problem. Any data-based optimization which would avoid using a specific ASR paradigm is desirable.

Here, using our data-driven design of temporal filters we find a set of filters that resemble filters used in previous filtering techniques confirming their validity for approaching the problem. The evaluation of our technique is based on comparisons against previously used filters (which have been successfully applied to ASR) and not in any particular ASR paradigm.

8.1 Motivation

Relatively unconstrained [1] data-driven systems are the mainstream in today's ASR. These systems acquire their parameter values from large amounts of training data and are susceptible to failure when used in situations that assume different conditions than those encountered during the training.

[1] ASR systems, like HMM-based recognizers, can be highly structured systems but their parameter values may not be constrained.

It is our belief that more knowledge-constrained designs will result in simpler and ultimately more reliable systems. However, if we are to hardwire any constraints into the system, it is crucial that these constraints be based on well tested, reliable and relevant knowledge.

Some reasonable constraints may be implied by properties of the human hearing process and researches have been relatively successful when incorporating them into ASR [23], [24]. On the other hand, it is hard to deny the power of real speech data. Thus, we support using the speech data, as long as they are used in a way to provide permanent and reusable knowledge.

Thinking along these lines, we came to realize that since speech developed to optimally use the properties of human auditory perception, any relevant auditory knowledge may have its counterpart in the structure of the acoustic speech signal. The constraints derived from the data may either correct or support knowledge-based constrained designs.

8.2 Filter Design by Constrained Optimization

In this section we find a set of time trajectory filters for channel normalization by constructing a constrained optimization program. The least squares technique used in previous chapters is not suitable for approaching this problem. This is because if our aim is to achieve channel invariance of the features, any objective function involving a particular channel will yield a processing strategy with poor generalization power.

To approach the problem, the filter design criterion is the minimization of distance between the processed features when they are obtained from speech corrupted by several communication channels.

In the procedure shown in Fig. 8.1, a speech signal $s(n)$ is corrupted by J different channels $H_j(z)$, with $j = 1, 2, ..., J$. After an appropriate feature extraction procedure we have a set of JK corrupted logarithmic time trajectories $X_j(n, k)$, where

$$X_j(n, k) = \log[C_j(n, k)], \tag{8.1}$$

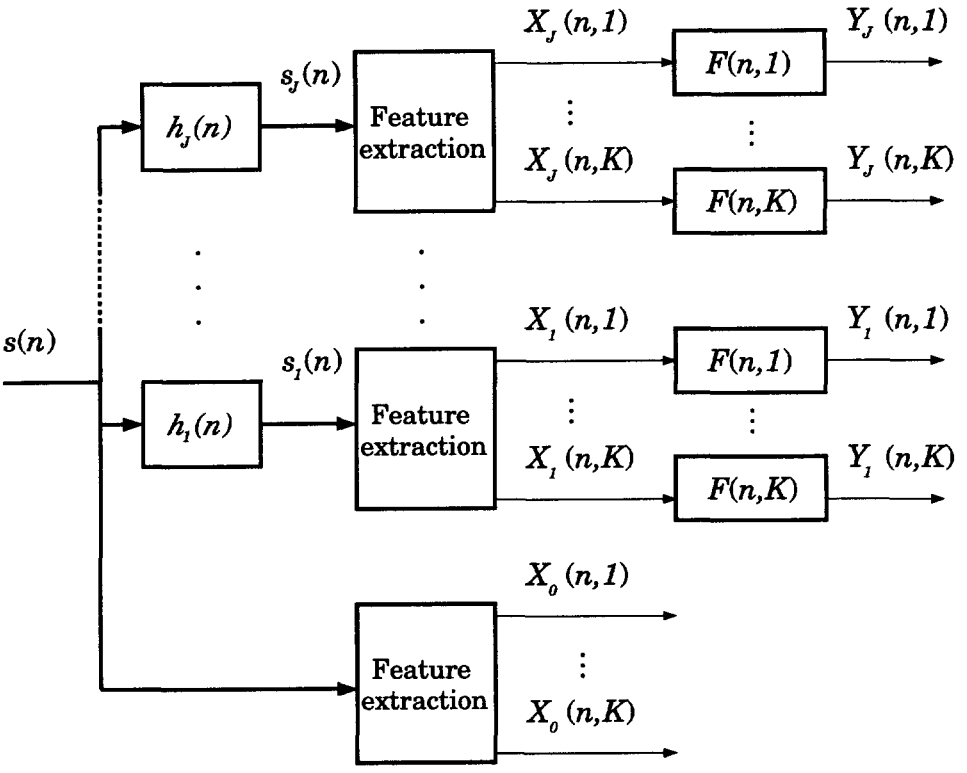


Figure 8.1: *Problem setup block diagram*

with simulated critical bands

$$C_j(n,k) = \sum_{\omega=\omega_i}^{\omega_j} \alpha_k(\omega)|S_j(n,\omega)|, \tag{8.2}$$

where the limits of the summation ω_i and ω_j correspond to the frequency interval over which each critical band k is integrated, and $\alpha_k(\omega)$ is a weighting factor. Notice that we have again dropped the STFT subindex "2" from the input speech STFT and simulated critical bands notation. Throughout this chapter we will use the subindex as a means to enumerate a feature set and not to indicate transformation with respect to an argument as in previous chapters.

From now on we will call $X(n,k)$ the critical band trajectories omitting the term logarithmic. Notice that in Fig. 8.1 we also show a set of trajectories of clean speech denoted by $X_0(n,k)$, which will be used to set the constraints described in the next section.

Objective

Ideally we would like to apply a CIT-MIF modification $F(n,k)$ (implemented as a set of temporal filters) on the time trajectories such that the outputs $Y_j(n,k)$ for the k^{th} feature are as similar as possible to each other, that is, we seek channel independence. A trivial solution to this problem is to set each filter to $F(n,k) = \bar{0}$, so the need to constrain the solution to some reasonable value is obvious. As will be seen later, the constraints will determine specific behavior of the filters.

8.2.1 Technique

Considering that the effect of the channel can be approximated as multiplicative in the short-time frequency domain, and approximately additive in the critical bands (see Chapter 5 and the discussion on this issue in [9]), we chose our CIT-MIF modification to be implemented as FIR filters. The objective function to derive the RASTA-like filter for each critical band can be written as

$$\mathcal{J}_k = E\left\{\sum_{j=1}^{J-1}\sum_{i=j+1}^{J}[Y_j(n,k) - Y_i(n,k)]^2\right\} \tag{8.3}$$

From (8.3) we see that the objective functions are defined as the expected value (with respect to time) of the Euclidean distance between the outputs $Y_j(n,k)$ of the $F(n,k)$ filter produced by $X(n,k)$ for $j = 1,...,J$. These quadratic functions have a global minimum at $F(n,k) = \mathbf{0}$, where $\mathbf{0}$ is a vector with all elements equal to 0. To avoid the trivial solution in which all filter coefficients are set to zero we need to impose a set of constraints.

8.2.2 Experimental Design

To derive the filters, three parallel speech recordings were used. A sample of clean speech was taken from the TIMIT database (approximately 2 minutes). The other two samples were taken from the corresponding speech of the NTIMIT database (telephone channel) and TIMIT recorded through a cellular telephone channel. Auditory frequency band trajectories for the three recordings were computed by a weighted sum of their short-term power spectrum as proposed in [23]. The logarithm of these trajectories was taken to

produce $X_0(n,k)$, $X_1(n,k)$ and $X_2(n,k)$ respectively. The subindex now serves as a label for each database. The same design was applied to each critical band independently. For simplicity we will drop the frequency index k and it should be understood that the following procedure was applied K times, one for each simulated critical band independently. This band independent design is just our choice; the technique is more general and can be applied to a Multi-Input Multi-Output (MIMO) system.

Using (8.3) we find the objective function for $J = 2$

$$\mathcal{J} = E\left\{[Y_1(n) - Y_2(n)]^2\right\}, \tag{8.4}$$

that in matrix notation can be written as

$$\mathcal{J} = E\left\{(\mathbf{x}_1^T(n)\mathbf{f} - \mathbf{x}_2^T(n)\mathbf{f})^T(\mathbf{x}_1^T(n)\mathbf{f} - \mathbf{x}_2^T(n)\mathbf{f})\right\}, \tag{8.5}$$

where

$$\mathbf{x}_j(n) = [X_j(n), X_j(n-1), ..., X_j(n-L+1)]^T,\ j = 1,2$$

and the filter vector

$$\mathbf{f} = [F(0), F(1), ...F(L-1)]^T.$$

Taking the expected value in (8.3) we arrive to our cost function in matrix notation

$$\mathcal{J} = \mathbf{f}^T R_{x_1,x_1}\mathbf{f} + \mathbf{f}^T R_{x_2,x_2}\mathbf{f} - \mathbf{f}^T R^v_{x_1,x_2}\mathbf{f}, \tag{8.6}$$

where$R^v = R + R^T$and R_{x_j,x_i} refers to the cross-correlation matrix between $X_j(n)$ and $X_i(n)$. Minimizing (8.4) leads to the following equation

$$(R_{x_1,x_1} + R_{x_2,x_2} - R^v_{x_1,x_2})\mathbf{f} = 0, \tag{8.7}$$

which has as solution $\mathbf{f} = \mathbf{0}$. It could be argued that if the matrix $(R_{x_1,x_1} + R_{x_2,x_2} - R^v_{x_1,x_2})$ is rank deficient, $\mathbf{f}$ would have a different solution, however there is no reason why such condition should hold. In fact, the experimental design revealed that the matrix is always full rank. It is now obvious why we need to constraint the solution. Two constraints (j=1,2), one for each of the outputs of the filter need to be set.

