Learning-Based Auditory Encoding for Robust Speech Recognition

Submitted in Partial Fulfillment of the Requirements for
the Degree of
Doctor of Philosophy
in
Electrical and Computer Engineering

Yu-Hsiang Bosco Chiu

B.S., National Tsing Hua University
M.S., National Tsing Hua University

Carnegie Mellon University
Pittsburgh, PA 15213

May, 2010

To my parents Chiu-Kuei Shen and Yi-Chuan Chiu

Abstract

While there has been a great deal of research in the area of automatic speech recognition (ASR) with substantial improvements in performance realized by current large vocabulary speech systems, the application of speech recognition to real environments remains limited because of serious degradation in accuracy. One of the most common causes for this loss of accuracy is a mismatch between training and testing environments. The goal of this thesis is to develop a set of new approaches to the signal processing used to extract features for speech recognition that are more robust to changes in the acoustical environment. We begin with an analysis of the relative effectiveness of the various stages of a popular physiologically-motivated model of feature extraction toward the improvement of recognition accuracy in the presence of additive noise. We then propose a new approach toward the extraction of speech features which is shown to be more robust to environmental distortion. Key parameters of the improved model are obtained using data-driven optimization rather by direct modeling of physiologically-measured data. In this work we focus our attention on (1) the nonlinear compressive function that relates the input signal level to the output level of neural activity in each frequency band, and (2) the modulation transfer function, which filters the filters that emerge from the output of the nonlinearity. Based on these analyses, we develop a set of algorithms that obtain the parameters that specify these modulation filters and rate-level nonlinearities. Finally, we discuss ways of reducing the computational complexity required to determine the optimal parameters for the feature extraction algorithms.

Acknowledgments

I don't know how to describe how lucky I was that I could have the chance to be part of Robust Speech Recognition group and guided by Prof. Richard M. Stern over these years. It seems just yesterday, that I was on the phone interview with Rich and was so nervous that I couldn't remember the name of a speech recognition text book and author. I still can remember in my first year, I was so nervous about my qualifying exam as I forgot almost everything during the past years of military service and Rich gave me help, prepared me such that I could be ready for the exam. Thanks so much for all the guidance and discussions which Rich gave me these years, that I could not only advance in the academic field, but also know more about life.

I would also like to thank Prof. Bhiksha Raj and Rita Singh. No matter what kinds of question I have, no matter what problem I face, Bhiksha is always available for discussion. Thanks for his guidance these years that I can't complete this thesis without his help.

I am also grateful to my other committee members, Prof. Vijayakumar Bhagavatula and Dr. Michael L. Seltzer, for their valuable feedback, suggestions, and time that helped me improve the quality of this work.

Thanks also to Ziad Al Bawab, Kshitiz Kumar and Chanwoo Kim. You are my best group members and officemates who helped me a lot all these years. I really enjoyed the discussions overnight and appreciated the help you gave me when I got stuck.

Best wishes to my friends Mei-Hsuan Lu, David Liu, Kevin Chang, Frank Wang, Xie Le, Chen-Ling Chou, Yen-Tzu Lin and Mike Kuo. Your help supported me that I could pass again and again through challenges all these years.

Finally I would like to thank my parents Yi-Chuan Chiu and Chiu-Kuei Maria Shen, and my sister Yu-Chen Joan Chiu. Without your support, I could not come over my Ph.D. study.

This work was sponsored by NSF Grants IIS-0420866 and IIS-0916918 and by the Draper Laboratory.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Human speech is highly evolved in order to make daily communication between one another effective under all kinds of environments. Auditory perception plays a major role (in addition to the speech production process) in recognizing what other people are trying to convey even under adverse conditions. In particular, if a sound which we produce cannot be easily or robustly perceived by other people under all kinds of situations, it will be discarded during the evolution of language as it cannot be conveniently used in our daily life to communicate with other people.

Although the mechanics of our speech production system have been studied extensively for quite a long time, the functioning of the human auditory system, and especially the interpretation of neural signals after cochlear processing, is still under debate. For example, in the representation of sound by the higher auditory system, it has been argued whether temporal processing (*i.e.* the neural response of temporal patterns synchronized to the temporal structure of the stimulus) or place information (*i.e.* the frequency at which the neural firing rate is maximum) is the dominant factor in representing the input sound (*e.g.* [2, 3]).

On the other hand, due to rapidly-evolving computer technologies, the use of automatic speech recognition (ASR) systems whose models are completely statistically driven can be extended to a wide variety of application areas including task-oriented dialog systems, meeting transcription, and telemetric assistance. In all these applications, the ASR system usually includes a feature extraction component, which incorporates engineering intuition about human auditory processing strategies, and an acoustic modeling component, which models the temporal evolution of speech. For example, the log-scale frequency analy-

sis and amplitude compression that are major components of auditory models are important components of conventional feature extraction schemes such as mel-frequency cepstral coefficients (MFCC) [4], and perceptual linear prediction (PLP) [5]. The Hidden Markov Model (HMM), a doubly stochastic process in which an underlying stochastic sequence of states can be observed only through a random observation that is emitted by the state transitions [6], is the most commonly used acoustic modeling approach for the recognition system.

Robustness to environmental or acoustical change is critically important to the application of speech recognition systems in our daily life. Although conventional MFCC and PLP methods for feature extraction function quite well when acoustical environments for training and testing are matched, their performance degrades seriously when they are applied in noisy environments, especially when training and testing conditions are mismatched.

The major goal of my thesis research is to develop computational models for audition. We humans are superb in recognizing speech from other people in all kinds of adverse environments. Motivated by human ability, in the past decades, the application of auditory models for speech recognition has enjoyed both widespread academic interest and experimental success. Conventional techniques try to model the human auditory system by fitting various functions to known neural responses. More specifically, the computational models obtained by mimicking the mechanical and chemical responses of human auditory system are optimized at the level of the parameters of the model such that the model responses are as close as possible to the physiologically measured data. However, the human auditory system is only one of many ways of deriving similar information from incoming signals. We believe that for certain computational models, the details of the human auditory system are less important than the overall framework of processing. So, instead of mimicking the human auditory system through a model, as show in figure 1.1, we present a larger framework for feature computation, within which the actual details of the model themselves can be learned from data.

1.1 Computational auditory model front end

A number of feature extraction methods that are motivated by results from auditory physiology have been developed over the years. These methods have yielded systems that outperform traditional approaches such as MFCC or PLP in the presence of noise and other adverse conditions (*e.g.* [7, 8, 9, 10, 11, 1, 12, 13,

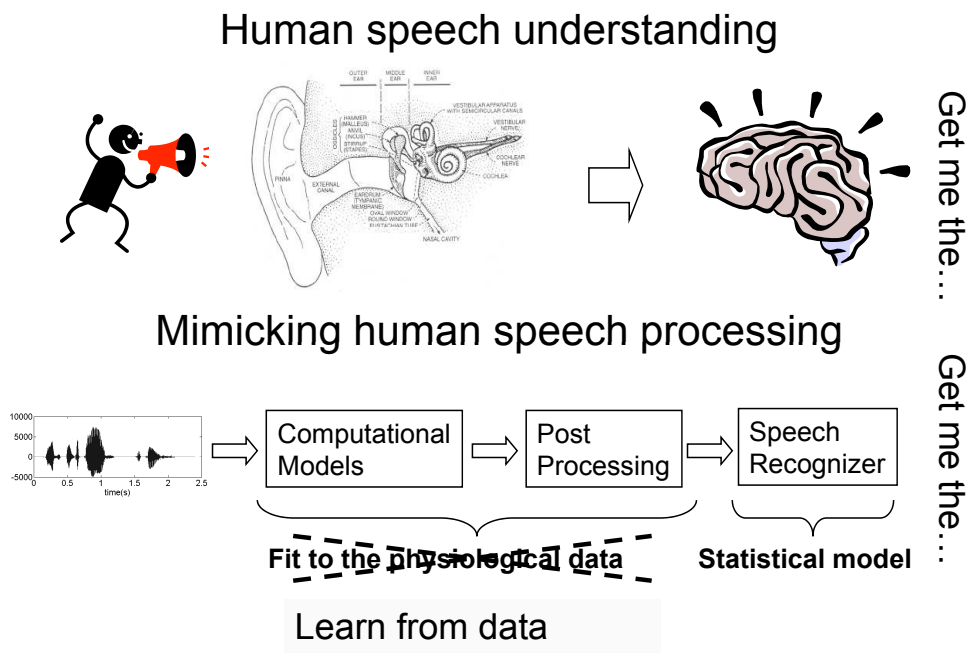

Figure 1.1: *Comparison of human auditory processing and computational auditory modeling for speech recognition task*

14, 15, 16, 17, 18, 19]). In these approaches, an auditory model is typically constructed in some fashion to model certain aspects of the human auditory system, and then followed by a feature extraction scheme. Two major approaches are synchrony-based processing [8, 9, 12, 13, 15, 16, 17] and modulation-based processing [10, 1, 14, 18, 19, 20, 21, 22]. But even though auditory models have been successful in reducing word error rate (WER) in ASR systems, it is still unclear which aspect of them really gives us the benefit, and how we can best exploit that particular property.

For example, Young and Sachs [3] have observed that sound representations (especially for vowels) are much more consistent in different in terms neural synchrony than mean rate of firing [8, 9, 12, 13, 15, 16, 17]. Though performance improvements under noisy conditions were obtained by using different approaches to extract this "synchrony" information, the real reason why we could obtain such performance improvements is still unclear.

Another example would be modulation frequency analysis. It is believed that the tuning characteristic towards certain modulation frequencies observed in the response of auditory-nerve fibers [23, 24] could be helpful for distinguishing speech signals from environmental sounds [10, 1, 14, 18, 19, 20, 21, 22]. In these

approaches either measurement from the auditory-nerve data [10, 1, 14] or by heuristic rules [18, 19, 20] are proposed for enhancing the discrimination of speech signal under advertise environments. Though to some extent, the performance improvements have been achieved over traditional processing such as MFCC or PLP, it's still not clear which set of coefficients or models we should use in order to obtain the best performance for a new task which hasn't yet been seen before. Even though there are some methods such as [21] that attempt to answer this question through data analysis, success is limited. One particular problem which draws our attention is that all these approaches are basically matched filters with fixed models and parameters for all kinds of environments, and the same set of filters is used without regarding to the environmental changes. The single set of parameters/models (such as the impulse responses of the filters) which is best for one environment might not be a good choice for another one. But how can we determine this from data? More specifically, how can we augment the benefit of modulation frequency analysis for ASR purposes from data with environmental information rather than by using measured physiological data or heuristic rules which might be suboptimal for the speech recognition purposes? This leads to an interesting and challenging problem, which is one of the main objectives of this thesis.

1.2 Overview of our learning based auditory front end

While a specific instance of a model that might be represented within the framework is one that most closely reproduces the processing details of the auditory system, the actual model that is learned for optimal classification of data may be different. Automatic gain control (AGC) is a characteristic of the auditory system. But experimentally we found that the details of AGC are less important than the fact that it results in noise flooring, which can be modeled by a sigmoid. Similarly, while equal loudness compression is a characteristic of the human auditory system, we hypothesize that it is the compressive effects of equal loudness that are key to capturing the underlying informative patterns in the speech signal. We model it by a nonlinearity and learn the parameters of the nonlinearity from data which optimizes the speech recognition performance.

In yet another instance, modulation frequency components in speech signals have long been believed to be important in human recognition of speech. Inspired by the findings that those are highly correlated with human speech perception [25] and speech recognition accuracy [26], we developed a technique for automatic design of a filter that operates in the modulation domain, which jointly minimizes the environmental

distortion as well as the distortion caused by the filter itself.

More generally, we have attempted to determine all aspects of the auditory system that contribute to robust speech recognition and developed a generalized framework within which similar effects could be produced and optimized. Toward this end, we have analyzed the effect of auditory modeling on speech recognition robustness by analyzing a detailed computational model, specifically the well-known Seneff model [8] of peripheral auditory processing. Based on this analysis, we have developed a statistically-driven approach within which we can optimize the rate-level nonlinearity that is an important part of the auditory model. We have also proposed a data-driven approach to optimize modulation spectral analysis. In related experiments on speech recognition, we measured recognition accuracy using the CMU SPHINX-III speech recognition system, and the DARPA Resource Management and Wall Street Journal speech corpus for training and testing. We showed that with our data driven approach, the performances are much better than with traditional Mel Frequency Cepstral Coefficients (MFCC) and deterministic initials of conventional computational auditory model front end under different types of adverse conditions [27, 28, 29, 30].

1.3 Thesis objectives and framework

The goal of this thesis is to improve the performance of speech recognition systems using knowledge from the human auditory modeling. This incorporates analysis of the computation modeling of human auditory system and exploration of how we can augment the benefit for the speech recognition system.

To achieve this, we first analyze the effect of auditory modeling on speech recognition by analyzing the well-known Seneff model of peripheral auditory processing [8]. In the first part of the thesis, we will analyze the contribution of each stage of the Seneff model to the robustness of speech recognition. Based this analysis, in the second part we propose a statistically-driven approach to optimizing the rate-level nonlinearity that is an important part of the auditory model. In the third part of the thesis we propose a data-driven approach to optimizing modulation spectral analysis. The collective effect of the proposed work will be to enhance speech recognition accuracy under different types of environmental distortions.

1.4 Thesis outline

In this chapter we introduced at a high level the problem we are solving and the solutions we develop in this thesis. In Chapter 2 we present the basic background the reader of this thesis needs. We discuss Mel Frequency Cepstral Coefficients, the computational auditory model front end by Dau and Seneff, and the conjugate gradient descent optimization. In Chapter 3 we discuss our first main contribution, analysis of noise robustness contribution of auditory model front end. We present recognition results and an analysis of contribution of different stages of auditory model front end by stage by stage comparison. In Chapter 4 we discuss our second contribution, optimizing the nonlinearity which we found to contribute most to the recognition robustness through data analysis. We describe how we formulate the learning problem as a joint training problem which iteratively optimizing the feature extraction and model learning as a whole framework. We also describe how we optimized the training speed through the use of word lattice and conjugate gradient descent algorithm. In Chapter 5 we discuss our third contribution, use data analysis approach to obtain the modulation filter which improve the recognition performance under noisy condition. More specifically, we developed a technique for automatic design of a filter that operates in the modulation domain, which jointly minimizes the environmental distortion as well as the distortion caused by filter itself. In Chapter 6 we summarize our contributions and propose future research directions.

Chapter 2

Background

In general, to extract features from an incoming speech signal for speech recognition, the incoming speech is segmented into short time segments. These segments are subjected to frequency analysis while the preserving time-varying characteristics that are inherent in the speech signal.

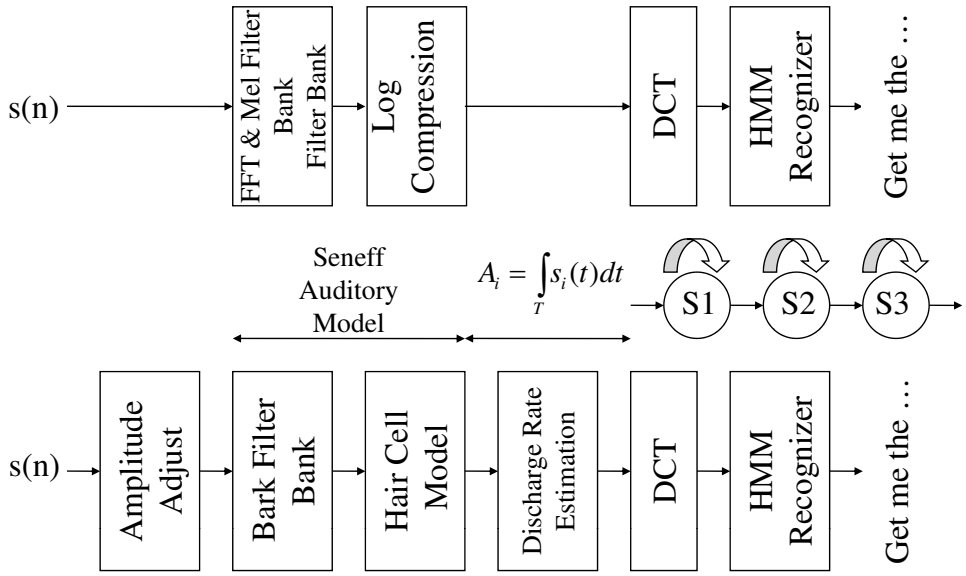

Figure 2.1: *Block diagram of traditional MFCC processing (upper panel) compared with a typical auditory-based ASR system (lower panel).*

2.1 Mel frequency cepstral coefficients

The most popular current method of extracting features for ASR is that of Mel-frequency cepstral coefficients (MFCC). MFCC processing includes the calculation of the power spectrum of successive brief segments of speech using the discrete Fourier transform (DFT), followed by a set of triangular weighting functions that provide frequency-specific estimates of energy in 40 frequency bands. This weighting is narrower at low frequencies than at higher frequencies, following results from studies of auditory physiology and perception. A logarithmic compression is applied to these energy estimates, simulating loudness compression in a primitive fashion, and the resulting log-energy estimates are then passed through a Discrete Cosine Transform (DCT) that reduces dimensionality and discards extraneous information:

$$C_t(k) = \beta(k) \cdot \sum_{i=0}^{N-1} log(E_t(i)) cos(\frac{\pi(2i+1)k}{2N}) \; where \; \beta(k) = \begin{cases} \sqrt{\frac{1}{N}}, \; k = 0 \\ \sqrt{\frac{2}{N}}, \; k \neq 0 \end{cases}$$

where $C_t(k)$ is the k^{th} cepstral coefficients of the t^{th} segment, $E_t(i)$ is the energy output of the i^{th} filter in the t^{th} segment, and N is the number of channels. These steps are summarized in the upper panel of Fig. 2.1.

2.2 Physiology of the auditory periphery

2.2.1 Sound encoding in the auditory periphery

In human auditory processing, sound enters the ear canal and is converted into mechanical motion by the eardrum. This sets the cochlea into motion, including the basilar membrane, which has mechanical resonant frequencies that vary systematically along its length: high-frequency components induce motion of the basilar membrane near its input end, and low-frequency components induce motion at the opposite end. Fibers of the auditory nerve are attached to local regions of the basilar membrane, and the motion of the membrane causes electrical spikes to be generated and propagated along the nerve fibers through an electrochemical process [31]. The response of fibers of the auditory nerves (measured in spikes per second) is frequency specific because of the frequency analysis provided by the frequency specific tuning of the basilar membrane. This frequency-specific neural response pattern (called tonotopic organization) is well preserved up to the auditory cortex [2], where sound and information from other sensory systems are processed.

2.2.2 Representation in terms of discharge rate

Even though detailed recordings from periphery auditory-nerve fibers have been obtained with the help of advanced technology, how complex sounds such as speech are encoded at the various stages of auditory processing in the brainstem and beyond still remains a challenge in auditory research. As observed from neural recordings in physiological experiments, we can describe the sound representation in higher stages of the auditory system by the number of firings within a short time interval in its response to sound stimuli, which is monotonically related to the intensity of the sound stimulus [32]. When the input stimulus is kept at an appropriate level to avoid saturation in the auditory-nerve fibers, this "firing pattern" characterized by the number of firings can preserve the frequency content and describe how sound is represented in higher stages of the auditory system in the human brain. As people can perceive human language even under noisy conditions, we believe that the representation of sound in the human auditory system is capable of capturing the most important aspects of speech and hence is potentially valuable as a model for signal processing for automatic speech recognition.

2.3 Models of peripheral processing

There have been a number of computational models proposed for peripheral auditory processing (*e.g.* [7, 8, 33, 34, 35]), but most of them are constructed to describe physiological observations rather than to provide a detailed analysis of the contribution of different stages of the models to speech recognition. Similar analyses of computational auditory modeling have been performed previously [11, 1]. For example, Ohshima and Stern [11] also analyzed the Seneff model that we will examine, but they considered only the short-term adaptation, lowpass filter, and automatic gain control (AGC) stages, which are found not to be of critical importance in our analysis. By analyzing the auditory model developed by Dau *et al.* [34, 36, 37, 38] to describe human performance in psychoacoustical experiments of speech recognition (shown in Fig. 2.2), Tchorz and Kollmeier [1] concluded that the adaptive compression stage of the auditory model by Dau *et al.* is of the greatest importance. Here we provide a more detailed description of the computational models of Dau *et al.* and Seneff.

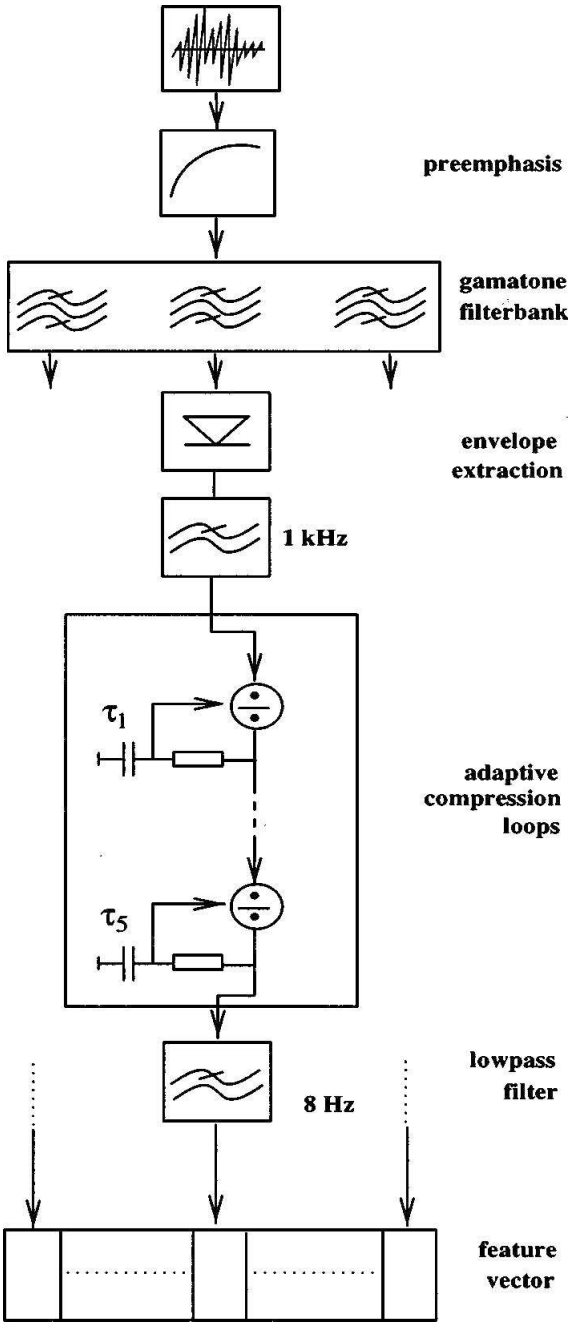

Figure 2.2: *Processing stage of auditory model used in [1]*

2.3.1 Auditory modeling by Dau and colleagues

As shown in Fig. 2.2, the model of Dau *et al.* consists of five main processing blocks. The first step is a pre-emphasis of the input signal with a first-order difference operation. The second step consists of a set of gammatone filterbank with 19 frequency channels equally spaced according to their equivalent regular bandwidth (ERB) [39]. After filtering, each frequency channel is half-wave rectified and filtered by a first-order lowpass filter with a cutoff frequency of 1000 Hz for envelope detection. After that, the amplitude of each channel output is compressed by an adaptation circuit consisting of five consecutive nonlinear adaptation loops. Each stage of the adaptive compression loops contains a divider ($output = \sqrt{input}$) and a lowpass filter with time constants tuned such that the model can best describe human performance in psychoacoustical spectral and temporal masking experiments. The last step of the auditory model is a first-order lowpass filter with a cutoff frequency of 8 Hz.

2.3.2 Auditory modeling by Seneff

Generally speaking, the auditory-based feature extraction proposed by Seneff [8] can be divided into two main stages. The first stage is a model of the auditory periphery to deal with sound transformations occurring in the early stages of the hearing process. The second stage is a series of operations intended to convert the auditory outputs into estimates of short-term average firing rate, and subsequently into features that are like cepstral coefficients. This processing is summarized in block diagram form in the lower panel of Fig. 2.1, and the Seneff auditory model is expanded in the block diagram in Fig. 2.3.

Amplitude adjustment

Because the auditory model is nonlinear, we must adjust the amplitude of the input wave to obtain a consistent input level. Here, without loss of generality, we simply adjust the input speech amplitude such that the magnitude of the maximum amplitude of input utterance is equal to one: $sig_{normalized}(t) = sig_{input}(t)/sig_{max}$ where sig_{max} is the maximum amplitude of the input speech signal.

