Feature Extraction for Automatic Speech Recognition in Noisy Acoustic Environments

Bojana Gajić

A DISSERTATION SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF

DOKTOR INGENIØR

Department of Telecommunications
Norwegian University of Science and Technology
Trondheim, Norway

2002

Norwegian University of Science and Technology
Department of Telecommunications
7491 Trondheim, Norway

Doktor ingeniøravhandling 2002:60
Repportnr. 420204

ISBN 82-471-5457-9
ISSN 0809-103X

Abstract

This thesis presents a study of alternative speech feature extraction methods aimed at increasing robustness of automatic speech recognition (ASR) against additive background noise.

Spectral peak positions of speech signals remain practically unchanged in presence of additive background noise. Thus, it was expected that emphasizing spectral peak positions in speech feature extraction would result in improved noise robustness of ASR systems. If frequency subbands are properly chosen, dominant subband frequencies can serve as reasonable estimates of spectral peak positions. Thus, different methods for incorporating dominant subband frequencies into speech feature vectors were investigated in this study.

To begin with, two earlier proposed feature extraction methods that utilize dominant subband frequency information were examined. The first one uses zero-crossing statistics of the subband signals to estimate dominant subband frequencies, while the second one uses subband spectral centroids. The methods were compared with the standard MFCC feature extraction method on two different recognition tasks in various background conditions. The first method was shown to improve ASR performance on both recognition tasks at sufficiently high noise levels. The improvement was, however, smaller on the more complex recognition task. The second method, on the other hand, led to some reduction in ASR performance in all testing conditions.

Next, a new method for incorporating subband spectral centroids into speech feature vectors was proposed, and was shown to be considerably more robust than the standard MFCC method on both ASR tasks. The main difference between the proposed method and the zero-crossing based method is in the way they utilize dominant subband frequency information. It was shown that the performance improvement due to the use of dominant subband frequency information was considerably larger for the proposed method than for the ZCPA method, especially on the more complex recognition task. Finally, the computational complexity of the proposed method is two orders of magnitude lower than that of the zero-crossing based method, and of the same order of magnitude as the standard MFCC method.

Preface

This dissertation is submitted in partial fulfillment of the requirements for the doctoral degree of *doktor ingeniør* at the Norwegian University of Science and Technology (NTNU). The advisors have been Professor Torbjørn Svendsen and Associated Professor Magne Hallstein Johnsen, both at the Department of Telecommunications, NTNU.

The work has been conducted in the period from June 1997 to May 2002. In addition to the research activity, the work included compulsory courses corresponding to one year full-time studies, as well as one year of teaching assistant duties. Most of the time I spent with the Signal Processing Group at the Department of Telecommunications, NTNU. In the period from September 1999 to April 2000 I stayed at Shannon Laboratory, AT&T Labs.–Research, Florham Park, New Jersey, USA, and worked under the supervision of Dr. Richard C. Rose. Furthermore, in the period from August 2000 to November 2000, I stayed at the Signal Processing Laboratory, Faculty of Microelectronics, Griffith University, Brisbane, Australia, and worked under the supervision of Professor Kuldip K. Paliwal.

The work has been funded by a scholarship from the Faculty of Electrical Engineering and Telecommunications, NTNU. In addition, my stay in USA was supported by the AT&T Labs.–Research, and a grant from the Norwegian Research Council. Finally, I received a grant from the Australian Research Council for my stay in Australia.

Acknowledgments

My deepest gratitude goes to Professor Kuldip K. Paliwal who has, with his ideas and valuable suggestions, significantly contributed to the outcome of this dissertation. I thank him for inviting me to stay with his group at Griffith University, and for sharing with me his knowledge and experience.

I am grateful to my advisor, Professor Torbjørn Svendsen, for his guidance throughout this work. Without his help and encouragement, this thesis would

have been difficult to accomplish. My sincere thanks also go to Associate Professor Magne Hallstein Johnsen who has always been available for questions and discussions. His suggestions and encouragement during the writing process were of great help.

A close collaboration with Dr. Richard C. Rose gave me a valuable insight into the field of robust automatic speech recognition, which served as a firm base for the rest of the work. I would like to thank him for making my stay at AT&T Labs.-Research possible, and for his help and guidance during the stay.

There is a number of people at the Department of Telecommunications, NTNU, as well as my visiting institutions in USA and Australia, who have given me a lot of help and encouragement along the way, and provided an enjoyable working atmosphere. I especially appreciate efforts made by Tor André Myrvoll, Helge Coward, Ingunn Amdal, Ole Morten Strand, and Kirsten Ekseth at the Department of Telecommunications, NTNU. I would also like to thank my friend Aida Omerović for proof-reading the manuscript.

At last, my deepest gratitude goes to my family and close friends for their encouragement, love and understanding. I am particularly grateful to my mother, Jasna Baturina–Bratanić, for her unconditional love and support along the way.

Trondheim, May 2002
Bojana Gajić

Contents

List of Figures

List of Tables

List of Abbreviations

ASR	Automatic Speech Recognition
AR	Autoregressive
BFCCF	Frequency-Domain Bark-Frequency Cepstrum Coefficients
BFCCT	Time-Domain Bark-Frequency Cepstrum Coefficients
CMN	Cepstral Mean Normalization
DCT	Discrete Cosine Transform
DFT	Discrete Fourier Transform
EIH	Ensemble Interval Histograms
EM	Expectation Maximization
FIR	Finite Impulse Response
FFT	Fast Fourier Transform
GSD	Generalized Synchrony Detector
HMM	Hidden Markov Model
HTK	Hidden Markov Model Speech Recognition Toolkit
LDA	Linear Discriminant Analysis
LP	Linear Prediction
LPCC	Linear Prediction Cepstral Coefficients
MAP	Maximum a Posteriori
MFCC	Mel-Frequency Cepstrum Coefficients
ML	Maximum Likelihood
MLLR	Maximum-Likelihood Linear Regression
PLP	Perceptual Linear Prediction
PMC	Parallel Model Combination
SBCOR	Subband Autocorrelation
SSC	Subband Spectral Centroids
SSCH	Subband Spectral Centroid Histograms
SNR	Signal-to-Noise Ratio
VTN	Vocal-Tract Normalization
WAC	Word Accuracy
ZC	Zero Crossings
ZCPA	Zero Crossings with Peak Amplitudes

Chapter 1

Introduction

Automatic speech recognition (ASR) makes it possible to extract the linguistic message from an acoustic speech signal, and perform a task associated to it. Thus, it enables a more natural way of human-machine communication. Its importance is especially large in the situations where other interaction modes are limited, for example in the emerging pocket-size electronic devices, or for people with certain disabilities.

A major limitation of state-of-the-art ASR systems is their lacking robustness against different acoustic environments. This problem has to be overcome before ASR can be widely used in human-machine interfaces. Much research during the last decade has been concentrated toward finding effective methods for increasing robustness of ASR systems. This thesis gives a small contribution in that direction.

1.1 Basic ASR Concepts

This section describes the main concepts of standard statistical approach to ASR based on hidden Markov modeling of speech, which has been used throughout this study. A comprehensive introduction to hidden Markov modeling and its use in ASR can be found in [91, 89].

Figure 1.1 illustrates the statistical approach to ASR. It consists of a training phase and a recognition phase. In the training phase, the system learns the characteristics of basic speech units from a large speech database with associated transcriptions. This knowledge is then used in the recognition phase, where unknown speech utterances are decoded in terms of the basic speech units.

Feature extraction is the common initial processing stage for both training and recognition phases. It normally involves converting the speech utterance

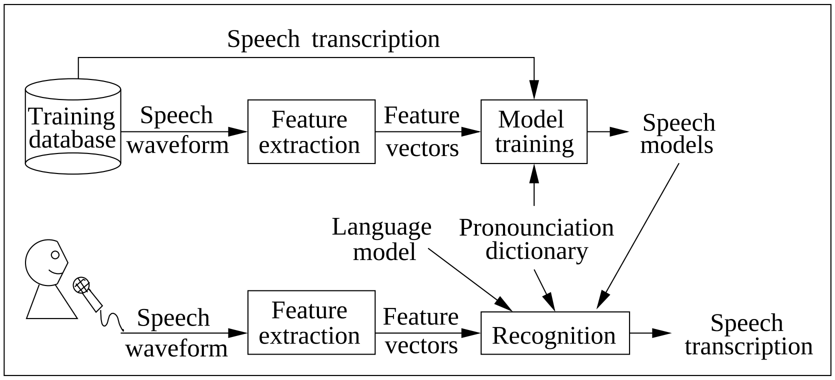

Figure 1.1: Statistical approach to ASR based on hidden Markov modeling

into a sequence of observation vectors $\boldsymbol{O} = \{\boldsymbol{o}_t\}$, also called feature vectors. Feature vectors should only contain the relevant information for distinguishing between different speech sounds. All the other information contained in the speech signal, such as speaker and environmental characteristics, should ideally be discarded. Speech feature extraction is discussed into more detail in Chapter 2.

Each basic speech unit is modeled by a hidden Markov model (HMM). The basic speech units are either whole words or subwords (e.g. phonemes, triphones). Hidden Markov models are stochastic parametric models that consist of a number of states. Each state corresponds roughly to one stationary part of the basic speech unit. An HMM can be seen as a finite state machine that generates speech observation vectors. Each state generates observation vectors according to the corresponding state probability density function. The underlying (hidden) state sequence is governed by the state transition probabilities, given by the state transition matrix. State probability density functions typically have the form of a weighted sum of Gaussian mixture components given by

$$b_j(\boldsymbol{o}_t) = \sum_{m=1}^{M} c_m \mathcal{N}(\boldsymbol{o}_t; \boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m), \tag{1.1}$$

where $b_j(\boldsymbol{o}_t)$ denotes the probability of generating observation vector $\boldsymbol{o}_t$ in state j, $\mathcal{N}(\,\cdot\,; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ is a multivariate Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$, and c_m is the weight of m-th Gaussian mixture component.

In the training phase, the parameters of HMMs for all basic speech units are estimated. This involves estimating the state transition matrices and the parameters associated with state probability density functions. Parameter estimation is normally performed according to the maximum-likelihood (ML) principle using a variant of the expectation maximization (EM) algorithm, referred to as Baum-Welch algorithm. A detailed description of the parameter estimation procedure can be found in [91, 89].

The goal of the recognition phase is to find the sequence of speech models $\boldsymbol{W}$ that maximizes the probability of having generated the speech utterance represented by $\boldsymbol{O} = \{\boldsymbol{o}_t\}$. Thus, the recognition task can be expressed by

$$\hat{\boldsymbol{W}} = \operatorname*{argmax}_{\boldsymbol{W}} P(\boldsymbol{W}|\boldsymbol{O}) = \operatorname*{argmax}_{\boldsymbol{W}} P(\boldsymbol{W})P(\boldsymbol{O}|\boldsymbol{W}) \tag{1.2}$$

The probability $P(\boldsymbol{O}|\boldsymbol{W})$ can be computed from the acoustic models obtained during the training phase, while $P(\boldsymbol{W})$ can be computed from the language model for the given ASR task. In the case of sub-word basic speech units, a pronunciation dictionary is also needed in order to describe all vocabulary words in terms of basic speech units. The Viterbi algorithm is usually used to approximate the probability $P(\boldsymbol{O}|\boldsymbol{W})$ as

$$P(\boldsymbol{O}|\boldsymbol{W}) \approx \max_{\boldsymbol{s}} P(\boldsymbol{O}, \boldsymbol{s}|\boldsymbol{W}), \tag{1.3}$$

where $\boldsymbol{s}$ is the underlying state sequence.

1.2 Robustness in Automatic Speech Recognition

Standard speech features used in ASR have a good ability to discriminate between different speech sounds. However, they are not invariant to speaker and acoustic environment characteristics. Thus, if an ASR system is used in conditions that differ from those observed in the training, there will be a mismatch between observed speech data and trained speech models that can cause a severe degradation of ASR performance.

Hence, increasing robustness in ASR refers to designing techniques that would make the performance of ASR systems less sensitive to the mismatch between training and operation conditions.

The particular speech realization depends on a large number of factors, such as speaker's voice characteristics, dialect, social class, emotional state and context. All those factors can cause mismatched conditions in ASR if they are not adequately represented in the training database.

Acoustic environment refers to all factors that influence the speech signal after it leaves the speaker's mouth, e.g. background noise, microphone and channel characteristics. Figure 1.2 illustrates a commonly used model of the acoustic environment. Background noise is modeled as additive noise, while the influence of the microphone and transmission channel is modeled by a linear filter.

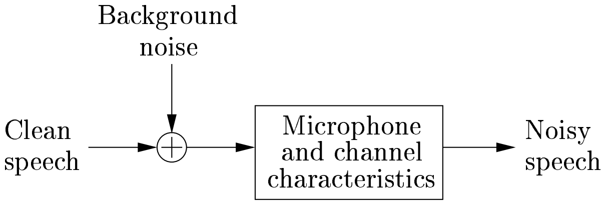

Figure 1.2: Model of acoustic environment

A number of different methods for increasing the robustness of ASR systems have been proposed. They operate either in the feature domain or in the model domain. The main research directions are summarized in the following:

Robust feature extraction aims at finding new speech features that would be less dependent on the particular speaker and acoustic-environment characteristics. They are usually motivated by some aspects of human speech recognition, due to the exceptional ability of humans to deal with all kinds of variabilities in speech signals.