Constraints

To avoid the trivial solution which sets all output signal values to zero, a reasonable constraint is to restrict the energy at the output of the filter $Y_j(n)$ to be a fraction of the energy of the input signal. While this constraint avoids the trivial solution, it imposes no restriction on the characteristics of the output signal and thus it will be less effective if we need to preserve relevant information about speech. Interestingly, by using this constraint we found that the filters had very similar responses (i.e. band-pass with strong dc suppression and narrow pass-band) as the so called delta cepstrum processing [20].

A more reasonable constraint was found by not allowing the distance between the filter outputs $Y_j(n,k)$ and the original uncorrupted speech features $X_0(n,k)$ (see bottom of Fig. 8.1) to be large. This similarity constraint can be more or less restrictive depending on the amount of error allowed, and the resulting filters will have different characteristics depending on this factor. Notice that the availability of clean speech is necessary only for this particular constraint and is not a general requirement of the technique. Speech corrupted by any another channel (with no strong zeros) can be used as a reference.

The constraints proposed above can be written as:

$$E\left\{[\tilde{X}_0(n) - \tilde{Y}_j(n)]^2\right\} < c_j. \tag{8.8}$$

In these constraints we removed the dc component from the original clean speech features and from the output of the filter to obtain $\tilde{X}_0(n)$ and $\tilde{Y}_j(n)$ respectively. This normalization is needed in order to make a fair comparison of the clean and corrupted features (since adding or removing a constant in the logarithmic domain corresponds to modifying the power of the signals in the linear domain). Writing (8.8) in matrix notation we get

$$P_{\tilde{x}_0} + \mathbf{f}^T R_{x_j,x_j}\mathbf{f} - 2\mathbf{f}^T \mathbf{u}_{\tilde{x}_0,x_j} - \mathbf{f}^T \tilde{\mathbf{x}}_j \tilde{\mathbf{x}}_j^T \mathbf{f} < c_j. \tag{8.9}$$

Here $P_{\tilde{x}_0}$ is the power of $\tilde{X}_0(n)$, $\tilde{\mathbf{x}}_j$ is a vector with all elements equal to the mean of $X_j(n)$, and $\mathbf{u}_{\tilde{x}_0,x_j}$ is the cross-correlation vector between $\tilde{X}_0(n)$ and $X_j(n)$.

The constraints were initially chosen setting c_j 3 dB below the power of the inputs to the filter (i.e. $||{X_j}^2||$) and were varied (decreased) systematically until no feasible solution

could be found. The last feasible point found was chosen to be the solution. We call the result the Constraint-OPtimized (COP) filter. The optimization problem described by (8.4) and (8.9) is non-linear (quadratic) with non-linear (quadratic) constraints and was solved using sequential quadratic programming (SQP)[2] [19].

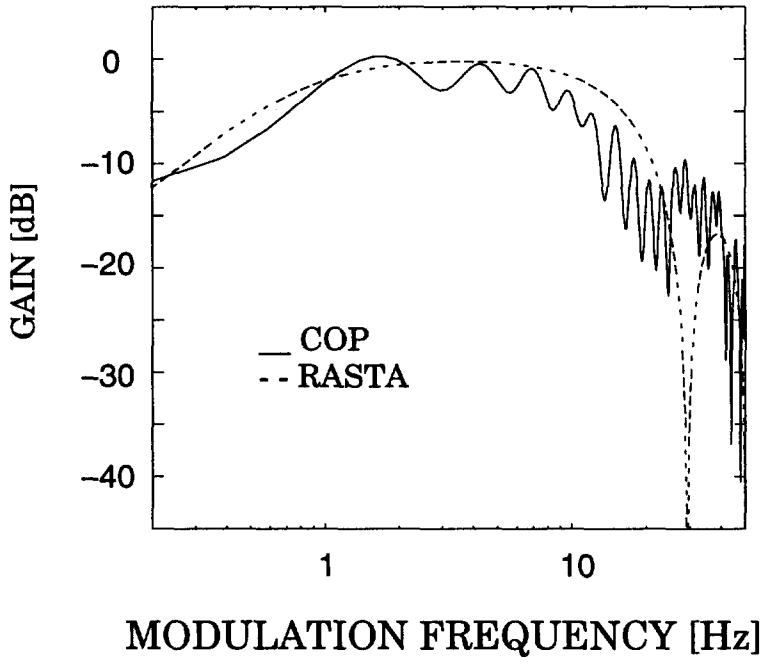


Figure 8.2: *Magnitude frequency response of COP and RASTA filters*

8.3 Results

The resulting filter for the critical band at 1 kHz can be seen in Fig. 8.2. In the figure, the magnitude frequency response of the COP filter is compared to the original RASTA filter. We can observe a close similarity. A dc suppression of modulation frequencies is evident, while attenuation of higher modulation frequencies is also achieved.

Another interesting property that we observed was that the COP filters did not differ significantly at different critical bands. COP filters for several bands are shown in Fig. 8.3. They seem to differ only in the dc gain factor which can be attributed to the differences

[2]We used the Matlab Optimization Toolbox for this purpose.

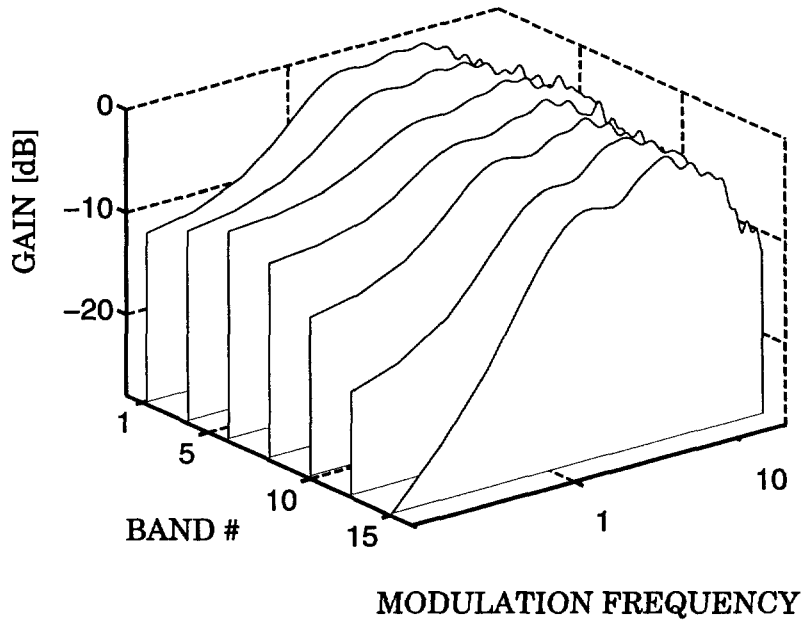


Figure 8.3: *Magnitude frequency response of COP filters for different critical bands*

in energy across frequency components inherent of speech signals.

In terms of the impulse response, we designed the filters to be non-causal by delaying the reference sequence $X_0(n)$ in (8.9). The delay corresponded to half the length of the FIR filter L which we chose to be about 1 s long (101 taps long at a feature extraction rate of 100 Hz). The resulting impulse response was near to symmetric.

8.3.1 Constraint effects

We mentioned before that the constraints determined some specific properties of the resulting COP filters. We observed that as the value of c_j decreased, i.e. tighter constraint, the filter gain increased approaching values close to 0 dB in the passband, and its frequency response became flatter (less dc suppression). For very relaxed constraints the gain was low and the passband of the filter narrowed.

8.4 ASR Experiment

To test the performance of the COP filters in a recognition task we performed an ASR experiment [3]. The experiment was conducted on the Bellcore isolated-digits database which consists of the ten isolated digits (zero, oh, one, two, three, four, five, six, seven, eight, nine) and two control words (yes,no). The training set consisted of 150 speakers and 50 speakers comprised the test set. Each speaker uttered the vocabulary once. The features used were 5^{th} order LPC cepstra and energy, along with their delta and acceleration features. The recognizer was an HTK/HMM-based isolated word recognizer. Each word model consisted of 8 states with 4 mixtures per state.

Two recognizers were trained and tested. One used RASTA filters (same filter at all subbands) and the other used the COP filters described in section 8.3. Both recognizers had similar performance (around 7% word error rate).

8.5 Summary

In this chapter we have derived a filter bank for channel normalization. To design the filters, a non-linear constrained optimization technique was used. The technique used real speech and channel data. The results indicate that the filters resemble those which have been successfully used in ASR recently (RASTA). It is evident that a dc suppression is necessary to approximate the channel independence condition. However, the low-pass response at higher modulation frequencies found in these data-derived filters is not an obvious characteristic and it is consistent with the ad hoc designed RASTA filter.

Although the ASR experiments showed no performance improvement (or deterioration) over RASTA, the usefulness of the work described in the chapter is related to validation and understanding of successful temporal processing strategies that are currently used for channel normalization.

[3]Thanks to Sangita Tibrewala for performing the experiment.