Modeling cochlear processing

After amplitude adjustment, the speech signal is then passed through a Bark-scaled filter bank of 40 bandpass filters with relatively narrow-band filters in the low-frequency region and wider-band filters in

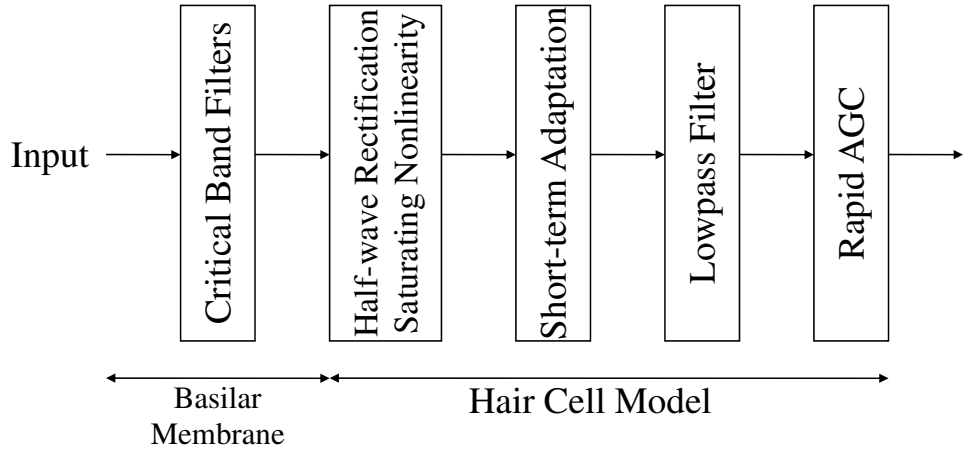

Figure 2.3: *Detailed structure of auditory modeling in Seneff auditory model*

the high frequency region, representing the frequency analysis provided by the basilar membrane in the cochlea. The bandwidth of the filters is designed to mimic human frequency resolution (like the similar mel scale that is part of the computation of MFCC features). Figure 2.4 illustrates the transfer function of each filter in the filter bank.

Hair cell synapse model

The hair cell synapse model attempts to describe the electro-chemical transformation that converts the vibration of the basilar membrane, which is represented by the output of the filter bank, to the time-varying neural firing rate of each fiber. It consists of several stages: (a) half-wave rectification with a compressive nonlinearity, to represent the inherently positive nature of the rate of spike generation and the input-output relationships between amplitude and spike rate, which is referred to here as the rate-level function, (b) short-term adaptation, which models certain aspects of the electrochemical spike generation process, (c) a lowpass filter, which represents the loss of detailed timing information at higher frequencies, and (d) a rapid automatic gain control (AGC) which represents, among other attributes, the limit on spike rate imposed by the inability to generate spikes in short succession. The panels of Fig. 2.5 illustrates the response of the system to a tone burst at 2000 Hz after the initial bandpass filtering, after the initial rectification and saturation, after the initial adaptation, and after the AGC, respectively.

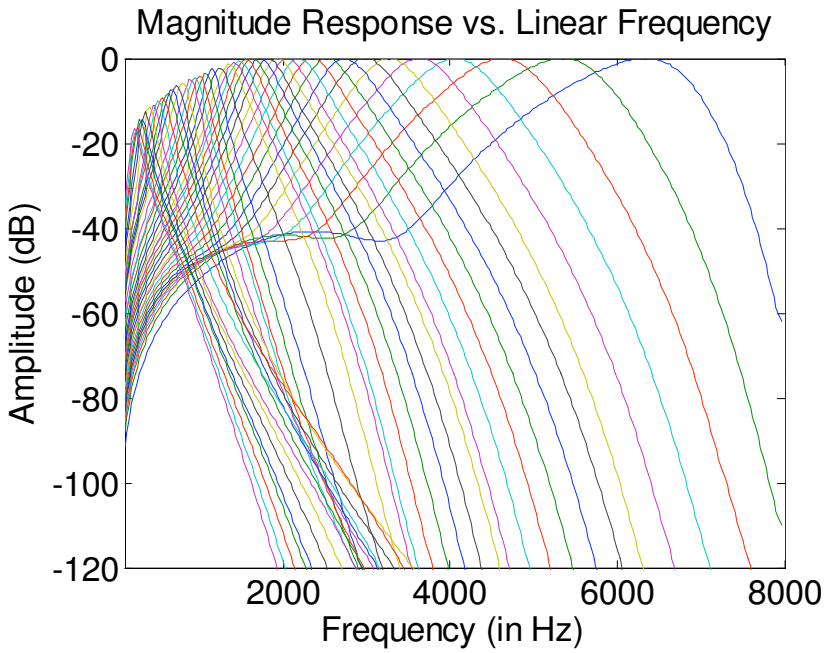

Figure 2.4: *Transfer functions of the 40-channel critical-band linear filter bank*

2.3.3 Discharge rate estimation

In order to be able to extract features relevant to perception, we need to estimate the short-time discharge rate from the outputs of the auditory model. Since the outputs of the auditory model are measured in spikes/second, we consider the discharge rate to be described by the number of spikes within a certain time interval that would be relevant to sound perception. For this purpose, we integrate the output of auditory model over a 20-ms frame:

$A_i = \int_T s_i(t)dt \quad for \quad i = 1, 2, ..., N$

where N is the number of channels. For the speech frame at time t, the corresponding feature coefficients are computed by from the DCT of the channel outputs as in MFCC processing to reduce the dimension and obtain the final features. Thus, at the end of the process we obtain features with more detailed properties of the human auditory system that resemble cepstral coefficients in other respects.

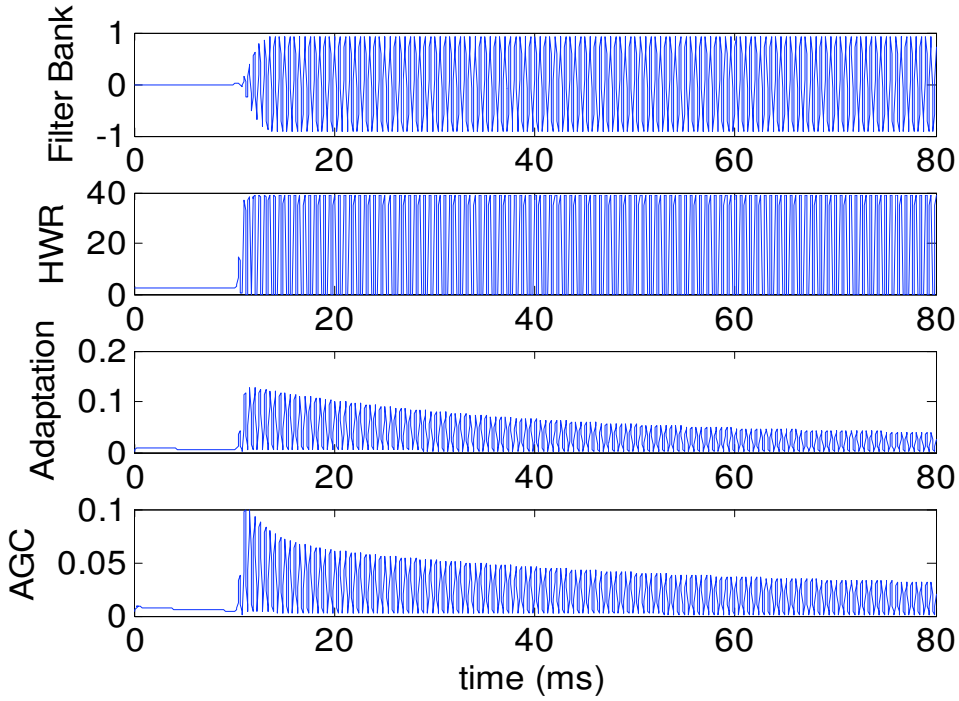

Figure 2.5: *Output of each intermediate stage in inner-hair-cell/synapse model in response to a 2k Hz input signal.*

2.4 Modulation spectrum analysis

Modulation spectrum analysis refers to the spectral components of either amplitude or frequency modulation of each frequency channel output of auditory periphery. Motivated by experimental observations that the neuronal response of mammalian auditory cortex is tuned to temporal modulation frequencies (*e.g.* [23]), and that humans are most sensitive to modulation frequencies in the range of 4 to 16 Hz (*e.g.* [40, 41]), a number of feature extraction methods have been proposed in recent years that exploit temporal information. For example, Hermansky and Morgan proposed RASTA processing [20] which employs band-pass filtering of time trajectories of speech feature vectors. The step response of these filters is comparable to physiological observations [42, 43], and RASTA processing does provide robustness to variations in noise conditions. By using a simplified adaptation model which models the synaptic adaptation observed in inner hair cells of the auditory system, Holmberg *et al.* [14] implemented a system which consistently provides better performance in various noisy conditions. The use of spectro-temporal response fields (STRFs) [19, 24], described by researchers at the University of Maryland and elsewhere, is another effective method that analyzes modulation frequency components of the auditory model output. These STRFs can be thought of as a set of two-dimensional filters with each filter combining gammatone-like response in the time domain and Gabor-like response in the frequency domain as shown in Fig. 2.6.

2.5 Conjugate gradient descent

The paper written by Jonanthan Shewchuk "An Introduction to the Conjugate Gradient Method Without the Agonizing Pain" [44] provides a good tutorial for the method of conjugate gradient descent. Here we just briefly summarize this well known optimization method which is related to our work.

2.5.1 Steepest descent

Given a function f, in steepest descent, we are starting from some point x_0 and trying to find either the maximum or minimum point by picking up the direction which increases or decreases most quickly. For example, in the case of a quadratic function:

$$f(x) = \frac{1}{2}x^T Ax - b^T x + c \tag{2.1}$$

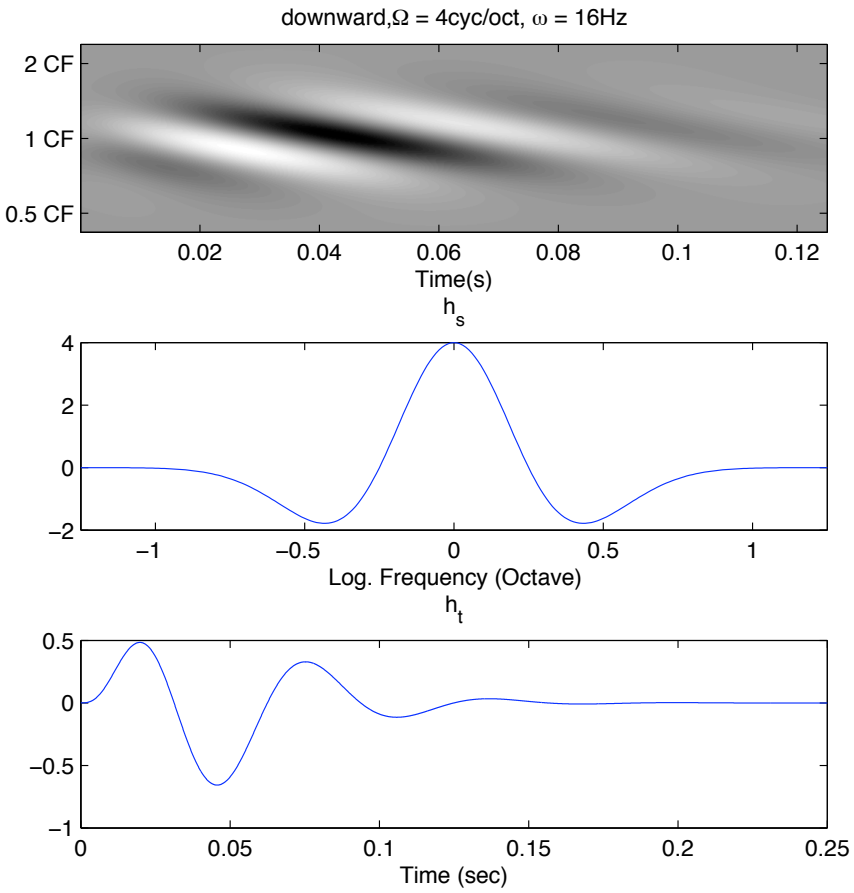

Figure 2.6: *A representative STRF and the seed functions of the spectrotemporal multiresolution cortical processing model. Upper panel: A representative STRF. This particular example is upward selective and tuned to (4 cyc/oct, 16 Hz). Middle and lower panels: Seed functions (non-causal h_s and causal h_t) of the model. The abscissa of each figure is normalized to correspond to the tuning scale of 4 cyc/oct or rate of 16 Hz.*

the direction we take for each step $x_1, x_2, x_3, ..., x_i, ...$ will be the opposite of $f'(x_i)$, which is $-f'(x_i) = b - Ax_i$. More specifically, for each step x_i, the next point which we are going to select is:

$$x_{i+1} = x_i + \alpha r_i \tag{2.2}$$

with the error:

$$e_i = x_i - x \tag{2.3}$$

where x is the optimum point we want to achieve and α is a small constant which we are going to step out. The residual $r_i = -f'(x_i) = b - Ax_i = -Ae_i$ is the steepest direction. Figure 2.7 shows an example of steepest descent optimization steps which starts at (-1,-3) and converge to the minimum point at (-2,2).

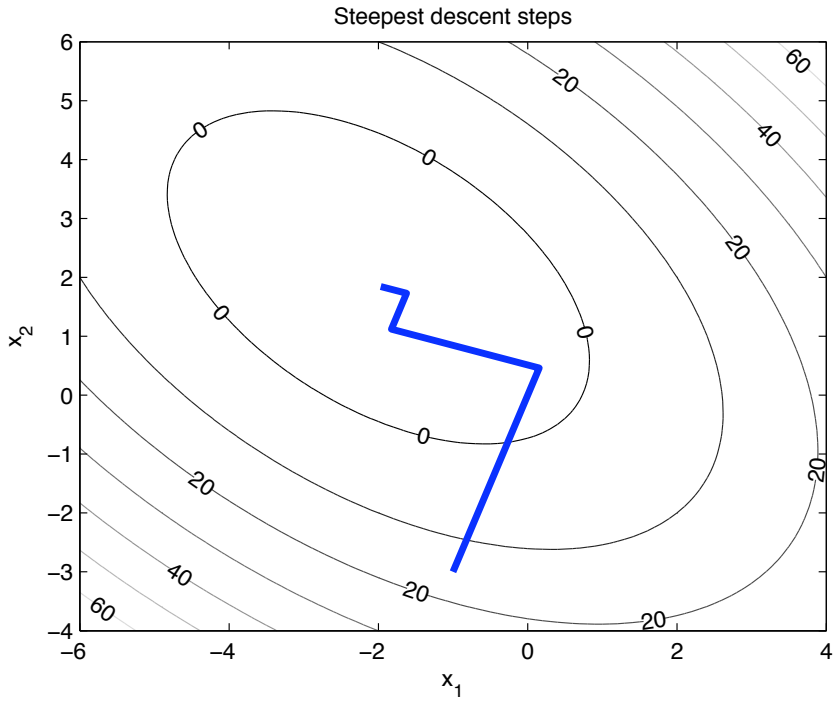

Figure 2.7: *An example of steepest descent optimization steps.*

2.5.2 Conjugate gradients

One problem for the steepest descent method is that the step it takes usually ends up taking the similar direction as previous steps. To solve this problem, one simple approach is to take a set of directions

$(d_0, d_1, ..., d_i, ...)$ which are *orthogonal* to each other as shown in the example of figure 2.8.

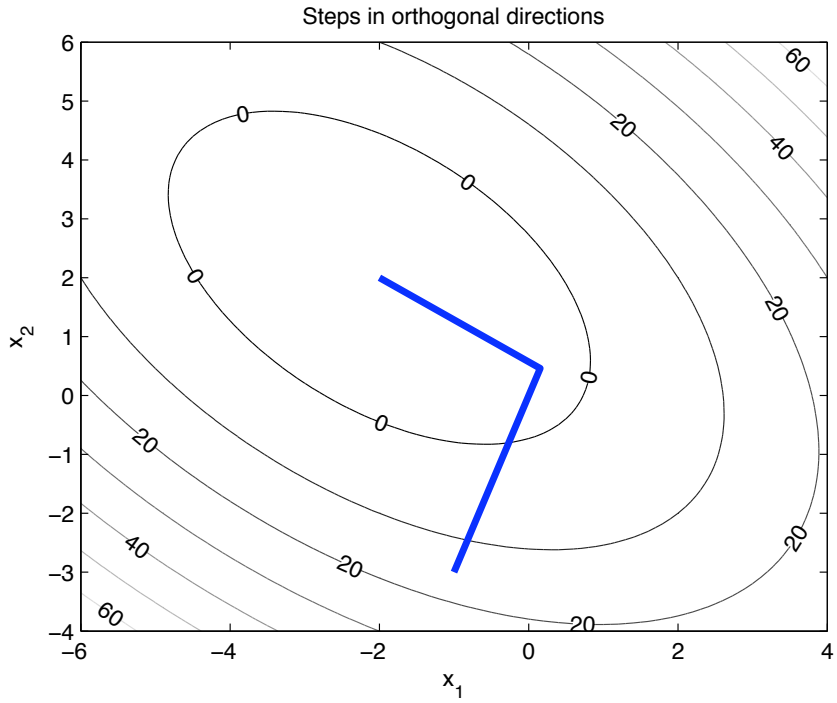

Figure 2.8: *An example of optimization steps in orthogonal directions.*

In order to let each step not repeat the same direction as in previous step, let:

$$d_i^T e_{i+1} = d_i^T (e_i + \alpha_i d_i) = 0$$
$$\alpha_i = -\frac{d_i^T e_i}{d_i^T d_i} \tag{2.4}$$

As we don't know e_i, instead of using orthogonal the original definition as in 2.4, the definition of *A-orthogonal, or conjugate* is used:

$$d_i^T A d_j = 0 \; \forall i \neq j \tag{2.5}$$

More specifically, we require e_{i+1} be A-orthogonal to d_i, that:

$$d_i = -\frac{d_i^T A e_i}{d_i^T A d_i} = \frac{d_i^T r_i}{d_i^T A d_i} \tag{2.6}$$

To obtain a set of A-orthogonal directions $\{d_i\}$, suppose we have a set of n linearly independent vectors $u_0, u_1, ..., u_{n-1}$, then we can construct d_i by taking u_i and subtracting out those which are not A-orthogonal to the previous d vectors. I.e. set $d_0 = u_0$ and for all $i > 0$:

$$d_i = u_i + \sum_{k=0}^{i-1} \beta_{ik} d_k \tag{2.7}$$

where β_{ik} are defined for $i > k$ with values can be obtained by:

$$d_i^T A d_j = u_i^T A d_j + \sum_{k=0}^{i-1} \beta_{ik} d_k^T A d_j = u_i^T A d_j + \beta_{ij} d_j^T A d_j = 0$$

$$\beta_{ij} = -\frac{u_i^T A d_j}{d_j^T A d_j} \tag{2.8}$$

Using this approach, we can set the search direction as the conjugation of the residuals, i.e. $u_i = r_i$ that:

$$\beta_{ij} = -\frac{r_i^T A d_j}{d_j^T A d_j} \tag{2.9}$$

to simplify,

$$r_{i+1} = -Ae_{i+1} = -A(x_{i+1} - x) = -A(x_i + \alpha_i d_i - x) = -A(e_i + \alpha_i d_i) = r_i - \alpha_i A d_i$$

$$r_i^T r_{j+1} = r_i^T r_j - \alpha_j r_i^T A d_j$$

$$\alpha_j r_i^T A d_j = r_i^T r_j - r_i^T r_{j+1}$$

$$r_i^T A d_j = \begin{cases} \frac{1}{\alpha_i} r_i^T r_i, & i = j \\ -\frac{1}{\alpha_{i-1} r_i^T r_i}, & i = j+1 \\ 0, & \text{else} \end{cases}$$

$$\beta_{ij} = \begin{cases} \frac{1}{\alpha_{i-1}} \frac{r_i^T r_i}{d_{i-1}^T A d_{i-1}}, & i = j \\ 0, & i > j+1 \end{cases} \tag{2.10}$$

or we can simply write:

$$\beta_i = \frac{r_i^T r_i}{d_{i-1}^T r_{i-1}} = \frac{r_i^T r_i}{r_{i-1}^T r_{i-1}} \tag{2.11}$$

and the conjugate gradient:

$$\begin{aligned} d_0 = r_0 &= b - Ax_0 \\ \alpha_i &= \frac{r_i^T r_i}{d_i^T A d_i} \\ x_{i+1} &= x_i + \alpha_i d_i \\ r_{i+1} &= r_i - \alpha_i A d_i \\ \beta_{i+1} &= \frac{r_{i+1}^T r_{i+1}}{r_i^T r_i} \\ d_{i+1} &= r_{i+1} + \beta_{i+1} d_i \end{aligned} \tag{2.12}$$

2.5.3 Nonlinear conjugate gradient method

In nonlinear conjugate gradient method, similar to the linear case, the residual is set to the opposite of the gradient, i.e. $r_i = -f'(x_i)$. The search directions are computed by Gram-Schmidt conjugation of residuals as with linear case. An outline of the nonlinear conjugate gradient could be seem as in the following:

$$\begin{aligned} d_0 = r_0 &= -f'(x_0) \\ \text{find } \alpha_i \text{ that minimize} &f(x_i + \alpha_i d_i) \\ x_{i+1} &= x_i + \alpha_i d_i, \\ r_{i+1} &= -f'(x_{i+1}), \\ \beta_{i+1} &= \frac{r_{i+1}^T r_{i+1}}{r_i^T r_i} \\ d_{i+1} &= r_{i+1} + \beta_{i+1} d_i \end{aligned} \tag{2.13}$$

where the line search, we can obtain the set α_i by:

$$\begin{aligned} f(x + \alpha d) &\approx f(x) + \alpha [\frac{d}{d\alpha} f(x + \alpha d)]_{\alpha=0} + \frac{\alpha^2}{2} [\frac{d^2}{d\alpha^2} f(x + \alpha d)]_{\alpha=0} \\ &= f(x) + \alpha [f'(x)]^T d + \frac{\alpha^2}{2} d^T f''(x) d \\ \frac{d}{d\alpha} f(x + \alpha d) &\approx [f'(x)]^T d + \alpha d^T f''(x) d = 0 \end{aligned} \tag{2.14}$$

To minimize $f(x + \alpha d)$, set equation 2.14 to zero and we get:

$$\alpha = -\frac{f'^T d}{d^T f'' d}$$

In Secant method, f'' is approximated by:

$$\frac{d^2}{d\alpha^2} f(x+\alpha d) \approx \frac{[\frac{d}{d\alpha} f(x+\alpha d)]_{\alpha=\sigma} - [\frac{d}{d\alpha} f(x+\alpha d)]_{\alpha=0}}{\sigma} = \frac{[f'(x+\sigma d)]^T d - [f'(x)]^T d}{\sigma} \tag{2.15}$$

where σ is a small constant not equal to zero and we can substitute this approximation into equation 2.14 and set to zero:

$$\begin{aligned} \frac{d}{d\alpha} f(x+\alpha d) \approx [f'(x)]^T d + \frac{\alpha}{\sigma}\{[f'(x+\sigma d)]^T d - [f'(x)]^T d\} = 0 \\ \alpha = -\sigma \frac{[f'(x)]^T d}{[f'(x+\sigma d)]^T d - [f'(x)]^T d} \end{aligned} \tag{2.16}$$

Typically, an arbitrary σ is chosen on the first iteration and $\sigma_{i+1} = -\alpha_i$ in later iterations.

2.6 Finite impulse response Wiener filter

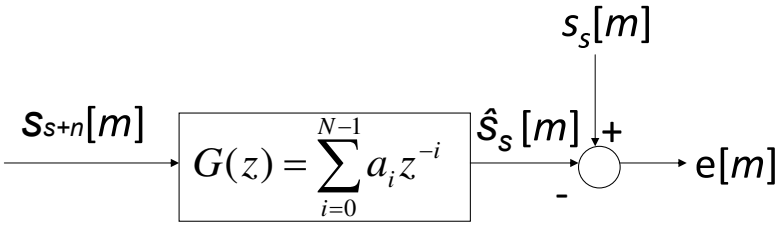

Figure 2.9: *The block diagram of Wiener filter processing.*

The finite impulse response (FIR) Wiener filter filters out interference to the desired signal. By using the statistics of the desired signal and interference, the Wiener filter finds the optimal tap weight for modifying the interference reference signal to cancel out the interference signal from the mixed input which contains both the desired signal and the interference signal.