Robust feature transformations refer to techniques that modify speech feature vectors in order to reduce the effect of the particular speaker and acoustic environment, and thus increase the correspondence between the feature vectors and given speech models. Those algorithms can often be seen as an integral part of feature extraction algorithms.

Model adaptation and compensation techniques modify the parameters of speech models, in order to make them more representative of the speech data in target conditions.

Chapter 3 gives an overview of the commonly used model-based algorithms and feature transformations for increasing robustness in ASR, while Chapter 4 is devoted to robust feature extraction.

1.3 This Thesis

This thesis deals with the problem of robust speech feature extraction. The main contributions are summarized in the following:

- Feature extraction techniques based on simulating the processing in the human auditory system are analyzed. Similarities and differences compared to conventional feature extraction methods are found. In particular, use of dominant subband frequency information in the auditory-based methods is pointed out as a possible explanation for their superior robustness against additive background noise.

- A new feature extraction algorithm is proposed that incorporates dominant subband frequency information into speech features in a computationally efficient way. The new algorithm is shown to significantly increase the ASR robustness in different background conditions compared to conventional methods. The advantage of the proposed algorithm becomes larger as we move from a small-vocabulary isolated-word task to a medium-vocabulary continuous speech recognition task, where it also greatly outperforms the computationally expensive auditory-based methods.

The thesis is organized as follows. Chapter 2 summarizes the basic concepts of speech feature extraction in ASR, and describes the commonly used feature extraction methods in state-of-the-art ASR systems. Chapter 3 gives an overview of the model-based methods and feature transformations that are commonly used for increasing the robustness in ASR systems. Chapter 4 describes several auditory-based feature extraction methods that have been shown to increase ASR robustness compared to the conventional methods. Furthermore, a new feature extraction method is proposed, that combines the advantages of auditory-based and conventional methods. Chapter 5 describes an experimental study that compares the ASR performance of the proposed method with the performance of the conventional and auditory based methods. Finally, the main conclusions are summarized in Chapter 6.

Chapter 2

Fundamentals of Speech Feature Extraction

The role of speech feature extraction in ASR is to extract from the speech waveform the information relevant for discriminating between different speech sounds. In order to find speech features with good discriminative properties, knowledge of human speech production and perception has been used in combination with signal processing techniques.

This chapter starts by summarizing the basic concepts from speech production, speech perception and signal processing, that serve as building blocks for a number of different feature extraction methods. Thereafter, some commonly used speech representations are described.

Speech representations should, ideally, be invariant to the particular acoustic environment, transmission channel and speaker characteristics. Feature extraction methods specially designed to improve ASR performance in adverse conditions are addressed in Chapter 4.

2.1 Human Speech Production

Characteristics of different speech sounds are directly related to the way they are produced in human speech production system. This section gives a brief description of speech production, and presents a simple model of speech production that has been used successfully both in speech recognition, synthesis and coding. Furthermore, it explains the motivation for use of short-term spectral analysis as basis for feature extraction in ASR. A detailed discussion of the topics covered by this section can be found in [32, 90, 83, 24].

Speech is produced by forcing the air stream from the lungs through the human speech production apparatus consisting of trachea, glottis with vocal cords, vocal tract and nasal tract. Vocal tract refers to the part of speech production system between vocal cords and lips, while nasal tract refers to the part between velum and nostrils. By varying the position of articulators in the vocal tract (e.g. tongue, jaw, lips), one can control the production of different speech sounds. Thus, the vocal tract can be considered as a time-varying filter whose current configuration decides the particular sound to be produced.

A simple model of speech production is shown in Figure 2.1. The glottal excitation signal, $e(n)$, at the input of the vocal tract, is modeled as an impulse train for voiced sounds (V), and as white noise for unvoiced sounds (U). The vocal tract is modeled as a time-varying filter with impulse response $h(n)$. Since movements of speech articulators are relatively slow, vocal-tract filter characteristics can be assumed to be constant during short time intervals. Thus, speech signal $s(n)$ can be seen as a realization of a quasi-stationary random process. The quasi-stationary property makes it possible to apply well developed signal processing tools for stationary signals to short speech frames.

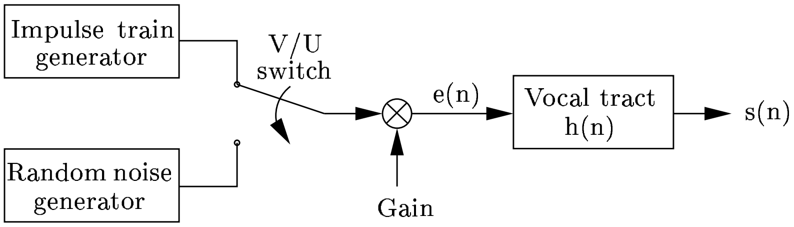

Figure 2.1: A simple model of speech production

According to the model in Figure 2.1, speech sounds can be fully determined by the underlying vocal-tract filter, and the nature (i.e. voiced or unvoiced) and gain of glottal excitation signal. The human ear is highly insensitive to the phase in the speech signals. Thus, only the magnitude response of vocal-tract filter, $|H(f)|$, is important for discriminating between different speech sounds. According to the model in Figure 2.1, the speech power spectrum $S_{ss}(f)$ is given by

$$S_{ss}(f) = |H(f)|^2 \cdot S_{ee}(f), \tag{2.1}$$

where $S_{ee}(f)$ denotes the power spectrum of excitation signal. Since the excitation signal is modeled either as impulse train or as white noise, the magnitude response of the vocal-tract filter can be estimated as the spectral envelope of the speech signal. This explains why spectral envelope estimation serves as a common basis for all widely used feature extraction methods in ASR.

2.2 Human Speech Perception

Humans' exceptional ability to recognize speech, even under adverse conditions, has motivated the use of human speech perception knowledge in ASR. Practically all speech feature extraction methods in use today, incorporate some properties of human speech perception, ranging from simple psycho-acoustic concepts to simulating the processes in human auditory system in great detail. This section summarizes the properties of human speech perception that are commonly used in ASR. It starts with an overview of the physiology of the human auditory system, followed by a description of the most important psycho-acoustic results.

Detailed studies of human speech perception started in the early 20th century by works of Fletcher [33] and von Békésy [108]. Fletcher's work has been recently reviewed in [6]. Good overviews of the topic can be found in [32, 31, 5, 83].

2.2.1 Physiology of Human Auditory System

Human speech perception consists of converting sound pressure waves into the corresponding linguistic messages. In the context of speech perception, human auditory system can be divided into two main parts, the pre auditory-nerve part and the post auditory-nerve part. The first part transforms the sound pressure waves into auditory-nerve activity, and is relatively well understood. The second part comprises higher level processing in the human brain, which transforms the auditory-nerve activity into a linguistic message. Very little is known about the latter part of the system.

The pre auditory-nerve part of human auditory system consists of the outer ear, middle ear, and inner ear. The outer ear directs the sound pressure waves toward the eardrum. The middle ear converts the vibrations of the eardrum into mechanical vibrations at the oval window on the input of the inner ear. It performs impedance matching between the air medium in the outer ear and fluid medium in the inner ear, and protects the inner ear from extensively intense sounds. Finally, the inner ear converts the mechanical vibrations on its input into electrical activity of auditory neurons.

The inner ear consists of the cochlea and auditory nerve. Cochlea is a fluid-filled tube longitudinally partitioned by the basilar membrane. The mechanical vibrations at the entrance of the inner ear excite the fluid inside the cochlea and cause the basilar membrane to vibrate. The displacement at a specific location along the basilar membrane is dependent on the frequency and the intensity of the input sound. Uniformly distributed along the basilar membrane are sensors, the inner hair cells, that transform the displacements of the basilar membrane into firings of the auditory neurons. It is said that a neuron fires when it exhibits an impulse in its electrical potential. The neural activity is processed further by the post auditory-nerve part of the human auditory system.

The frequency response of the basilar membrane varies along its length. The positions closest to the cochlea input are most sensitive to high frequencies, while those close to the apex are most sensitive to the low frequencies. In addition, frequency resolution of human hearing is largest in the low-frequency region, and it decreases gradually toward higher frequencies. Thus, the cochlea can be modeled by a filter bank, where each filter models basilar membrane frequency response at certain position. The logarithm of filter center frequencies is approximately proportional to the corresponding distance from the apex of the basilar membrane. Furthermore, filter bandwidths are proportional to the corresponding center frequencies, causing high frequency resolution at low frequencies and vice versa. A filter bank specially designed to model the frequency response along the basilar membrane into great detail is usually referred to as a cochlear filter bank.

Neural activity generally increases with increased sound intensity due to the increased amplitude of the basilar membrane vibration. The extent of neural activity can be modeled as the logarithm of sound intensity. Furthermore, for frequencies up to 4 kHz, neural firings tend to be time-synchronized with the displacements of the basilar membrane in one direction. This synchrony property is utilized in several speech representations described in Chapter 4, that are based on temporal characteristics of neural firing patterns.

2.2.2 Some Important Results from Psycho-Acoustics

Psycho-acoustics is the study of human auditory perception that relates acoustic signals to what the human listeners perceive. Results from psycho-acoustics help distinguish the properties of speech signals that are important for human perception from those that are irrelevant. This section summarizes several psycho-acoustic results that have successfully been used in ASR.

2.2.2.1 Loudness as a Function of Sound Intensity and Frequency

Sound intensity is measured in terms of sound pressure level relative to a well defined reference level, and is expressed in decibels. Perceived loudness is directly related to sound intensity, and is usually approximated as the logarithm of speech signal power. Alternatively, it can be estimated as the cubic root of signal power [100, 3].

Perceived loudness depends also on frequency. The sensitivity of human hearing is gradually reduced for frequencies lower than approximately 400 Hz and greater than 5 kHz [83].

2.2.2.2 Masking, Critical Bands and Bark Scale

Masking is an important phenomenon in hearing that denotes the fact that a tone (probe) that is clearly perceivable when presented in isolation, becomes imperceivable when presented together with another tone (masker). Consequently, the intensity of the probe has to be raised above the hearing threshold by a certain amount, called amount of masking, in order to be heard. The amount of masking increases with increased masker intensity, and with reduced difference between probe and masker frequencies.

Many masking phenomena can be explained using the notion of critical bands [83]. For example, a band of noise kept at constant intensity while its bandwidth is increased is perceived to have constant loudness until the critical bandwidth is reached. Thereafter, the loudness increases. Furthermore, when two sounds have energies inside the same critical band, the sound with higher energy inside the band dominates the perception and masks the other sound. Critical bandwidths are commonly approximated by the following expression [116]

$$CB = 25 + 75\ \left[1 + 1.4\,(F/1000)^2\right]^{0.69}, \tag{2.2}$$

where CB is critical bandwidth and F is frequency, both given in Hertz.

Bark scale is a perceptually-warped frequency scale designed such that critical bandwidths have a constant value of 1 Bark along the entire scale. The mapping from the linear frequency scale to Bark scale is commonly approximated by the following expression [116]

$$F_{Bark} = 13\arctan\left(0.76\,F/1000\right) + 3.5\arctan\left(F/7500\right)^2, \tag{2.3}$$

where F is frequency given in Hertz, and F_{Bark} is the corresponding perceptual frequency given in Bark. However, in this study, an older approximation [63] was used given by

$$F_{Bark} = 6\ \ln\left(F/600 + \sqrt{(F/600)^2 + 1}\right). \tag{2.4}$$

The difference between the approximations in Equations 2.4 and 2.3 is very small in the low frequency region (e.g. approximately 0.2 Bark at 500 Hz). It increases gradually with increased frequency up to approximately 2000 Hz, and remains approximately constant at 1.5 Bark for higher frequencies.

Critical bandwidths correspond approximately to 1.5 mm spacing along the basilar membrane. Since the typical length of the basilar membrane is 35 mm, it is usually modeled by a set of 24 critical-bandwidth filters uniformly distributed along the Bark scale. Such filter bank is referred to as critical-band filter bank, and is commonly used in speech feature extraction. Note that it is similar to the cochlear filter bank introduced in Section 2.2.1, but the motivation for its use comes from psycho-acoustics rather than physiology of the auditory system.

2.2.2.3 Frequency Perception and Mel Scale

Probably the most commonly used perceptually-warped frequency scale, the mel-scale, evolved from a set of experiments on human frequency perception [101]. The perceived frequency (pitch) of a 1 kHz tone at 40 dB sound pressure was defined as a reference point and assigned the value of 1000 mel. Listeners were then asked to adjust the tone frequency until the pitch they perceived was twice the reference, half the reference and so on. The obtained frequencies were labeled 2000 mel, 500 mel, etc. In this way, the mapping between linear and mel frequency scales was found to be approximately linear for frequencies up to 1000 Hz and logarithmic for frequencies above 1000 Hz. A commonly used analytic approximation [83] of the mapping is given by

$$F_{mel} = 2595 \log_{10}(1 + F/700), \tag{2.5}$$

where F is the linear frequency in hertz, and F_{mel} is the perceived frequency in mel. The relationship between mel and Bark scales is approximately linear. Thus, it is of little importance which of the scales is used in a practical application.