Chapter 9

Multiresolution Channel Normalization for ASR in Reverberant Environments

In this chapter we show that by using high frequency resolution (long-time window) analysis during the channel normalization steps of the feature extraction process, the performance of a speech recognizer under reverberant conditions is significantly increased. The technique is based on multirate signal processing concepts. High frequency resolution (large number of bands) is used at initial analysis stages where normalization is performed. Then, a frequency-time resolution trade-off is used to increase the rate at which the time information is sampled (short-time domain), yielding an appropriate domain to derive ASR features. For a reverberation time of about 0.5 s the new technique achieves significant performance improvement of a speech recognizer under reverberation, while gracefully decreasing performance on clean speech.

Our main contribution in this chapter is that we introduce the concept of using long analysis windows when the transmission channel involved has a long impulse response. While no data-driven designs were performed, the technique is suitable for that approach. The time trajectory filters that we use to prove our concept are fixed mean removal high-pass filters.

9.1 Introduction

Reverberant environments are common and can severely impair the performance of automatic speech recognition systems. One of the main problems is related to the long impulse

response involved in the reverberation process, which in general is longer than the time intervals over which the speech signal is considered stationary.

Conventional channel normalization techniques, such as RASTA and CMS have been successful in reducing the artifacts due to channels such as handset microphones and telephone lines, which in general have short impulse responses and can be considered as invariant or at least slowly varying. As we discussed in Chapter 5, the reason why such techniques work is that with the currently used analysis parameters (20 ms time windows and 10 ms overlap) the effect of the channel can be considered as multiplicative in the short-time frequency representation of speech (see the approximation in (5.17)).

However, the impulse response of a room can be rather long (up to several seconds, depending on wall reflectivities, distance between source and receiver, etc.). The conventional short-time analysis, with its fine time resolution (of the order of 10 ms, inherited from speech coding and dictated by the quasi-stationarity assumptions for speech signals) is not able to capture the impulse response properties in a single frame.

From the frequency domain point of view, frequency resolution of the conventional short-time analysis is not high enough to reflect all details of the transfer function of the reverberant environment. This means that for a given frame, the transfer function is undersampled and the additional frames needed to resolve it make the channel effects closer to convolutional in the time dimension of the STFT, rather than multiplicative in the frequency dimension of the STFT (see (5.12)).

Either way we look at it, the effect of the reverberation can not be approximated as multiplicative within a single analysis window of the conventional short-term analysis. Therefore, techniques which are typically applied in handling convolutional distortions such as RASTA or CMS, and which assume this condition do not perform well on the long impulse responses associated with room acoustics.

9.1.1 Background

In spite of the efforts by many researches during the last 50 years, there has been very little success in reducing reverberation in speech communications. The effects of reverberation on the details of the amplitude and phase of the speech signal are complicated and difficult

to reverse if no knowledge of the room response is available.

In ASR applications, the efforts for reduction of the effects of reverberation have been mainly adaptations of techniques used for reverberation reduction in speech enhancement, such as microphone arrays [36], and channel identification and inversion procedures. Such techniques attempt to recover the speech signal with good perceptual quality and intelligibility.

In ASR there is no need to resynthesize a speech signal, thus the short-time phase of the signal is typically not required, and the exact recovery of the spectral envelope is also not necessary. As a matter of fact, the frequency resolution of some of the most successful ASR analysis techniques (such as PLP [23], or mel cepstral analysis [16]) is rather low. So in ASR, the deterioration of phase or any spectral details of speech caused by reverberation may not be as damaging as in speech enhancement applications.

9.1.2 Problem

As we pointed out at the beginning of this chapter, the main problem that ASR researchers have faced when dealing with reverberation is related to the time-frequency resolution trade-offs of the analysis techniques. It is a fact that short-time spectral information is required in current ASR systems, but if we wish to apply traditional channel normalization to this representation the results may not be satisfactory. However, if classical channel normalization techniques were applied in a medium-time[1] representation, the approximation in (5.17) would be better. The problem is that these medium-time parameters may not have enough temporal resolution for recognition purposes. Next we describe our approach to solve this problem.

9.2 Multiresolution Concept

A signal can be described by many different invertible time-frequency representations. In Fig. 9.1 the signal $x(n)$ has been described by two different time-frequency representations,

[1] We use this term to refer to a short-time analysis where the window is at least 10 times longer than the 20 ms windows of typical short-time analyses.

$X(n_2, \omega_k)$ and $X(n_1, \theta_m)$. If the time-frequency representations are properly sampled (see [3] for a discussion on this point) then the signal $x(n)$ can be recovered from both.

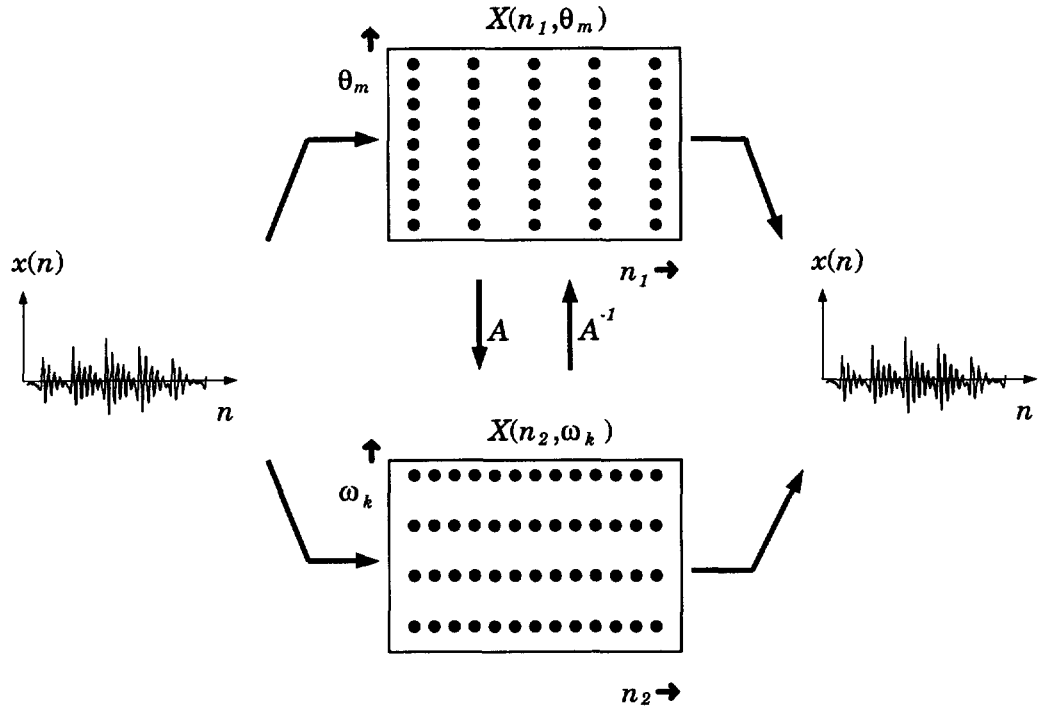


Figure 9.1: *Multiresolution Processing Concept.*

In principle, if both time-frequency representations are linear and invertible, then we can obtain one from the other by a linear transformation matrix [2]. In Fig. 9.1 the transformation matrix is denoted by A. Thus the signal $x(n)$ can be first transformed to the time-frequency representation $X(n_1, \theta_m)$, and recovered from $X(n_2, \omega_k)$ by going through the transform matrix A.

The idea of applying this transformation between time-frequency representations for channel normalization is the following. The impulse response of a room is long compared to the analysis window length commonly used in feature extraction for ASR (e.g. 25 ms).

[2] Although this is intuitively correct we did not prove that this can be accomplished for all possible cases. For the purposes of this dissertation we found that the matrix could be found for the time-frequency representations that we used.

So, in order to satisfy the condition (5.17), which represents the best case for channel normalization, we need to use a long analysis window. We can do this by using a medium-time transform $X(n_1, \theta_m)$. For example, for an impulse response 0.5 s long, an analysis window 2 s long would be adequate to render the channel as an approximately multiplicative term. This representation has low time resolution and high frequency resolution (e.g. the window length is 2^{14} time samples at 8 kHz sampling rate). In this domain we can apply temporal processing as we would when reducing the effects of the handset microphone or telephone channels in conventional ASR.

The problem is now that the normalized parameters are not in a domain appropriate for ASR. To solve this we could go to the time domain signal and then compute the proper short-time transform for ASR, i.e. $X(n_2, \omega_k)$. However, we can go directly to the short-time transform if we apply the transformation matrix A directly to the modified medium-time transform, avoiding the resynthesis of a time-domain signal. We call this last step *partial resynthesis* for reasons that will become clear when we describe the technique formally in section 9.3.

The important point is that the partial resynthesis trades time resolution for frequency resolution. Based on the conceptual idea that we have just described, we now present an outline of the algorithm.

9.2.1 The Algorithm

The algorithm is basically an extension of the traditional short-term analysis and temporal-processing-based channel normalization techniques. It does, however, use rather long effective analysis windows compared to the impulse response of the room. It can be summarized in the following steps:

1. Perform a high frequency resolution (long time window) analysis.
2. Apply temporal processing for channel normalization.
3. Partial resynthesis to obtain a short-time representation.
4. Compute features for ASR.