To derive the coefficients of the Wiener filter, as shown in Fig. 2.9, we consider a noisy input signal $s_{s+n}[m]$ being fed to a Wiener filter of order N and with coefficients $a_i, i = 0, ..., N$. The output of the filter is denoted $\hat{s}_s[m]$, which is given by the expression

$$\hat{s}_s[m] = \sum_{i=0}^{N-1} a_i s_{s+n}[m-i] \tag{2.17}$$

After subtracting this estimated clean signal from the mixed input $s_{s+n}[m]$, the residual signal $e[m]$ is defined as $e[m] = s_{s+n}[m] - \hat{s}_s[m]$, which is the error between the estimated clean signal and the true clean signal. The Wiener filter is designed so as to minimize the mean square error (MMSE):

$$a_i = \text{argmin} E\left\{e^2[m]\right\} \tag{2.18}$$

where $E\{\cdot\}$ denotes the expectation operator. In general, the coefficients a_i may be complex and may be derived for the case where $s_{s+n}[m]$ and $s_s[m]$ are complex as well. For simplicity, we only consider the case where all of these quantities are real. The mean square error in the above equation can also be written as:

$$\begin{aligned} E\left\{e^2[m]\right\} &= E\left\{(s_s[m] - \hat{s}_s[m])^2\right\} \\ &= E\left\{s_s^2[m]\right\} + E\left\{\hat{s}_s^2[m]\right\} - 2E\left\{s_s[m]\hat{s}_s[m]\right\} \\ &= E\left\{s_s^2[m]\right\} + E\left\{(\sum_{i=0}^{N-1} a_i s_{s+n}[m-i])^2\right\} - 2E\left\{\sum_{i=0}^{N-1} a_i s_{s+n}[m-i] s_s[m]\right\} \end{aligned} \tag{2.19}$$

Taking the directive with respect to a_i we get:

$$\begin{aligned} \frac{d}{da_i} E\left\{e^2[m]\right\} &= 2E\left\{(\sum_{j=0}^{N-1} a_j s_{s+n}[m-j]) s_{s+n}[m-i]\right\} - 2E\left\{s_{s+n}[m-i] s_s[m]\right\} \\ &= 2\sum_{j=0}^{N-1} E\left\{s_{s+n}[m-j] s_{s+n}[m-i]\right\} a_j - 2E\left\{s_{s+n}[m-i] s_s[m]\right\} \\ &= 2\sum_{j=0}^{N-1} r_{s_{s+n}}[j-i] a_j - 2r_{s_{s+n}s_s}[i] \\ &= 0, \quad \forall i = 0, .., N-1 \end{aligned} \tag{2.20}$$

Note that $r_{s_{s+n}}[m] = E\left\{s_{s+n}[k] s_{s+n}[k+m]\right\}$ and $r_{s_{s+n}s_s}[m] = E\left\{s_{s+n}[k] s_s[k+m]\right\} = r_{s_s s_{s+n}}[-m]$ so we can obtain:

$$\sum_{j=0}^{N-1} r_{s_{s+n}}[j-i] a_j = r_{s_{s+n}s_s}[i], \quad \forall i = 0, ..., N-1 \tag{2.21}$$

We can also write the above expression as:

$$\begin{bmatrix} r_{s_{s+n}}[0] & r_{s_{s+n}}[1] & \cdots & r_{s_{s+n}}[N-1] \\ r_{s_{s+n}}[1] & r_{s_{s+n}}[0] & \cdots & r_{s_{s+n}}[N-2] \\ \vdots & \vdots & \ddots & \vdots \\ r_{s_{s+n}}[N-1] & r_{s_{s+n}}[N-2] & \dots & r_{s_{s+n}}[0] \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_{N-1} \end{bmatrix} = \begin{bmatrix} r_{s_{s+n}s_s}[0] \\ r_{s_{s+n}s_s}[1] \\ \vdots \\ r_{s_{s+n}s_s}[N-1] \end{bmatrix}$$

or

$$\mathbf{A} = \mathbf{R}_{s_{s+n}}^{-1} \mathbf{r}_{s_{s+n}s_s} \tag{2.22}$$

where $\mathbf{A} = [a_0 \;\; a_1 \;\; ... \;\; a_{N-1}]^T$ is the vector of FIR Wiener filter coefficients, $r_{s_{s+n}}[k] = E\{s_{s+n}[m]s_{s+n}[m+k]\}$ represents the autocorrelation function of the reference interference and $r_{s_{s+n}s_s}[k] = E\{s_{s+n}[m]s_s[m+k]\}$ represents the cross correlation between the reference interference and the mixed noisy signal.

2.7 Databases

To evaluate the performance of our proposed methods, we conducted speech recognition experiments on three standard databases, the DARP Resource Management database, the DARPA Wall Street Journal database, and the AURORA 2 database.

2.7.1 Resource Management database

The DARPA Resource Management (RM) database consists of digitized and transcribed speech data for use in designing and evaluating speech recognition systems. There are two main sections, often referred to as RM1 and RM2. RM1 contains three sections, Speaker-Dependent (SD) training data, Speaker-Independent (SI) training data and test and evaluation data. RM2 has an additional and larger SD data set, including test material.

All RM material consists of read sentences modeled after a naval resource management task. The complete corpus contains over 25000 utterances from more than 160 speakers representing a variety of American dialects. The material was recorded at 16 kHz, with 16-bit resolution, using a Sennheiser HMD-414 headset microphone. All discs conform to the ISO-9660 data format.

The Speaker-Dependent (SD) Training Data contains 12 subjects, each reading a set of 600 “training sentences“, two “dialect“ sentences and ten “rapid adaptation“ sentences, for a total of 7,344 recorded sentence utterances. The 600 sentences designated as training cover 97 of the lexical items in the corpus.

The Speaker-Independent (SI) Training Data contains 80 speakers, each reading two ”dialect” sentences plus 40 sentences from the Resource Management text corpus, for a total of 3,360 recorded sentence utterances. Every sentence from a set of 1,600 Resource Management sentence texts was recorded by two subjects, with no sentence read twice by the same subject.

RM1 contains all SD and SI system test material used in 5 DARPA benchmark tests conducted in March and October of 1987, June 1988, and February and October 1989, along with scoring and diagnostic software and documentation for those tests. Documentation is also provided outlining use of the Resource Management training and test material at CMU in development of the SPHINX system. In our experiments, we use the RM1 corpus as our data set for training and testing.

2.7.2 Wall Street Journal database

During 1991, the DARPA Spoken Language Program initiated efforts to build a new corpus to support research on large-vocabulary Continuous Speech Recognition (CSR) systems.

The first two CSR Corpora consist primarily of read speech with texts drawn from a machine-readable corpus of Wall Street Journal news text and are thus often known as WSJ0 and WSJ1. (Later sections of the CSR set of corpora, however, consist of read texts from other sources of North American business news and eventually from other news domains).

The texts to be read were selected to fall within either a 5,000-word or a 20,000-word subset of the WSJ text corpus. Some spontaneous dictation is included in addition to the read speech. The dictation portion was collected using journalists who dictated hypothetical news articles.

Two microphones are used throughout: a close-talking Sennheiser HMD414 and a secondary microphone, which may vary. The corpora are thus offered in three configurations: the speech from the Sennheiser, the speech from the other microphone and the speech from both; all three sets include all transcriptions, tests, documentation, etc.

In general, transcriptions of the speech, test data from DARPA evaluations, scores achieved by various speech recognition systems and software used in scoring are included on separate discs from the waveform

data.

2.7.3 AURORA 2 database

Another speech database with which we have been evaluated our proposed methods is the AURORA 2 database, which supports speaker-independent recognition of digit sequences. All speech data are derivatives of the TIdigits database at a sampling frequency of 8 kHz.

The original TIdigits database contains speech which was originally designed and collected at Texas Instruments, Inc. (TI) for the purpose of designing and evaluating algorithms for speaker-independent recognition of connected digit sequences. There are 326 speakers (111 men, 114 women, 50 boys and 51 girls) each pronouncing 77 digit sequences. Each speaker group is partitioned into test and training subsets. The corpus was collected at TI in 1982 in a quiet acoustic enclosure using an Electro-Voice RE-16 Dynamic Cardiod microphone, digitized at 20 kHz. The waveform files are in the NIST SPHERE format.

In the AURORA2 database, two training modes are considered: training on clean data and multi-condition training on noisy data.

"Clean" corresponds to TIdigits training data downsampled to 8 kHz and filtered with a G712 characteristic. "Noisy" data corresponds to TIdigits training data downsampled to 8 kHz, filtered with a G712 characteristic and with noise artificially added at several SNRs (20 dB, 15 dB, 10 dB, 5 dB, and clean, which no noise added). Four noises are used: recording inside a subway, babble, car noise, recording in an exhibition hall. So, in total, data from 20 different conditions are taken as input for the multi-condition training mode.

Three different sets of speech data are taken for the recognition.

Set "a" consists of TIdigits test data downsampled to 8 kHz, filtered with a G712 characteristic and with noise artificially added at the same SNRs. The noises are the same as for the multi-condition training.

Set "b" consists of TIdigits test data downsampled to 8 kHz, filtered with a G.712 characteristic and with noise artificially added at the same SNRs. The noises are: restaurant, street, airport, and train station. Those noises shall represent realistic scenarios for using a mobile terminal.

Set "c" consists of TIdigits test data downsampled to 8 kHz, filtered with a MIRS characteristic and noise artificially added at the samr SNRs. The noises are: subway and street. The noises are the same as used in test set "a" and "b". The intention of test set "c" is the consideration of a different frequency

characteristic (MIRS instead of G.712). This should simulate the influence of terminals with different characteristics.

2.8 Motivations

2.8.1 Auditory model analysis

As mentioned previously, we will begin our work for this thesis by conducting an analysis of the relative contributions of the various components of the Seneff model to improved ASR accuracy. This work is necessary, even though similar analyses have been performed by [11, 1] and others. Specifically, Ohshima and Stern [11] considered only the short-term adaptation, lowpass filter, and automatic gain control (AGC) stages, which are not critical in our analysis.

We also disagree with the conclusions of the thorough analysis conducted by Tchorz and Kollmeier. In their analysis, they modified the original auditory model into several different versions to study their contribution to robust speech signal representation. As a result of these modifications they concluded that the temporal processing in each frequency channel of the auditory model plays the most important role for robust representation of speech. They arrived at this conclusion because they observed that changing the original filter parameters to better reflect the average modulation spectrum of speech further enhanced the performance of digit recognition in noise in their experiments, and the adaptive compression stage encodes the dynamic evolution of the input signal. We believe that this is not the only reason for the observed performance, as their experimental results also indicate that performance improves dramatically after the second version of their modification (which replace the adaptation compression with simple log compression, *i.e.* the nonlinear compression changes abruptly from zero to one). If the temporal processing reflected all the benefits of auditory model, then a similar amount of effect should be observed in other stages as well, especially by increasing or decreasing the number of adaptation loops as in other versions of their modification. We use the Seneff model for analysis because its structural simplicity (with only one property per stage and without interaction between stages) facilitates stage-by-stage analysis. In contrast, other previous models have either complex interaction between stage (*e.g.* [35]) or multiple properties per stage (*e.g.* [1]). We conclude that the rate-level nonlinearity described in Chapter 3 is of the greatest importance.

2.8.2 Deriving the modulation filter

Another component of the proposed work is a data-driven approach to the design of the modulation filter. Systems employing modulation spectrum analysis typically provide a recognition accuracy that exceeds the accuracy that is obtained using MFCC or PLP features in the presence of noise and other adverse conditions, especially if they are combined in some fashion with a system based on traditional features. But even after a great deal of previous work it is still not clear how can we best exploit modulation spectra or design filter coefficients for the filters that implement modulation spectral analysis that optimize ASR accuracy. For example, in RASTA processing, we need to use different parameters for different compression front end (log or cubic power) to have the best performance. A natural question that arises is that how we can know whether the filter parameters are best suited for the data set under consideration. Even though there have been some attempts to address this question by finding the best-matched temporal patterns for each frequency channel of the speech signal from training data (as in the TRAPS method [21]), accuracy still suffers when the training and testing environments are different. We attempt to address this problem by a data-driven approach to obtain our filter coefficients. Our approach develops optimal modulation filter coefficients considering the information from testing sentence as well as the information from the training data.

2.9 Conclusions

In this section we have reviewed selected relevant background for the analysis of the contribution of auditory modeling to speech recognition as well as for modulation frequency analysis. We also briefly discussed different approaches in these areas along with their advantages and disadvantages. In Chapter 3 we conduct an analysis to examine which component of the computational auditory model contributes the most to the robustness of speech recognition. Based on this analysis, in Chapter 4 we propose a data-driven approach to optimize for speech recognition accuracy. Also the optimization on improving the training speed also done by the use of word lattice and conjugate gradient descent. In Chapter 5, we propose and implement a data-driven modulation filter that improves the robustness of speech recognition systems. Finally in Chapter 6 we summarize our work, including a tabulation of our main contributions and some directions for future research.

Chapter 3

Analysis of the Seneff auditory model

In this part of the thesis we address the issue of which part of the Seneff auditory model provides the greatest benefit of noise robustness for speech recognition. We attempt to answer these questions through a statistical analysis of recognition results.

The first step in our approach is to analyze the contribution that each stage of auditory modeling plays in robust speech recognition in the presence of environmental distortion. This analysis will convey information about which attribute of auditory modeling contributes the most to robustness.

In the second step, based on the analysis results we have, we characterize the properties which contribute to the speech recognition robustness through statistical data-driven analysis. Objective functions are defined based the results of auditory-model analysis, and data-driven approaches are utilized in the next chapter to maximize the effect of those components of auditory processing that are the most effective.

3.1 Comparing performance with MFCC processing

The feature extraction scheme described above was applied to the DARPA Resource Management (RM) database described in Sec. 2.7.1. 1600 randomly selected utterances from RM1 training set are used as our training set and 600 randomly selected sentences from the RM1 testing set are used as our testing utterances (72 speakers in the training set and another 40 speakers in the testing set, representing a variety of American dialects). To evaluate the performance under noise, white noise from the NOISEX-92 database [45] was artificially added to the testing set with energy adjusted according to a pre-specified noise level (with SNRs of 0 dB, 5 dB, 10 dB, 15 dB, and 20 dB). We used CMU's SPHINX-III speech

recognition system. Cepstral-like coefficients were obtained for the auditory model by computing the DCT of the outputs of the estimator of discharge rate in each frequency band, as in the lower panel of Fig. 2.1. Seven such coefficients were obtained for each frame in the auditory model, compared to thirteen cepstral coefficients for traditional MFCC processing. Cepstral mean normalization (CMN) was applied in both cases. A comparison of speech recognition accuracy obtained with the auditory model with the corresponding accuracy obtained using traditional MFCC processing is shown in Fig. 3.1. (The term "accuracy" here is defined to be 100% minus the word error rate [WER].)

As can be seen from Fig. 3.1, speech recognition accuracy in the presence of background noise is greater when the auditory model is used than when traditional MFCC processing is employed, especially at SNRs of approximately 10-dB noise level (resulting in about a 7-dB improvement in effective SNR over MFCC processing). Next, we consider feature extraction at different stages of the auditory model output to determine which component has the greatest impact on recognition accuracy.

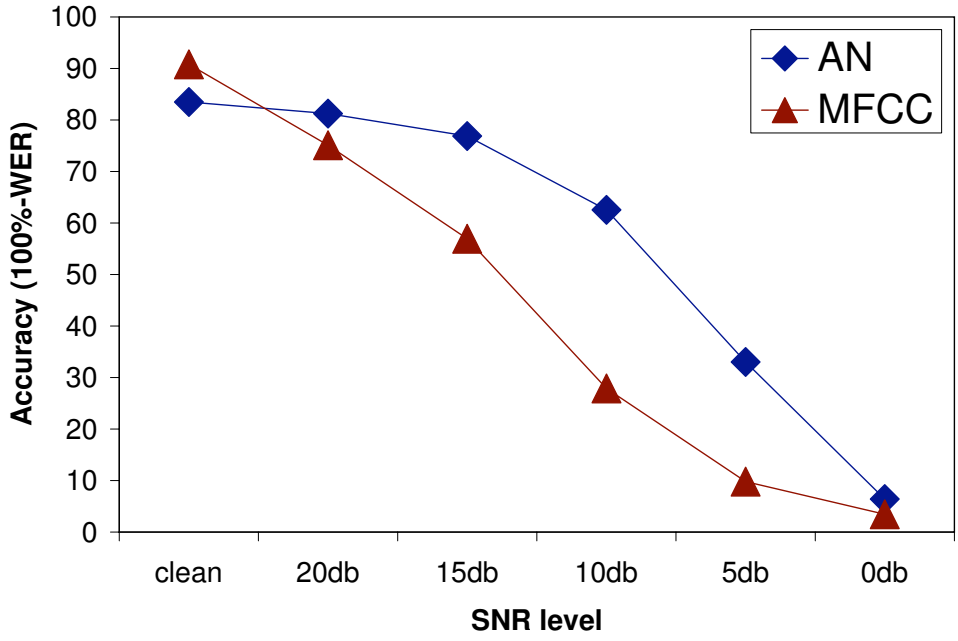

Figure 3.1: *Comparison of the recognition accuracy (100% minus the word error rate) using features based on auditory processing (diamonds) and MFCC processing (triangles) for the DARPA Resource Management (RM) database.*

3.2 Significance of each stage

To understand why using auditory processing could give us such improvement in the presence of noise, it is helpful to evaluate the contribution of each of its stages. Since the auditory model is fine-tuned to the physiological data and each stage depends on appropriate input from the previous stage, removing any of the stages is likely to cause the system to malfunction, making meaningful analysis impossible. To analyze the effect of each stage while maintaining the functionality of the auditory model, we compared the performance of each stage after the filter bank by integrating its output over 20 ms as in Fig. 3.2. The sole exception is the filterbank output which was obtained by calculating the short-term energy of each bandpass filter output, taking the log, and computing the DCT, in a fashion similar to that of traditional MFCC processing. These results evaluated on the RM database using the SPHINX-III speech recognition system are shown in Fig. 3.3 and are discussed in the following paragraphs.

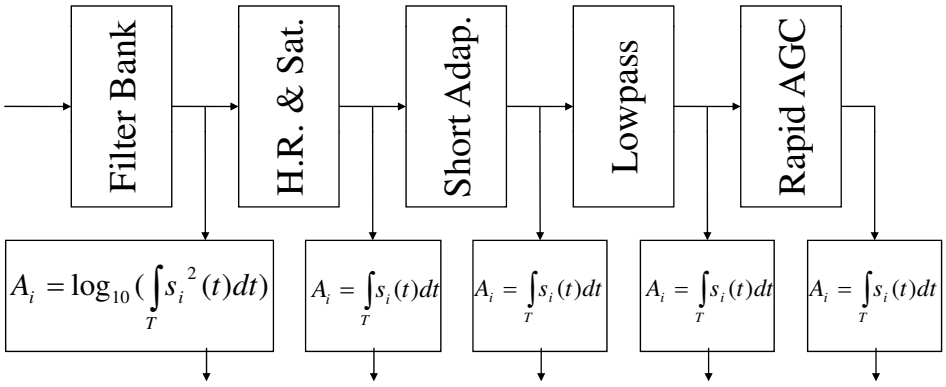

Figure 3.2: *Features extracted from each stage of the auditory model.)*

3.2.1 Effect of the rectification and nonlinearities

To evaluate the effect of the rate-level function, we first compare the recognition accuracy with features extracted before and after the half-wave rectification/saturating nonlinearity stage. As can be seen from Fig. 3.3, extracting features directly from the outputs of the filter bank (circles) provides performance that is quite similar to the result of MFCC processing (filled triangles). This result is somewhat expected as both are based on similar concepts (the filter bank simulates the frequency resolution of human ear while the log operation simulates the loudness curve). On the other hand, if we compare the result of features

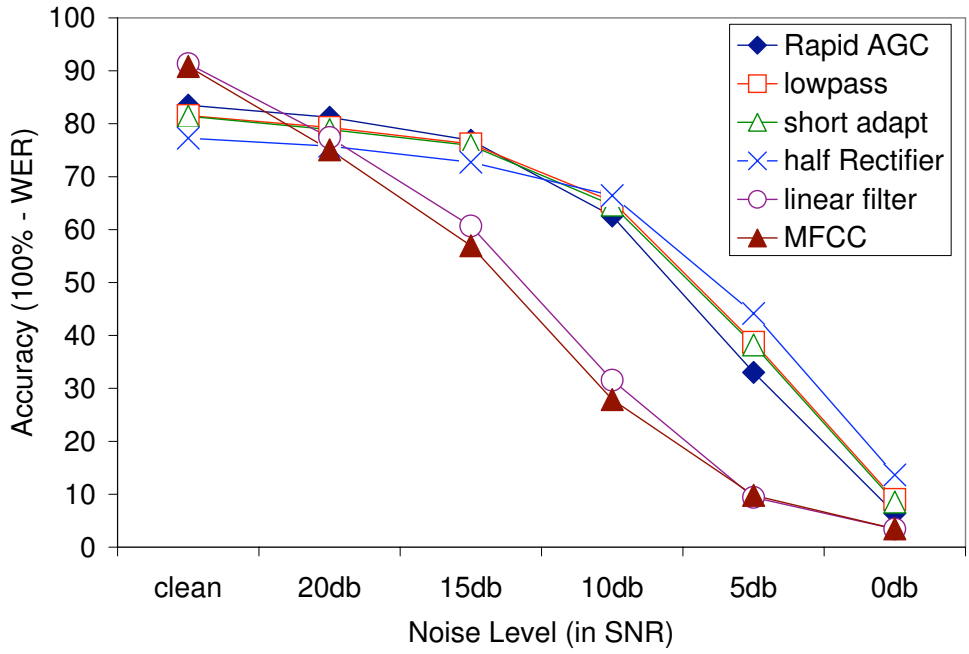

Figure 3.3: *Comparison of recognition accuracy for the RM database using features extracted from outputs of each stage of the auditory model. (See legend for details.)*

extracted from the outputs of the rectification/saturating nonlinearity stage (crosses) with the result of the filter bank outputs, the performance is much improved under noisy condition while somewhat degraded under clean speech.

As shown in Fig. 3.4, the rate-level function operates as a soft clipping mechanism, which limits both small and large amplitudes of sound. Because small-amplitude sounds are more easily affected by noise, this mechanism could help reduce the effects of degradation by noise. For example, the lower panel of Fig. 3.4 depicts the amplitude histogram of clean speech in the training data. For certain noise levels, such as 10^{-3}, speech signals with large amplitude such as 10^{-2} will only be slightly affected by additive noise after compression. In contrast, for speech signals with small amplitudes such as 10^{-4} (which is close to the silence region), an additive noise of 10^{-3} is 10 times larger than clean speech and would cause a large amount of degradation after compression. Attenuating the waveform during small-amplitude segments of sound can help reduce the degradation caused by noise, but the resultant deliberate signal distortion can degrade recognition accuracy for clean speech.

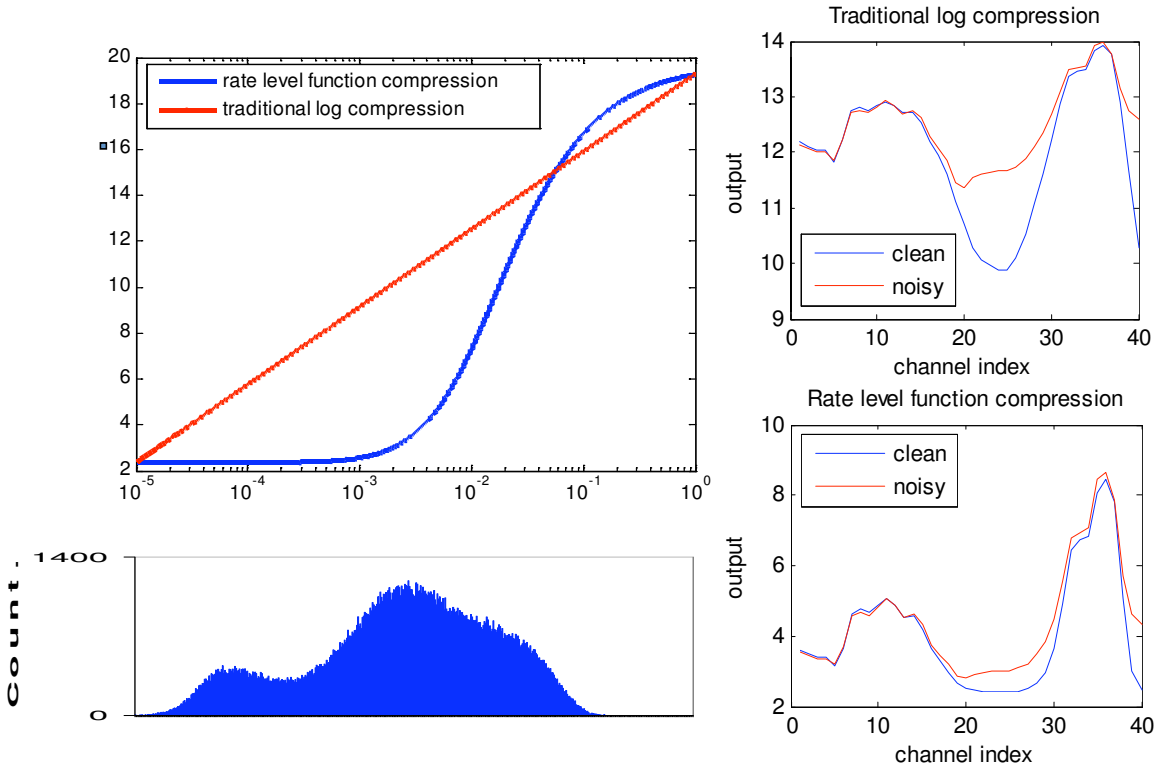

Figure 3.4: *Upper panel: rate-level function (line) in the half wave rectification stage compared with traditional log compression (dots). Lower panel: magnitude (rms) histogram for clean speech.)*

3.2.2 Effect of short term adaptation

As in the previous stages, we can assess the effect of short-term adaptation by comparing results obtained from features derived from the outputs of the half wave rectifiers (crosses in Fig. 3.3) and the outputs of short term adaptation (open triangles). These are the inputs and outputs of the short-term adaptation stage of the auditory model. The transient enhancement produced by the short-term adaptation improves recognition accuracy only slightly, as seen in Fig. 3.3. This finding is somewhat different from the conclusions in [1] and [14]. Our implementation includes both integration (which is lowpass in nature with a cutoff frequency around 50 Hz) and CMN (which is highpass, removing the DC component). The net effect of these modules is that of a bandpass filter which emphasizes the low frequencies that are most significant in modulation-spectrum analysis. This may limit the potential benefit of short-term adaptation, which is believed by at least some researchers (e.g. [1, 14]) to have a similar effect on the incoming signal.

3.2.3 Effect of the lowpass filter

For the third step, we compare features directly from the short-term adaptation stage output with features from the outputs of the lowpass filters to examine the effect of the lowpass filter stage in auditory model. As shown in Fig. 3.3 (triangles versus squares), the presence of the lowpass filter has little effect on the results that are obtained. This is somewhat as one would expect, as the feature extraction includes integration over the output, which could also be seen as a kind of lowpass filtering. Since the cutoff frequency of the lowpass filter stage (around 4 kHz) is much greater than the cutoff frequency of integration (around 50 Hz for a 20-ms period), the removal of the lowpass filter here will not have much effect on performance.