2.3 Spectral Analysis

Although some attempts have been made to extract relevant features for ASR directly from the speech waveform, all commonly used speech representations today are based on some kind of spectral analysis. The motivation for use of spectral analysis in ASR is found both in human speech production and perception. We have seen in Section 2.1 that the envelope of speech spectrum can be used as an estimate of the magnitude response of underlying vocal-tract filter. Furthermore, Section 2.2 showed that human ear can be viewed as a spectral analyser.

This section gives a summary of the spectral analysis techniques that are commonly used in ASR, and describes the most widely used feature extraction methods. A nice overview of the spectral analysis techniques used in ASR can be found in [85], while an in-depth description of spectral estimation in general can be found in [68, 80].

2.3.1 The Spectral Estimation Problem

The power spectrum of a wide sense stationary random process x(n) is defined as

$$S_{xx}(f) = \mathcal{F}\{r_{xx}(k)\} = \sum_{k=-\infty}^{\infty} r_{xx}(k) e^{-j2\pi fk}, \tag{2.6}$$

where $\mathcal{F}\{\cdot\}$ is Fourier transform operator, and $r_{xx}(k)$ is the autocorrelation function of the random process. Alternatively, under the assumption that the autocorrelation function decays sufficiently rapidly, the power spectrum can be expressed as [68]

$$S_{xx}(f) = \lim_{M\to\infty} E\left\{ \frac{1}{2M+1} \left| \sum_{n=-M}^{M} x(n) e^{-j2\pi fn} \right|^2 \right\}, \tag{2.7}$$

where $E\{\cdot\}$ is the expectation operator.

Since the statistical properties of speech signals change with time, spectral estimation has to be done on short signal intervals, called frames. Extracting a signal frame can be regarded as multiplication of the signal by a window function. Windowing gives raise to spectral smoothing. In order to obtain power spectrum estimates closest to the actual power spectrum, the window functions having frequency responses with narrow main lobe and large attenuation in the side lobes are desired. The Hamming window given by

$$w(n) = \begin{cases} 0.54 - 0.46\cos\left(\frac{2\pi n}{N-1}\right) & 0 \le n \le N-1 \\ 0 & \text{otherwise} \end{cases} \tag{2.8}$$

achieves a good compromise between the two requirements, and has become a de facto standard in ASR. The width of the main lobe decreases with increased frame length N. However, too long analysis frames violate the stationarity assumption. As a compromise, frame lengths of 20-30 ms are typically used.

In order to track spectral changes in the speech signal, spectral analysis is repeated on successive analysis frames. The distance between successive analysis frames is referred to as frame shift, and is typically set to 10 ms.

Voiced sections of speech have a negative spectral slope of approximately 20 dB per decade due to the physiological characteristics of the speech production system. A preemphasis filter is often applied to the speech signal before spectral analysis to offset this natural slope. This can potentially improve the quality of the spectral estimate in the high-frequency region. A commonly used preemphasis filter is given by

$$H_{pre}(z) = 1 - a_{pre}z^{-1}, \qquad 0.9 \leq a_{pre} \leq 1 \tag{2.9}$$

A detailed discussion of the effects of the preemphasis filter can be found in [79, 24].

Three main classes of spectral estimation techniques used in ASR are linear prediction analysis, filter-bank analysis and Fourier transform. They are described in the following sections.

2.3.2 Linear Prediction Analysis

Linear prediction (LP) analysis was first applied to speech processing in early 1970s [11, 10]. It provides a compact, low-cost representation of speech spectral envelope, which has been used successfully both in speech recognition, synthesis and coding. However, the use of LP in ASR has decreased during the last decade, mainly due to its relatively poor ability to model speech under noisy conditions. Detailed description of the use of LP in speech processing can be found in [78, 79, 90, 24].

Application of LP analysis to speech signals was motivated by the speech production model in Figure 2.1, and the assumption that the vocal tract can be appropriately modeled by an all-pole filter

$$H(z) = \frac{1}{A(z)} = \frac{1}{1 - \sum_{i=1}^{p} a_i z^{-1}}. \tag{2.10}$$

This is equivalent to assuming that speech samples can be approximated by a linear combination of p previous samples

$$s(n) = \sum_{i=1}^{p} a_i s(n-1) + e(n), \tag{2.11}$$

where p is the prediction order, $\{a_i\}_{i=1}^{p}$ are the LP coefficients, and $e(n)$ is prediction error. The optimal LP coefficients are found by minimizing the prediction error energy over the given analysis frame. Efficient algorithms exist for computing the LP coefficients. For an analysis frame of length N approximately $p^2 + N(p+1)$ operations are required for computation of the prediction coefficients. Typical values of predictor order used in ASR are between 8 and 14.

If the prediction order is properly chosen, LP coefficients provide a compact and precise representation of the speech spectral envelope, particularly for vowel-like sounds that are completely characterized by their spectral peaks. However, the quality of the speech representation is highly dependent on the proper choice of model order. If the model order is too low, not all of the prominent spectral peaks will be properly modeled. If it is too high, the LP model tends to match random variations in the speech spectrum. Furthermore, the all-pole model is not well suited for modeling nasal sounds, which are characterized by spectral zeros. Finally, the major problem of LP analysis is its sensitivity to noise.

Normally, LP coefficients are derived based on the unwarped speech spectrum. However, it is possible to obtain an LP representation corresponding to the perceptually-warped spectrum through the bilinear transform. The complexity of this method makes it less attractive compared to the simpler perceptual warping provided by the filter-bank and Fourier analyses.

2.3.3 Filter-Bank Analysis

Filter-bank based spectral analysis is the oldest type of spectral analysis used in ASR. It consists of passing the speech signal through a bank of bandpass filters covering the speech frequency range. This is usually followed by computation of a short-term power estimate for each subband. The set of subband power estimates provides a compact representation of the speech spectral envelope.

Filter-bank analysis enables simple incorporation of different perceptually-motivated processing steps into spectral estimation. For example, perceptually-based frequency warping is achieved simply by using a critical-band filter bank. Furthermore, it is possible to obtain frequency-dependent time-frequency resolution by choosing different analysis frame lengths for different subband signals.

Since filter-bank analysis corresponds to the way speech is processed by the human auditory system, it serves as basis for a number of alternative speech representations based on detailed modeling of the human auditory system. Several such representations are described in Chapter 4.

The computational cost of filter-bank analysis is dependent on the particular implementation of the filter bank. Generally, it is much more computationally expensive than the alternative spectral estimation methods. For example, for an FIR filter bank consisting of K filters of order L, a total of $K \cdot L$ operations per speech sample are needed to obtain subband signals, and additional K operations per speech sample for computation of subband powers. For a typical choice of the parameters, this results in the order of 10^3 operations per speech sample, compared to the order of 10 operations for LP analysis.

2.3.4 Fourier Transform

The most common way of performing spectral estimation in ASR is based on short-term Fourier transform of speech signals. From the definition of power spectrum in Equation 2.7, it follows that a reasonable spectral estimate can be obtained as

$$\hat{S}_{ss}(f) = \frac{1}{N}\left|\sum_{n=0}^{N-1} s(n)e^{-j\omega n}\right|^2 = \frac{1}{N}\left|\mathcal{F}\{s(n)\}\right|^2, \tag{2.12}$$

where $\mathcal{F}\{\cdot\}$ is the Fourier transform operator, and N is the length of the analysis frame. This is known as periodogram spectral estimate.

The fast Fourier transform (FFT) can be used for efficient computation of the above spectral estimate at N equally spaced frequencies

$$\hat{S}_{ss}(k) = \hat{S}_{ss}(f)|_{f=k/N} \qquad k = 0, \ldots, N-1 \tag{2.13}$$

The FFT reduces the number of computations from N^2 to $N \log_2 N$. However, it sets limitations on the evaluation frequencies.

A number of different speech representations can be obtained from the spectral estimate in Equation 2.12. The most common is to compute a set of subband power estimates similar to those described in Section 2.3.3. This is achieved by multiplying the FFT-based spectral estimate by the frequency response of a subband filter bank. The resulting subband spectral estimates are then integrated to obtain subband power estimates. Perceptual warping of the frequency axis can easily be achieved through a proper choice of the filter bank. However, the time-frequency resolution in the Fourier analysis is fixed. The computational cost of this method is similar to that of LP analysis.

2.3.5 Cepstrum

Cepstral representation of speech signals is usually computed as the discrete cosine transform (DCT) of the logarithm of the speech spectral representation. When the speech spectrum is estimated by a vector of subband power estimates $\{p_k\}_{k=1}^{K}$, cepstral coefficients are computed as

$$c(m) = \frac{1}{K}\sum_{k=1}^{K} \log(p_k)\cos\left(\frac{\pi m(k-0.5)}{K}\right), \quad 1 \leq m \leq M. \tag{2.14}$$

where M is the total number of cepstral coefficients. Alternatively, cepstral coefficients can be computed directly from the LP coefficients using the following recursion [85]

$$c(m) = \begin{cases} a(m) & m = 1 \\ a(m) + \sum_{l=1}^{m-1} \left(1 - \frac{l}{m}\right) a(l)c(m-l) & 2 \leq m \leq M. \end{cases} \tag{2.15}$$

Cepstral transformation was originally used in order to achieve separation between vocal-tract characteristics and glottal excitation. The two components of the speech signal become additive in the cepstral domain, and the contribution of glottal excitation can be removed by simple windowing in the cepstral domain. This is referred to as liftering.

However, the original motivation does not hold for cepstral representation obtained from perceptually warped speech spectrum. Nevertheless, cepstral transformation plays an important role in speech feature extraction mainly due to good decorrelation properties of the DCT transform. Decorrelated feature vectors are desirable in the HMM framework, since they can be modeled by diagonal covariance matrices. This reduces considerably the computation complexity in both training and recognition phase.

Cepstrum is a strongly decaying function, and only a small number of coefficients is required to represent the spectral content. Typically, 12 cepstral coefficients are used to represent a speech frame.

2.3.6 Capturing Spectral Dynamics

All speech representations discussed so far are computed from a single speech frame. However, it has been shown [34] that ASR performance can be considerably improved by incorporating information about spectral changes into speech feature vectors. This is achieved by estimating time derivatives of short-time spectral representations and augmenting them to feature vectors.

Estimates of spectral time derivatives, referred to as delta parameters, are usually obtained by linear regression over a number of successive speech frames. If the speech representation at time frame t is given by cepstral vector $\boldsymbol{c}_t$, the corresponding vector of delta parameters is given by

$$\boldsymbol{\Delta}_t = \frac{\sum_{l=1}^{L} l \left(\boldsymbol{c}_{t+l} - \boldsymbol{c}_{t-l}\right)}{2 \sum_{l=1}^{L} l^2}, \tag{2.16}$$

where parameter L determines the number of speech frames used in the linear regression (typically L=2). Alternatively, delta parameters can be computed by taking simple differences

$$\boldsymbol{\Delta}_t = \frac{\boldsymbol{c}_{t+L} - \boldsymbol{c}_{t-L}}{2L}. \tag{2.17}$$

Estimates of the second derivatives, referred to as delta-delta coefficients, can be obtained by applying Equations 2.16 or 2.17 to the delta coefficients. It is common to include both delta and delta-delta parameters in speech feature vectors.

2.4 Conventional Speech Representations

Previous sections of this chapter have summarized common signal processing steps used in feature extraction for ASR, and explained motivation for their use based on human speech production and perception. They serve as building blocks for a number of different feature extraction methods used in ASR. This section gives a summary of the conventional feature extraction methods, while Chapter 4 describes several alternative methods, specially designed to improve ASR performance in the presence of noise.

2.4.1 Representations Based on LP Analysis

Several different representations derived from the LP coefficients have been used in speech recognition including reflection coefficients, line-spectrum pair parameters, perceptually warped LP coefficients and LP-based cepstral coefficients [24]. Among them, linear prediction cepstral coefficients (LPCC), given by Equation 2.15, have been most successful, and are the only LP-based features commonly used today. LPCC are usually augmented by delta and delta-delta parameters.

2.4.2 Representations Based on Subband Power Estimates

Mel-frequency cepstral coefficients (MFCC) computed from FFT-based subband power estimates [22] are currently the most popular features used in ASR. This is due to their superior noise robustness compared to LP-based features, the simplicity with which perceptual warping can be employed, and the low computational cost. A standard procedure for their computation is illustrated in Figure 2.2. For each analysis frame $s(n)$, an FFT-based spectral

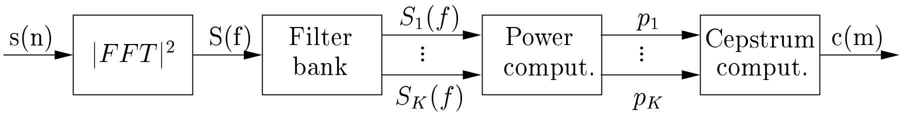

Figure 2.2: FFT-based MFCC computation

estimate $S_{ss}(f)$ is computed first, and then passed through a critical-band filter bank. The filter bank consists of K overlapping filters having constant bandwidth on the mel scale and center frequencies uniformly distributed on the mel scale. Next, a power estimate p_k is computed for each subband signal $S_k(f)$. Finally, a set of cepstral coefficients $c(m)$ is derived from the vector of subband power estimates. Delta and delta-delta coefficients usually augment the feature vector. Note that similar representations can be obtained by using other perceptual warping functions (e.g. Bark scale).