For step 1 we found that a window at least twice as long as the reverberation time (T_{60}) of the room was adequate. This step yields the proper medium-time representation. The temporal processing in step 2 can be any standard method such as mean removal or some adaptation of RASTA [3]. In principle we could also apply our data-driven filter design for this purpose.

After normalization, the high frequency resolution of the initial medium-time analysis is traded for time resolution which is needed to compute the ASR features. This step involves transforming the modified medium-time representation to a short-time representation. Conceptually this is accomplished by the transformation matrix A. The term *partial resynthesis* indicates the intuition behind this operation. One can think of the partial resynthesis as combining a set of adjacent frequency bands to yield a single frequency band. The procedure reduces the frequency resolution which can then be traded for time resolution (uncertainty principle), yielding a time trajectory with the time-frequency dimensions suitable for ASR.

The next section describes the algorithm in a formal mathematical notation. We chose to use an M^{th} band filter bank model [61], to gain intuition into the time-frequency trade-offs involved in the technique. Also, the filter bank model is very general and leads naturally to the partial resynthesis concept.

9.3 Technique

Now we describe in detail the multiresolution technique for channel normalization. It uses a high frequency resolution (i.e. very long effective analysis window) filter bank analysis at the first stage. So if $s(n)$ is the speech signal, $h(n)$ is the impulse response of the room, and $x(n) = s(n) * h(n)$ is the corrupted speech input, then the filter bank outputs are given by

[3]Note that the RASTA filter was designed for a particular sampling rate of the time trajectories (100 Hz). If we want to apply any RASTA-like filtering we need to consider the sampling rate of the medium-time trajectories, which can be considerably lower.

$$X(n_1, \Omega_m) = \sum_{r=-\infty}^{\infty} x(nM - r) w_m(r), \tag{9.1}$$

where $n_1 = nM$ is the decimated time, $m = 1, 2, ..., M$, and $w_m(n)$ are band-pass filters with discrete center frequencies $\Omega_m = \frac{2\pi m}{M}$ and passbands equal to $(\Omega_m - \frac{\pi}{2M}) < \Omega_m < (\Omega_m + \frac{\pi}{2M})$ to avoid aliasing due to the decimation by M. The variable Ω_m denotes the sampled frequency for the initial high resolution analysis.

Conceptually (9.1) represents a critically sampled M^{th} band filter bank [61], [58]. It should be understood that different types of filter banks may offer other advantages, and we just used this particular one to simplify the visualization of the idea. The M^{th} band filter bank offers the advantage that the modulation properties of the decimation and interpolation operations, together with appropriate band-pass filters, avoids the introduction of frequency modulators in our analysis [15].

If aliasing is neglected, and the number of bands M is high, then the effect of reverberation becomes approximately multiplicative:

$$X(n_1, \Omega_m) \simeq S(n_1, \Omega_m) H(\Omega_m). \tag{9.2}$$

This is because, as we discussed in Chapter 5, if the number of bands is high, then their bandwidths are narrow (see (5.20) and the effective window length is large so that the approximation (9.2) can be made. In fact, for a critically decimated filter bank the effective window is at least as long as the decimation ratio, in this case M.

Once the multiplicative property is achieved, we can apply temporal processing for channel normalization to the bank outputs. We will apply normalization to the envelope of the time trajectories, and keep the original phase (possibly adding a delay to compensate for the group delay introduced by temporal processing) for the partial resynthesis. This is an important point, since disregarding the phase would prevent us from preserving the timing information of the short-time trajectories.

If we write the logarithmic envelope medium-time trajectories as

$$L_x(n_1, \Omega_m) = \log \{|X(n_1, \Omega_m)|\}, \tag{9.3}$$

then the normalized trajectories can be written as

$$\widetilde{S}(n_1, \Omega_m) = \exp\left\{ \sum_{r=-\infty}^{\infty} F(r, \Omega_m) L_x(n_1 - r, \Omega_m) \right\} e^{j\phi(n_1, \Omega_m)}, \tag{9.4}$$

where the time trajectory filters are denoted by $F(r, \Omega_m)$ and the phase term $\phi(n_1, \Omega_m)$ is that of the original medium-time trajectory (9.2).

To arrive to the appropriate short-time representation necessary for feature extraction, we need to increase the time resolution of the medium-time trajectories. This is only possible if we integrate adjacent medium-time trajectories into a lower frequency resolution representation (uncertainty principle [14]).

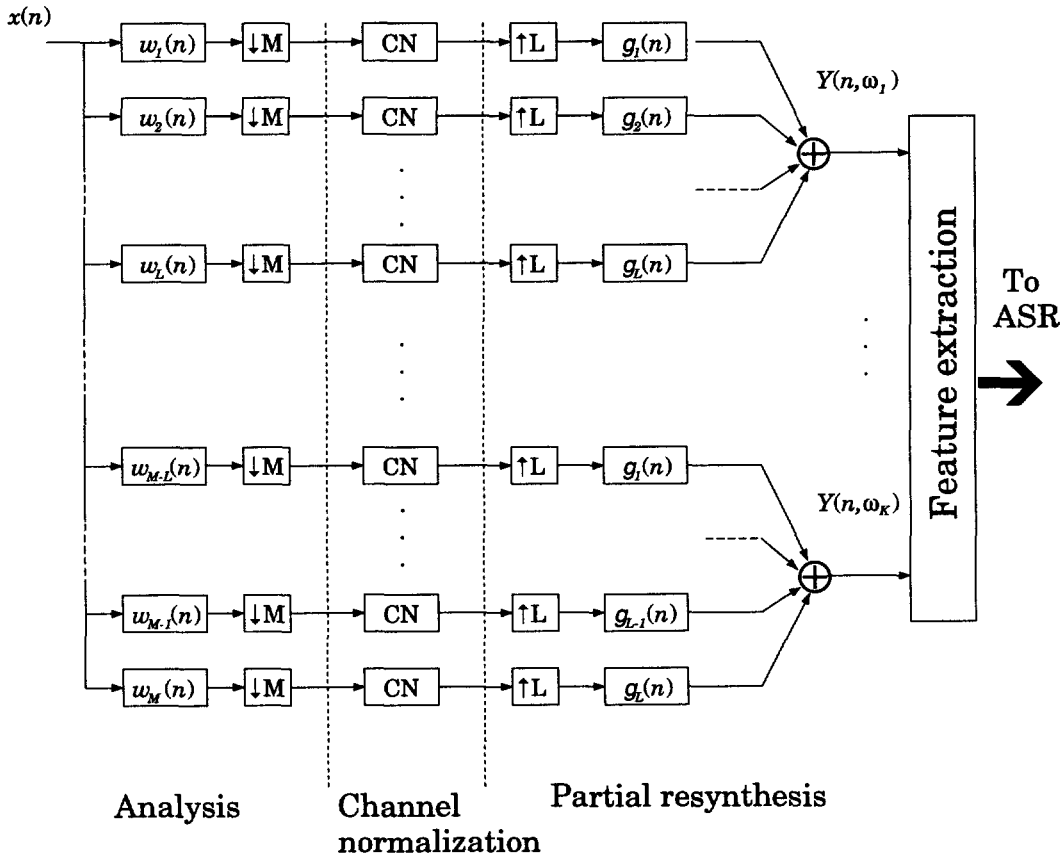


Figure 9.2: *Block diagram of the multiresolution normalization technique.*

The time-frequency resolution trade-off is done with a filter bank-based partial synthesis. The new short-time trajectories can be obtained as

$$Y(n_2, \omega_k) = \sum_{l=1}^{L} \sum_{r=-\infty}^{\infty} \widetilde{S}(r, \Omega_{L(k-1)+l}) g_l(n_1 - rL), \tag{9.5}$$

where $L = M/K$ is an integer, $n_2 = n_1/L$ is the interpolated time, and $g_l(n_1)$ are interpolation bandpass filters (see Fig. 9.2). The center frequencies of these filters are equally spaced and their bandwidths equal to $\frac{\pi}{L}$. So, to obtain the new frequency sampling $\omega_k = \frac{2\pi k}{K}$, we upsample and add L adjacent time trajectories. Upsampling with bandpass filters serves as a frequency modulation step that allocates the interpolated trajectories in the proper frequency band before addition.

The new frequency sampling ω_k is L times coarser than the medium-time frequency sampling Ω_m. The time resolution increase is also equal to L. The new resolution of these normalized trajectories is adequate for feature extraction and subsequent steps in the ASR system (e.g. typical short-time analysis).

A block diagram of the multiresolution normalization technique is depicted in Fig. 9.2. We would like to stress the point that this is a conceptual description and the technique is more general. The sampling rate conversion operations illustrate the changes in resolution which are the key points of the method. Other filter bank configurations are possible, and the decimation/interpolation operations are not strictly required.

9.3.1 Implementation

Analysis

For the current implementation we have used a DFT-based filter bank [15]. The analysis window length was chosen to be larger than at least 2 times the reverberation time (T_{60} in samples) of the room response considered. In practice we could estimate the reverberation time and determine the necessary number of bands. A fixed system with a large initial number of bands may be more practical. The overestimation of the number of bands can only lead us to a better approximation to the multiplicative condition (5.17).

However, such a filter bank would introduce unnecessary delay, not mentioning the arithmetic complexity increment and the slight performance degradation on clean speech (see section 9.4).