3.2.4 Effect of AGC

Because the effect of the AGC is similar to that of short-term adaptation (as can be seen in Fig. 2.5), recognition accuracy is only slightly improved for clean speech due to transient enhancement, compared to the results obtained directly from the lowpass filter output before the final AGC stage (squares and diamonds in Fig. 3.3).

3.3 Applying a nonlinear transformation to the log Mel spectrum

We argued above that the most important aspect of the auditory model was the nonlinearity associated with the hair cell model, as depicted in Fig. 3.2.1. To the extent that this is true, we should be able to obtain a similar benefit by applying such a nonlinearity to conventional MFCC-like feature extraction. Toward this end we interposed the logistic function in the upper panel of Fig. 3.4 between the log of the triangularly-weighted frequency response and the subsequent DCT operation in traditional MFCC processing as shown in Fig. 3.5. Specifically, after windowing the incoming signal into frames of brief duration, a short-time Fourier Transform is applied to obtain the magnitude spectrum of each frame. Each frequency component is weighted by the weighting function shown in Fig. 3.6 to account for the equal loudness curve in the human auditory system [46]. After applying the triangularly-shaped Mel-scale filter with log compression, a logistic function is introduced to model the nonlinear function that relates the observed average auditory-nerve response as a function of the input level in decibels:

$$x_t[i] = \frac{\alpha[i]}{1 + exp(w_1[i] \cdot y_t[i] + w_0[i])} \tag{3.1}$$

where $y_t[i]$ is the i^{th} log Mel-spectral value and $x_t[i]$ is the corresponding sigmoid-compressed value of frame t. The parameters of the non-linearity, $\alpha[i] = 0.05$; $w_0[i] = 0.613$; $w_1[i] = -0.521$ $\forall i$ were determined empirically by evaluation using the Resource Management development set in additive white noise at 10-dB SNR. These values are used in all our experiments. Note that these values are the same for all Mel-frequency components, *i.e.* they are frequency independent. $y_t[i]$ is the log of the output of the i^{th} channel. Finally, cepstral-like coefficients were obtained by applying the DCT transform to the out of the rate-level nonlinearity.

Results shown in Fig. 3.7 for speech in the presence of white noise, pink noise, and "buccaneer" noise from the NOISEX-92 database indicate a similar improvement in recognition accuracy to that seen in Fig. 3.3, corresponding to about 7-dB improvement around the 10-dB white noise level. In other words, the benefit of the auditory nonlinearity can be obtained without incurring the computational complexity associated with other aspects of auditory modeling, at least to some extent.

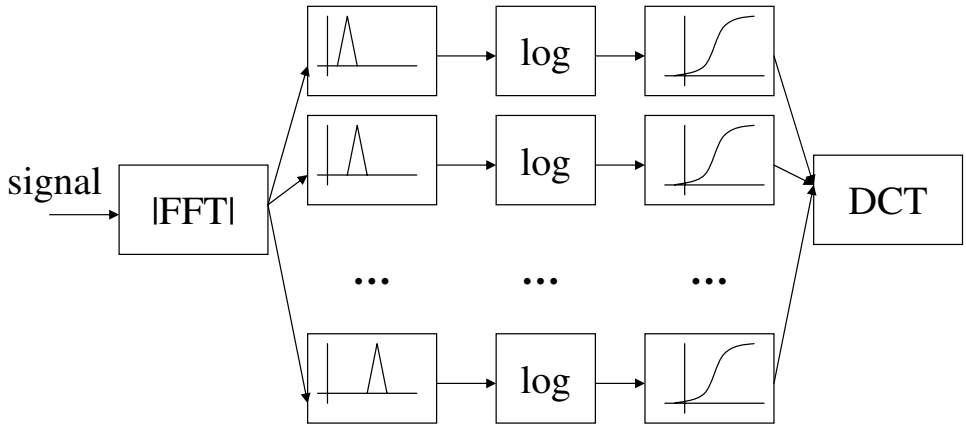

Figure 3.5: *Block diagram of the feature extraction system.*

3.4 Kullback-Leibler divergence

Another way of measuring the robustness of the feature set is by comparing the Kullback-Leibler (KL) divergence of the probability distributions of the feature sets of the same training data under clean con-

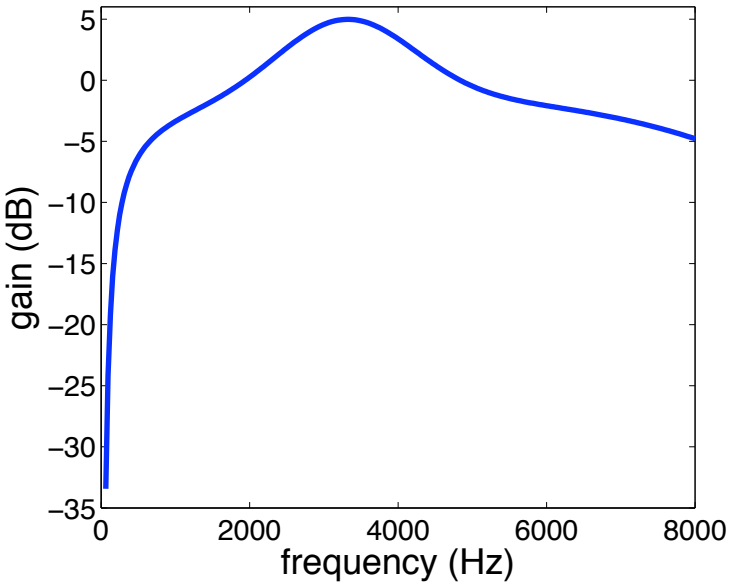

Figure 3.6: *the weighting applied to the frequency components that models the equal loudness curve of the human auditory system.*

ditions and in the presence of noise. The KL divergence measures the distance between two probability distributions which are of interest, as in the following equation:

$$D(P(X_1,...,X_n)||Q(X_1,...,X_n)) = E_P\left[log\frac{P(X_1,...,X_n)}{Q(X_1,...,X_n)}\right]$$
$$= \sum_i P(phoneme_i)D(P(X_1,...,X_n|phoneme_i)||Q(X_1,...,X_n|phoneme_i)) \quad (3.2)$$

where P is the density of $phoneme_i$ under clean conditions and Q is the density of the corresponding phoneme under noisy conditions over random variable $X_1,...,X_n$. Since the system was trained using clean data, we want the probability distribution under noisy condition to be as close as possible to the model we constructed under clean conditions. Fig. 3.8 compares the histograms and KL divergence between clean speech and speech in several different types of noise. Data from the RM training set were corrupted at 10-dB SNR using traditional MFCC processing and the features with the rate-level nonlinearity. As we can see from the figure, the KL divergences of features obtained using the rate-level nonlinearity are smaller than those obtained with traditional MFCC processing in all cases. This is consistent with our experimental results in that features with the rate-level nonlinearity perform better when there exists a

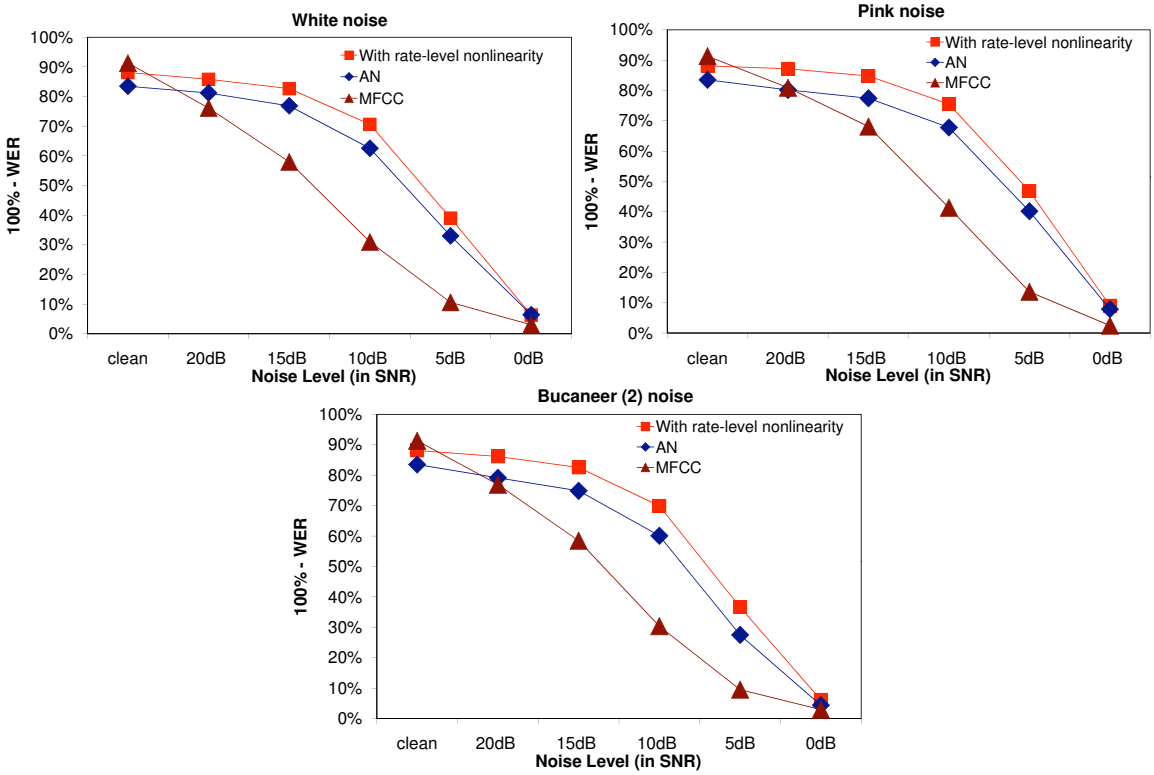

Figure 3.7: *Comparison of recognition accuracy for the RM database obtained by applying the auditory rate-level nonlinearity directly to log Mel spectral values (squares), with the entire auditory processing model (diamonds), and with traditional MFCC processing (triangles).*

mismatch between the training and testing data.

3.5 Conclusions

We have examined the relative effectiveness of the various stages of the model of the auditory periphery proposed by Seneff for improving the recognition accuracy of speech in the presence of broadband noise. Detailed robustness contributions from each stage of auditory model are also described and discussed. Results obtained using the DARPA Resource Management database with CMU's SPHINX-III recognition system indicate an improvement of about 7 dB for the Seneff model for these maskers. We also found that the saturating nonlinearity contributes the most to robustness at lower SNRs while transient enhancement in the rapid AGC and short term adaptation, on the other hand, enhance recognition accuracy only for

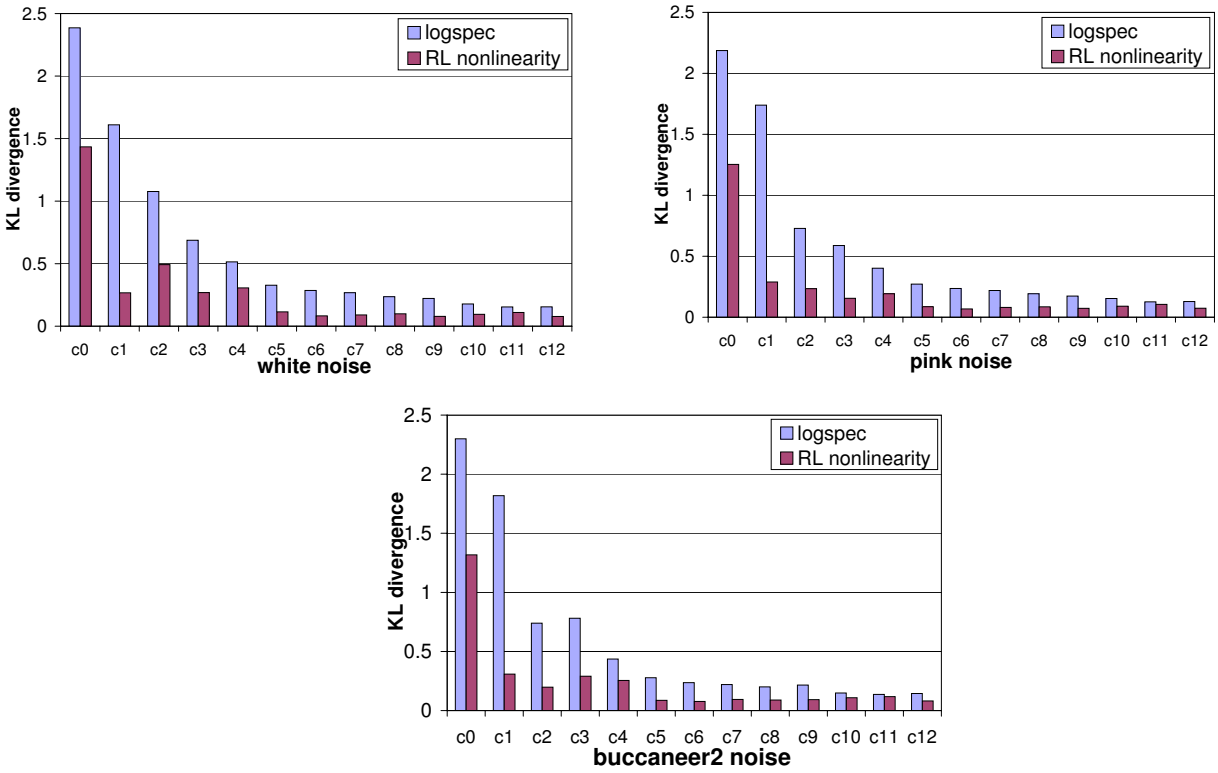

Figure 3.8: *Comparison of KL divergence between clean and noisy conditions for the RM training set. These results were obtained by applying the auditory rate-level nonlinearity directly to log Mel spectral values as in Fig. 3.5, in the presence of traditional MFCC processing under white, pink and buccaneer2 noise with SNR fixed at 10 dB.*

clean speech. By applying the same nonlinearity to the log Mel spectrum, one can achieve similar results with conventional MFCC processing. In the next chapter, we conduct a data-driven approach for optimizing the parameterized rate-level nonlinearity, i.e. the logistic function, which we proposed in this chapter for improving speech recognition accuracy.

Chapter 4

Optimizing the nonlinearity

In previous chapters we have determined that the rate-level nonlinearity that models the nonlinear relationship between the input signal level and the auditory neural spike rate is a major contributor to robustness in speech recognition. In other physiological studies in cats it has been observed that the distribution of the spontaneous rates of firing of auditory neurons depends on the noise in the environment in which the animal was raised in [47], indicating that the rate levels at least partially depend on the "training" data the animal was exposed to. Motivated by these observations, we investigate a technique for automatically learning the parameters of a nonlinear compressive function that mimics the rate-level nonlinearity to optimize recognition performance in noise.

4.1 Effect of the nonlinearity parameter on recognition accuracy

As discussed in Sec. 3.3, we added an equal-loudness curve to our system, which will not affect performance obtained using MFCC features. In the absence of the sigmoidal nonlinearity, the equal loudness weighting emerges as an additive constant after the logarithmic compression and would get eliminated by the cepstral mean normalization (CMN) that is routinely used in speech recognition. The sigmoidal nonlinearity serves to combine the gain into the features in a nonlinear manner such that it cannot be eliminated by CMN. Fig. 4.1 compares speech recognition accuracy of the proposed system with and without the equal loudness curve in the presence of four different types of background noise. As can be seen from the figure, the equal loudness curve (which can be thought of as a set of different w_0's in the different frequency channels) provide substantial benefit for the robustness of speech recognition especially for those natural

environmental sounds like market or theater noises. We will describe our efforts to obtain optimal values of these parameters in the remainder of this section.

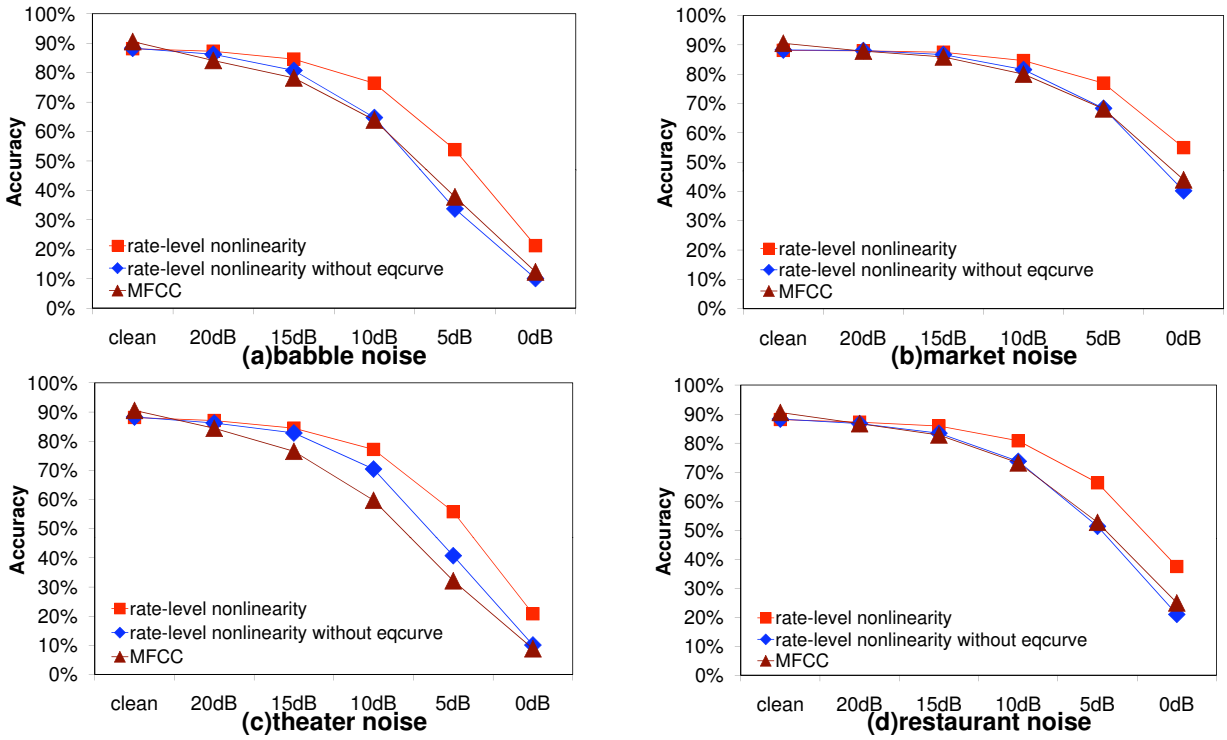

Figure 4.1: *Comparison of performance of the proposed system without equal loudness curve (diamonds), the original system(squares) and baseline MFCC processing (triangles) for the RM database in the presence of four different types of background noise.*

4.2 Learning the rate level nonlinearity

We would like to optimize the parameters of the nonlinearity to optimize recognition accuracy. However, the statistical models used for automatic speech recognition are highly complex, including hidden Markov models for the various phonemes and a language model, and it is difficult to obtain a simple update mechanism that can relate recognition accuracy to the parameters of the sigmoidal nonlinearity. Instead, we use a simple Bayesian classifier for sound classes in the language as a simple substitute for the recognizer itself. Each sound class is modelled by a Gaussian distribution, computed from the training data for that sound class. We use a maximum-mutual information (MMI) criterion to estimate the parameters of the nonlinearity such that the posterior probabilities of the phonemes on their own training data is maximized.

The actual optimization is performed using a gradient descent algorithm. This is illustrated by Figure 4.2.

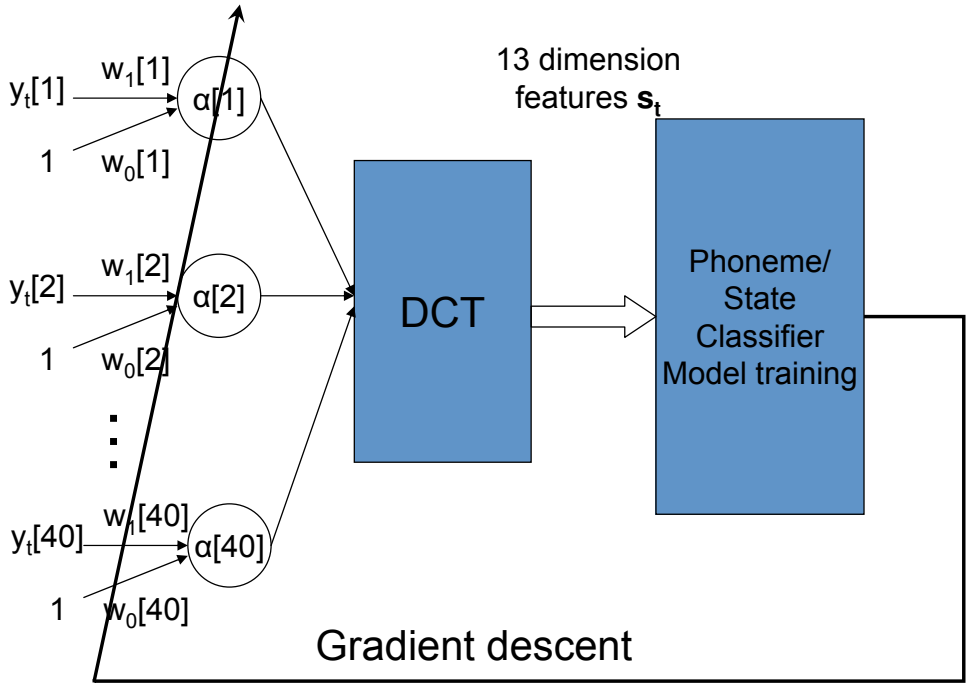

Figure 4.2: *The system to train the nonlinearity parameters.*

The procedure for optimizing the nonlinearity is as follows. Let $\boldsymbol{\mu}_C$ be the mean vector and $\boldsymbol{\sigma}_C$ be the covariance of the feature vectors for any sound class C. The likelihood of any vector $\mathbf{s}$, as computed by the distribution for that sound class is assumed to be given by a Gaussian density $N(\mathbf{s}|\boldsymbol{\mu}_C, \boldsymbol{\sigma}_C)$. The posterior probability of any sound class C, given a specific observation $\mathbf{s}$ is given by

$$P(C|\mathbf{s}) = \frac{P(\mathbf{s}|C)P(C)}{\sum_{C'} P(\mathbf{s}|C')P(C')} = \frac{P(\mathbf{s}|C)}{\sum_{C'} P(\mathbf{s}|C')} = \frac{N(\mathbf{s}|\boldsymbol{\mu}_C, \boldsymbol{\sigma}_C)}{\sum_{C'} N(\mathbf{s}|\boldsymbol{\mu}_{C'}, \boldsymbol{\sigma}_{C'})} \tag{4.1}$$

with the assumption that the prior probabilities of each class are equal.

We assume that we have a collection of training data, and that for each analysis frame of these data we know the identity of the correct sound class. We initialize the parameters of the feature computation with the values from Sec. 3.3. Each recording from the training data is parameterized using the initial values. CMN is performed on every training recording, in order to stay consistent with the processing that is performed in a complete speech recognition system.

Let $\mathbf{s}_{u,t}$ be the feature vector obtained for the t^{th} analysis frame of the utterance u. Let $C_{u,t}$ be

the sound class that the corresponding segment of speech belongs to. The overall accumulated posterior probability of the entire training data is given by

$$P = \prod_{u,t} \frac{N(\mathbf{s}_{u,t}|\boldsymbol{\mu}_{C_{u,t}}, \boldsymbol{\sigma}_{C_{u,t}})}{\sum_C N(\mathbf{s}_{u,t}|\boldsymbol{\mu}_C, \boldsymbol{\sigma}_C)} \tag{4.2}$$

The parameters of the distributions of each sound class, and those of the sigmoidal nonlinearity in the feature computation, are now jointly estimated to maximize $\log(P)$.

4.2.1 Estimating sound-class distribution parameters

The model parameters $\boldsymbol{\mu}_C$ and $\boldsymbol{\sigma}_C$ for each sound class is obtained using the same objective criterion employed by the speech recognizer. For maximum-likelihood training, this is given by:

$$\begin{aligned} \mu_C &= \frac{1}{\sum_u \sum_t I(\mathbf{s}_{u,t} \in C)} \sum_u \sum_t I(\mathbf{s}_{u,t} \in C)\mathbf{s}_{u,t}, \\ \boldsymbol{\sigma}_C &= \frac{1}{\sum_u \sum_t I(\mathbf{s}_{u,t} \in C)} \sum_u \sum_t I(\mathbf{s}_{u,t} \in C)\left(s_{u,t} - \boldsymbol{\mu}_C\right)\left(s_{u,t} - \boldsymbol{\mu}_C\right)^T \end{aligned} \tag{4.3}$$

where $I(\mathbf{s} \in C)$ is an indicator function that takes a value of 1 if $\mathbf{s}$ belongs to sound class C and 0 otherwise.