Alternatively, subband filtering and power estimation can be done in the time domain, resulting in the feature extraction method shown in Figure 2.3. Note, however, that computational cost is greatly increased in this case due to time-domain filtering (see Section 2.3.3).

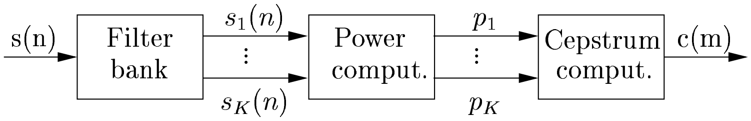

Figure 2.3: Time-domain based MFCC computation

Chapter 3

Common Methods for Increasing Robustness in ASR

This chapter gives an overview of several well known methods for increasing robustness in automatic speech recognition that have been shown to be beneficial on a number of different recognition tasks. The methods for increasing robustness in ASR can be classified into three main classes, namely, model-based techniques, robust feature transformations and robust feature extraction methods. Model-based techniques aim to modify the parameters of acoustic speech models in order to make them more representative with respect to observed speech data. The goal of robust feature transformations, on the other hand, is to modify the speech feature vectors such that the mismatch between the speech data and given acoustic models is reduced. Finally, robust feature extraction refers to a number of methods that utilize different aspects of knowledge of human speech perception, in order to derive speech features that would be less dependent on the acoustic background environment.

The chapter starts with an overview of speech endpoint detection in Section 3.1, since several methods for increasing robustness in ASR systems depend on a reliable discrimination between speech and background events. Then, several model-based techniques are described in Section 3.2, followed by an overview of commonly used robust feature transformations in Section 3.3. Robust feature extraction, which is the main interest of this thesis, is discussed in Chapter 4.

3.1 Robust Speech Endpoint Detection

A reliable method for discriminating between speech and background intervals in an acoustic signal is an important part of an ASR system. It can considerably reduce the amount of computation, and improve the system accuracy. Furthermore, classification of signal frames into speech and background noise is required by several methods for improving ASR robustness described later in this chapter. An ideal endpoint detector should be reliable, robust, accurate, adaptive, simple, able to perform in real time, and require no a priori knowledge of noise [95]. The relative importance of the different requirements depends on the particular recognition task.

A simple and reliable way of speech endpoint detection is the use of the push-to-talk mode, that requires a button to be pressed in order to start recognition. However, this simple solution is not convenient for all applications. Systems that require a continuously listening mode need an automatic algorithm for separating speech and background events.

Speech endpoint detectors consist of a feature extraction block followed by a simple two-class classifier. Conventional speech endpoint detectors are based on short-term energy, possibly combined with zero-crossing rate and different duration constraints [88, 72, 112, 95]. The classification is based on a set of fast or adaptive thresholds. These algorithms work well only at high SNRs.

Several alternative approaches have been proposed in order to improve robustness of speech endpoint detection against changes in the acoustic environment. Some of the major directions are summarized in the following:

- One idea is to use alternative features for endpoint detection that are less dependent on the particular acoustic environment. For example, use of cepstral difference measure was proposed in [54], and a good separation between speech and background regions was demonstrated for both clean and noisy speech.

- Since the detection of speech intervals corresponding to vowels is easier than the general problem of speech detection, another approach to endpoint detection is to first locate the vowel intervals in the speech signal, and then apply some refinement procedure. In [106], the detection of vowel intervals was based on a signal periodicity measure, while [64] relied on measuring energy in the frequency region covering the first three speech formants. Possible low-intensity regions in the beginning and end of the utterance can be accounted for by incorporating security margins before the first vowel region and after the last vowel region. This procedure is appropriate for many applications where the exact determinations of speech endpoints is not critical.

- In the HMM framework, it is common to train models that represent the acoustic background. In this case, speech endpoint detection can be achieved implicitly through the recognition process, by searching for the best path in the recognition network consisting of both speech and background models [111]. The disadvantage of this method is its high computational cost, since the entire signal has to be passed through the recognizer. Furthermore, the introduction of background models makes the recognition process more complex. However, the implicit endpoint detection may lead to greater accuracy in presence of noise than the energy-based methods.

- HMM modeling can also be used to improve the quality of explicit endpoint detection. This can be done by training a simple two-class classifier based on only two HMMs, one representing background and the other one representing speech. Any set of features can be used with this approach. For example, normalized log-energy and delta log-energy were used in [1], while cepstrum and delta-cepstrum coefficients were used in [35, 30]. The parameters of the HMMs can be dynamically updated using the EM algorithm. This method will be referred to as probability based speech endpoint detection.

In an earlier study [35], a comparison between the implicit, energy-based, and probability-based endpoint detection methods was performed on an ASR task. The evaluation was done both on clean speech and in presence of additive car noise. In addition, the ASR performance on hand-labeled data was evaluated. The ASR task consisted in recognizing strings of four Japanese digits. The resulting recognition error rates are shown in Table 3.1. It can be seen that the probability-based approach performed best both in clean and noisy conditions. Its performance was in turn very close to that achieved by hand labeling.

Table 3.1: Performance comparison of different endpoint detection methods

	Recognition error rate [%]	
Method	SNR=25 dB	SNR=12 dB
Hand labeling	2.37	12.62
Probability-based	3.10	14.76
Energy-based	5.24	22.65
Implicit	12.40	27.44

3.2 Model Adaptation and Compensation

This section describes several model-based techniques for increasing robustness in ASR, which have been shown to be beneficial on a number of different recognition tasks. They are based on modifying the parameters of speech acoustic models in order to make them more representative with respect to the observed speech data. All of the methods rely on a small amount of adaptation data collected in the target operating conditions.

3.2.1 Maximum a Posteriori Adaptation

Maximum a posteriori (MAP) adaptation [45] represents an efficient method for adjusting model parameters to new operation conditions if only a small amount of speech data from the new operating conditions is available.

Let $\mathbf{\Phi}$ denote the parameter vector of an acoustic model $\Lambda = \Lambda(\mathbf{\Phi})$, and let $\boldsymbol{O}_a$ be the available adaptation data from target operating conditions. Then, the adapted model parameter vector is estimated by maximizing a posteriori probability of the parameter vector given the adaptation data

$$\hat{\mathbf{\Phi}} = \underset{\mathbf{\Phi}}{\operatorname{argmax}}\, P(\mathbf{\Phi}|\boldsymbol{O}_a) = \underset{\mathbf{\Phi}}{\operatorname{argmax}}[P(\boldsymbol{O}_a|\mathbf{\Phi})P(\mathbf{\Phi})]. \tag{3.1}$$

It can be seen from Equation 3.1 that MAP adaptation utilizes the prior information about the distribution of parameter vectors, $P(\mathbf{\Phi})$. This information can be estimated using unadapted acoustic models. If no prior information is available, MAP adaptation reduces to standard maximum-likelihood (ML) parameter estimation.

MAP adaptation formulas for HMMs with Gaussian mixture state observation densities were derived in [45]. They show that adapted model parameters are computed as a weighted sum of the prior parameters and the ML-estimate computed using the adaptation data. Thus, as the amount of adaptation data increases, the adapted parameters converge to the ML estimates. On the other hand, if no adaptation data is available the prior value is adopted.

Two major limitation of the MAP adaptation were stated in [58]. First, it requires an accurate initial guess for the prior distribution $P(\mathbf{\Phi})$, which is often difficult to obtain. Second, only model parameters that are observed in the adaptation data can be modified from their prior values. In order to overcome these problems, several modifications of the standard MAP algorithm have been proposed [2, 97].

3.2.2 Maximum-Likelihood Linear Regression

Maximum-likelihood linear regression (MLLR) [76, 40] is an alternative approach to adapting model parameters to new operating conditions. The aim of MLLR is to obtain a set of transformation matrices for the model parameters that maximizes the likelihood of the adaptation data. The adaptation of model means is given as

$$\hat{\boldsymbol{\mu}} = \boldsymbol{A}\boldsymbol{\mu} + \boldsymbol{b} = \boldsymbol{W}\boldsymbol{\mu}', \tag{3.2}$$

where $\boldsymbol{W} = [\boldsymbol{b} \;\; \boldsymbol{A}]$ is the transformation matrix and $\boldsymbol{\mu}' = [1 \;\; \boldsymbol{\mu}^T]^T$ is extended mean vector. The transformation matrix $\boldsymbol{W}$ is estimated by maximizing the likelihood of the adaptation data $\boldsymbol{O}_a$, given the adapted model $\Lambda_a = \Lambda_a(\boldsymbol{W})$

$$\hat{\boldsymbol{W}} = \operatorname*{argmax}_{\boldsymbol{W}} P(\boldsymbol{O}_a | \Lambda_a(\boldsymbol{W})). \tag{3.3}$$

The maximization is performed using the EM algorithm. The adaptation of model variances can be obtained in a similar manner [40, 39]. However, most of the performance gain obtained by MLLR adaptation is due to adaptation of the means.

As with MAP adaptation, MLLR adaptation also requires a small amount of adaptation data from target operation conditions. However, the MLLR method adapts all model parameters irrespective of whether they have been observed in the adaptation data. This is achieved through transformation sharing, i.e. the same transformation is used for a number of model parameters. All model parameters sharing the same transformation constitute a regression class. The number of different regression classes is dependent on the amount of the adaptation data available. It can be increased dynamically, as more adaptation data become available [75]. The performance of MLLR adaptation can be further improved by incorporating ideas from MAP adaptation into the MLLR framework [18, 19, 99], using prior distributions of the transformation matrix parameters.

Both MLLR and MAP require knowledge of the transcriptions of the adaptation data in order to align the data to the correct acoustic models. If the correct transcriptions are not available, they can be estimated through an initial recognition pass using the original acoustic models.

3.2.3 Parallel Model Combination

Parallel model combination (PMC) [41, 44] is a noise compensation technique aimed at reducing the mismatch between clean speech models and speech data observed in a noisy acoustic environment. This is achieved by combining the clean speech models and a model of the acoustic environment into a new set of acoustic models that better represents noisy speech.

PMC assumes that a small amount of background data from the target environment is available in order to train the background model. The model can be dynamically updated to compensate for slow changes in the acoustic environment.

It is assumed that speech and background noise are additive in the linear spectral domain and uncorrelated. Consequently, the statistics of noisy speech features in linear spectral domain can be obtained by simple addition of the corresponding speech and background feature statistics

$$\begin{aligned}\boldsymbol{\mu}_x &= \boldsymbol{\mu}_s + \boldsymbol{\mu}_n \qquad (3.4)\\ \boldsymbol{\Sigma}_x &= \boldsymbol{\Sigma}_s + \boldsymbol{\Sigma}_n, \qquad (3.5)\end{aligned}$$

where subscripts x, s and n denote noisy speech, clean speech and additive background noise, respectively. However, the model parameters are usually in the cepstral domain. Thus, in order to combine speech and background model parameters, they have to be converted into linear spectral domain, then combined, and finally converted back into cepstral domain. Closed form expressions for those conversions were derived in [41].

The above procedure is repeated for all combinations of speech and background Gaussian components. Thus, the complexity of the resulting models is directly dependent on the chosen structure of the background model. Consequently, there is a trade-off between the quality of noise modeling and system complexity.

It is implicitly assumed in the PMC procedure that the combination of two Gaussian distributions results in a new Gaussian distribution. However, this assumption is not always realistic. A modified, data-driven, approach was suggested in [42], which does not require this assumption. In this approach, speech and background acoustic models are used to generate a number of observation vectors, which are combined according to the procedure described above to produce noisy speech observation vectors. Then, noisy speech models can be trained using the generated noisy speech observation vectors. Using this procedure, delta and delta-delta parameters can also be successfully updated [44].

An advantage of the PMC method compared to the MAP and MLLR adaptation methods, is the fact that PMC does not require any speech data for noise compensation. Thus, noise compensation can be done during speech pauses. A drawback of this method compared with the adaptation procedures is that it can only be used for compensation of additive noise. Furthermore, the method is limited to cepstral coefficients. However, PMC has been successfully combined with procedures for convolutional noise compensation in order to simultaneously compensate for both additive and convolutional noise [43, 81].

PMC has been shown to be very successful on several recognition tasks, where different noise types and levels were artificially added to clean speech data [41, 44, 113]. Furthermore, the efficiency of PMC combined with ML cepstral bias estimation for channel compensation, has been demonstrated on a speech database collected in a real noisy environment where both transducer and acoustic environment mismatch were present [38].

3.3 Robust Feature Transformations

In this section, several commonly used feature vector transformations are described that have been proven successful in increasing robustness of ASR systems. The first three methods normalize the feature vectors with respect to some environment or speaker dependent characteristics estimated from the speech signal, while the last method finds a linear transformation of the feature space that improves separability between different speech classes.

3.3.1 Cepstral Mean Normalization

Cepstral mean normalization (CMN) [9] is a widely used, simple and effective method for removing the effect of microphone and transmission channel characteristics from speech cepstral representation. It can also be useful for reducing inter-speaker variability in the speech representation.