Channel Normalization

At the channel normalization step we have experimented with fixed length mean subtraction. The channel normalization was performed on the logarithm of the medium-time magnitude components. After mean removal we reconstruct the complex signal using the phase of the original medium-time representation (see (9.4)). Modification of the phase is not done for the same reasons as in the noise reduction system (Chapter 6). Phase is not a bounded, and its modification can cause destruction of the temporal information during partial resynthesis.

Added to the inherent time delay of the long analysis window, we have to estimate the mean of the trajectories which is a non-causal long-delay procedure. The total delay introduced makes this system less suitable for real time applications.

For the initial testing of our algorithm we used fixed room impulse responses. The use of mean subtraction in this case is justified by the fact that the channel modifies only the dc component of the medium-time modulation spectrum (ignoring analysis artifacts). However, in real time situations, where the room impulse response may be slowly varying, a different time trajectory filtering strategy may be required. The data-driven approach which we demonstrated in this dissertation could be used to obtain the optimal temporal processing strategy. At this point, the lack of realistic data has prevented us from optimizing the system with the data-driven technique.

Partial Resynthesis

After normalizing and reconstructing the medium-time trajectories, we proceed with the partial resynthesis. Since we used a DFT-based analysis bank, the partial resynthesis can be accomplished easily by designing a transformation matrix A that maps the medium-time representation to the short-time representation. The computation of this matrix is given in Appendix C.

Once we have a short-time representation the following steps will depend on the ASR system and feature extraction procedure. In general, ASR does not require the phase at this stage. What is important is that now we have a short-time representation which has

been normalized and in shape to be fed to a standard recognizer.

9.4 Experimental Results

In this section we show the results obtained with the multiresolution normalization technique. First we show how the technique achieves channel independence by inspecting spectrograms. Then we describe a preliminary ASR experiment with results that support our observations and the technique in general.

9.4.1 Channel Independence

For this experiment speech was artificially degraded by convolving it with a fixed impulse response of a reverberant room [4]. The room had a reverberation time of about $T_{60} = 0.56$ seconds. We applied the multiresolution technique using the parameters in Table 9.1. The DFT-based filter bank used a 2 s Hanning window with 50% frame overlap.

Table 9.1: Multiresolution Normalization Parameter Values

Parameter	Value
Reverberation time T_{60}	0.56 s
Mean computation interval	10 s
Sampling frequency	8 kHz
Analysis window length	2 s
Medium-time sampling frequency	1 s
Number of medium-time bands M	16384
Interpolation ratio L	64
Short-time sampling frequency	10 ms
Number of short-time bands K	256

The normalization used was mean subtraction over a sample length of 10 s. The transformation matrix A for partial resynthesis transforms each high frequency resolution frame from the medium-time representation $X(n_1, \theta_m)$ to a set of low frequency resolution frames of the short-time representation $X(n_2, \omega_m)$ (see Appendix C).

[4] The room impulse response used was obtained in the varechoic chamber at Bell Laboratories in Murray Hill. The data were made available by Jim West and Gary Elko.

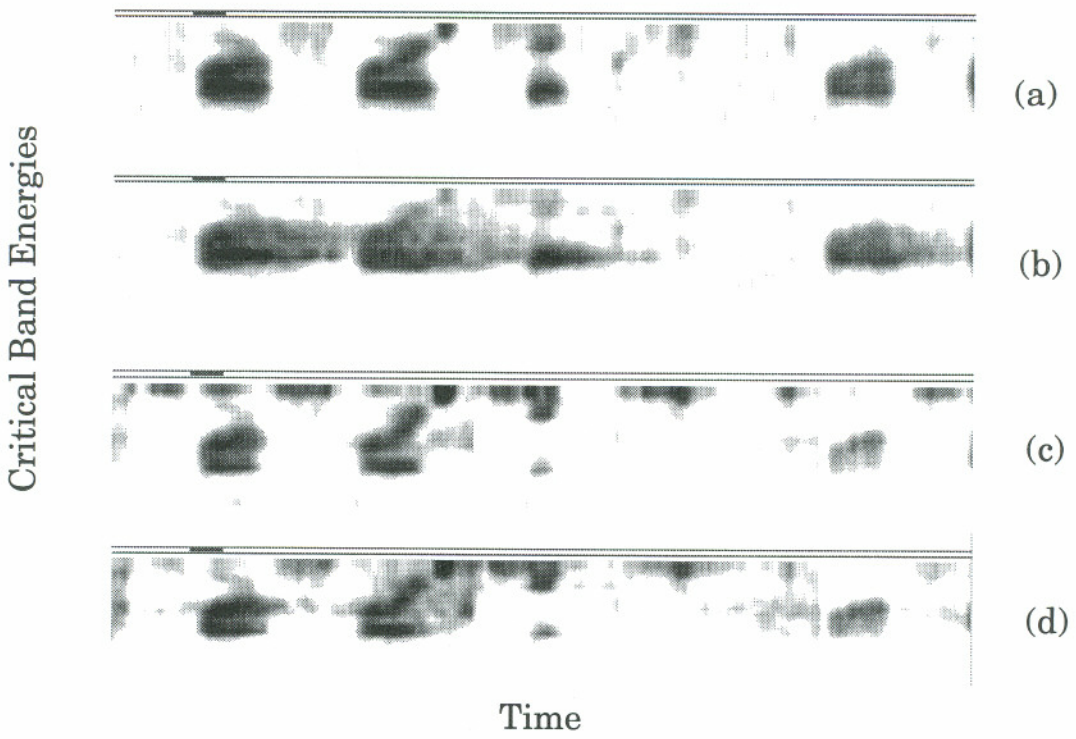


Figure 9.3: *Channel independence results for multiresolution normalization. Critical band energy spectrograms of (a) clean and (b) the corresponding reverberant speech. Critical band spectrograms of (c) clean and (d) reverberant speech after multiresolution normalization.*

After partial resynthesis, we integrated the resulting short-time magnitude into critical bands using the simulated filters reported in [23]. We applied the procedure to the clean and reverberated speech samples. For comparison we computed a similar critical band spectrum directly from the clean and reverberated speech but without applying any processing.

In Fig. 9.3(a) and (b) we show critical band energy spectrograms of the clean and the reverberant speech respectively. Below we show the critical band energies for (c) clean and (d) reverberant speech after the multiresolution normalization was applied. Similar results were observed in a variety of experiments where we tested the channel independence for different reverberation times.

We can observe that the reverberation causes a considerable time smearing of the features (Fig. 9.3(b)). In contrast, if the multi-resolution processing is applied, the clean

and processed features are very similar, thus indicating that the technique could bring advantages in improving the performance of a speech recognizer, i.e. the features are more channel independent.

9.4.2 ASR Experiments

We conducted preliminary speech recognition experiments to test the performance of the new technique [5]. The baseline system was an HMM/MLP hybrid speaker independent recognizer trained on the numbers database corpus from CSLU. The database consists of telephone-quality continuous digit strings. The features used were 8^{th} order RASTA-PLP cepstral coefficients and first order delta coefficients.

Table 9.2: % Word Recognition Error

	BASELINE	MULTIRESOLUTION
Clean	8.6	13.5
Reverberant	34.8	22.8

The same recognizer was then trained on the features derived by the new technique from clean speech. Both recognizers were tested with reverberant data. The results in Table 9.2 show that for reverberant data there is a considerable performance improvement, 55% reduction in error rate over the conventional baseline, when the new multiresolution technique is applied. However, we can also observe a slight degradation on the performance for clean speech.

To the best of our knowledge, the only technique that has been compared to ours is based on the modulation spectrogram [32]. In a personal communication with the authors of that work, they reported a word recognition error of 10.2% on clean and 29.3% on reverberant speech, using the same data of our experiment.

[5]Thanks to Sangita Tibrewala for conducting the experiment

Experiments Conclusion

Further optimization of the technique parameters and the corresponding recognition experiments might be needed to improve the performance obtained in this preliminary test. Properties of the new features obtained with multiresolution normalization are not yet investigated. While the appear to be very similar and have about the same information content of regular critical band energies, their discriminative power may not be as good. The original medium-time phase was not modified in these experiments. It is not clear how should it be manipulated, and this fact undoubtedly limits the effectiveness of our system.

Another possibility is that the filter bank structure and parameters used might still need to be further optimized. We observed that the power spectrum trajectories had some artifacts related to the aliasing of the particular filter bank utilized in this experiment. In any case the recognition improvement for reverberant speech is significant and suggests that multiresolution normalization is a feasible technique which deserves further work and attention from the ASR community.

9.5 Summary

In this section we have described the motivation and implementation of a multiresolution channel normalization technique. The technique is able to overcome the problems related with long impulse responses found when traditional normalization procedures are applied to the short-time representation of speech.

Recognition experiments indicate that if an algorithmic delay of a few seconds is permitted, the technique alleviates the effects of reverberation improving the recognition performance. A degradation of accuracy on clean speech was found in the preliminary experiments. This suggests that the technique reduces the discriminability of speech. However, the filter bank parameters were not optimized and the reduction of performance can also be attributed to analysis artifacts.

It was shown that the technique is basically a generalization of temporal processing, where the time and frequency resolution of the time trajectories is exchanged in order

to, first obtain the desired conditions for normalization, and then generate a short-time representation adequate to ASR.

While no data were used to design the temporal filters in our experiments with fixed impulse responses, the optimization of the system for more realistic data (time-varying impulse responses) could in principle be achieved through our data-driven approach.