4.3 Estimating sigmoidal parameters

The parameters for the logistic function $\mathbf{F}= \{\boldsymbol{\alpha}, \boldsymbol{w_0}, \boldsymbol{w_1}\}$ are estimated to maximize $log(P)$ using a gradient descent approach. Taking the derivative of the objective function with respect to $\mathbf{F}$, the nonlinear parameters are updated as in Eq. 4.4 below. Note that step sizes are adjusted such that the convergence rate for each individual set of parameters are roughly equal.

$$\begin{aligned} \boldsymbol{\alpha}^{new} &= \boldsymbol{\alpha}^{old} + 0.001 step \frac{\partial \log P}{\partial \boldsymbol{\alpha}} \\ \boldsymbol{w}_0^{new} &= \boldsymbol{w}_0^{old} + step \frac{\partial \log P}{\partial \boldsymbol{w}_0} \\ \boldsymbol{w}_1^{new} &= \boldsymbol{w}_1^{old} + 0.2 step \frac{\partial \log P}{\partial \boldsymbol{w}_1} \end{aligned} \tag{4.4}$$

Note that the inverse of the Hessian matrix is approximated by the weighting shown above such that the convergence rate for each individual set of parameters is roughly the same and step size equal 0.05 in our

experiments. After each step of gradient descent according to the previous equations on the noisy training set, the model parameters are updated by using Eq.(7.6) on the clean training set only. Finally, after training is done and the objective function has converged, the nonlinearity parameters $\mathbf{F}=\{\boldsymbol{\alpha}, \boldsymbol{w_0}, \boldsymbol{w_1}\}$ are retained for the feature extraction process. The model parameters are then retrained using the whole speech recognition system on the clean training set. The entire learning algorithm is shown in Algorithm 2 below. Here $s_{u,t}$ represents the acoustic signal corresponding to the t^{th} analysis window of the u^{th} utterance, $s_{u,t}$ the feature vector computed from it, and $C_{u,t}$ the corresponding sound class.

Input: $\mathbf{F}$,$\{(\boldsymbol{y}_{u,t}, C_{u,t}),\ u = 1..U, t = 1..T_U\}$

Output: $\mathbf{F}$

while *not converged* **do**

1 Compute feature vector $\{\mathbf{s}_{1,1}, ..., \mathbf{s}_{U,T_U}\}$ using Eq.(3.1) and DCT with CMN

2 Estimate $\{\boldsymbol{\mu}_C, \boldsymbol{\sigma}_C\}\ \forall C$ using Eq.(7.6) on clean training set

3 Compute $\log(P)$ using Eq.(4.2) on both clean and noisy training set

4 $\mathbf{F}_{new} \leftarrow \mathbf{F}_{old} + \frac{\partial \log P}{\partial \mathbf{F}}$ using Eq.(4.4) on both clean and noisy training set

end

Algorithm 1: *Algorithm for learning the parameters of the sigmoidal nonlinearity.*

Note that the learned parameters $\mathbf{F}$ are different for each Mel-spectral channel.

4.4 Reducing computational complexity by using a word lattice

As mentioned earlier, a complete MMI solution, that compares each "true" class label for *all* competing classes can become extremely computationally expensive. More specifically, the amount of computation for calculating derivatives for each set of parameters at each iteration will be on the order of $\Theta(KLMN)$, where K is the number of cepstral dimensions, L is the number of channels, M is the number of Gaussian mixtures, and N is the number of sound classes. As the number of sound classes and Gaussian mixtures increases with the complexity of the speech recognizer, the amount of computation becomes too large for machines to handle. To overcome this problem, rather than using all sound classes as the denominator for the MMI updates, we use word lattice as shown in Fig. 4.3 to include only the sound classes that are determined to be "competitors" by the decoder as the competing classes. The word lattices used in our

experiments were generated by running the SPHINX decoder on the training data using the same initial acoustic model that had been used for generating the forced alignment. Once obtained, these lattices remained fixed throughout the optimization process.

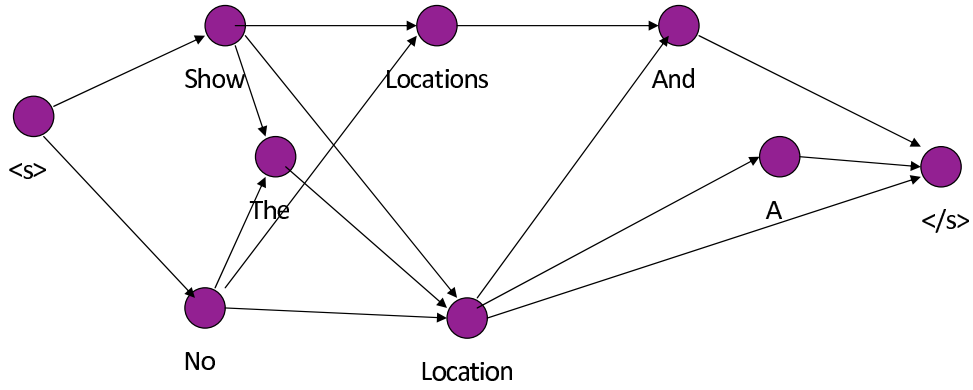

Figure 4.3: *Example of a word lattice to reduce the computational complexity by including only decoder-identified candidates as the competing classes.*

4.5 Optimizing converging speed using conjugate gradient descent

Another problem which catch our attention is that in traditional gradient descent optimization approach, even though we take the direction which increase or decrease most for getting to the optimum point. As each search direction is independently determined only through its location on parameter space, there could be redundancy between search directions, i.e. the current search direction might waste some effort searching towards the direction which similar to the previous one. To solve this problem, we use the conjugate gradient descent algorithm which finds the search direction not only with fast increase or decrease, but each search direction is orthogonal to the previous one such that no effort will be wasted between consecutive search as we described in Sec. 2.5. As the figure 2.7 shows the optimization steps with gradient descent search and the figure 2.8 shows the conjugate gradient descent search, we can achieve the optimum point with much less steps compared to the traditional gradient descent approach. And the actual implementation of the conjugate gradient descent approach can be summaries in Algorithm 2.

4.6 Results of experiments

4.6.1 Resource Management database

Experiments were run on the DARPA Resource Management database described in Sec. 2.7.1 to evaluate the proposed method. The Sphinx-III continuous-density HMM-based system was used in all experiments. Similar to the parameter settings described in Sec. 3.1, HMMs with 1000 tied states, each modeled by a mixture of 8 Gaussians, were trained for recognition experiments. The feature extraction employed a 40-filter Mel filter bank covering the frequency range 130 Hz to 6800 Hz.

In order to train the rate-level nonlinearity, pink noise from the NOISEX-92 database was artificially added to the original clean training set at 10-dB SNR to create the noisy training set. The class labels were the 1000 tied states generated by force aligning the clean training set using previously-trained models. The noisy test sets were created by artificially adding babble noise from the NOISEX-92 databases using the market, theater and restaurant noises from real environmental recordings according to the specified SNRs with respect to the original clean testing data.

Figure 4.4 shows the rate-level nonlinearities that were learned. Fig. 4.4(a) is a 3-D plot showing the nonlinearities for all 40 Mel-frequency channels. Figures 4.4(b) shows a few cross sections of this plot. Figures 4.4(c)-(e) show the individual parameters of the rate-level nonlinearities as a function of frequency. We note that the estimated optimal rate-level functions vary greatly across frequencies in all aspects, including gain, slope and attack. While we have not compared these to physiological measurements, we do note that these responses can be roughly clustered into low, mid and high spontaneous rate responses.

Once the parameters of the feature computation module were learned, the feature computation module was employed to derive features from a *clean* version of the RM training set, from which HMM model parameters were retrained.

Recognition experiments were run on speech corrupted to various SNR levels by a variety of noises. The performance metric shown is recognition accuracy, which is computed as 100% minus the word error rate, where the latter includes insertion deletion and substitution errors. Figure 4.5 shows the recognition results that were obtained. Note that none of the noises used in these experiments were used to train the rate-level nonlinearity. The plots of Figure 4.5 show three sets of recognition results. As a baseline, the recognition accuracy obtained using conventional Mel-frequency cepstral coefficients is shown. As a second comparison, the performance obtained with an implementation of [30], which also employed equal-loudness

weighting and a rate-level nonlinearity is also shown. Here, however, both the equal-loudness weighting was set to be the *approximated* loudness weighting curve [46], while the rate-level nonlinearity was also set to model physiological data with some empirical tuning. Finally the results obtained using the learned values for the rate-level nonlinearity are shown.

We note that even the equal-loudness weighting and rate-level nonlinearity obtained by fitting to physiological measurements greatly improve noise robustness compared to the MFCC baseline. The automatically-learned parameters, however, result in the best performance of all.

The improvement over the training speed can be shown in two major factors: The reduction of total training iterations over the whole training process and the time required for each iteration. Figure 4.6 shows the comparison of number of iterations needed to achieve convergence criterion. The red dashed line shows the convergence of the conjugate gradient descent method and the solid blue line shows the convergence using gradient descent on the Resource Management database. As we can see from the figure, by using conjugate gradient descent, the number of iterations required to achieve convergence is reduced by 10 times compared to the gradient descent case.

Another important component of the speedup is the use of the lattice structure as described before which reduces the processing time per iteration by reducing the number of competing candidate hypotheses needed to be considered. In empirical comparisons of the processing time with and without the lattice representation we observed that the use of the word lattice reduces the processing time for the gradient descent step of each iteration of the optimization by a factor of approximately 2.5, as shown in Table 4.1.

4.6.2 Wall Street Journal database

We also benchmark our proposed method on the standard Wall Street Journal database as described in Sec. 2.7.2. The training set we used consisted of 7024 speaker-independent utterances from 84 speakers. The test set consisted of 330 speaker-independent utterances using the 5000-word vocabulary, read from the si_et_05 directory using non-verbalized punctuation. As in the case of the Resource Management database, a noisy test set was created by artificially adding babble noise from the NOISEX-92 database and market noise from real environment recordings at pre-specified SNRs at 0 dB, 5 dB, 10 dB, 15 dB, and 20 dB. The noisy training set was created by adding 10-dB pink noise from the NOISEX-92 database to the

original clean training set. The SPHINX-III trainer and decoder were implemented using 4000 tied states, a language model weight of 11.5, and 16 GMMs, with no further attempt made to tune system parameters. Other conditions are the same as in the RM case.

Figure 4.7 shows similar results obtained using the familiar DARPA Wall Street Journal database. The results of the figure indicate that recognition accuracy using the Wall street journal database follows similar trends to what had been previously described for the resource management database. The optimization process provides an additional increase of 2-4 dB in SNR compared to the SNR obtained using the deterministic initial values of the parameters of the rate-level nonlinearity and an improvement of 3-5 dB compared to the baseline MFCC results.

4.6.3 AURORA 2 database

We also obtained bench mark results using the AURORA 2 database as described in Sec. 2.7.3. HMMs with 1000 tied states, each modeled by a mixture of 8 Gaussians for MFCC, and 32 Gaussians for the rate-level nonlinearity, were trained for recognition experiments. The feature extraction employed a 23-filter Mel filter bank covering the frequency range 64 Hz to 4000 Hz. The number of cepstral coefficients for best recognition accuracy was determined empirically to be 10 for MFCCs and 11 for the rate-level nonlinearity. The initial nonlinearity parameters were set to $w_0 = -0.110, w_1 = -0.521, \alpha = 0.05$ to account for the change of sampling rate from 16 kHz to 8 kHz.

Figure 4.8 show results obtained using the AURORA 2 database with clean training. The results of the figure indicate that recognition accuracy using the AURORA 2 database follows similar trends to what had been previously described for the resource management database and the Wall Street Journal database. The optimization process provides an additional 2-4 dB increase in SNR compared to the SNR obtained using the deterministic initials of rate-level nonlinearity and an improvement of 5-7 dB compared to the baseline MFCC results.

Figure 4.9 show results obtained using the AURORA2 database with multi-condition training. The results for the learned rate level nonlinearity are actually very close to those obtained with MFCC features under multi-condition training.

4.7 Discussion

4.7.1 Learned rate-level nonlinearity

As we can see from Figure 4.10, even though the detailed shapes of rate-level nonlinearities learned in the presence of different type of background noises are different, the general trends are similar with a shallow slope in the middle to capture the large dynamic range of speech frequency components in the mid frequencies and a steeper slope in both the low and high frequency regions.

4.7.2 Recognition accuracy as a function of the number of iterations

Figure 4.11 shows the recognition accuracy (100% - WER) as a function of SNR for babble noise for three different numbers of iterations. As can be seen in the figure, the performance improvements under different level of noise reach their final values after the first few iterations.

4.7.3 Conclusions

Our initial results indicate that the integrated approach which we use to learn the rate-level nonlinearity parameters automatically has led to substantially improved speech recognition accuracy compared to traditional MFCC processing, and better results than had been obtained using the deterministic rate-level nonlinearity estimated directly from physiological measurement under different types of background noise. More importantly, the results indicate that a statistical model can be used successfully to maximize the benefit of properties from the auditory system for speech recognition purposes. Though we can obtain quite good improvement training clean speech, unfortunately, the performance obtained using multi-condition training is approximately the same as is obtained using traditional MFCC processing. We also trained the nonlinearity parameters directly on the multi-condition training set, but this did not improve the results, either.

In the future, we will refine our current training system to further improve the recognition accuracy. For example, we can include multiple Gaussians rather than using single Gaussians as we use here to further improve the performance. In the next chapter, we will discuss ways of designing a modulation filter using data-driven approaches.

Input: $\mathbf{F}$,$\{(\boldsymbol{y}_{u,t}, C_{u,t}),\ u = 1..U, t = 1..T_U\}$

Output: $\mathbf{F}$

$r \leftarrow \frac{\partial \log P}{\partial \mathbf{F}}$

$s \leftarrow Mr$ where M is the weighting shown in Eq.4.4

$d \leftarrow s$

$\delta_{new} \leftarrow r^T d$

while *not converged* **do**

  $j \leftarrow 0$

  $\gamma \leftarrow \sigma_0$

  while $j < j_{max}$ **do**

    Compute feature vector $\{\mathbf{s}_{1,1}, ..., \mathbf{s}_{U,T_U}\}$ using Eq.(3.1) and DCT with CMS

    Estimate $\{\boldsymbol{w}_C, \boldsymbol{\mu}_C, \boldsymbol{\sigma}_C\}\ \forall C$ on clean training set

    Compute $\log(P)$ using Eq.(4.2) on both clean and noisy training set

    $\eta \leftarrow [\frac{\partial \log P}{\partial \mathbf{F}}]^T d$

    if $j \neq 0$ **then**

      $\gamma \leftarrow \gamma \frac{0.5\eta}{\eta' - \eta}$

    end

    $\mathbf{F}_{new} \leftarrow \mathbf{F}_{old} + \gamma d$

    $\eta' \leftarrow \eta$

    $j \leftarrow j + 1$

  end

  $r \leftarrow \frac{\partial \log P}{\partial \mathbf{F}}$

  $\delta_{old} \leftarrow \delta_{new}$

  $\delta_{mid} \leftarrow r^T s$

  $s \leftarrow Mr$ where M is the weighting shown in Eq.4.4

  $\delta_{new} \leftarrow r^T s$

  $\beta \leftarrow \frac{\delta_{new} - \delta_{mid}}{\delta_{old}}$

  if $\beta \leq 0$ **then**

    $d \leftarrow s$

  else

    $d \leftarrow s + \beta d$

  end

end

Algorithm 2: *Algorithm for learning the parameters of the sigmoidal nonlinearity where* $\sigma_0 = 0.05$ *and* $j_{max} = 5$.

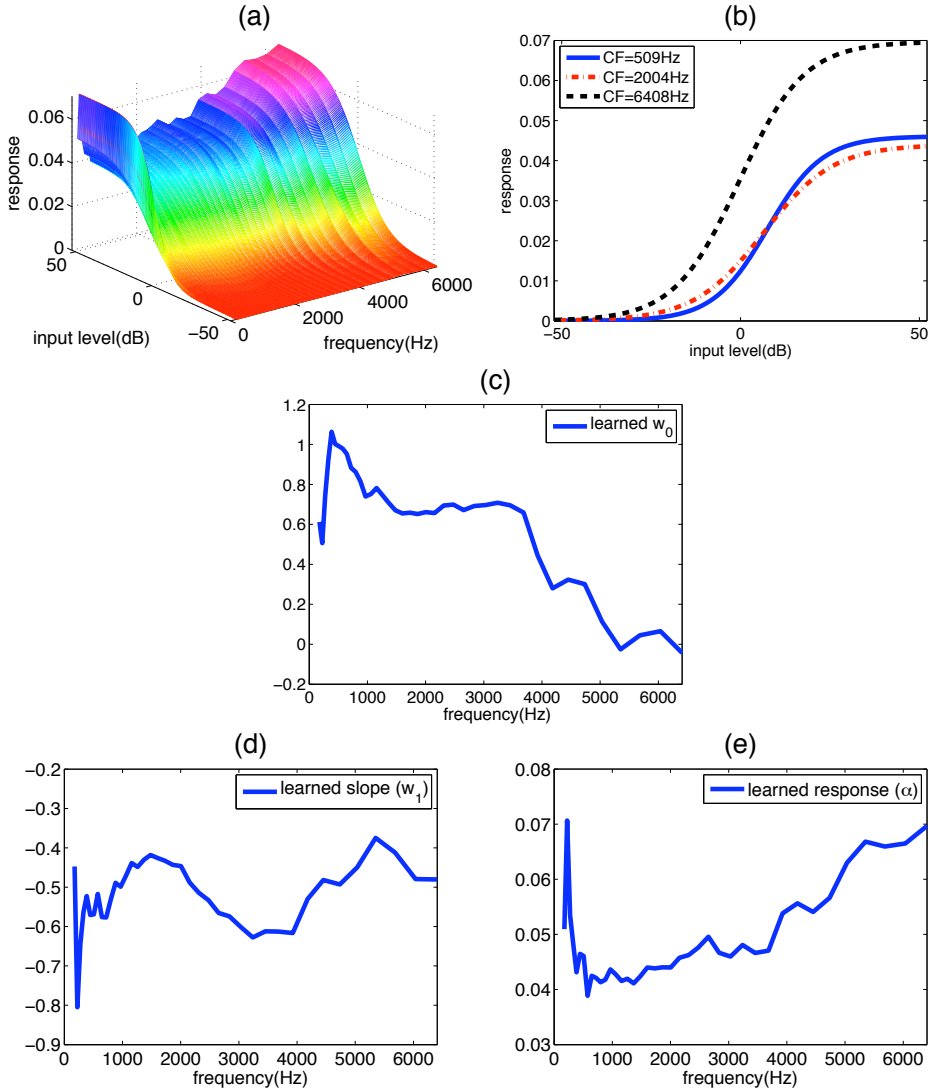

Figure 4.4: *(a)The trained RL nonlinearity over channels. (b)Examples of trained RL nonlinearity at low, mid and high frequency region: CF = 509Hz, CF = 2004Hz, CF = 6408Hz. (c)The trained w_0's over frequency channels. (d)The trained w_1's over frequency channels. (e)The trained α's over frequency channels.*

	Time
Optimization without lattice	727 sec.
Optimization with lattice	291 sec.

Table 4.1: *Tables of comparison of using and not using word lattice representation when do the training about the time required for each iteration.*

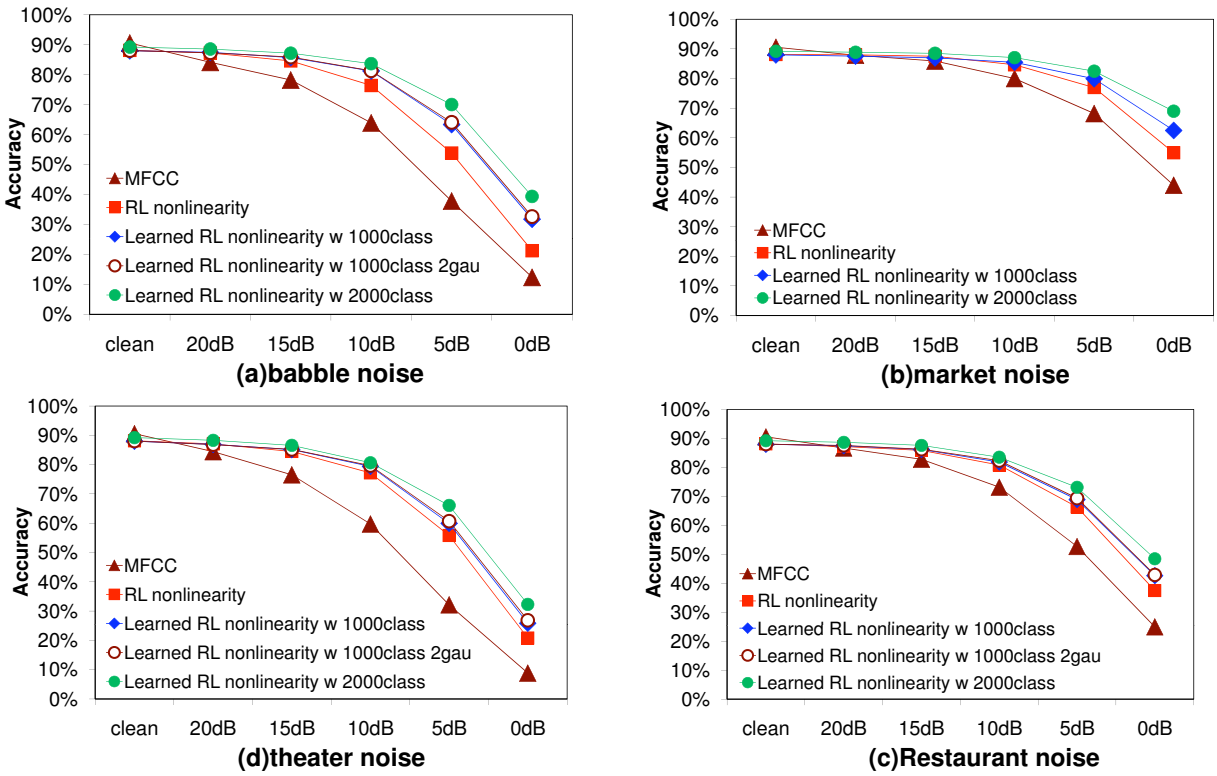

Figure 4.5: *Comparison of recognition accuracy for the same systems as in Fig. 3.5 in the presence of four types of background noise using the RM corpus. WER obtained training and testing under clean conditions: MFCC: 9.45%, RL nonlinearity: 11.88%, RL nonlinearity from learning: 10.88%*

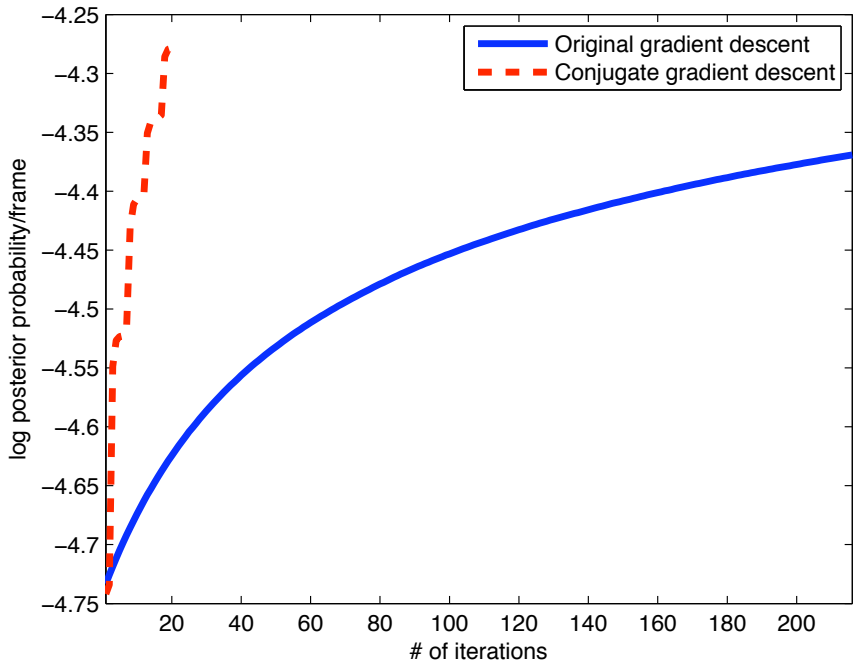

Figure 4.6: *The number of iterations required to achieve the convergence criterion using the traditional gradient descent and conjugate gradient descent methods on the Resource Management database.*

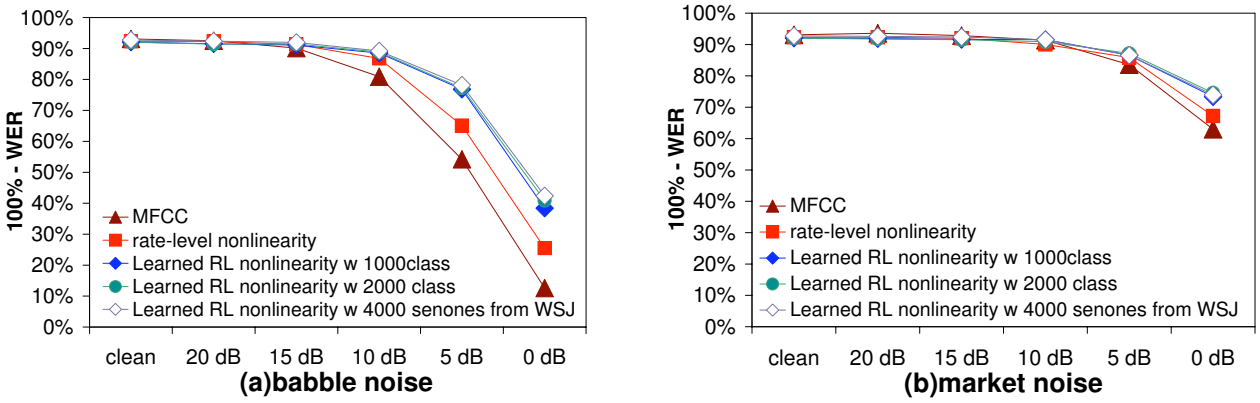

Figure 4.7: *Comparison of recognition accuracy in the presence of two types of background noise on the WSJ corpus. WER obtained training and testing under clean conditions: MFCC: 6.91%, RL nonlinearity: 7.66%, RL nonlinearity learned from RM, 1000 tied states 7.94%, 2000 tied states: 7.96%, from WSJ 4000 tied states: 7.25%*