CMN is based on the fact that any convolutional distortion in the time domain transforms to additive distortion in cepstral domain, i.e.

$$\boldsymbol{c}_t^x = \boldsymbol{c}_t^s + \boldsymbol{c}_t^h, \tag{3.6}$$

where $\boldsymbol{c}_t^x$ is cepstral representation of clean speech corrupted by convolutional noise measured at time t, $\boldsymbol{c}_t^s$ is cepstral representation of the clean speech, and $\boldsymbol{c}_t^h$ is cepstral bias due to the convolutional noise. Under assumption that the microphone, channel and speaker characteristics remain approximately constant during entire utterance, the part of the cepstrum corresponding to the clean speech can be estimated as

$$\hat{\boldsymbol{c}}_t^s = \boldsymbol{c}_t^x - \frac{1}{T}\sum_{\tau=1}^{T} \boldsymbol{c}_\tau^x, \tag{3.7}$$

where T is the number of observations in the utterance.

An improved version of CMN is obtained by computing the cepstral mean only over the signal frames that contain speech [92]. This modification requires the use of a speech detector (see Section 3.1). In order to make the use of CMN more suitable for real-time applications, the long-term average in Equation 3.7 can be replaced by a short-term average computed over a given number of signal frames [92]. A number of more complex methods for cepstral normalization have been proposed in the literature. They are summarized in [63].

3.3.2 Vocal-Tract Normalization

Differences in vocal-tract lengths represent one of the major sources of inter-speaker variability in speech signals. The goal of vocal-tract normalization (VTN) is to compensate for these differences in the speech analysis stage, thus reducing the inter-speaker variability of speech feature vectors.

The formant frequencies of speech sounds are inversely proportional to the speaker's vocal-tract length. Thus, VTN can be done by simple linear warping of speech spectra

$$S_{ss}^{\alpha}(f) = S_{ss}(\alpha f), \tag{3.8}$$

where α is a speaker-dependent frequency warping factor. Two main approaches to warping factor estimation have been proposed. The first one is based on explicit estimation of formant frequencies [109, 29]. The optimal warping factor is estimated by comparing average formant positions for the given speaker with corresponding average formant positions computed across all training speakers. The second approach estimates the optimal warping factor by maximizing the probability of warped observation vectors $\boldsymbol{O}^{\alpha}$, with respect to the corresponding transcriptions W, and a set of speaker-independent acoustic models Λ [7, 73, 74]

$$\hat{\alpha} = \underset{\alpha}{\operatorname{argmax}}\, P(\boldsymbol{O}^{\alpha}|W, \Lambda). \tag{3.9}$$

In practice, the optimal warping factor is estimated by evaluating the probability in Equation 3.9 for a number of different values for α, and choosing the value that maximizes the probability.

The procedure for implementing VTN in ASR is summarized in the following. In the training phase, the optimal warping factor is estimated for each speaker based on all utterances corresponding to that speaker. Then, each training utterance is normalized by the corresponding warping factor, and a set of normalized acoustic models Λ_N is trained. In the recognition phase, the warping factor has to be estimated based on the single input utterance, since the identity of the speaker is usually unknown. The normalized utterance is then recognized using the set of normalized acoustic models Λ_N.

In the ML-based approach, computation of warping factors is dependent on an existing set of acoustic models. In the training phase, the acoustic models trained on unwarped utterances are used for estimation of initial warping factors. Then, normalized acoustic models and improved warping factors are estimated by an iterative procedure. In the test phase, the correct speech transcription, needed for estimation of the warping factor, is typically not known. Thus, a two-pass recognition procedure is used. The unknown transcription is estimated from the unwarped speech utterance during the first recognition pass, it is used for estimation of the warping factor, and the final transcription is estimated in the second recognition pass using the warped utterance.

An alternative procedure for estimating warping factors in the recognition phase that does not require estimation of the speech transcription, was proposed in [74]. The idea is to divide all speakers into a number of classes based on their corresponding warping factors. In the training phase, one Gaussian mixture was trained for each warping factor. Unwarped feature vectors belonging to the utterances that have been assigned to the particular warping factor were used in the training. Then, during the recognition stage, the Gaussian mixture that maximizes the probability of the incoming utterance was found and the speech utterance is warped using the corresponding warping factor.

Considerable improvements of ASR performance due to the use of VTN have been reported on a number of different recognition tasks [29, 74, 115, 87, 82]. In addition, VTN was shown to outperform CMN and gender dependent modeling [74, 87]. The ML-based approach has been shown to be superior to the formant-based approach [115], but it is also more computationally expensive.

3.3.3 Spectral Subtraction

Spectral subtraction is a computationally efficient technique for suppressing additive noise from noisy speech signals. It is used in ASR to reduce the mismatch between noisy speech data and clean speech models. It is assumed that clean speech and noise are uncorrelated, so that the clean speech power spectrum, $S_{ss}(f)$, can be computed by subtracting the noise power spectrum, $S_{nn}(f)$, from the noisy speech power spectrum, $S_{xx}(f)$, i.e.

$$S_{ss}(f) = S_{xx}(f) - S_{nn}(f). \tag{3.10}$$

This is done on the frame-by-frame basis. An estimate of the noise spectrum is obtained during non-speech intervals, by averaging short-time spectra over a number of non-speech frames. Thus, spectral subtraction assumes that the noise is stationary or slowly varying, and that there exists a reliable method for speech endpoint detection.

There are two major problems associated with the spectral subtraction method. First, the subtraction in Equation 3.10 can result in the appearance of negative values in the resulting clean speech spectral estimate. This problem was originally solved by setting the negative values to zero. Second, spectral subtraction results in the appearance of, so called, musical noise. It is due to the noise residual, which is characterized by a large number of random narrow-band spectral peaks. They appear in the signal intervals dominated by noise, as a result of subtracting a smooth noise spectral estimate from a highly varying short-term spectral estimate.

A number of algorithms have been proposed to deal with the above problems [13, 12, 77]. In [13], it was suggested to subtract an overestimate of the noise power spectrum in order to reduce the magnitude of the spectral peaks associated with residual noise. At the same time, a spectral floor was introduced to prevent the resulting spectral components from taking values below a minimum level. This resulted in the following algorithm:

$$D(f) = S_{xx}(f) - \alpha S_{nn}(f) \tag{3.11}$$

$$S_{ss}(f) = \begin{cases} D(f) & \text{if } D(f) > \beta S_{nn}(f) \\ \beta S_{nn}(f) & \text{otherwise} \end{cases} \tag{3.12}$$

where $\alpha \geq 1$ and $0 < \beta \ll 1$. A generalization of the above algorithm was proposed in [77], referred to as nonlinear spectral subtraction. The idea is to substitute the second term in Equation 3.11 by a nonlinear function of the noise power spectrum and local SNR at a given frequency. The nonlinear function is chosen such that more noise is subtracted in low-SNR regions than in high-SNR regions. In that way, noise subtraction and noise masking are combined in the same framework.

The ability of spectral subtraction to improve ASR performance in mismatched environmental conditions has been demonstrated in several studies [77, 93, 21]. Furthermore, it has been shown that spectral subtraction can be efficiently combined with other noise compensation techniques. Finally, nonlinear spectral subtraction has been shown to outperform linear spectral subtraction [77]. However, the main contribution compared to the non-compensated case was achieved by linear spectral subtraction.

3.3.4 Linear Discriminant Analysis

The aim of linear discriminant analysis (LDA) [26, 102] is to find a linear transformation of feature vectors that results in improved class separability. This can be written as

$$\boldsymbol{y} = \boldsymbol{A}^T \boldsymbol{x}, \tag{3.13}$$

where $\boldsymbol{x}$ is a feature vector of length N in the original feature space, $\boldsymbol{y}$ is the transformed feature vector of length M in the new feature space, and $\boldsymbol{A}$ is the transformation matrix having dimension NxM. Choosing $M < N$ leads to a dimensionality reduction, in addition to improved class separability.

The improved class separability is achieved by maximizing the following function

$$J = tr(\boldsymbol{W}^{-1}\boldsymbol{B}), \tag{3.14}$$

where $\boldsymbol{W}$ is the within-class scatter matrix, $\boldsymbol{B}$ is between-class scatter matrix, and $tr(\cdot)$ denotes matrix trace. The matrices $\boldsymbol{W}$ and $\boldsymbol{B}$ are defined as

$$\boldsymbol{W} = \frac{1}{N}\sum_{k=1}^{K}\sum_{n=1}^{n_k}(\boldsymbol{o}_{kn} - \boldsymbol{\mu}_k)(\boldsymbol{o}_{kn} - \boldsymbol{\mu}_k)^T \tag{3.15}$$

$$\boldsymbol{B} = \frac{1}{N}\sum_{k=1}^{K} n_k(\boldsymbol{\mu}_k - \boldsymbol{\mu})(\boldsymbol{\mu}_k - \boldsymbol{\mu})^T, \tag{3.16}$$

where N denotes the total number of training vectors, K is the number of classes, n_k is the number of training vectors in the k-th class, $\boldsymbol{o}_{kn}$ is the n-th vector in the k-th class, $\boldsymbol{\mu}_k$ is the mean of the k-th class given by

$$\boldsymbol{\mu}_k = \frac{1}{n_k}\sum_{n=1}^{n_k}\boldsymbol{o}_{kn}, \tag{3.17}$$

and $\boldsymbol{\mu}$ is the overall mean given by

$$\boldsymbol{\mu} = \frac{1}{N}\sum_{k=1}^{K} n_k\,\boldsymbol{\mu}_k. \tag{3.18}$$

It can be shown that the optimization criterion given by Equation 3.14 leads to a transformation matrix $\boldsymbol{A}$ whose columns are equal to the M eigenvectors of matrix $\boldsymbol{W}^{-1}\boldsymbol{B}$ that correspond to the M largest eigenvalues. Computation of the eigenvectors and eigenvalues of the matrix $\boldsymbol{W}^{-1}\boldsymbol{B}$ can be done using the procedure described in [8].

The procedure for applying LDA in ASR is summarized in the following. First, a set of HMMs trained on the original feature vectors is used to segment the training data into classes. The resulting segmentation is used for computation of the transformation matrix $\boldsymbol{A}$, as described above. The transformation is then applied to all feature vectors in the training database, and a new set of HMMs is trained on the transformed data. Finally, in the recognition stage, each feature vector is multiplied by the transformation matrix $\boldsymbol{A}$ prior to recognition.

The use of LDA has resulted in improved recognition performance, in addition to reduced feature dimensionality on a number of different recognition tasks, both in clean and noisy conditions [60, 59, 53, 52]. Robustness of features obtained using LDA in presence of different types and levels of additive noise has been investigated in [98]. It has been shown that these features retain good separability even at very low SNRs in the case of matched conditions between training and test environments. However, the features exhibited much greater sensitivity to mismatched environmental conditions compared to the standard MFCC features.

Chapter 4

Robust Feature Extraction

Speech features extracted using the conventional methods described in Chapter 2 are very sensitive to changes in the acoustic environments. Thus, they can provide a suitable basis for automatic speech recognition only in the situations where there is a correspondence between environmental conditions in training and test speech.

Humans have an exceptional ability to recognize speech in adverse environmental conditions. Thus, one approach to increasing the robustness in ASR has been to employ a more detailed knowledge on human speech processing in the speech feature extraction. As a result, a number of alternative feature extraction methods have been proposed that model different aspects of human speech perception. While some methods use certain psycho-acoustical concepts, others simulate physiological processes in the human auditory system in great detail.

This chapter starts by presenting several feature extraction methods based on auditory processing that have been shown to increase robustness in ASR. Several of the methods are based on extracting the dominant frequencies information from the speech signal. This information has not been used in the conventional feature extraction methods. Dominant speech frequencies (i.e. formant frequencies) remain practically unchanged in presence of moderate levels of additive background noise, and their use in feature extraction is a possible reason for the greater robustness of auditory-based methods compared to conventional feature extraction methods.

It has recently been shown [84] that subband spectral centroids (SSC) can serve as reasonable estimates of dominant subband frequencies, both on clean speech and in presence of additive noise. The initial studies based on using SSC as additional features in ASR are reviewed in Section 4.7. Finally, a new feature extraction method is proposed in Section 4.8. It is based on combining the subband power information used by conventional methods with the dominant-frequency information provided by SSC. Compared to the earlier proposed robust feature extraction methods, the new method stands out for its simplicity and low computational cost.

4.1 Multiband Speech Recognition

Multiband speech recognition was inspired by Fletcher's study of human speech perception [33], that has recently been reviewed in [6]. Fletcher studied the effects of filtering and noise on human speech recognition accuracy for nonsense syllables, words and sentences. One of the important findings in his work was the fact that humans process the information from different frequency subbands independently. The recombination of the recognition results from the different subbands is done at some higher level in such a way that the global error rate can be approximated with the product of error rates in the different subbands. This is in contrast to the conventional processing in ASR systems, where the information from all frequency bands is combined in a single feature vector before recognition.