Chapter 10

Conclusion and Future Directions

In this dissertation we showed that temporal processing of spectral features is a relatively new and interesting domain, and a fertile ground for innovative applications in the area of speech processing. In our attempt to fulfill this purpose, we developed a series of algorithms which alleviate the detrimental effects of several environmental factors on speech communications.

As one our contributions to the speech processing field, we performed a detailed analysis of several properties of temporal processing setting a theoretical background which was lacking in the understanding of this technique.

Another major contribution was the development of techniques to design temporal filters from realistic data. The data-driven approach which we pursued in this work proved to be of great value, not only in the optimization of our algorithms, but in increasing our understanding of the temporal properties of speech signals in adverse environments.

While our work was focused on the human speech signal, the techniques and theory described here can be applied to other signals. The hope of this author is that this work will stimulate an interest in temporal processing, not only for speech researchers, but for scientists and engineers working in other areas where these concepts could be advantageously applied.

10.1 Summary and Future Work

Based on prior work on temporal processing in the area of automatic speech recognition [24], we applied linear filters to the time trajectories of speech. An interesting and useful

approach was to derive temporal filters from real data. Previous investigators had used ad hoc filters, and the data-driven design opened a whole new set of possibilities for temporal processing. The analysis of the resulting filters raised many questions regarding the properties of temporal processing. In this dissertation we answered some of these questions and with the new knowledge designed useful speech processing algorithms.

Next we will comment on our accomplishments, summarize our contributions, and indicate possible future research on the applications described in the dissertation.

10.1.1 Noise Reduction for Speech Enhancement

Noise reduction for speech enhancement has been a topic of interest for many years. In section 6.1 we discussed this problem and briefly described common approaches to deal with it. What we consider to be a favorable outcome of our research in this topic is that we were able to design a system which is tailored to the speech signal. By this we mean that the data-driven design procedures used yielded a processing which exploited specific properties of speech, namely the behavior of the modulation spectrum of speech under different noise conditions.

An issue that we did not address in detail in this dissertation was the formal perceptual evaluation of the system. Even when the disturbance level is always reduced by our system and our informal listening tests indicate quality improvement, we can not conclude on the basis of a formal evaluation. Future research in this system should include perceptual evaluations as well as the investigation of ways of reducing or modifying the residual noise.

Real time implementation issues were not considered in detail in our work. In principle our system has a short algorithmic delay and requires simple basic signal processing operations available in any digital signal processor. Interesting research would be its implementation in speech coders, where algorithmic delays must be kept very low.

Another topic which we just briefly studied was the implementation of temporal processing by non-linear systems, such as artificial neural networks. Research in this area is still open and may yield improved noise reduction, specially if one utilizes some adaptive strategy such as the one we described in section 6.5.

10.1.2 Reverberation Reduction for Speech Enhancement

Reducing the effects of reverberation from a reverberated speech signal has been a topic of intense research in the past, and unfortunately there are very few approaches which yield acceptable results.

In our study of temporal filtering we found that reverberation reduction is a natural extension of the noise reduction algorithm. Reverberation modifies the modulation spectrum in a characteristic manner. We found that applying our data-designed filters indeed modified the modulation spectrum in the expected way. However, for the reasons that we mentioned in section 7.5.2 the system did not improve the speech in a noticeable way.

For future work, the analysis parameters should be investigated. As we showed in Chapter 9, promising results in ASR are obtained for long impulse responses when the analysis windows are longer. A topic of future research is indeed the application of the multiresolution concepts to the reverberation reduction problem.

10.1.3 Data-Driven Design of Temporal Filters for Channel Normalization

As another extension of data-driven temporal processing design we considered channel normalization filters for ASR. In this dissertation we developed a technique to automatically derive filters from training data, regardless of any particular speech recognition paradigm. The value of the technique is in that it avoids cumbersome and time consuming ASR experiments needed to optimize the time trajectory filters, and it provides an understanding of the conditions necessary for the feature trajectories to be channel independent.

Analysis of the resulting filters also provide us with a better understanding of the reasons why previously used filters, which were heuristically derived, have been successful.

The set of constraints that we imposed on the filter design were just a few among a large number of alternatives. Work in the selection of other meaningful constraints needs to be considered for the future. The features used, as well as the data, were our choice. Results may be different if other features and/or databases are used (e.g. other ASR system requirements), and future research should also consider these differences.

The optimization technique that we used to solve the constrained optimization program

was chosen for its availability and known capabilities to handle non-linear problems. If other concerns, such as speed or accuracy are important, then future work should focus on the optimization procedure.

As our cost function we used a mean squared error difference between time trajectories. Other cost functions based on maximizing discriminability between classes (e.g. linear discriminant analysis), should be pursued. In fact, ongoing work in our laboratory is focusing on this kind of cost functions [12].

10.1.4 Multiresolution Channel Normalization for Reverberation Reduction in ASR

The effect of room acoustics on the speech signal is detrimental to ASR, and new applications in hands-free environments require a solution to this problem. The multiresolution technique which we developed in Chapter 9 is an important contribution towards the solution of the problem. As a generalization of temporal processing, our research in multiresolution normalization provided us with alternative perspectives on how to approach certain problems related to traditional speech analysis techniques.

The technique suffers from the inherent delay of temporal processing. What makes it more or less practical in real applications will be determined by the length of the impulse response of the channel, since this will determine the minimum window length. Also, the type of filter used will introduce delay. In the case of mean subtraction, the delay could make the technique completely inadequate for real time situations. However, if a causal high-pass filter with short impulse response could be used instead, the algorithmic delay could be significantly reduced.

In this dissertation we did not apply the data-driven approach to design the temporal filters. Our goal was mainly to prove the concept of using long windows. The choice of the fixed mean removal filters simplified our experiments and allowed us to test and optimize other parts of the system, necessary to prove the concept. A data-driven filter design technique, like the one presented in Chapter 5, can be applied to optimize the multiresolution technique This is where we consider that much of the future efforts should be focused on.

Multiresolution normalization, while theoretically well based, has implementation issues that still need to be studied. Future work is needed to improve the analysis and partial resynthesis filter banks. Aliasing and noise have not been considered in our analysis, and while our experiments served to prove our idea, we believe that the recognition results could be significantly improved by further investigating these problems.

Bibliography

[1] ALLEN, J. B. Short-term spectral analysis and synthesis and modification by discrete Fourier transform. *IEEE Trans. Acoustics, Speech, and Signal Processing ASSP-25*, 3 (June 1977), 235–238.

[2] ALLEN, J. B., BERKELEY, D., AND BLAUERT, J. Multimicrophone signal processing technique to remove room reverberation of speech signals. *Journal of the Acoustical Society of America 62*, 2 (October 1977), 912–915.

[3] ALLEN, J. B., AND RABINER, L. R. A unified approach to short-time Fourier analysis and synthesis. *Proceedings of the IEEE 65*, 11 (November 1977), 1558–1564.

[4] ARAI, T., PAVEL, M., HERMANSKY, H., AND AVENDANO, C. Intelligibility of speech with filtered time trajectories of spectral envelopes. *Proceedings of the International Conference on Speech and Language Processing 1996* (October 1996), 2490–2493.

[5] ATAL, B. S. Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification. *Journal of the Acoustical Society of America 55*, 6 (June 1974), 1304–1312.

[6] ATAL, B. S., AND HANAUER, S. Speech analysis and synthesis by linear prediction of the speech wave. *Journal of the Acoustical Society of America 50*, 3 (April 1971), 637–655.

[7] AVENDANO, C., BENESTY, J., AND MORGAN, D. R. A least squares approach to blind identification of multichannel FIR filters using component normalization. *Technical Memorandum BL011331-970203-01TM, Lucent Technologies* (March 1996).

[8] AVENDANO, C., AND HERMANSKY, H. Study on the dereverberation of speech based on temporal envelope filtering. *Proceedings of the International Conference on Speech and Language Processing 1996* (October 1996), 889–892.

[9] AVENDANO, C., AND HERMANSKY, H. On the effects of short-term spectrum smoothing in channel normalization. *to appear in IEEE Trans. on Speech and Audio Processing* (July 1997).

[10] AVENDANO, C., HERMANSKY, H., VIS, M., AND BAYYA, A. Adaptive speech enhancement using frequency-specific SNR estimates. *Proceedings of the IEEE Third Workshop Interactive Voice Technology for Telecommunications Aplications* (October 1996), 65–68.

[11] AVENDANO, C., HERMANSKY, H., AND WAN, E. A. Beyond Nyquist: Towards the recovery of broad-bandwidth speech from narrow-bandwidth speech. *Proceedings of the 4th European Conference on Speech Communication and Technology EUROSPEECH'95 I* (September 1995), 165–168.

[12] AVENDANO, C., VAN VUUREN, S., AND HERMANSKY, H. Data-based RASTA-like filter design for channel normalization in ASR. *Proceedings of the International Conference on Speech and Language Processing 1996* (October 1996), 2087–2090.

[13] BOLL, S. F. Suppression of acoustic noise in speech using spectral subtraction. *IEEE Trans. Acoustics, Speech, and Signal processing ASSP-27*, 2 (April 1979), 113–120.

[14] COHEN, L. *Time-Frequency Analysis.* Prentice Hall, 1995.