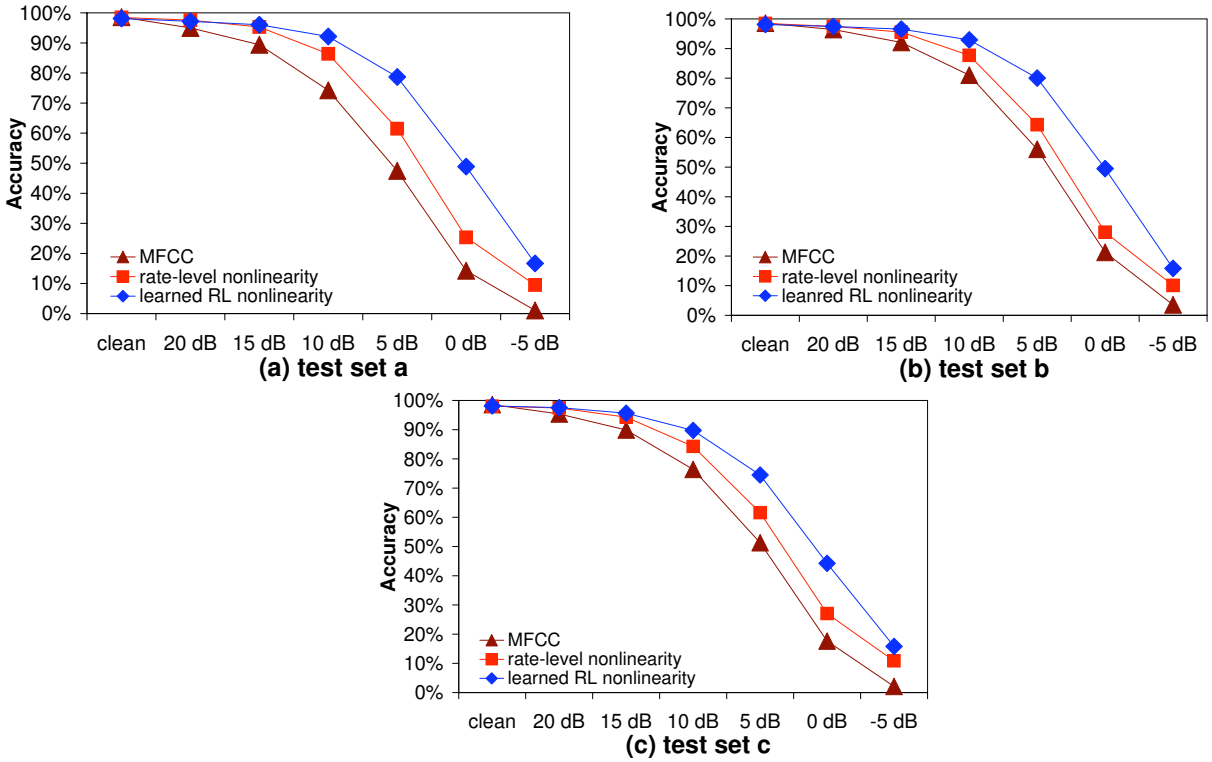

Figure 4.8: *Comparison of recognition accuracy in the presence of three sets of background noise on the AURORA 2 corpus. WER obtained training and testing under clean conditions: MFCC: test a 1.43%, test b 1.43%, test c 1.42%,RL nonlinearity: test a 1.54%, test b 1.54%, test c 1.93%, learned RL nonlinearity: test a 1.86%, test b 1.86%, test c 1.86%*

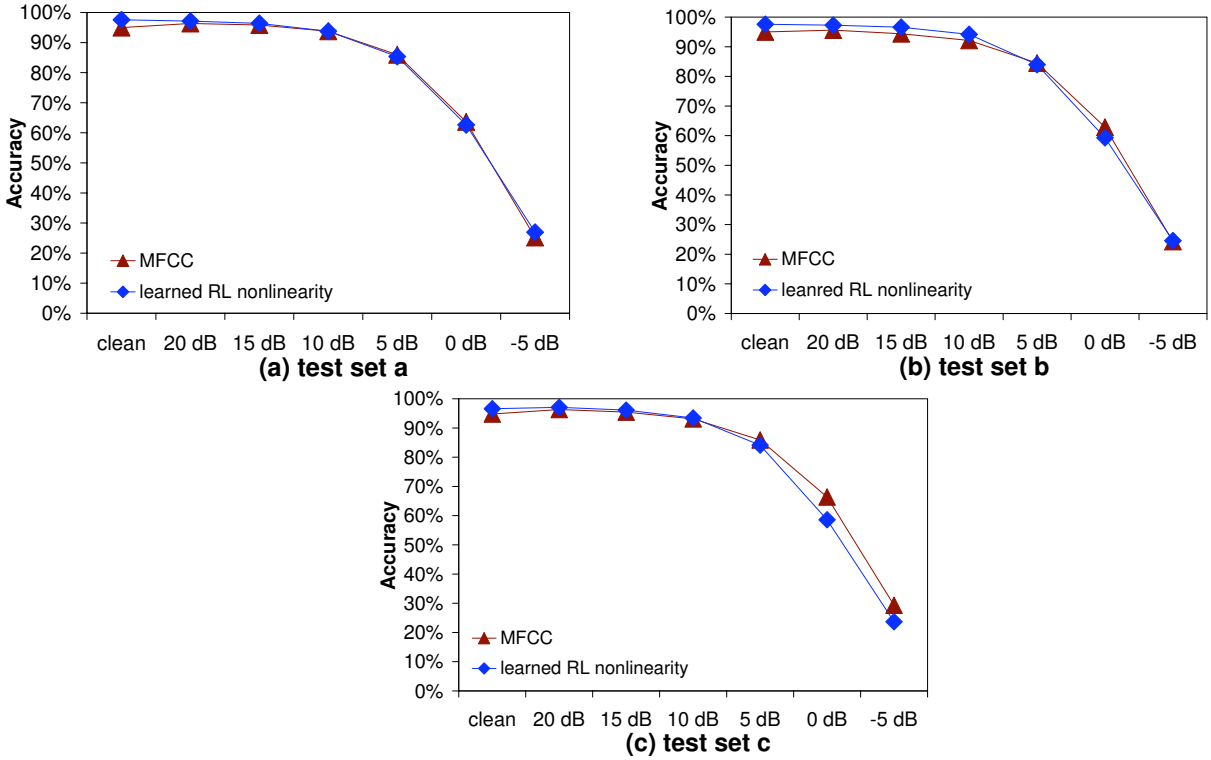

Figure 4.9: *Comparison of recognition accuracy in the presence of three sets of background noise on the AURORA2 corpus. WER obtained training and testing under clean conditions: MFCC: test a 4.99%, test b 4.99%, test c 5.22%, learned RL nonlinearity: test a 2.42%, test b 2.42%, test c 3.40%, which are significantly better than results of MFCC*

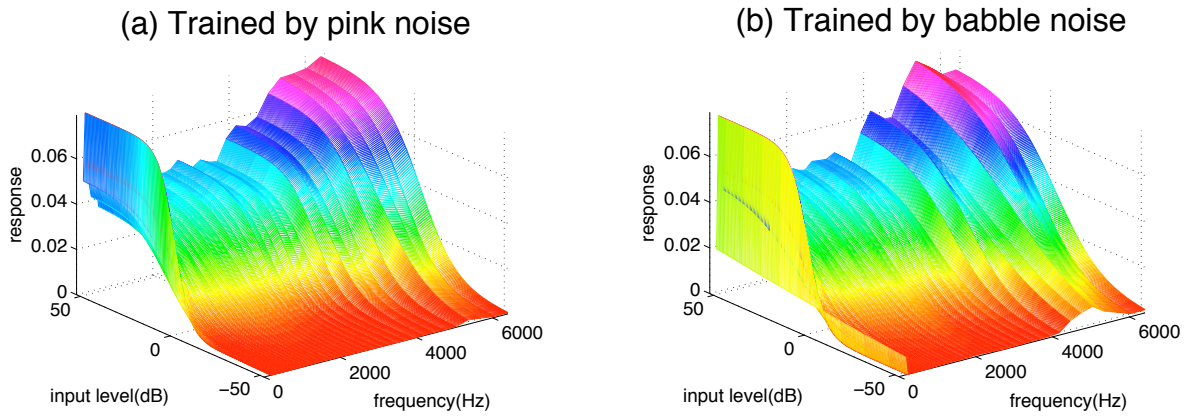

Figure 4.10: *Comparison of learned rate-level nonlinearity from different types of noises: left: from 10 dB pink noise, right: from 10 dB babble noise*

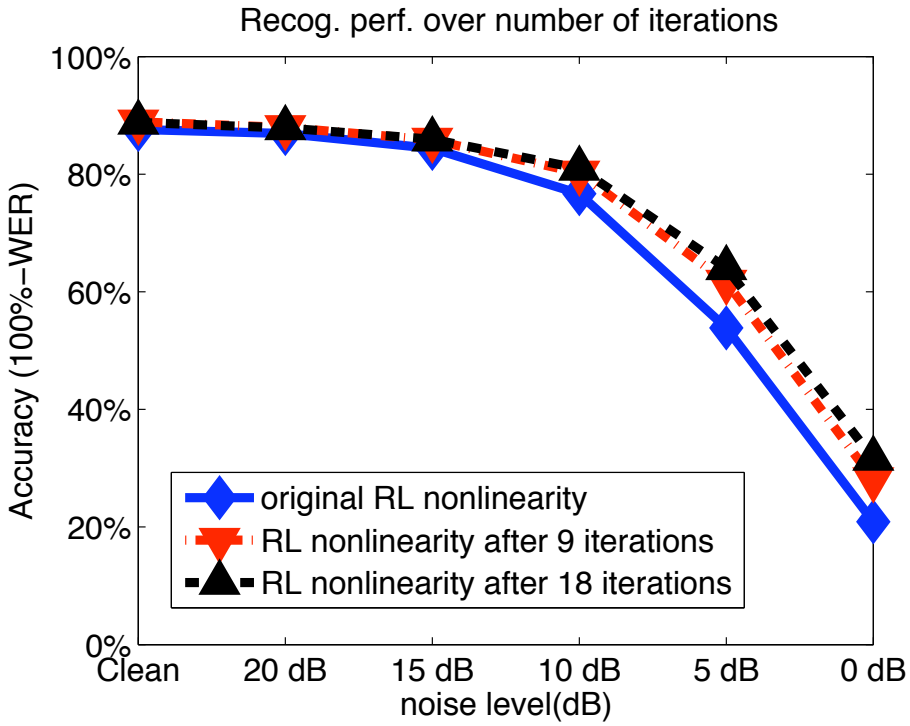

Figure 4.11: *Recognition accuracy over number of training iterations .*

Chapter 5

Minimum-variance modulation filter

In the previous chapters describing auditory model analysis, our results disagreed with the conclusions of some other researchers [1, 14] in that in our results, the short term-adaptation which accounts for the modulation transfer function does not contribute much to the robustness of speech recognition under noise. We suspect that this is due to the fact that we are using short-time integration in combination with CMN, so the net effect functions like a bandpass filter with emphasis on the low frequencies of the speech modulation spectrum. In this section, we address this issue and propose a data-driven approach that augments the benefit of modulation frequency analysis for robust speech recognition.

5.1 Modulation frequency analysis

Modulation frequency components in speech signals have long been believed to be important in human recognition of speech. For example, by assessing the change of modulation index under environmental distortion, Steeneken and Houtgast proposed the Speech Transmission Index (STI) which is highly correlated with subjective scores under different distortions [25]. In addition, by studying the contribution of different modulation frequency bands to automatic speech recognition accuracy, Kanedera *et al.* concluded that modulation frequency components in the range of 1 to 16 Hz contribute the most to ASR accuracy [26]. Inspired by these results, we focus in this section on the design of a filter that operates in the modulation domain with three objectives in mind. First, as different sentences could be subjected to different types of distortion, we want our modulation filter to be data driven, so that the filter's frequency response would be appropriate to different environmental conditions. Second, we define the environmental distortion (which

we attempt to minimize on our work) as the change of the modulation frequency components of the power spectrum. Finally, the filter itself should cause as little distortion as possible when the input signal is relatively undistorted.

5.1.1 Filter design by modulation frequency analysis

With the three objectives mentioned above in mind, we obtain the filter that minimizes the statistic

$$\rho = \lambda \int_{-\pi}^{\pi} |H(\omega)|^2 P_N(\omega) d\omega + \int_{-\pi}^{\pi} |1 - H(\omega)|^2 P_S(\omega) d\omega \tag{5.1}$$

The parameter λ controls the balance between the degree of minimization of distortion caused by the environment ($P_N(\omega)$ in the first term of the expression for ρ) and the distortion of the original modulation spectrum caused by the filter:

$$|M_S(\omega) - M_S(\omega)H(\omega)|^2 = |1 - H(\omega)|^2 |M_S(\omega)|^2 = |1 - H(\omega)|^2 P_S(\omega) \tag{5.2}$$

where $M_S(\omega)$ is the modulation spectrum, obtained by computing the Fourier transform of the output after the nonlinear in each frequency channel with clean speech as the input. Note that both phase and magnitude are considered. The frequency response of the filter is

$$H(\omega) = \sum_{l=-(L-1)/2}^{(L-1)/2} h(l) e^{-j\omega l} \tag{5.3}$$

We assume that a Type I linear phase filter with L odd and $h(l) = h(-l)$ can be utilized to achieve our goal, without providing any further constraints on its frequency response at the outset. The expression that minimizes ρ can also be expressed as

$$\begin{aligned}
\rho &= \lambda \int_{-\pi}^{\pi} \Big(\sum_{k=-\frac{L-1}{2}}^{\frac{L-1}{2}} h(k) e^{-j\omega k} \Big) \Big(\sum_{l=-\frac{L-1}{2}}^{\frac{L-1}{2}} h(l)^* e^{j\omega l} \Big) P_N(\omega) d\omega + \int_{-\pi}^{\pi} \Big(1 - \sum_{k=-\frac{L-1}{2}}^{\frac{L-1}{2}} h(k) e^{-j\omega k}\Big) \Big(1 - \sum_{l=-\frac{L-1}{2}}^{\frac{L-1}{2}} h(l)^* e^{j\omega l}\Big) P_S(\omega) d\omega \\
&= \lambda \sum_{k=-\frac{L-1}{2}}^{\frac{L-1}{2}} \sum_{l=-\frac{L-1}{2}}^{\frac{L-1}{2}} h(k) h(l)^* \int_{-\pi}^{\pi} P_N(\omega) e^{j\omega(l-k)} d\omega - \sum_{k=-\frac{L-1}{2}}^{\frac{L-1}{2}} h(k) \int_{-\pi}^{\pi} P_S(\omega) e^{-j\omega k} d\omega - \sum_{l=-\frac{L-1}{2}}^{\frac{L-1}{2}} h(l)^* \int_{-\pi}^{\pi} P_S(\omega) e^{j\omega l} d\omega \\
&+ \sum_{k=-\frac{L-1}{2}}^{\frac{L-1}{2}} \sum_{l=-\frac{L-1}{2}}^{\frac{L-1}{2}} h(k) h(l)^* \int_{-\pi}^{\pi} P_S(\omega) e^{j\omega(l-k)} d\omega + \int_{-\pi}^{\pi} P_S(\omega) d\omega = \lambda h^T R_N h - 2 h^T r_S + h^T R_S h + \int_{-\pi}^{\pi} P_S(\omega) d\omega
\end{aligned} \tag{5.4}$$

where $r_S = \begin{bmatrix} r_S(\frac{L-1}{2}) \\ \vdots \\ r_S(0) \\ \vdots \\ r_S(\frac{L-1}{2}) \end{bmatrix}$, $r_S(k) = r_S(-k)$ assuming that h is real. The matrices R_N and R_S represent the autocorrelation matrices of the distortion and speech components, respectively, of the inputs to the filter $H(\omega)$. If we further assume that the distortion and speech modulation frequency components are uncorrelated, *i.e.* $R_{N+S} = R_N + R_S$, the above equation can also be written as:

$$\rho = \lambda h^T(R_{N+S} - R_S)h + h^T R_S h - 2h^T r_S + \int_{-\pi}^{\pi} P_S(\omega)d\omega = \lambda h^T R_{N+S} h + (1-\lambda)h^T R_S h - 2h^T r_S + \int_{-\pi}^{\pi} P_S(\omega)d\omega \quad (5.5)$$

Taking the derivative with respect to h and setting it equal to zero we obtain

$$\frac{\partial \rho}{\partial h} = 2\lambda R_{N+S} h + 2(1-\lambda)R_S h - 2r_S = 0 \quad (5.6)$$

producing the filter coefficients

$$h = (\lambda R_{N+S} + (1-\lambda)R_S)^{-1} r_S \quad (5.7)$$

In the expression above, the $(i,j)^{th}$ element of the $L \times L$ autocorrelation matrix R_{N+S} of incoming noisy speech is denoted by $r_{N+S}(i-j)$, and corresponding element of the $L \times L$ Toeplitz autocorrelation matrix R_S from the clean speech used to train the system is $r_S(i-j)$. The elements $r_S(k)$ and $r_{N+S}(k)$ are obtained by:

$$r_S(k) = \frac{1}{\sum_{u=1}^{U}(T_u - k)} \sum_{u=1}^{U} \sum_{t=1}^{T_u - k} x_S(t) x_S(t+k) \quad (5.8)$$

$$r_{N+S}(k) = \frac{1}{T-k} \sum_{t=1}^{T-k} x_{N+S}(t) x_{N+S}(t+k) \quad (5.9)$$

where U is the number of training utterances and T_u is the number of frames of each training utterance and T is the number of frames of the incoming utterance. The observations $x_S(t)$ and $x_{N+S}(t)$ are the inputs to $H(\omega)$ in each channel (with mean subtraction) when the system inputs are training and testing utterances, respectively.

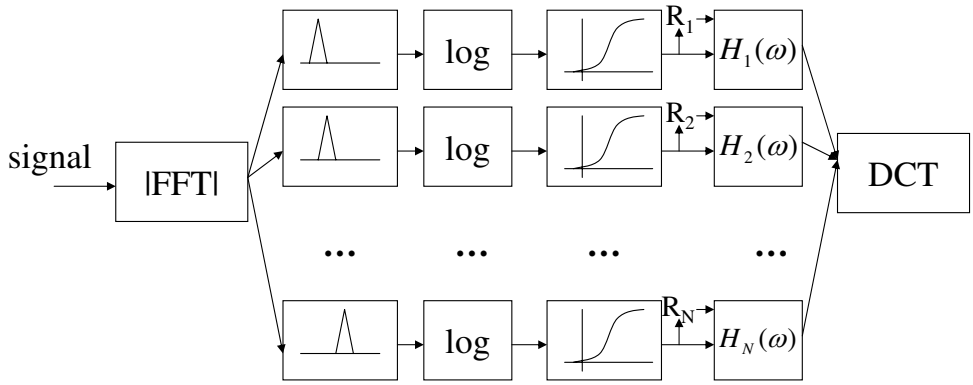

Figure 5.1: *Block diagram of the feature extraction system.*

5.1.2 System implementation

We apply the filter described above to the output of each channel of the system shown in Fig. 5.1, which is based on the system described in Sec. 3.3. After the rate-level nonlinearity, the autocorrelation matrix elements $r_S(k)$ and $r_{N+S}(k)$ are estimated according to Eq. 5.8 and 5.9 to obtain the coefficients of the filter in each channel through which the outputs of the nonlinearities are passed.

5.1.3 Effects of Modulation filter

To better understand what the filters look like under different environmental conditions and the corresponding effects on modulation spectrum, we show some examples of the speech frame under different environmental conditions and the corresponding filer responses both in time and frequency. Figure 5.2 show the impulse response of our filter under clean conditions, and in white noise at SNRs of 10 and 0 db noise. As we can see, under clean conditions, the filter response is approximately a delta function, which passes the modulation components without any change. The curves in the lower panel show the magnitude of the frequency response of the filter under the same three environmental conditions. The filter response is flat for clean speech. As the noise level increases the filter becomes more lowpass, reflecting the low pass nature of the modulation frequencies of speech.

The effect of the modulation filter on the modulation spectrum can be seen in Figure 5.3 which shows the modulation spectrum of the output of the filters before and after processing. Note that the red and black curves (which represent speech in noise) are much closer to the blue curve (which represents clean

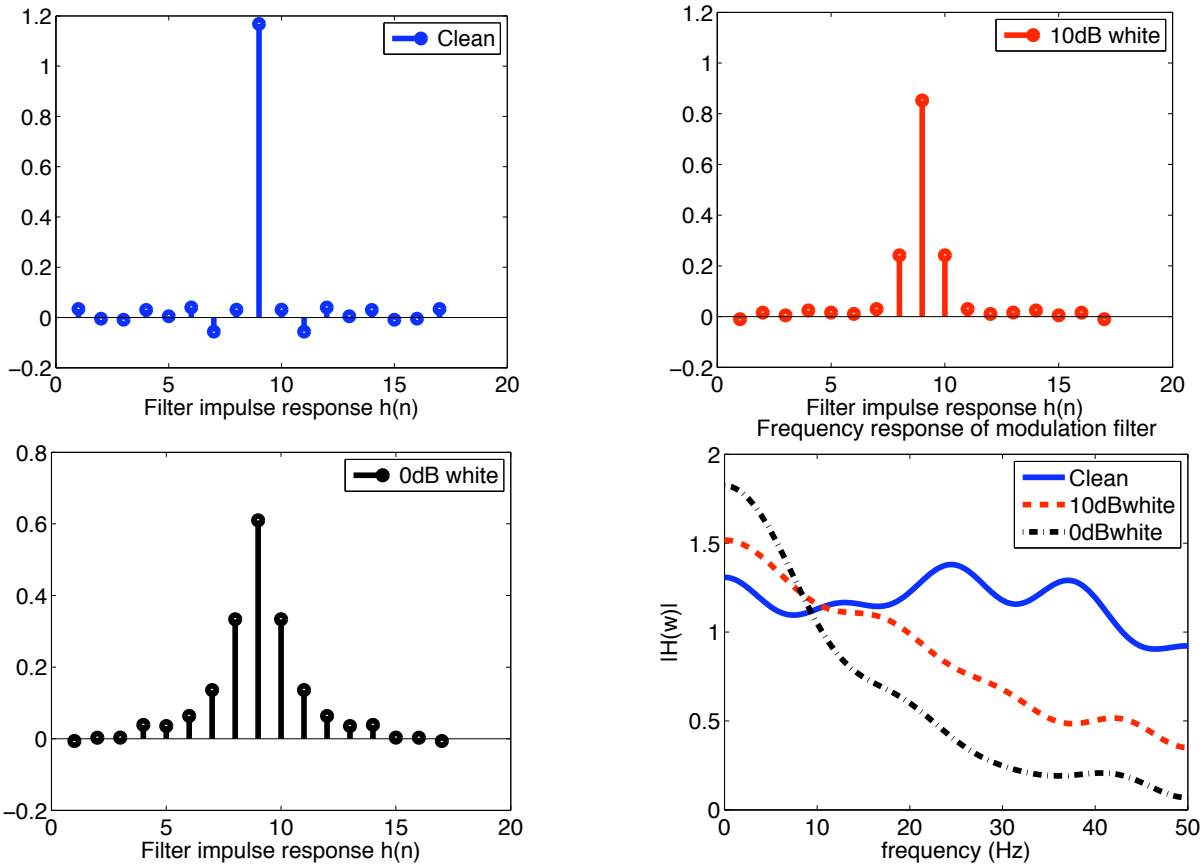

Figure 5.2: *Filter response under different environmental conditions both in time (the first three figures), and frequency (the last figure).*

speech) after the processing.

5.2 Experimental results

5.2.1 Recognition accuracy using the RM database

The feature extraction scheme described above was applied to the DARPA Resource Management (RM) database described in Section 2.7.1 and Section 4.6.1. Each utterance is normalized to have zero mean and unit variance before multiplication by a 25.6-ms Hamming window with 10 ms from frame to frame. We used CMU's SPHINX-III speech recognition system (with a language model weight of 9.5). Cepstral-like coefficients were obtained for the proposed system by computing the DCT of the outputs of the filters

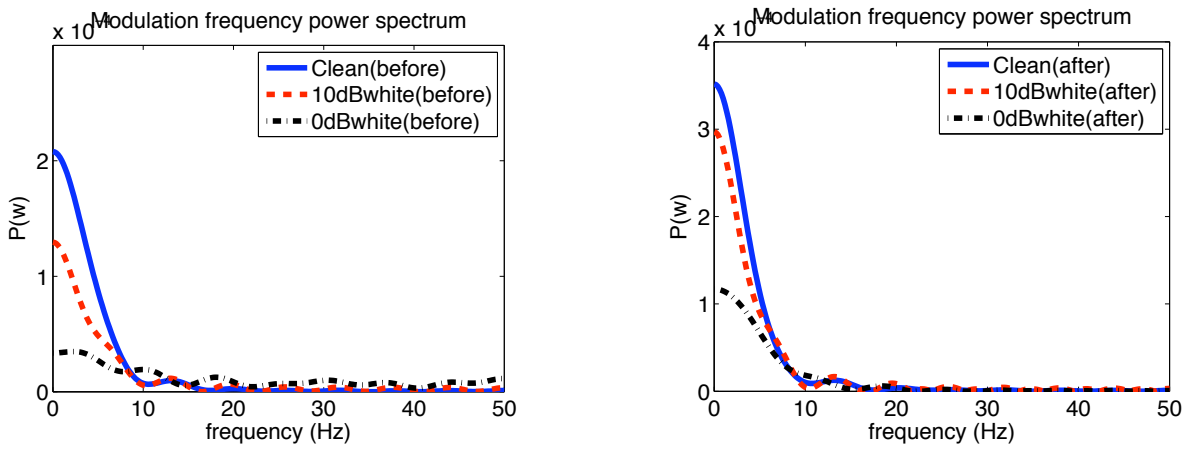

Figure 5.3: *The modulation spectrum of the output of the filters before (left) and after (right) processing.*

described in Sec. 5.1.1 (with L = 17 and $\lambda = 0.49$, as chosen empirically by evaluation of the development set). The major differences between traditional MFCC processing and our present approach is in the use of the rate-level nonlinearity and modulation filter described above. Cepstral mean normalization (CMN) was applied, and delta and delta-delta cepstral coefficients were developed in both cases.