Multiband speech recognition was first proposed in [25], and then further formalized in [15]. The idea is to recognize the information from different frequency subbands independently, and then recombine the recognition results in some efficient way to yield the final recognition result. This is illustrated in Figure 4.1.

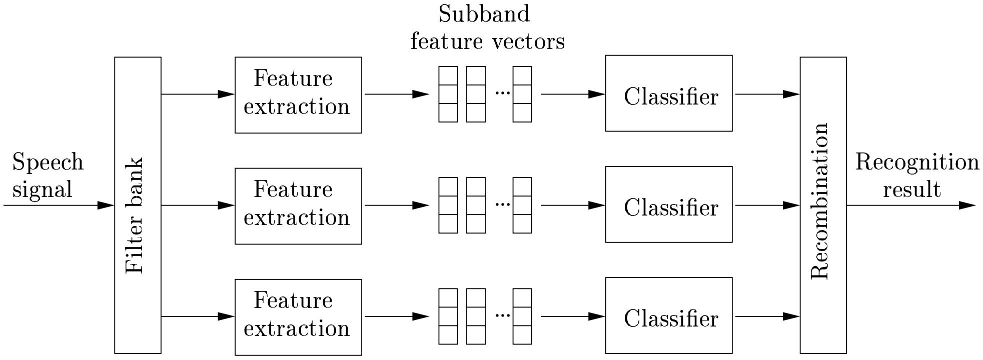

Figure 4.1: Multiband speech recognition

This approach is of special interest for robust speech recognition in narrowband noise. In the conventional, cepstrum-based feature extraction, a noise degradation in a narrow frequency region affects all the components of the feature vector. In the multiband approach, only a few subband recognition results would be affected. During the recombination, it is possible to assign lower weights to the subbands with lower confidence level. Fletcher has shown that signal-to-noise ratio computed over a subband can be used as a measure of confidence of the recognition result in that subband. Furthermore, it is possible to optimize the processing in each subband independently, which opens for use of different features in different subbands. Note, however, that reliable recognition of subband signals is a more difficult task than the recognition based on the original speech signal, due to less information available for discrimination between different speech units.

The multiband approach performs considerably better than the conventional methods in presence of narrow-band noise, while there is no significant difference in the clean speech performance [15, 14, 103]. Furthermore, there has been no documented improvement in the presence of broad-band noise. However, this interesting approach is still in its initial stage, and some promising results might be expected in the future. Many problems still have to be solved, the major one being to find an efficient recombination procedure. This problem was addressed in [57, 104, 16, 17, 110]. Unlike the other feature extraction methods described in this chapter, this approach requires a new structure of the speech recognizer.

An interesting generalization of the approach is the so called multistream speech recognition [28]. Instead of processing the information from different frequency subbands in the different streams, other types of information can be extracted from the speech signal and processed by separate recognizers. For example, the speech signal can be processed with different time resolutions in the different streams [27]. In this way, it is possible to locate rapid spectral changes in one stream, while the information evolving over longer time intervals would be extracted in another stream.

4.2 Perceptual Linear Predictive Analysis

Perceptual linear predictive (PLP) analysis [56, 62, 65, 63] is a variation of the FFT-based critical-band analysis, that differs from the MFCC computation in two main aspects:

1. Psycho-acoustic concepts are more accurately modeled.
2. The perceptually-modified spectrum is fitted by an all-pole model.

The processing steps used in PLP analysis are illustrated in Figure 4.2. As in MFCC computation, the analysis begins with FFT-based spectral computation, followed by critical-band filtering and subband power computation. A minor difference compared to the MFCC method is in the shape of filter

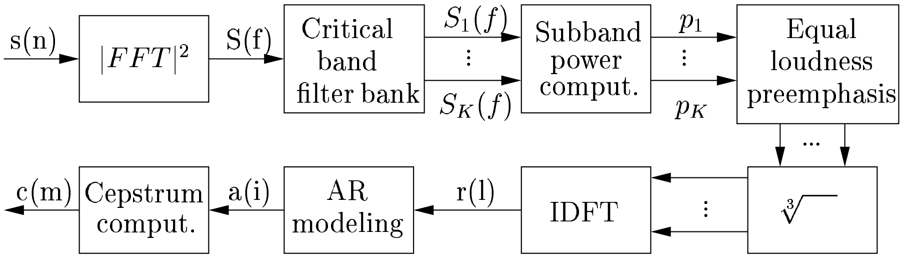

Figure 4.2: PLP method for speech feature extraction

frequency responses, which are chosen to better match the shape of masking curves. In addition, the Bark scale is used instead of the mel scale. The next processing step consists of equal-loudness preemphasis of the resulting spectral estimate. This is done in order to compensate for reduced sensitivity of the ear at low and high ends of the speech frequency range. Next, cubic-root power compression is applied to the modified spectrum in order to approximate intensity-to-loudness conversion. Finally, the perceptually modified spectral estimate is fitted by an all-pole model. This is done by taking the inverse DFT of the spectral estimate in order to obtain a pseudo-autocorrelation function, followed by AR-modeling. This results in a set of LP-parameters $\{a(i)\}_{i=1}^{p}$. Cepstrum coefficients are normally derived from the smoothed spectral estimate using Equation 2.15.

Seventeen critical-band filters with center frequencies linearly spaced on the Bark scale are typically used in PLP computation. Note that the all-pole modeling of the perceptually modified spectrum can no longer be justified by the speech production model in Figure 2.1. Instead, it can be seen as fitting a smooth parametric curve to the perceptually modified spectrum, such that more weight is given to the high-energy parts of the spectrum than to the low-energy parts. The order of the all-pole model is lower than the number of critical-band filters, and is typically between five and eight.

An advantage of PLP compared to the conventional LP analysis is the use of critical-band filtering and perceptual warping prior to all-pole modeling. The filtering provides spectral smoothing, which reduces the influence of the irrelevant spectral fine structure on the all-pole model. The perceptual warping reduces the weight given to the model fit at higher frequencies. This is in accordance with the reduced spectral resolution of human hearing at higher frequencies.

PLP was shown to outperform the conventional LP analysis in the ASR context both in clean speech and in presence of additive noise. The computational complexity of the two methods is approximately the same.

4.3 Joint Synchrony/Mean-Rate Auditory Model

Auditory models represent a class of speech representations that are based on simulating physiological processes in human auditory system in great detail. Since human speech recognition is extremely robust, it is believed that the use of auditory models in ASR would lead to improved recognition performance in adverse conditions. However, since very little is known about human speech feature extraction beyond the auditory nerve level, auditory models include a considerable amount of heuristics. Description of several different auditory models can be found in [51].

The joint synchrony/mean-rate auditory model, proposed by Seneff in [96], consists of three processing stages illustrated in Figure 4.3. In the first stage, the input signal is divided into a number of overlapping frequency bands using a cochlear filter bank. Forty bandpass filters with bandwidths 0.5 Bark were used, covering the frequency range between 130 Hz and 6400 Hz. This stage models the processes on the basilar membrane. The second stage consists of several nonlinearities that model the transformation from the basilar membrane vibrations to the probability of neural firings. The last stage consists of two branches. The first one computes an overall energy measure for each channel by finding the average rate of neural firings. The result is referred to as mean-rate spectrum. The second one measures the extent of dominance of periodicities at subband center frequencies. It is computed using a set of generalized synchrony detectors (GSD), one for each channel.

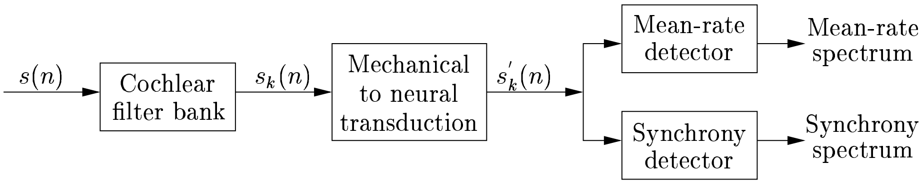

Figure 4.3: Joint synchrony/mean-rate auditory model

A simplified structure of a GSD is illustrated in Figure 4.4. It has two inputs, namely, the output signal from the second stage and its delayed version. The delay is equal to the inverse of the subband center frequency. The output from the GSD is the ratio between the sum and the difference of the two input signals. The speech parameterization that consists of the outputs from all GSDs is referred to as the synchrony spectrum.

An important property of the synchrony spectrum is the fact that it enhances spectral peaks, while suppressing the features associated with glottal excitation. This is explained in the following. If a bandpass signal has a domi-

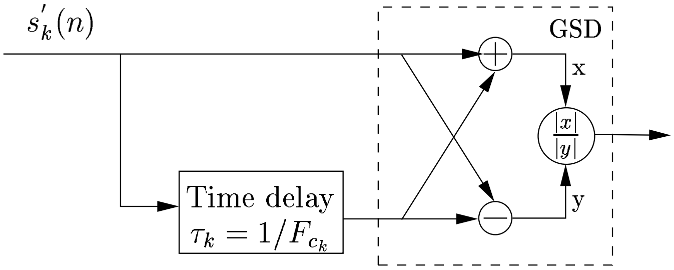

Figure 4.4: Simplified structure of generalized synchrony detector

nant periodicity close to the subband center frequency, the difference between the two input signals to the GSD becomes very small, and the output from the GSD attains a large value. Thus, the output from the channel with the center frequency closest to a spectral peak position has a considerably larger value than the outputs from the neighboring channels.

4.4 Subband-Autocorrelation Analysis

Subband autocorrelation (SBCOR) analysis, proposed by Kajita and Itakura [66, 67], is a simplification of the synchrony spectrum computation described in the previous section. The main idea of measuring the dominance of periodicities at subband center frequencies remains the same, but the particular choice of the dominance measure is different. Furthermore, the computation is simplified by skipping the intermediate stage that simulates the mechanical-to-neural transduction in the inner ear.

SBCOR analysis starts by passing the speech signal through a filter bank of bandpass filters with center frequencies $\{F_{c_k}\}_{k=1}^{K}$. Then, for each subband, the normalized autocorrelation coefficient is computed at the time equal to the inverse of the center frequency, i.e.

$$\rho_k(\tau_k) = \frac{r_k(\tau_k)}{r_k(0)}, \qquad \text{for } \tau_k = 1/F_{c_k}, \tag{4.1}$$

where $r_k(\cdot)$ denotes the autocorrelation function of the k-th subband signal. The resulting speech representation, $\{\rho_k(\tau_k)\}_{k=1}^{K}$, is referred to as SBCOR spectrum. Alternatively, SBCOR analysis can be done in the spectral domain. In this case, an FFT-based spectral estimate is computed first, followed by subband filtering in the spectral domain. Finally, subband autocorrelation functions are computed by taking the inverse DFT of the subband spectra. Note that SBCOR analysis is similar to the conventional subband analysis described in Section 2.4.2. The only difference is in the type of subband features used. While subband power, $r_k(0)$, is used in the conventional methods, the autocorrelation coefficient at time $\tau_k = 1/F_{c_k}$ is used in the SBCOR analysis.

Generally, a spectral peak at frequency F gives rise to peaks in the autocorrelation function at times $\tau = n/F$, where n is an integer value. Consequently, the value of subband autocorrelation coefficient at time $1/F_{c_k}$ indicates the extent of dominance of the subband center frequency in the subband signal. Thus, the SBCOR spectrum provides a good indication of the positions of speech spectral peaks.

Several different filter banks for use with SBCOR analysis were compared in [66]. It was found that a fixed Q filter bank (i.e. fixed ratio between bandwidth and center frequency) with center frequencies uniformly spaced on the Bark scale gave the best results. Furthermore, it is not crucial whether the filter shape is similar to the cochlear filter or not. Both 128 and 16 filters were used in the recognition experiments with SBCOR analysis, but no attempt to optimize the number of filters has been reported.

SBCOR analysis was shown to outperform conventional speech feature extraction methods based on subband power estimates [66]. In addition, robustness of SBCOR spectrum against different types of speech distortion was proven in [67].

4.5 Ensemble Interval Histograms

Ensemble Interval Histogram (EIH) [47, 49] is probably the best known auditory model used in ASR. It is based on temporal information in simulated neural firing patterns, similarly as the synchrony spectrum described in Section 4.3. However, the two auditory models differ in the way neural firing patterns are computed, and in the way temporal information is extracted.