[15] CROCHIERE, R. E., AND RABINER, L. L. *Multirate Digital Signal Processing.* Prentice Hall, 1983.

[16] DAVIS, S. B., AND MERLMENSTEIN, P. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Trans. Acoustics, Speech, and Signal processing ASSP-28*, 4 (August 1980), 357–366.

[17] EPHRAIM, Y. Statistical-model-based speech enhancement systems. *Proceedings of the IEEE 80*, 10 (October 1992), 1526–1555.

[18] FLANAGAN, J. L. Parametric coding of speech spectra. *Journal of the Acoustical Society of America 68*, 2 (August 1980), 412–419.

[19] FLETCHER, R. *Practical Methods of Optimization.* John Wiley and Sons, 1991.

[20] FURUI, S. Speaker-independent isolated word recognition using dynamic features of speech spectrum. *IEEE Trans. on Acoustics, Speech and Signal Processing ASSP-34*, 1 (February 1986), 52–59.

[21] GRIFFIN, D. W., AND LIM, J. S. Signal estimation from modified short-time Fourier transform. *IEEE Trans. Acoustics, Speech, and Signal processing ASSP-32*, 2 (April 1984), 236–243.

[22] HAYKIN, S. *Adaptive Filter Theory.* 2nd ed., Engelwood Cliffs, 1991.

[23] HERMANSKY, H. Perceptual linear predictive (PLP) analysis of speech. *Journal of the Acoustical Society of America 87*, 4 (April 1990), 1748–1752.

[24] HERMANSKY, H., AND MORGAN, N. RASTA processing of speech. *IEEE Trans. on Speech and Audio Processing 2*, 4 (October 1994), 578–589.

[25] HERMANSKY, H., MORGAN, N., AND HIRSCH, H.-G. Recognition of speech in additive and convolutional noise based on rasta spectral processing. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing 1993 II* (April 1993), 83–86.

[26] HERMANSKY, H., WAN, E. A., AND AVENDANO, C. Speech enhancement based on temporal processing. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing 1995 I* (May 1995), 405–408.

[27] HIRSCH, H. G. *Signal Processing IV: Theories and Applications. Automatic Speech Recognition in Rooms.* J.L Lacome, A. Chehilian, N.martin and J.Malbos (editors), Elsevier Science Publishers B. V., 1988.

[28] HIRSCH, H. G. *Estimation of noise spectrum and its application to SNR estimation and speech enhancement.* Tech. Report TR-93-012, International Computer Science Institute Berkeley, CA., 1993.

[29] HOUTGAST, T., AND STEENEKEN, H. J. M. A review of the MTF concept in room acoustics and its use for estimating speech intelligibility in auditoria. *Journal of the Acoustical Society of America 77*, 3 (March 1985), 1069–1077.

[30] KAILATH, T. *Channel Characterization:Time-Variant Dispersive Channels, in Lectures on Communication System Theory.* Elie J. Baghdady, editor, McGraw-Hill, 1961.

[31] KANG, G., AND FRANZEN, L. Quality improvement of LPC-processed noisy speech by using spectral subtraction. *IEEE Trans. Acoustics, Speech, and Signal processing ASSP-37*, 6 (June 1989), 939–942.

[32] KINGSBURY, B. E., MORGAN, N., AND GREENBERG, S. Improving asr performance for reverberant speech. *to appear in Proceedings of the ESCA Tutorial and Research Workshop on Robust Speech Recognition for Unknown Communication Channels* (1997).

[33] LANGHANS, T., AND STRUBE, H. W. Speech enhancement by nonlinear multiband envelope filtering. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing 1982 I* (April 1982), 156–159.

[34] LIM, J. S. Spectral root homomorphic deconvolution system. *IEEE Trans. Acoustics, Speech, and Signal processing ASSP-27*, 3 (June 1979), 223–232.

[35] LIM, J. S., AND OPPENHEIM, A. V. Enhancement and bandwidth compression of noisy speech. *Proceedings of the IEEE 67*, 12 (December 1979), 1586–1604.

[36] LIN, Q., YUK, D., DE VRIES, B., PARSON, J., AND FLANAGAN, J. Robust distant-talking speech recognition. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing 1996 I* (1996), 21–24.

[37] MAKHOUL, J. Linear prediction: A tutorial review. *Proceedings of the IEEE 63*, 4 (April 1975), 561–580.

[38] MIYOSHI, M., AND KANEDA, Y. Inverse filtering of room acoustics. *IEEE Trans. on Acoustics, Speech and Signal Processing ASSP 36*, 2 (February 1991), 145–152.

[39] MOORE, B. C. J. *An Introduction to the Psychology of Hearing.* Academic Press, 1992.

[40] MORGAN, N. *Personal communications* (1996).

[41] NADEU, C., AND JUANG, B.-H. Filtering of spectral parameters for speech recognition. *Proceedings of the International Conference on Speech and Language Processing 1994* (1994), 1927–1930.

[42] NAWAB, S. H., AND QUATIERI, T. F. *Advanced Topics in Signal Processing: Short-Time Fourier Transform.* Prentice Hall, J. S Lim and A. V. Oppenheim, editors, 1988.

[43] NELSON, A. T., AND WAN, E. A. Neural speech enhancement using dual extended kalman filtering. *to appear in Proceeding Neural Information Processing Systems Conference NIPS'96* (1996).

[44] NEUMEYER, L. G., DIGALAKIS, V. V., AND WEINTRAUB, M. Training issues and channel equalization techniques for the construction of telephone acoustic models using a high quality speech corpus. *IEEE Trans. on Speech and Audio Processing 2*, 4 (October 1994), 590–597.

[45] OPPENHEIM, A. V., AND SCHAFER, R. W. *Discrete-Time Signal Processing.* Prentice Hall, 1989.

[46] OPPENHEIM, A. V., SCHAFER, R. W., AND STOCKHAM, T. G. Nonlinear filtering of multiplied and convolved signals. *Proceedings of the IEEE 56*, 8 (August 1968), 1264–1291.

[47] PETROPULU, A. P., AND SUBRAMANIAM, S. Cepstrum based deconvolution for speech dereverberation. *Proceedings IEEE International Conference on Acoustics, Speech and Signal Processing 1994 I* (1994), 9–13.

[48] PORTER, J. E., AND BOLL, S. F. Optimal estimators for spectral restoration of noisy speech. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing 1984 I* (1984), 18A.2.1–18A.2.4.

[49] PORTNOFF, M. Time-frequency representation of digital signals and systems based on short-time Fourier analysis. *IEEE Trans. on Acoustics, Speech and Signal Processing ASSP-28*, 1 (February 1980), 55–69.

[50] RABINER, L. R., AND SCHAFER, R. W. *Digital Processing of Speech Signals.* Prentice Hall, 1978.

[51] ROSENBERG, A. E., LEE, C. H., AND SOONG, F. K. Cepstral channel normalization techniques for hmm-based speaker verification. *Proceedings of the International Conference on Speech and Language Processing 1994* (1994), 1835–1838.

[52] ROUCOS, S. R., AND WILGUS, A. M. High quality time-scale modification from speech. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing 1985 I* (April 1985), 493–496.

[53] SCHAFER, R. W., AND RABINER, L. R. Design and simulation of a speech analysis-synthesis system based on short-time Fourier analysis. *IEEE Trans. on Audio and Electroacoustics AU-21*, 1 (June 1973), 165–174.

[54] SCHROEDER, M. R. Modulation transfer functions: Definition and measurement. *Acoustica 49* (1981), 179–182.

[55] SCHUSSLER, H., AND STEFFEN, P. *Advanced Topics in Signal Processing: Some Advanced Topics in Filter Design.* Prentice Hall, J. S Lim and A. V. Oppenheim, editors, 1988.

[56] STEPHENNE, A., AND CHAMPAGNE, B. Cepstral prefiltering for time delay estimation in reverberant environments. *Proceedings IEEE International Conference on Acoustics, Speech and Signal Processing 1995*, 5 (May 1995), 3055–3058.

[57] STOCKHAM, T. G., CANNON, T. M., AND INGEBRETSEN, R. B. Blind deconvolution through digital signal processing. *Proceedings of the IEEE 63*, 4 (April 1975), 678–692.

[58] STRANG, G., AND NGUYEN, T. *Wavelets and Filter Banks.* Wellesley-Cambridge Press, 1996.

[59] TOHYAMA, M., LYON, R. H., AND KOIKE, T. Pulse waveform recovery in a reverberant condition. *Journal of the Acoustical Society of America 91*, 5 (May 1992), 2805–2812.

[60] TOHYAMA, M., LYON, R. H., AND KOIKE, T. Source waveform recovery in a reverberant space by cepstrum dereverberation. *Proceedings IEEE International Conference on Acoustics, Speech and Signal Processing 1993* (1993), I 157–160.

[61] VAIDYANATHAN, P. P. *Multirate Systems and Filter Banks.* Prentice Hall, 1993.

[62] WANG, D. L., AND LIM, J. S. The unimportance of phase in speech enhancement. *IEEE Trans. Acoustics, Speech, and Signal Processing ASSP-30*, 4 (August 1982), 679–681.

[63] WATKINS, A. J. Central auditory mechanisms of perceptual compensation for spectral-envelope distortion. *Journal of the Acoustical Society of America 90*, 6 (December 1991), 2942–2955.