Background noise

To evaluate recognition accuracy in background noise, we selected segments of white, pink, and babble noise from the NOISEX-92 database and segments of music from the DARPA Hub 4 Broadcast News database. These noise samples were artificially added to the test speech with energy adjusted according to obtain SNRs of 0, 5, 10, 15, and 20 dB.

Speech recognition accuracy in background noise (100% minus the word error rate [WER]) is summarized in Fig. 5.4. Each panel compares the recognition accuracy obtained using MFCC coefficients, MFCC coefficients augmented by the nonlinearity described in Sec. 3.3, and MFCC coefficients augmented by both that nonlinearity and the modulation filter described in this paper. As can be seen from that figure, recognition accuracy in the presence of background noise obtained with our proposed system is significantly greater than the accuracy obtained using traditional MFCC processing for all four types of noise. At a WER of approximately 50%, the use of the modulation filtering provides an effective improvement of approximately 1 to 4.5 dB of SNR compared to baseline MFCC processing with CMN (depending on the type of noise), in addition to the improvement of approximately 3 to 7 dB obtained through the use of the

rate-level nonlinearity described in Sec. 3.3.

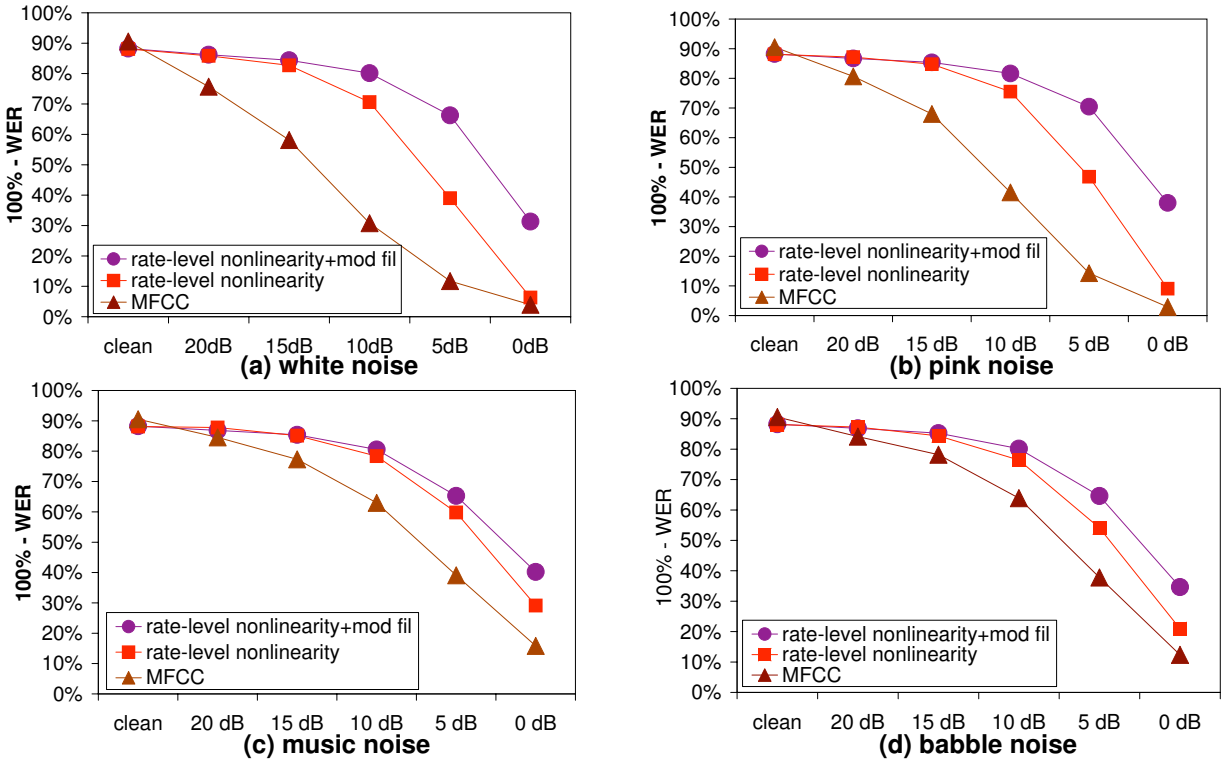

Figure 5.4: *Comparison of recognition accuracy of the proposed system with modulation filtering and peripheral nonlinearity (circles), MFCC processing with nonlinearity (squares) and baseline MFCC processing (triangles) for the RM database in the presence of four different types of background noise. Clean-condition WER: MFCC: 9.45%, RL nonlinearity: 11.88%, RL nonlinearity with modulation filter: 11.78%*

Reverberation

To evaluate the recognition accuracy of our proposed system in reverberant environments, simulated reverberated speech was obtained by convolving clean speech with a room impulse response developed from the room simulator *RIR* based on the image method [48]. The dimensions of the simulated room were $5 \times 4 \times 3$m, with a single microphone at the center of the room and 1 m from the source, with 8 virtual sources included in the simulation. Examples of the simulated room impulse response are shown in Fig. 5.5. The reverberation time (RT_{60}, the time required for the acoustic signal power to decay by 60 dB from the instant a sound source is turned off) was set to 0.3, 0.5, 1.0 and 2.0 s.

Fig. 5.6 describes experimental results as a function of the reverberation time of the simulated room.

Again, the proposed system shows substantial improvement for all reverberation conditions; about 37% relative improvement in WER compared to the MFCC processing and 24% relative improvement compared to the use of rate-level nonlinearity observed for the case of 0.3 s reverberation time.

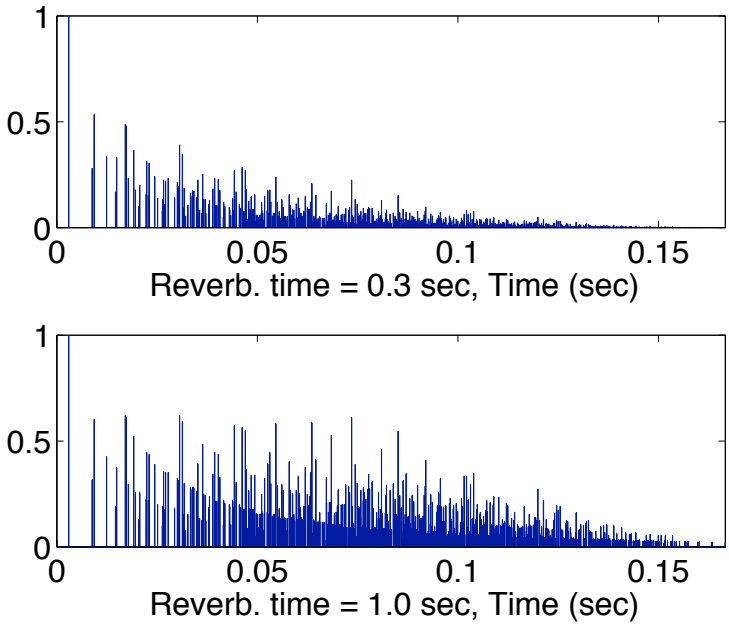

Figure 5.5: *Simulated room impulse response (upper panel: RT = 0.3s, lower panel: RT = 1.0s).*

Effect of the parameter λ

We measured the effect of the mixing parameter λ by adding the same four noise sources described above to speech from our development set a a 10-dB SNR. Fig. 5.7 summarizes the results from these experiments. While the detailed shape of the curves are different for each type of noise, the general trends are similar showing that values of λ in the range of 0.4 to 0.6 provides a broad minimum in WER, at least for the RM database.

Figure 5.8 shows the dependence of λ under different levels of pink noise. As we can see from the figure, under clean conditions, the WER is quite flat for a wide range of λ and it increases when λ becomes closer to 1. Under noisy condition, as the SNR becomes lower, the optimum λ for achieving lowest WER also increases, reflecting the fact that in our filter design, we will prefer a higher value of λ to get rid of noise distortion.

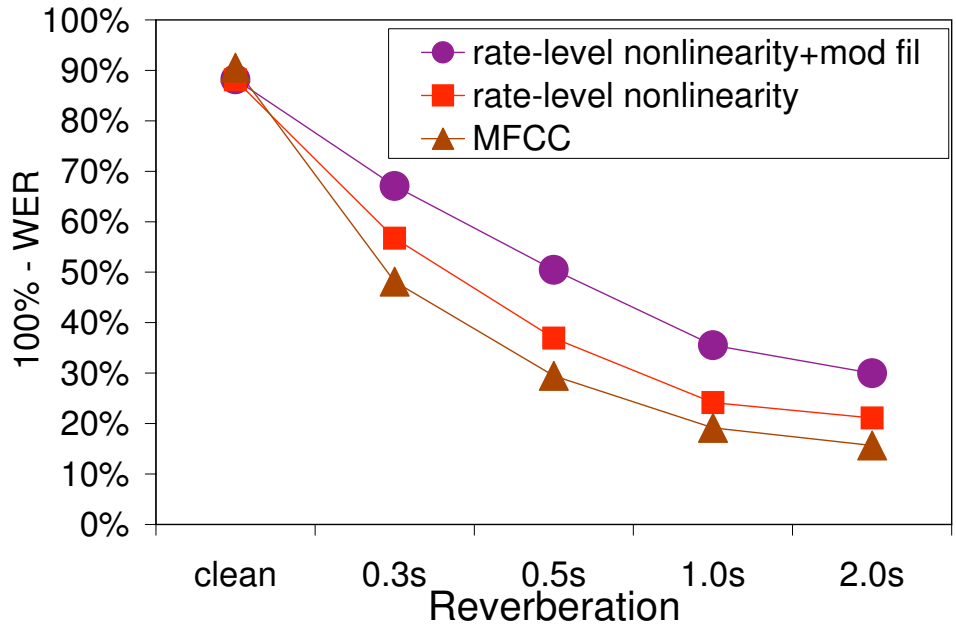

Figure 5.6: *Comparison of recognition accuracy for the same systems as in Fig. 3 as a function of simulated reverberation time using the RM corpus. Clean condition-WERs are the same as in Fig. 3.*

Multi-condition training

To evaluate the performance under multi condition training, the original RM database was down sampled to a sampling rate of 8 kHz. The original clean training set with 1600 utterances was randomly partitioned into 17 equal partitions, each consisting of a single noise type and SNR (with noise levels: 20 dB, 15 dB, 10 dB, 5 dB, and clean speech, and noise type: babble, car, exhibition, and subway). The MFCC features are extracted with frequency range from 130 to 3700 Hz, 13 dimensions, and 31 Mel frequency channels. The parameters for the recognizer are the same as before. Similar to the case of AURORA 2 database in Section 4.6.3, the nonlinearity parameters are set to $w_0 = -0.110, w_1 = -0.521, \alpha = 0.05$ to account for the change of sampling rate from 16 kHz to 8 kHz.

Figure 5.9 shows the recognition accuracy obtained for three types of signal processing: MFCC, rate-level nonlinearity and rate-level nonlinearity with modulation filtering for multi conditioned cases. Test set A contains noises that are the same type as in the training set: babble, car exhibition and subway. Test set B contains: airport, restaurant, street, and train noise. The SNRs are 20 dB, 15 dB, 10 dB, 5 dB, 0 dB, -5 dB. As shown in the figure, with modulation filtering we can obtain better recognition accuracy

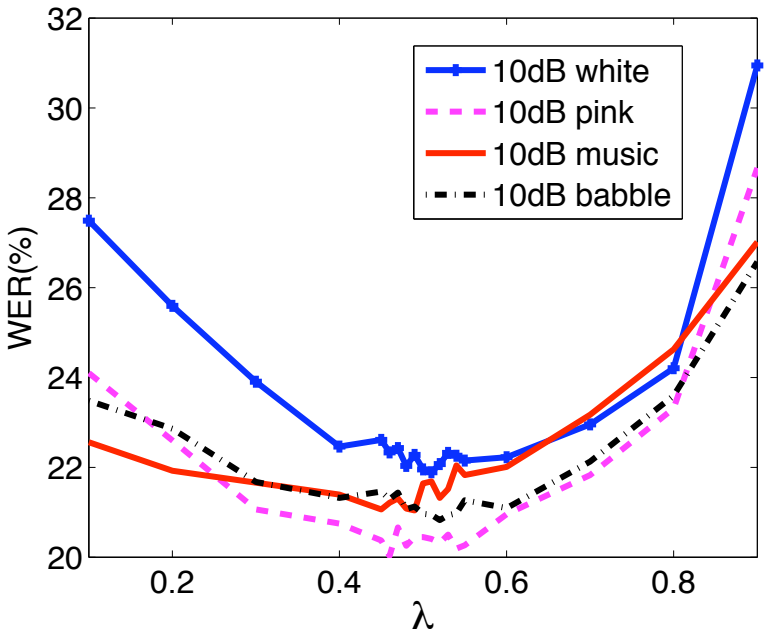

Figure 5.7: *Dependence of WER using the RM development set on the value of the mixing parameter as function of λ under different types of background noise (with SNR fixed at 10 dB).*

compared to both MFCC processing and MFCC processing augmented by the rate-level nonlinearity in the case of multi-condition training, although the improvement is greater when the training and testing environments are different.

Table 5.1 shows significance test results for pairs of three sets of experiments in figure 5.9: MFCC and rate-level nonlinearity, MFCC and rate-level nonlinearity with modulation filtering, and rate-level nonlinearity and rate-level nonlinearity with modulation filtering for multi condition training cases. As can be seen from table, the difference between MFCC and rate-level nonlinearity with modulation filtering is quite significant under noisy conditions with SNRs below 10 dB for Test Set A and below 5 dB for Test Set B.

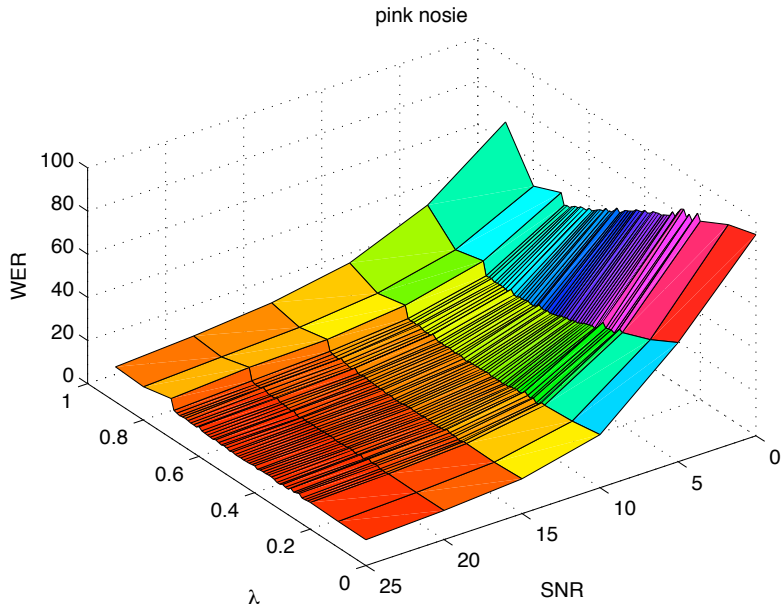

Figure 5.8: *Dependence of WER using the RM development set on the value of the mixing parameter as a function of λ under different level of pink noise.*

5.2.2 Wall Street Journal database

We also evaluated the proposed system on the DARPA Wall Street Journal WSJ0 (WSJ) database. The training set consisted of 7024 speaker-independent utterances from 84 speakers. The test set consisted of 330 speaker-independent utterances using the 5000-word vocabulary, read from the si_et_05 directory using non-verbalized punctuation. Another similar set of 409 speaker-independent utterances from the si_dt_05 directory were used as our development set. The signals were corrupted by white noise and background music maskers, obtained as described as above. Additionally, 10-dB pink noise (also from the NOISEX-92 database) was added to the development set to obtain the λ parameter value of 0.51 used in filter design, as depicted in Fig. 5.10. The SPHINX-III trainer and decoder were implemented with 4000 tied states, a language model weight of 11.5 and 16 GMMs with no further attempt made to tune system parameters. Other conditions are the same as in the RM case.

The results of Fig. 5.11 indicate that the recognition accuracy for the WSJ database follows similar trends to what had been previously described for the RM database, with the modulation filter providing an additional 2-4 dB increase in SNR compared to the SNR obtained using the rate-level nonlinearity (and

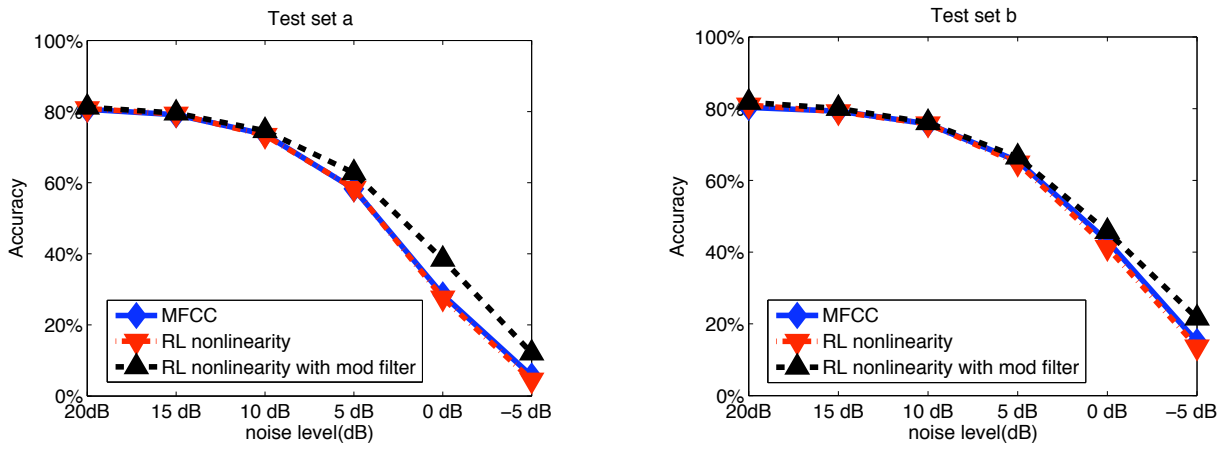

Figure 5.9: *Comparison of recognition accuracy of the proposed system with modulation filtering and peripheral nonlinearity (triangles), MFCC processing with nonlinearity (reverse triangles), and baseline MFCC processing (diamond) for the RM database in the presence of four different types of background noise using multi-condition training.*

an improvement of 5-10 dB compared to the baseline MFCC results).

5.3 Comparison with Wiener filtering

We also compared our proposed modulation filtering with the well-known Wiener filter algorithm as described in Sec. 2.6. To obtain the best possible Wiener filter coefficients without needing to estimate the reference interference correctly, we directly obtained the noisy information from the noise we added to the original clean testing data set. In other words, we used Oracle information for obtaining the optimal Wiener filter coefficients. Figure 5.12 compares the recognition accuracy obtained with the Resource Management database using our proposed modulation filter and the Wiener filter approach with Oracle parameters, with the nonlinearity included in both systems. As we can see from the figure, when the interference is stationary noise like white or pink noise, the performance obtained using the Wiener filter and our proposed filtering is very similar, while under non-stationary noises such as music or babble, our proposed filter performs better than the Wiener filter algorithm. Note also that the choice of the Wiener filter coefficients was based on Oracle information while our filter design is learned from the data.

	SNR	20 dB	15 dB	10 dB	5 dB	0 dB	-5 dB
Test Set A	MFCC v.s. RL	0.569	0.503	0.589	0.497	0.044	<0.001
	MFCC v.s. RL with mod fil	0.101	0.144	0.004	<0.001	<0.001	<0.001
	RL v.s. RL with mod fil	0.142	0.308	<0.001	<0.001	<0.001	<0.001
Test Set B	MFCC v.s. RL	0.063	0.803	0.841	0.384	<0.001	<0.001
	MFCC v.s. RL with mod fil	<0.001	0.033	0.222	0.002	<0.001	<0.001
	RL v.s. RL with mod fil	0.009	0.002	0.144	<0.001	<0.001	<0.001

Table 5.1: *Tables of comparison of statistical significance test results (the probability of having the same recognition performance) on the tasks in figure 5.9.*

5.4 Discussion

Our results indicate that the data-driven approach which we use to design the modulation filter has led to substantially improved speech recognition accuracy compared to traditional MFCC processing under both different types of background noise and different level of reverberation conditions. More generally, the results indicate that a statistical model can be used with good results to maximize the benefits for speech recognition of properties of auditory models.

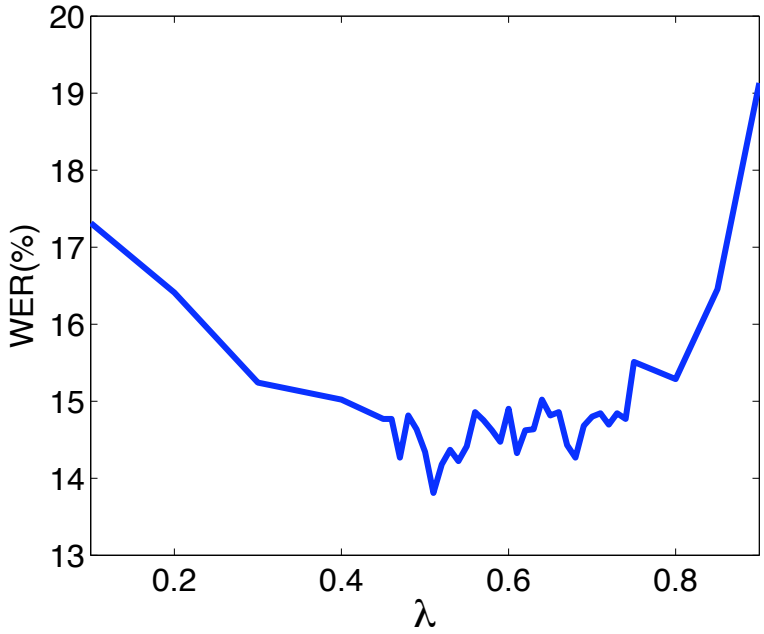

Figure 5.10: *Dependence of WER using the WSJ development set on the value of the mixing parameter as function of λ under pink noise with SNR fixed at 10 dB.*

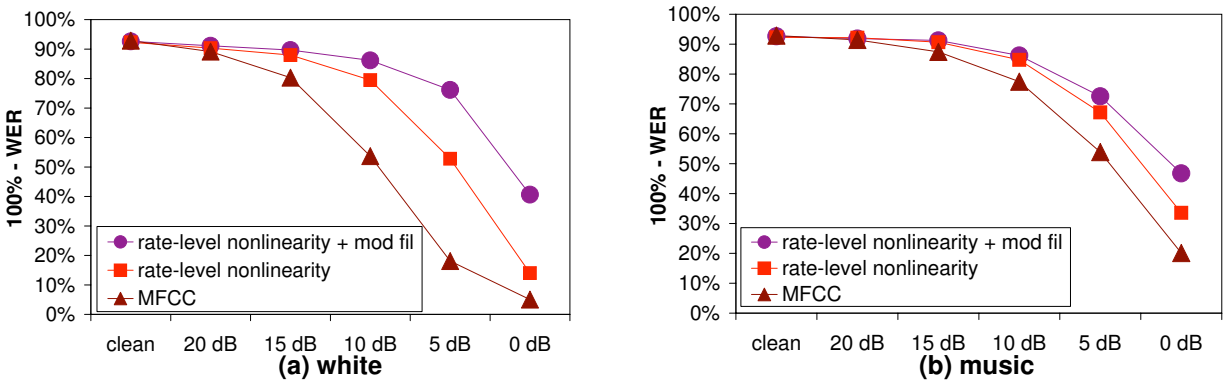

Figure 5.11: *Comparison of recognition accuracy for the same systems as in Fig. 3 in the presence of two types of background noise using the WSJ corpus. Clean-condition WER: MFCC: 7.14%, RL nonlinearity: 7.70%, RL nonlinearity with modulation filter: 7.38%*

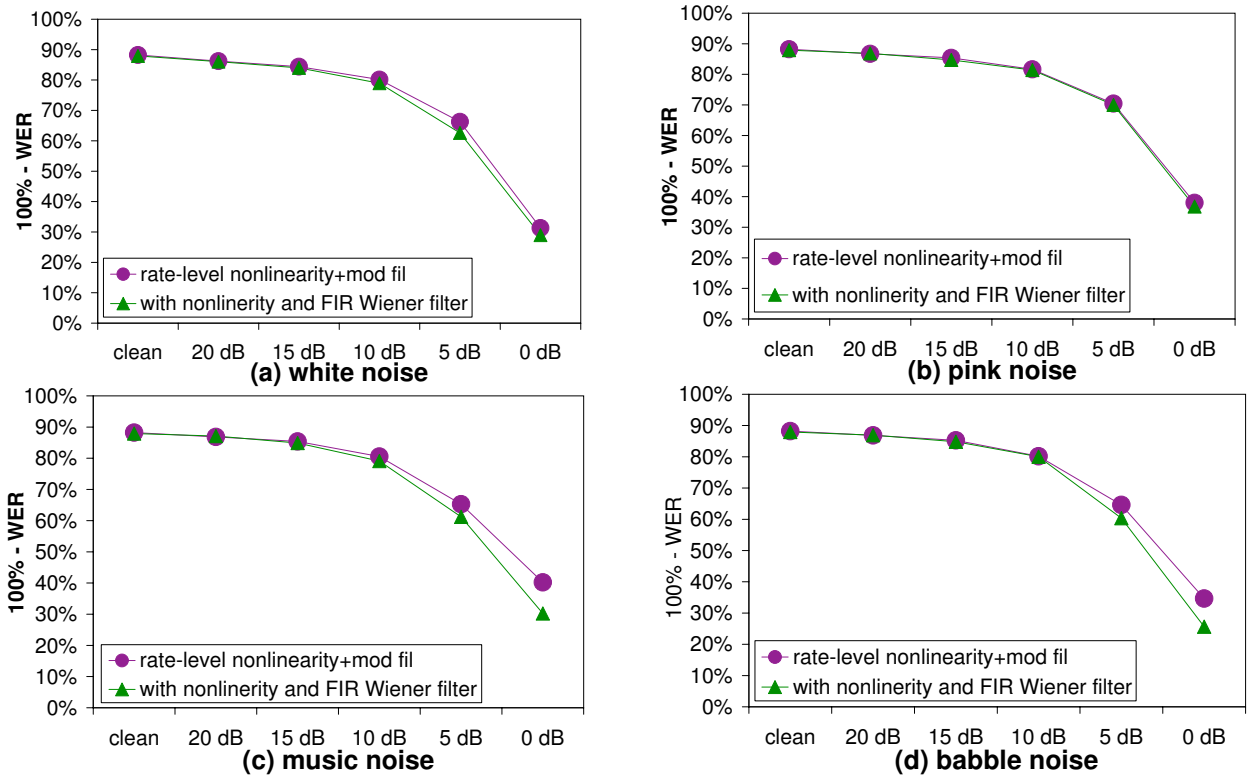

Figure 5.12: *Comparison of recognition accuracy of the proposed system with modulation filtering and peripheral nonlinearity (circles) and system with nonlinearity and Wiener filtering (using Oracle information for obtaining Wiener filter coefficients).*

Chapter 6

Summary and Conclusions

6.1 Introduction

The major goal of this thesis research is to develop computational models for audition. We humans are superb in recognizing speech from other people in all kinds of adverse environments. Motivated by human abilities, in the past decades the application of auditory models for speech recognition has enjoyed both widespread academic interest and experimental success. Conventional techniques attempt to model the human auditory system by fitting various functions to known neural responses. More specifically, the computational models obtained by mimicking the mechanical and chemical responses of human auditory system are optimized at the level of the parameters of the model such that the model responses are as close as possible to the physiologically-measured data. However, the human auditory system is only one of many ways of deriving similar information from incoming signals. We believe that for certain computational models, the details of the human auditory system are less important than the overall framework of processing. So, instead of mimicking the human auditory system through a model, we present a larger framework for feature computation, within which the actual details of the models themselves can be learned from data.