The procedure for EIH computation is illustrated in Figure 4.5. The speech signal, $s(n)$, is passed through a filter bank of K bandpass filters that simulates the frequency response of the basilar membrane. The resulting subband signals, $s_k(n)$, model vibrations at different locations along the basilar membrane. Transduction between basilar membrane vibrations and neural firings is modeled by an array of L level-crossing detectors, where different levels correspond to different neural fiber firing thresholds. Neural firings are simulated as positive-going level crossings. Temporal information is extracted from the neural firing patterns by measuring the inverse interval lengths between successive positive crossings of the same level, i.e.

$$f_{kl}(i) = \frac{1}{n_{kl}(i+1) - n_{kl}(i)}, \tag{4.2}$$

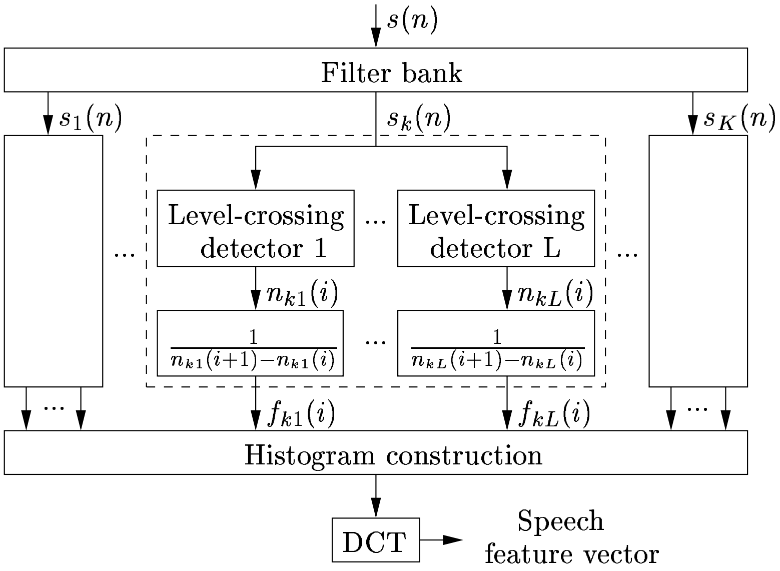

Figure 4.5: EIH method for speech feature extraction

where $n_{kl}(i)$ denotes the location of i-th positive-going crossing of the l-th level in the given frame of k-th subband signal. Next, the frequency axis is divided into a number of histogram bins, R_j, and a histogram of the inverse interval lengths corresponding to all levels of all subband signals is then constructed. The count of j-th histogram bin is computed as

$$\text{count}(j) = \sum_{k=1}^{K} \sum_{l=1}^{L} \sum_{i=1}^{I_{kl}-1} \Psi_j\{f_{kl}(i)\}, \tag{4.3}$$

where

$$\Psi_j\{f_{kl}(i)\} = \begin{cases} 1 & f_{kl}(i) \in R_j \\ 0 & \text{otherwise,} \end{cases} \tag{4.4}$$

and I_{kl} is the total number of positive crossings of l-th level in k-th subband signal. The histogram is usually normalized by the sum of all histogram bin counts, and a DCT is performed for decorrelation purposes.

Note that the inverse level-crossing intervals are closely related to instantaneous dominant subband frequencies. Thus, histogram bins having large counts indicate dominant frequency regions, and EIH can be seen as an alternative spectral representation of speech that emphasizes spectral peaks.

ASR performance of EIH features depends on the choice of the analysis frame lengths, filter bank, number and location of the levels, and particular histogram bin allocation. Analysis frame lengths that are inversely proportional to subband center frequencies have been used in order to ensure that analysis frames for different subband signals incorporate approximately the same number of signal periods (e.g. 10 or 20). The filter bank typically consists of 85 bandpass filters uniformly distributed on a perceptually based scale in the range [0,4000 Hz]. Cochlear filters were originally used, but it was later shown that the particular filter shape was not important for good ASR performance [46]. Five level-crossing detectors have been used, with levels uniformly distributed on the logarithmic scale. However, the optimal choice of the levels is dependent on the signal intensity, and there is no well defined procedure for optimal level determination. It was shown in [70] that the performance of EIH is highly dependent on the choice of number of levels and their particular values. Two different bin allocation schemes were compared in [49]. In the first one, 128 bins were uniformly distributed on the linear frequency scale, while in the second one 32 bins were uniformly distributed on a perceptually-based frequency scale. They led to approximately the same recognition performance.

In addition to the problem with level determination, a major drawback of the EIH method is its high computational cost compared to conventional methods. This is due to the high cost of time-domain filtering, as well as the need for heavy oversampling of the high-frequency subband signals in order to increase accuracy of measured level-crossing locations.

The performance of EIH has been compared to that of conventional methods on several ASR tasks [47, 48, 61, 94]. A general conclusion is that EIH outperforms conventional methods in noisy conditions, while it performs slightly worse in clean conditions. However, EIH performance in noisy conditions is still considerably lower than that of human listeners [49]. EIH has also been shown to outperform both PLP and SBCOR methods on a small vocabulary ASR task in presence of different types additive noise [70]. Finally, the performance of EIH improves considerably when delta and delta-delta parameters are included in the feature vector [94, 70]. However, the improvement is much lower than for conventional methods.

4.6 Zero Crossings with Peak Amplitudes

Kim at. al. [70] derived an analytic expression for the variance of level-crossing interval lengths of a sinusoidal signal in presence of white Gaussian noise. It showed that the variance increases with increased level value. Consequently, lower level values provide more reliable level-crossing interval lengths in presence of noise. On the other hand, experimental results on an ASR task revealed the importance of the intensity information provided by properly chosen higher level values.

Motivated by these results a modification of the EIH method was proposed [70]. The set of level-crossing detectors was exchanged by a single zero-crossing detector, while intensity information was preserved by measuring peak amplitudes between successive zero crossings. Thus, the resulting speech parameterization, referred to as zero crossings with peak amplitudes (ZCPA), provides more reliable interval lengths in noisy conditions without sacrificing the intensity information. In addition, it circumvents the problem of proper level choices of the EIH method, and reduces the computational cost as well as the number of free parameters compared to the EIH method.

4.6.1 Method Description

The procedure for ZCPA computation is illustrated in Figure 4.6. The input speech signal, $s(n)$, is passed through a filter bank consisting of K bandpass filters. Each subband signal, $s_k(n)$, is processed by a zero-crossing detector, in order to determine the positions of all positive-going zero crossing, $z_k(i)$, on the given analysis frame. Then, for each pair of successive zero crossings, $z_k(i)$ and $z_k(i+1)$, the peak signal value, $p_k(i)$, and the inverse zero-crossing

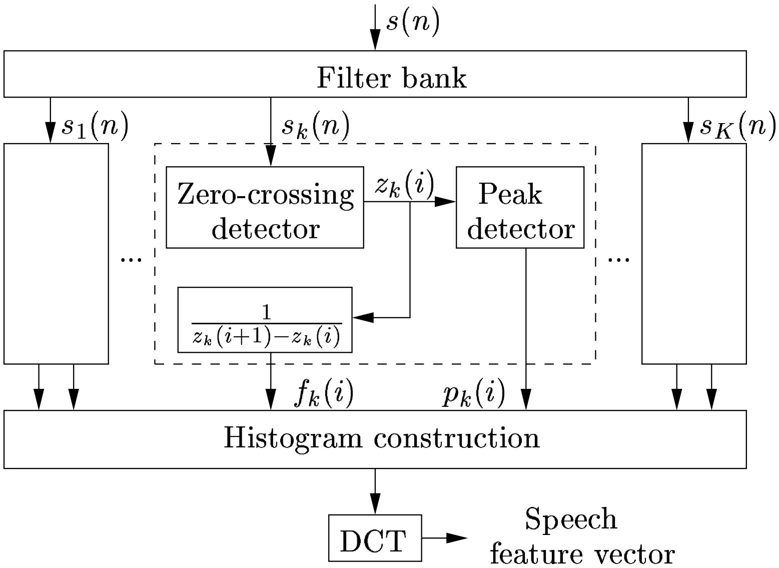

Figure 4.6: ZCPA method for speech feature extraction

interval length, $f_k(i)$, are found by

$$p_k(i) = \max_{z_k(i) \leq n < z_k(i+1)} \{s_k(n)\} \tag{4.5}$$

$$f_k(i) = \frac{1}{z_k(i+1) - z_k(i)}. \tag{4.6}$$

Next, the frequency axis is divided into a number of histogram bins, R_j, and a histogram of the inverse zero-crossing interval lengths is collected over all subband signals. However, instead of increasing the bin counts by one, they are increased by the logarithm of the corresponding signal peak amplitude, $\ln(p_k(i))$. Thus, the count of j-th histogram bin is computed as

$$\text{count}(j) = \sum_{k=1}^{K} \sum_{i=1}^{I_k - 1} \Psi_j\{f_k(i)\}, \tag{4.7}$$

where

$$\Psi_j\{f_k(i)\} = \begin{cases} \ln(p_k(i)) & f_k(i) \in R_j \\ 0 & \text{otherwise,} \end{cases} \tag{4.8}$$

and I_k is the total number of positive zero crossings of k-th subband signal. Finally, DCT is performed on the histogram for decorrelation purposes.

The ZCPA representation depends on the particular choice of the filter bank, analysis frame lengths and histogram bin allocation. Section 5.2.1 presents the results of an experimental study performed in order to optimize the parameter values with respect to the ASR performance.

In an experimental study on a small-vocabulary speaker-independent ASR task [70], ZCPA features were compared to LPCC, MFCC, SBCOR, PLP and EIH features. ZCPA features were shown to greatly outperform all of the other feature types in presence of additive background noise.

4.6.2 Computational Complexity

The computational cost of the ZCPA method is considerably lower than that of the EIH method and other auditory based methods. However, compared to the standard MFCC method, the computational complexity is still prohibitively high. This is due to the use of time-domain filtering, and a need for heavy interpolation of high-frequency subband signals in order to reliably determine zero-crossing positions. The interpolation provides a larger number of points between subsequent zero crossings, and thus better frequency resolution. The computational cost depends mainly on the number and order of subband filters, required interpolation factors, and the order of the interpolation filter. With the parameter choices used in the experimental study described in Section 5.3, the computational cost of the ZCPA method is two orders of magnitude higher than that of the standard MFCC method.

4.6.3 Relationship to Spectral Analysis

The ZCPA feature extraction method was derived from the EIH method which was motivated by physiological processes in the human auditory system. However, from the signal processing point of view, ZCPA histograms can be seen as alternative short-term spectral representations of speech. This is explained in the following.

The dominant frequency principle [69] states that if there is a significantly dominant frequency in the signal spectrum, then the inverse zero-crossing interval lengths tend to take values in the vicinity of the dominant frequency. Thus, the inverse zero-crossing interval lengths, $f_k(i)$, of the k-th subband signal can be seen as estimates of the dominant subband frequency. Furthermore, the peak signal value between subsequent zero crossings, $p_k(i)$, can be seen as a measure of signal power in the subband signal. Consequently, the construction of ZCPA histograms consists of assigning subband power estimates to frequency bins corresponding to dominant subband frequencies. Standard MFCC method, on the other hand, assigns subband power estimates to entire subbands, without taking into account the power distribution within subbands. Thus, the ZCPA representation can be seen as an alternative spectral representation of speech that emphasizes spectral peaks, while deemphasizing the information in spectral valleys, which is usually corrupted by noise.

The major differences between the ZCPA method and the standard MFCC method are summarized in the following:

- ZCPA combines subband power information with dominant subband frequency information, while MFCC uses subband power information alone.

- ZCPA uses instantaneous estimates of subband power and dominant frequency, while MFCC uses power estimates averaged over the entire analysis frame.

- ZCPA features are derived entirely in the time domain, while MFCC features are derived in the frequency domain.

- The computational cost of the ZCPA method is two orders of magnitude higher that that of the MFCC method.

4.7 Subband Spectral Centroids

The presence of additive background noise results in changes of the speech power spectrum. The effect is largest on spectral valleys, while the positions of dominant spectral peaks remain practically unaffected, as long as the noise is added at moderate levels and does not have strong spectral peaks. The auditory models and their simplifications presented earlier in this chapter all utilized the information about dominant frequencies in the speech signal. This might be a major reason for increased robustness of auditory models compared to conventional speech features. Recall that the conventional feature extraction methods do not utilize dominant-frequency information.

Reliable estimation of spectral peak positions is a difficult task, especially in presence of noise. Thus, rather than estimating spectral peak positions directly, Paliwal [84] was concerned with deriving features for use in ASR that would convey the information about spectral peak positions (i.e. dominant signal frequencies). He studied the properties of subband spectral centroids (SSC), that are computed as the first moment of the speech power spectrum over different frequency subbands. If $S(f)$ is the speech power spectrum and $H_k(f)$ is the frequency response of the k-th subband filter, then the centroid of the k-th subband is computed as

$$C_k = \frac{\int_0^{\frac{1}{2}} f H_k(f) S^{\gamma}(f) df}{\int_0^{\frac{1}{2}} H_k(f) S^{\gamma}(f) df}, \tag{4.9}$$

where γ is a constant controlling the dynamic range of the power spectrum, and f is the normalized frequency. Note that SSC depend on the particular filter bank and power spectrum estimate used for their computation. It was shown in [84] that SSC are closely related to spectral peak positions, and their robustness to additive white Gaussian noise was demonstrated. Furthermore, an evaluation on a small-vocabulary isolated-word speaker-dependent ASR task showed that augmenting three SSC to the standard LPCC features improved the recognition performance on clean speech. In this experiment, SSC were computed from the LP-based spectral estimate using filters with overlapping triangular frequency responses uniformly spaced on the linear frequency scale between 0 and 4 kHz. The dynamic range constant γ was set to 0.5. The result indicates that SSC provide useful additional information for speech recognition.