Appendix A

Derivation of (3.7) and (3.8)

Here we derive the proof of (3.7) and (3.8). Recall the synthesis equation (3.6)

$$y(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi} \sum_{l=-\infty}^{\infty} q(n-l) \sum_{r=-\infty}^{\infty} F_2(l-r,\omega) S_2(r,\omega) e^{j\omega n} d\omega.$$

Replacing the definition of the STFT of $s(n)$ (2.6) in the equation above yields

$$y(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi} \sum_{l=-\infty}^{\infty} q(n-l) \sum_{r=-\infty}^{\infty} F_2(l-r,\omega) \sum_{m=-\infty}^{\infty} w(r-m) s(m) e^{-j\omega m} e^{j\omega n} d\omega, \quad \text{(A.1)}$$

interchanging the order of summation and integration we get

$$y(n) = \sum_{m=-\infty}^{\infty} \sum_{r=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} w(r-m) q(n-l) \left[\frac{1}{2\pi}\int_{-\pi}^{\pi} F_2(l-r,\omega) e^{j\omega(n-m)} d\omega\right] s(m), \quad \text{(A.2)}$$

and recognizing the term in brackets as the inverse Fourier transform of the modification $F_2(n,\omega)$ (see also (3.9)) we obtain

$$y(n) = \sum_{m=-\infty}^{\infty} \sum_{r=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} w(r-m) q(n-l) f(l-r, n-m) s(m). \quad \text{(A.3)}$$

By making the changes of variables $m' = n-m$, $r' = n-r$, and $l' = n-l$ and changing the order of the summations we finally get

$$y(n) = \sum_{m'=-\infty}^{\infty} \sum_{r'=-\infty}^{\infty} w(m'-r') \sum_{l'=-\infty}^{\infty} q(l') f(r'-l', m') s(n-m'), \quad \text{(A.4)}$$

from which (3.7) and (3.8) immediately follow.

Appendix B

Derivation of (4.12)

To derive (4.12) first recall the synthesis equation (4.11)

$$y(n) = \sum_k \sum_{r=-\infty}^{\infty} f_{\omega_k}(n, n-r) S_2(r, \omega_k) e^{j\omega_k n},$$

Introducing the definition of the STFT (2.6) into the equation above we get

$$y(n) = \sum_k \sum_{r=-\infty}^{\infty} f_{\omega_k}(n, n-r) \sum_{m=-\infty}^{\infty} w(r-m) s(m) e^{-j\omega_k m} e^{j\omega_k n}, \tag{B.1}$$

after changing the summation order and making the changes of variable $m' = n - m$ and $r = r'$ we obtain

$$y(n) = \sum_{m'=-\infty}^{\infty} s(n-m') \sum_k \sum_{r'=-\infty}^{\infty} f_{\omega_k}(n, n-r') w(m'-n+r') e^{j\omega_k m'}, \tag{B.2}$$

which can be further rearranged by changing variables $r = n - r'$ and $m' = m$ to yield

$$y(n) = \sum_{m=-\infty}^{\infty} s(n-m) \sum_k \left[\sum_{r=-\infty}^{\infty} f_{\omega_k}(n, r) w(m-r) \right] e^{j\omega_k m}. \tag{B.3}$$

We can see from (B.3) that the term in brackets can be interpreted as a time-varying filter which now is a function of n and m. By expanding that term we obtain

$$\sum_{r=-\infty}^{\infty} f_{\omega_k}(n, r) w(m-r) = \sum_{r=-\infty}^{\infty} F_2(r, \omega_k) e^{j[\phi(n,\omega_k) - \phi(n-r,\omega_k)]} w(m-r), \tag{B.4}$$

for which we can assign a time-varying function

$$g_{\omega_k}(n,m) = \sum_{r=-\infty}^{\infty} F_2(r,\omega_k) e^{j[\phi(n,\omega_k)-\phi(n-r,\omega_k)]} w(m-r), \tag{B.5}$$

and by substituting this new function into (B.3) we finally obtain (4.12).

Appendix C

The Transformation Matrix A

In this appendix we show how the transformation matrix A in Fig. 9.1 can be obtained. We show the simple case where the medium-time and short-time analyses are obtained by rectangular windows with no overlap. The same procedure can be used to obtain matrices for transformations requiring other conditions (e.g. zero-padded sequences, overlap, non-rectangular windows, etc.).

A frame of the medium-time transform is obtained by applying the DFT to a windowed segment of the time domain signal $x(n)$. If the window is rectangular with length N, then at time $n = n_0$ the frame can be written as

$$\mathbf{u}^T = \mathbf{x}^T D_N, \tag{C.1}$$

where

$$\mathbf{x} = [x(n_0), x(n_0 + 1), ..., x(n_0 + N - 1)]^T, \tag{C.2}$$

and the DFT matrix D_N has elements

$$D_N(j,k) = e^{-j\frac{2\pi jk}{N}}, \quad j,k = 0, 1, ..., N-1.$$

A frame from the short-time transform can be obtained in the same way. Let the short-time window length be $M = N/L$ so that several short-time frames can be computed from the same segment $\mathbf{x}$ used to obtain a single frame from the medium-time transform. Each short-time frame can then be written as

$$\mathbf{v}_i^T = \mathbf{x}_i^T D_M, \quad i = 0, 1, ..., L-1, \tag{C.3}$$

where

$$\mathbf{x}_i = [x(n_0 + iM), x(n_0 + iM + 1), ..., x(n_0 + iM + M - 1)]^T, \tag{C.4}$$

and D_M is the M -dimensional DFT matrix. In this particular case, where there is no window overlap in the short-time analysis and L is an integer, we can use (C.2) and (C.4) to write the following identity

$$\mathbf{x} = [\mathbf{x}_0^T \ \mathbf{x}_1^T \ \cdots \ \mathbf{x}_{L-1}^T]^T. \tag{C.5}$$

A more general case where the short-time analysis uses window overlap can be written as

$$\mathbf{x}^T = [\mathbf{x}_0^T \ \mathbf{x}_1^T \ \cdots \ \mathbf{x}_{L-1}^T]B, \tag{C.6}$$

where B is a sparse matrix that depends on the window overlap conditions. For (C.5) matrix B reduces to an N -dimensional identity matrix.

Our goal is to find a transformation matrix A such that we can obtain L frames of the short-time frequency representation from a single frame of the medium-time transform, i.e.

$$\mathbf{u}^T A = [\mathbf{v}_0^T \ \mathbf{v}_1^T \ \cdots \ \mathbf{v}_{L-1}^T]. \tag{C.7}$$

Using (C.3) and (C.1) we can write (C.7) in terms of the DFT matrices as

$$\mathbf{x}^T D_N A = [\mathbf{x}_0^T \ \mathbf{x}_1^T \ \cdots \ \mathbf{x}_{L-1}^T] \begin{bmatrix} D_M & 0 & \cdots & 0 \\ 0 & D_M & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & \cdots & 0 & D_M \end{bmatrix}. \tag{C.8}$$

The sparse matrix in the right-hand side of (C.8) is an $[LM, LM]$ matrix, where 0 are $[M, M]$ matrices with all elements equal to 0. Using identity (C.5) it follows that the transformation matrix is

$$A = D_N^{-1} \begin{bmatrix} D_M & 0 & \cdots & 0 \\ 0 & D_M & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & \cdots & 0 & D_M \end{bmatrix}. \tag{C.9}$$

We see that the matrix A is an $[N, N]$ square matrix that depends only on the DFT matrices D_N and D_M. In this example the the only matrix that needs to be inverted is D_N which, by virtue of the unitary property of the DFT, is invertible. However, other analysis conditions (e.g. non-rectangular windows, non-zero overlap, zero-padded DFTs, etc.) may lead to non-square matrices (overdetermined systems), which would require pseudoinverse methods for their inversion [22]. It is out of the scope of this dissertation to discuss such cases, and for purposes of the implementation described in Chapter 9 we found that least squares solutions provided the desired results.

Biographical Note

Carlos was born in May 8th, 1968 in México City. After completing high school, and motivated by his interest in music and sound processing, he enrolled into the Electronics and Communications Engineering program at the Instituto Tecnológico y de Estudios Superiores de Monterrey CEM, one of the most prestigious technical universities in México. After completing the program in December 1991, he received with honors the degree of B.S. in Electrical Engineering.

In September 1992, after working as a consultant for a telecommunications licensing company in México City, he moved to Portland, Oregon to begin his graduate studies. In June 1993 he received the M.S. in Electrical Engineering degree from the Oregon Graduate Institute of Science and Technology. That same year he enrolled into the Ph.D. program offered by the institute.

In search for a thesis topic, and persuaded by Prof. Hynek Hermansky, Carlos began his Ph.D. research in the area of speech processing. Among his areas of interest he studied speech enhancement systems, and presented his research at several conferences in the United States and Europe.

In June 1996 he received the *Student Achievement Award* from the Quantum Society of the Oregon Graduate Institute of Science & Technology. During the summer of 1996 he was a student intern at the Acoustics and Audio Communications Research Department, Bell Laboratories in Murray Hill, where he worked on blind channel identification techniques under supervision of Dr. Bishnu S. Atal.

During his Ph.D. studies, Carlos coauthored 10 scientific papers in several international conferences and prestigious journals, and applied for two United States patents.