In the remainder of this chapter we summarize the findings and the contributions of this thesis, discuss some of the remaining open questions in this area, and we point out possible directions for future research.

6.2 Summary of the major contributions of the thesis

Analysis of a computational auditory model: We have analyzed the Seneff computational auditory model and concluded that the rate-level nonlinearity which is a part of the nonlinear saturating stage contributes the most to the robustness of auditory processing.

Quantification and optimization: We quantified the rate-level nonlinearity using the logistic function. The parameters of the logistic functions in each frequency channel are optimized using the maximum mutual information (MMI) approach, which maximizes the log posterior probability of the sound classes based on the training data. We showed that this optimization approach is promising and has led to substantially improved recognition accuracy compared to both MFCC and initial results with deterministic initials. More importantly, it indicates that a statistical model can be used to maximize the benefit of property from auditory system for speech recognition purposes.

Optimizing training speed: By using the word lattice to reduce competing-class computation and conjugate gradient descent to reduce the number of iterations, we improved our training speed and made training with a large corpus possible.

Data-driven modulation filter design: We have presented a data-driven algorithm for designing a modulation filter. The filter we proposed has led to substantially improved speech recognition accuracy compared to traditional MFCC processing in the presence of both different types of background noise and different degrees of reverberation.

6.3 Directions for future research

While the algorithms developed in this thesis have been quite successful at improving the performance of recognition system under adverse environments, there is still ample room for additional improvement.

Additional research in the following areas may have the potential to provide additional improvement:

More detailed analysis on the types and optimality of models

In our thesis work, we assume certain form of model properties of auditory model such as logistic function for modeling the rate-level nonlinearity. But there could be different types of models that generalize various characteristics of the auditory system, to determine how closely an optimal instance of these models (in

the classification sense) might resemble human response. For example, like different forms of kernels which have been widely used in machine learning area (in Support Vector Machines (SVM) etc.), the variation produced by different forms of nonlinearity might be helpful in the recognition robustness. More specifically, understanding what would be the effect of different forms of nonlinearity on speech recognition performance and how to incorporate them into training automatic speech recognition (ASR) systems would be an important topic when we are going to investigate more in the analysis. Another aspect of model analysis which could be helpful is the incorporation of temporal information. As in my previous research for the modulation filter design, the temporal information also plays an important role in ASR performance. By incorporating both spectral and temporal information into training process, we should be able to better estimate the models and achieve higher performance.

Extending the ideas to other forms of perception

We can also extend the ideas which we use to analyze the audio data for auditory perception to other forms of perception. For instance, in computer vision, it has been shown that by using the second order statistics and spatial arrangement of structure in the scene, we can reliably estimate and represent the spatial structure of a scene so as to reliably retrieve images that share the same semantic category [49]. With the assumption that our perceptions are formed to help us better recognize and understand the world, we can understand human perceptions better by learning (rather than reverse engineering) from data.

Machine learning in general

Our ultimate goal is to build a formalism for automatic learning systems, or machines that can perceive the world as we humans do. To achieve this requires a good mix of futuristic and present day research. One part of this research could focus on fundamentally different proposals and radical solutions. As an example, we ask whether we can eventually create a real-time learning system that can quickly adapt to the change in the surrounding environments and learn the context information as we humans do in our daily life? It also includes more subtle problems that have greater immediate relevance and impact in industry such as achieving better recognition performance in real applications. In general, although we have achieved much better performance compared with traditional approaches by optimizing parameters directly from data, we are still far from achieving computational perception which allows computers or

machines to have human recognition performance.

Chapter 7

Appendix

With the assumption that the prior probabilities of each class are equal and the observation probability $P(\mathbf{s}|C_i)$ is a single Gaussian. The feature vector $\mathbf{s}$ of the classifier can be computed from the input vector $\mathbf{x}$ by the rate-level nonlinear and DCT transformation:

$$s[k] = \beta[k] \sum_{n=1}^{N} x[n] cos \frac{\pi(2n-1)(k-1)}{2N}, \tag{7.1}$$

$$k = 1, ..., 13, \; with \; \beta[k] = \begin{cases} \sqrt{\frac{1}{N}}, & if \; k = 1 \\ \sqrt{\frac{2}{N}}, & otherwise \end{cases}$$

where $x[n]$ as shown in Eq.(3.1). The overall accumulated posterior probability can be written as:

$$P = \prod_{t} \frac{N(\mathbf{s}_t|\boldsymbol{\mu}_i, \boldsymbol{\sigma}_i)}{\sum_j N(\mathbf{s}_t|\boldsymbol{\mu}_j, \boldsymbol{\sigma}_j)} \tag{7.2}$$

taking the log and directive with respect to $\mathbf{F} = \{\boldsymbol{\alpha}, \boldsymbol{w_0}, \boldsymbol{w_1}\}$ we get:

$$\begin{aligned} Objective &= \max \log P \\ &= \max \sum_t \left[-\frac{1}{2} \sum_{k=1}^{13} [\log \sigma_i[k]^2 + \frac{||s_t[k] - \mu_i[k]||^2}{\sigma_i[k]^2}] \right. \\ &\quad \left. -\log \sum_j \prod_{k=1}^{13} \frac{1}{\sqrt{2\pi\sigma_j[k]^2}} e^{-\frac{||s_t[k]-\mu_j[k]||^2}{2\sigma_j[k]^2}} \right] \end{aligned} \tag{7.3}$$

$$\frac{\partial \log P}{\partial \mathbf{F}} = \sum_t \left[-\frac{1}{2} \sum_{k=1}^{13} \left[\frac{\partial \log \sigma_i[k]^2}{\partial \mathbf{F}} + \frac{\partial}{\partial \mathbf{F}} \frac{||s_t^c[k] - \mu_i[k]||^2}{\sigma_i[k]^2} \right] - \frac{\partial}{\partial \mathbf{F}} \log \sum_j \prod_{k=1}^{13} \frac{1}{\sqrt{2\pi\sigma_j[k]^2}} e^{-\frac{||s_t^c[k]-\mu_j[k]||^2}{2\sigma_j[k]^2}} \right] \tag{7.4}$$

$$\begin{aligned}
\frac{\partial -\frac{1}{2}\log\sigma_i[k]^2}{\partial \mathbf{F}} &= -\frac{1}{2\sigma_i[k]^2} \frac{\partial \sigma_i[k]^2}{\partial \mathbf{F}}, \\
\frac{\partial}{\partial \mathbf{F}} \left[-\frac{1}{2} \frac{||s_t^c[k] - \mu_i[k]||^2}{\sigma_i[k]^2} \right] &= \\
-\frac{1}{2\sigma_i[k]^4} \Bigg[2(s_t^c[k] - \mu_i[k])(& \frac{\partial s_t^c[k]}{\partial \mathbf{F}} \\
-\frac{\partial \mu_i[k]}{\partial \mathbf{F}})\sigma_i[k]^2 - ||s_t^c[k] - \mu_i[k]||^2 & \frac{\partial \sigma_i[k]^2}{\partial \mathbf{F}} \Bigg], \\
\frac{\partial}{\partial \mathbf{F}} \left[\log \sum_j \prod_{k=1}^{13} \frac{1}{\sqrt{2\pi\sigma_j[k]^2}} e^{-\frac{||s_t^c[k]-\mu_j[k]||^2}{2\sigma_j[k]^2}} \right] & \\
= \frac{\frac{\partial}{\partial \mathbf{F}} [\sum_m \prod_{k=1}^{13} \frac{1}{\sqrt{2\pi\sigma_m[k]^2}} e^{-\frac{||s_t^c[k]-\mu_m[k]||^2}{2\sigma_m[k]^2}}]}{\sum_j \prod_{k=1}^{13} \frac{1}{\sqrt{2\pi\sigma_j[k]^2}} e^{-\frac{||s_t^c[k]-\mu_j[k]||^2}{2\sigma_j[k]^2}}} & \\
= \sum_m \Bigg[\sum_{k=1}^{13} [-\frac{1}{2\sigma_m[k]^2} \frac{\partial \sigma_m[k]^2}{\partial \mathbf{F}} & \\
-\frac{1}{2} \frac{\partial}{\partial \mathbf{F}} (\frac{||s_t^c[k] - \mu_m[k]||^2}{\sigma_m[k]^2})] \Bigg] & \\
\cdot \frac{\prod_{l=1}^{13} \frac{1}{\sigma_m[l]} e^{-\frac{||s_t^c[l]-\mu_m[l]||^2}{2\sigma_j[l]^2}}}{\sum_j \prod_{l=1}^{13} \frac{1}{\sigma_j[l]} e^{-\frac{||s_t^c[l]-\mu_j[l]||^2}{2\sigma_j[l]^2}}} &
\end{aligned} \tag{7.5}$$

where $s_t^c[k]$ denote the cepstral mean normalization (CMN) being applied on the incoming utterance. The model parameters $\boldsymbol{\mu}_i$'s and $\boldsymbol{\sigma}_i$'s were obtained in maximum likelihood sense in the same fashion as in the training of speech recognizer:

$$\sigma_i[k] = \frac{1}{\sum_u \sum_{t=1}^{U_T} I(\mathbf{s}_t \in C_i)} \sum_u \sum_t^{U_T} I(\mathbf{s}_t \in C_i) \cdot \left(s_t[k] - \frac{1}{U_T} \sum_{t=1}^{U_T} s_t[k] - \mu_i[k] \right)^2,$$

$$\mu_i[k] = \frac{1}{\sum_u \sum_{t=1}^{U_T} I(\mathbf{s}_t \in C_i)} \sum_u \sum_{t=1}^{U_T} I(\mathbf{s}_t \in C_i) \cdot \left(s_t[k] - \frac{1}{U_T} \sum_{t=1}^{U_T} s_t[k] \right) \tag{7.6}$$

and the partial directive of mean and variance of each class and feature vector $s_t^c[k]$ over $\mathbf{F}$ can be written as:

$$\frac{\partial \sigma_i[k]^2}{\partial \mathbf{F}} = \frac{2}{\#offrames \in C_i} \sum_u \sum_{t=1}^{U_T} I(\mathbf{s}_t \in C_i) \left(\frac{\partial s_t[k]}{\partial \mathbf{F}} - \frac{1}{U_T} \sum_{t=1}^{U_T} \frac{\partial s_t[k]}{\partial \mathbf{F}} - \frac{\partial \mu_i[k]}{\partial \mathbf{F}} \right) \left(s_t[k] - \frac{1}{U_T} \sum_{t=1}^{U_T} s_t[k] - \mu_i[k] \right),$$

$$\frac{\partial \mu_i[k]}{\partial \mathbf{F}} = \frac{1}{\#\, offrames \in C_i} \sum_u \sum_{t=1}^{U_T} I(\mathbf{s}_t \in C_i) \frac{\partial s_t[k]}{\partial \mathbf{F}} - \frac{1}{total\ \#\, offrames} \sum_u \sum_{t=1}^{U_T} \frac{\partial s_t[k]}{\partial \mathbf{F}}$$

$$\frac{\partial s_t^c[k]}{\partial \mathbf{F}} = \frac{\partial s_t[k]}{\partial \mathbf{F}} - \frac{1}{U_T} \sum_{t=1}^{U_T} \frac{\partial s_t[k]}{\partial \mathbf{F}} \tag{7.7}$$

where U_T is the number of frames in each utterance and:

$$\frac{\partial s_t[k]}{\partial \alpha[o]} = \frac{\partial}{\partial \alpha[o]} \cdot \left(\beta[k] \sum_{n=1}^{N} \frac{\alpha[n]}{1 + e^{w_1[n] \cdot y_t[n] + w_0[n]}} cos \frac{\pi(2n-1)(k-1)}{2N} \right) = \beta[k] \frac{1}{1 + e^{w_1[o] \cdot y_t[o] + w_0[o]}} cos \frac{\pi(2o-1)(k-1)}{2N}$$

$$\frac{\partial s_t[k]}{\partial w_0[o]} = -\beta[k] \frac{e^{w_1[o] \cdot y_t[o] + w_0[o]}}{\left(1 + e^{w_1[o] \cdot y_t[o] + w_0[o]}\right)^2} cos \frac{\pi(2o-1)(k-1)}{2N}$$

$$\frac{\partial s_t[k]}{\partial w_1[o]} = -\beta[k] \frac{y_t[o] \cdot e^{w_1[o] \cdot y_t[o] + w_0[o]}}{\left(1 + e^{w_1[o] \cdot y_t[o] + w_0[o]}\right)^2} cos \frac{\pi(2o-1)(k-1)}{2N} \tag{7.8}$$

Bibliography

[1] J. Tchorz and B. Kollmeier, "A model of auditory perception as front end for automatic speech recognition," *J. Acoust. Soc. Amer.*, vol. 106, pp. 2040–2050, 1999.

[2] C.J. Plack, *The Sense of Hearing*, Lawrence Erlbaum Associates, 2005.

[3] E.D. Young and M.B. Sachs, "Representation of steady state vowels in the temporal aspects of the discharge patterns of populations of auditory-nerve fibers," *J. Acoust. Soc. Amer.*, vol. 66, pp. 1381–1403, 1979.

[4] S.B. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllable word recognition in continuously spoken sentences," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 28, pp. 357–366, August 1980.

[5] H. Hermansky, "Perceptual linear predictive (plp) analysis of speech," *J. Acoust. Soc. Amer.*, vol. 87, pp. 1738–1752, 1990.

[6] L.R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proceedings of the IEEE*, pp. 257–286, 1989.

[7] R.F. Lyon, "A computational model of filtering, detec-tion, and compression in the cochlea," in *Proc. IEEE Int. Conf. on Acoust. Speech, and Sig. Proc. (ICASSP)*, Paris, France, May 1982.

[8] S. Seneff, "A joint synchrony/mean rate model of audi-tory speech processing," *J. Phonetics*, vol. 16, pp. 55–76, 1988.

[9] O. Ghitza, "Temporal non-place information in the auditory-nerve firing patterns as a front-end for speech recognition in a noisy environment," *J. Phonetics*, vol. 16, pp. 109–123, 1988.

[10] J. R. Cohen, "Application of an auditory model to speech recognition," *J. Acoust. Soc. Amer.*, vol. 85, pp. 2623–2629, 1989.

[11] Y. Ohshima and R. Stern, "Environmental robustness in automatic speech recognition using physiologically motivated signal processing," in *Proc. Int. Conf. on Spoken Language Proc. (ICSLP)*, Yokohama, Japan, September 1994.

[12] A.M.A. Ali, J.V.D. Spiegel, and P. Mueller, "Robust auditory-based speech processing using the average localized synchrony detection," *IEEE Trans. on Speech and Audio Processing*, vol. 10, pp. 279–292, 2002.

[13] C. Kim, Y.-H. Chiu, and R. Stern, "Physiologically-motivated synchrony-based processing for robust automatic speech recognition," in *Proc. ICSLP*, Pittsburgh, USA, September 2006.

[14] M. Holmberg, D. Gelbart, and W. Hemmert, "Automatic speech recognition with an adaptation model motivated by auditory processing," *IEEE Trans. on Speech and Audio Processing*, vol. 14, pp. 43–49, 2006.

[15] P. Cosi, *Visual Representation of Speech Signals*, chapter Auditory modeling for speech analysis and recognition, John Wiley & Sons Inc, 1993.

[16] D.-S. Kim, S.-Y. Lee, and R.M. Kil, "Auditory processing of speech signals for robust speech recognition in real-world noisy environments," *IEEE Trans. on Speech and Audio Processing*, vol. 7, pp. 55–69, 1999.

[17] M. Ghulam, J. Horikawa, and T. Nitta, "A pitch-synchronous peak-amplitude based feature extraction method for robust asr," in *Proc. Int. Conf. on Spoken Language Proc. (ICSLP)*, Lisbon, Portugal, September 2005.

[18] B.E.D. Kingsbury, N. Morgan, and S. Greenberg, "Robust speech recognition using the modulation spectrogram," *Speech Communication*, vol. 25, pp. 117–132, 1998.

[19] N. Mesgarani, M. Slaney, and S.A. Shamma, "Discrimination of speech from nonspeech based on multiscale spectro-temporal modulations," *IEEE Trans. on Audio, Speech and Language Processing*, vol. 14(3), pp. 920–930, May 2006.

[20] H. Hermansky and N. Morgan, "Rasta processing of speech," *IEEE Trans. on Speech and Audio Processing*, vol. 2(4), pp. 578–589, 1994.

[21] H. Hermansky and S. Sharma, "Traps-classifiers of temporal patterns," in *ICSLP*, Sydney, Australia, November 1998.

[22] Fan-Gang Zeng, Kaibao Nie, Ginger S. Stickney, Ying-Yee Kong Michael Vongphoe, Sshish Bhargave, Chaogang Wei, and Keli Cao, "Speech recognition with amplitude and frequency modulations," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 102(7), pp. 2293–2298, 2005.

[23] L.M. Miller, M.A. Escabi, H.L. Read, and C.E. Schreiner, "Spectrotemporal receptive fields in the lemniscal auditory thalamus and cortex," *J. Neurophysiol.*, vol. 87(1), pp. 516–527, 2002.

[24] T. Chi, P. Ru, and S.A. Shamma, "Multiresolution spectrotemporal analysis of complex sounds," *J. Acoust. Soc. Amer.*, vol. 118(2), pp. 887–906, 2005.

[25] H.J.M. Steeneken and T. Houtgast, "A physical method for measuring speech-transmission quality," *J. Acoust. Soc. Amer.*, vol. 67(1), pp. 318–326, 1980.

[26] N. Kanedera, T. Arai, H. Hermansky, and M. Pavel, "On the relative importance of various components of the modulation spectrum for automatic speech recognition," *Speech Communication*, vol. 28, pp. 43–55, 1999.

[27] Y.-H. Chiu and R. Stern, "Analysis of physiologically-motivated signal processing for robust speech recognition," in *Proc. ICSLP*, Brisbane, Australia, September 2008.

[28] Y.-H. Chiu and R. Stern, "Towards fusion of feature extraction and acoustic model training: A top down process for robust speech recognition," in *Proc. ICSLP*, Brighton, United Kingdom, September 2008.

[29] Y.-H. Chiu and R. Stern, "Learning based auditory encoding for robust speech recognition," in *Proc. ICASSP*, Dallas, USA, March 2010.

[30] Y.-H. Chiu and R. Stern, "Minimum variance modulation filter for robust speech recognition," in *Proc. ICASSP*, Taipei, Taiwan, Aprial 2009.

[31] B. Gold and N. Morgan, *Speech and Audio Signal Processing- Processing and Perception of Speech and Music*, John Wiley and Sons, Inc., 2000.

[32] M.B. Sachs and E.D. Young, "Encoding of steady-state vowels in the auditory nerve: Representation in terms of discharge rate," *J. Acoust. Soc. Amer.*, vol. 66, pp. 470–479, 1979.

[33] S. Shamma, R. Chadwick, J. Wilbur, K. Morrish J., and Rinzel, "A biophysical model of cochlear processing: Intensity dependence of pure tone responses," *J. Acoust. Soc. Amer.*, vol. 80, pp. 133145, 1986.

[34] T. Dau, D. Püschel, and A. Kohlrausch, "A quantitative model of the effective signal processing in the auditory system: I. model structure," *J. Acoust. Soc. Amer.*, vol. 99, pp. 36153622, 1996.

[35] X. Zhang, M.G. Heinz, I.C. Bruce, and L.H. Carney, "A phenomenological model for the response of audi-tory-nerve fibers: I. nonlinear tuning with compression and suppression," *J. Acoust. Soc. Amer.*, vol. 109, pp. 648–670, 2001.

[36] T. Dau, D. Püschel, and A. Kohlrausch, "A quantitative model of the 'effective' signal processing in the auditory system: Ii. simulations and measurements," *J. Acoust. Soc. Amer.*, vol. 99, pp. 3623–3631, 1996.

[37] T. Dau, B. Kollmeier, and A. Kohlrausch, "Modeling auditory processing of amplitude modulation: I. modulation detection and masking with narrow-band carriers," *J. Acoust. Soc. Amer.*, vol. 102, pp. 28922905, 1997.

[38] T. Dau, B. Kollmeier, and A. Kohlrausch, "Modeling auditory processing of amplitude modulation: Ii. spectral and temporal integration," *J. Acoust. Soc. Amer.*, vol. 102, pp. 29062919, 1997.

[39] B.C.J. Moore and B.R. Glasberg, "A revision of zwicker's loudness model," *Acta Acustica*, vol. 82, pp. 335–345, 1996.

[40] R. Drullman, J.M. Festen, and R. Plomp, "Effect of temporal envelope smearing on speech reception," *J. Acoust. Soc. Amer.*, vol. 95(2), pp. 1053–1064, 1994.

[41] R. Drullman, J.M. Festen, and R. Plomp, "Effect of reducing slow temporal modulations on speech reception," *J. Acoust. Soc. Amer.*, vol. 95(2), pp. 2670–2680, 1994.

[42] R. Kay, "Hearing of modulation in sounds," *Physiolog. Rev.*, vol. 62(3), pp. 917, July 1982.

[43] Q. Summertield, A. Sidwell, and T. Nelson, "Auditoly enhancement of changes in spectral amplitude," *J. Acoust. Soc. Amer.*, vol. 81(3), pp. 700–708, Mar 1987.

[44] J. R. Shewchuk, "n introduction to the conjugate gradient method without the agonizing pain," *Technical Report: CS-94-125*, 1994.

[45] A. Varga, H.J.M. Steeneken, M. Tomlinson, and D. Jones, "The noisex-92 study on the effect of additive noise on automatic speech recognition," in *Tech. Rep., DRA Speech Research Unit*, Malvern, England, 1992.

[46] E. Terhardt, "Calculating virtual pitch," *Hearing Research*, vol. 1, pp. 155–182, 1979.

[47] M.C. Liberman, "Auditory nerve response from cats raised in a low noise chamber," *J. Acoust. Soc. Amer.*, vol. 63, pp. 442–455, 1978.

[48] S.G. McGovern, "A model for room acoustics," *http:// 2pi.us/rir.html*.

[49] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic represenation of the spatial envelope," *Int. J. Comp. Vision*, vol. 42(3), pp. 145–175, 2001.