Tsuge at al. [105] tested the effect of augmenting six SSC and their first derivatives to the conventional MFCC representation in presence of additive background noise. The SSC were computed from the FFT-based spectral estimate, using disjoint rectangular filters uniformly distributed on the linear frequency scale between 0 and 8 kHz. The dynamic range constant γ was set to 0.5. The test was performed on a large-vocabulary, continuous-speech, speaker-independent ASR task. Workstation noise was artificially added to the clean speech database. The ASR performance in noise was improved as a result of adding SSC to the feature vector, except at the highest SNR. Furthermore, speaker normalized SSC were introduced, which led to a considerable further improvement of performance at all SNRs. Speaker normalization was done using the linear frequency warping technique described in Section 3.3.2. The warping factor for a given speaker was computed as the ratio between the second formant frequency averaged over all vowels uttered by the given speaker, and the second formant frequency averaged over all vowels and all speakers in the training set.

Another study of the effect of augmenting three SSC to the standard MFCC representation was reported in [4, 23]. The SSC were computed in two different ways, namely, from the FFT-based spectrum and from the ZCPA histograms described in Section 4.6. Overlapping triangular filters were used, with bandwidths corresponding to the possible ranges of the first three speech formants (i.e. 0–1175 Hz, 315–2860 Hz and 1175–4000 Hz). The new features were compared to standard MFCC features on a speaker-independent ASR task, both on clean telephone speech and in presence of artificially added background noise. Two different test sets were used, consisting of 475 and 9329 vocabulary words, respectively. SSC computed from ZCPA histograms improved the ASR performance both in clean and noisy conditions. However, SSC computed from FFT-based spectra improved the ASR performance in clean conditions, but deteriorated the performance in presence of noise. The last result is inconsistent with the results reported by Tsuge [105]. The superior performance of the SSC computed from the ZCPA histograms is not surprising, since ZCPA histograms enhance the locations of spectral peaks. However, this method is much more computationally expensive than that of computing SSC from FFT-based spectra.

4.8 Subband Spectral Centroid Histograms

This section describes a new speech feature extraction method, which combines the subband power information used by conventional methods with the dominant subband frequency information provided by SSC in a simple and computationally efficient way. This is achieved through the construction of subband spectral centroid histograms (SSCH) [36, 37]. There were two main motivations for designing the new method:

1. The robustness of the ZCPA method in presence of additive noise has

indicated the positive effect of integrating dominant frequency information and intensity information into speech features for ASR. However, the high computational cost of the ZCPA method makes this method less attractive in practical applications. Thus, the idea was to develop a method that would utilize the same conceptual information as the ZCPA method, but have a low computational cost similar to that of the conventional feature extraction methods.

2. Subband spectral centroids were shown to provide reasonable estimates of dominant signal frequencies both on clean speech and in presence of additive noise. They can be computed efficiently from any frequency-domain representation of speech. Furthermore, the use of SSC as additional features was shown to have a positive effect on ASR performance. All this made them promising candidates for providing dominant frequency information in the new speech parameterization.

4.8.1 Method Description

The procedure for SSCH computation is illustrated in Figure 4.7, and it is summarized in the following.

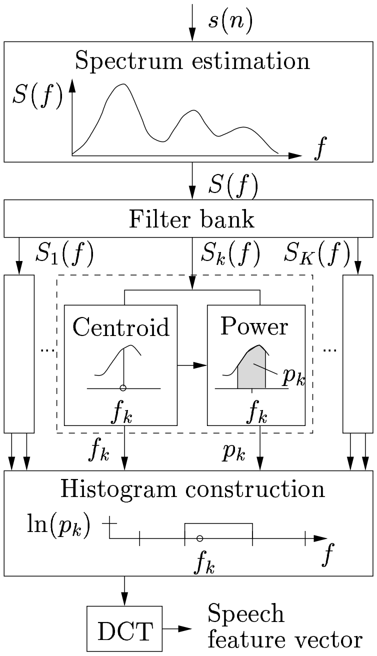

Figure 4.7: SSCH method for speech feature extraction

Power spectrum estimation: First, a short-term power spectrum estimate, $S(f)$, is computed for the given speech frame. Both FFT-based and LP-based spectral estimates can, in principle, be used. However, using FFT-based spectral estimates is simpler and more computationally efficient. In addition, LP-estimates are not reliable in noisy conditions. Thus, SSCH features used in this study were derived from FFT-based spectral estimates.

Centroid computation: Given a filter bank of K subband filters, with the frequency response of k-th filter equal to $H_k(f)$, subband spectral centroids are computed according to Equation 4.9. In practice, only a finite number N of frequency samples is available. Thus, Equation 4.9 is approximated as

$$C_k = \frac{\sum_{i=0}^{N-1} i H_k(i) S^{\gamma}(i)}{\sum_{i=0}^{N-1} H_k(i) S^{\gamma}(i)} \ , \tag{4.10}$$

where $S(i)$ is the power spectrum estimate of speech signal, and γ is the dynamic range constant.

Subband power computation: Subband power estimates are computed by integrating the power spectrum over each subband, i.e.

$$p_k = \sum_{i=0}^{N-1} H_k(i) S(i). \tag{4.11}$$

Instead of integrating over the entire subband, one can alternatively integrate over a smaller frequency range centered around the subband spectral centroid. The latter might provide more robust estimates since the frequency area around the dominant frequency is less influenced by noise than the other parts of the subband. However, smaller integration areas lead to less reliable estimates.

Histogram construction: The speech frequency range is divided into a number of histogram bins, R_j, and a histogram of subband spectral centroids is computed in the following way. For each centroid, C_k, the corresponding histogram bin is found, and its count is increased by the logarithm of the power estimate, p_k, normalized by the subband bandwidth. The normalization is done in order to avoid biasing of the histograms toward higher frequencies due to increased filter bandwidths. Thus, the count of j-th histogram bin is computed as

$$\text{count}(j) = \sum_{k=1}^{K} \Psi_j\{C_k\}, \tag{4.12}$$

where

$$\Psi_j\{C_k\} = \begin{cases} \ln(p_k/N_k) & C_k \in R_j \\ 0 & \text{otherwise,} \end{cases} \tag{4.13}$$

where N_k is the number of frequency samples between lower and upper cut-off frequencies of the k-th bandpass filter.

Decorrelation: Finally, the DCT of the histogram is computed for decorrelation purposes.

4.8.2 Computational Complexity

The SSCH method differs from the MFCC method only in two additional processing steps, namely the centroid computation and the histogram construction. The histogram construction requires only a small number of operations compared to the other processing steps, while the cost of the centroid computation depends on the particular choice of filter bank. If rectangular filter frequency responses are used, the number of operations needed for centroid computation is small compared to the cost of the spectral estimation. In this case, the computational complexity of the SSCH method is mainly given by the FFT order, and is only slightly higher than that of the MFCC method using the same FFT order. However, regardless of the filter bank choice, the computational complexity of SSCH and MFCC methods is of the same order of magnitude.

4.8.3 Relationship to ZCPA

Both SSCH and ZCPA methods combine the dominant subband frequency information and subband power information into speech feature vectors. However, the way this information is estimated from the speech signal is different. The main differences between the two methods are summarized in the following:

- The SSCH method estimates dominant subband frequency information by subband spectral centroids, while the inverse zero-crossing interval lengths are used in the ZCPA method.

- The SSCH method estimates subband power information as signal frame energy normalized by the frame length, while this information is provided by peak signal amplitudes between subsequent zero crossings in the ZCPA method.

- The ZCPA method is based on instantaneous estimates of the dominant frequency and power of subband signals, while SSCH uses estimates averaged over the entire analysis frame.
- The ZCPA method operates entirely in the time domain, while the the SSCH method operates in the frequency domain.
- Computational complexity of the SSCH method is of the same order of magnitude as that of the MFCC method, while the computational complexity of the ZCPA method is two orders of magnitude higher.

4.8.4 Relationship to MFCC

SSCH and MFCC feature extraction methods have several common processing steps, i.e. spectral estimation, subband filtering and subband energy computation. However, the SSCH method incorporates two additional steps, namely, centroid computations and histogram construction.

Similarly as the ZCPA method, the SSCH method can be seen as an alternative way of performing spectral analysis, which emphasizes spectral peaks. While the MFCC method assigns the subband power estimate to the entire frequency subband, in the SSCH method it is assigned to the histogram bin that contains the dominant subband frequency. In this way, the locations of spectral peaks are much better preserved, while the information in spectral valleys is deemphasized. This is advantageous in presence of additive background noise, which has the most serious effect on spectral valleys. However, it is important to remember that SSC are only estimates of spectral peak positions computed from the speech spectra. Thus, they are affected by noise even if the true spectral peaks remain unchanged.

Chapter 5

Experimental Study

This chapter presents the results of an experimental study aimed at evaluating ZCPA and SSCH speech feature extraction methods in the ASR context. The evaluation was performed on two different recognition tasks in various noisy conditions. The main objective of this study was to determine the influence of incorporating dominant-frequency information into speech parameterization on the robustness of ASR systems.

The chapter starts with a description of the ASR tasks and databases used in this study in Section 5.1. Section 5.2 presents the results of an experimental study aimed at optimizing a number of free parameters in ZCPA and SSCH feature extraction methods. In Section 5.3, the performance of ZCPA features is evaluated, while Section 5.4 presents an evaluation of SSC-based features.

5.1 Recognition Tasks and System Design

This section describes two speech recognition tasks used for evaluation of the different speech feature extraction algorithms compared in this study. The first one is a small-vocabulary isolated-word task, while the second one is a medium-vocabulary continuous-speech task. Furthermore, it describes the algorithm for generating noisy speech, together with the different noise sources used in this study. Some statistical considerations regarding the reliability of the recognition results obtained on the two recognition tasks are presented in Appendix A.

5.1.1 Small-Vocabulary Isolated-Word Task

The small-vocabulary isolated-word task used in this study was based on ISOLET Spoken Letter Database [20] down-sampled to 8 kHz. The vocabulary consists of the 26 letters from English alphabet. Two repetitions of each letter spoken in isolation were recorded for each speaker in a quiet room using a noise-canceling microphone. Utterances from 90 speakers (subsets ISOLET-1, ISOLET-2 and ISOLET-3) were used for model training, while utterances from 30 additional speakers (subset ISOLET-5) were used for evaluation. This gives a total of 4680 training utterances and 1560 testing utterances, which corresponds to approximately 48 minutes of training speech and 16 minutes of test speech. In spite of a very small vocabulary, this is not a simple recognition task, since the vocabulary words are very short and highly confusable.

Model training and recognition were performed using hidden Markov model speech recognition toolkit (HTK 3.0) [114]. One hidden Markov model (HMM) was trained for each vocabulary word. Each model consisted of five states and five Gaussian mixture components per state. The models had left-to-right structure with no skip transitions. A variance floor was set to 0.01 times the global variance. Single mixture models were trained first, using HTK tools HInit and HRest. HInit estimates the initial model parameters by iteratively applying the Viterbi algorithm to find the optimal segmentation of the training data. HRest performs several iterations of Baum-Welch re-estimation until a given stop criterion is reached. Next, the number of mixtures was increased by one and model parameters were re-estimated using HRest. This was repeated until the final number of mixtures was reached.

Recognition was done using Viterbi algorithm (HTK tool HVite). ASR performance was measured as a percentage of correctly recognized utterances, i.e. word accuracy (WAC)

$$\mathrm{WAC} = \frac{N_c}{N} \cdot 100\%, \tag{5.1}$$

where N_c is the number of correctly recognized utterances, and N is the total number of utterances.

5.1.2 Medium-Vocabulary Continuous-Speech Task

The medium-vocabulary continuous-speech recognition task used in this study, was based on the speaker-independent part of DARPA Resource Management (RM) database [86], down-sampled to 8 kHz. The vocabulary consists of 991 words needed to make queries about ships, ports, etc., along with commands to control a graphics display system. The data was collected in a quiet room using a close-talking noise-canceling microphone. The training set consists of 3990 sentences uttered by 109 speakers. Evaluation was performed on the February '89 test set, that consists of 300 sentences spoken by 10 speakers different from the ones used in training. There is a total of 34722 words in the training database, and 2561 words in the testing database, which corresponds to approximately 228 minutes of training speech and 16 minutes of test speech.

Model training and recognition were performed by closely following the RM Recipe supplied with the HTK distribution. A set of initial three-state single-mixture monophone models, trained on the phonetically balanced TIMIT database [71], is supplied with the HTK distribution, as well as a pronunciation dictionary and a word-pair language model for the RM task. The initial models are based on the standard MFCC parameterization with augmented frame energy, delta and delta-delta parameters. Starting from the initial models, a set of six-mixture tied-state cross-word triphone models was trained following the RM Recipe (steps 1, 7 and 9). State tying was performed using decision tree clustering.

Given the set of the well-trained models, a new set of models based on a different parameterization was obtained using single-pass retraining (option -r of HTK tool HERest) [114], followed by three iterations of embedded Baum-Welch algorithm (HTK tool HERest). This was repeated for each speech parameterization. It was assumed that the models obtained by single-pass retraining would give similar ASR performance compared to the models obtained by repeating the entire training procedure for each speech parameterization.

Testing was performed according to RM Recipe, using Viterbi algorithm with the word-pair language model supplied with the HTK distribution. Recognized utterances were aligned with corresponding reference utterances using a dynamic programming algorithm (HTK tool HResult). The ASR performance was measured in terms of word accuracy (WAC) defined as

$$\mathrm{WAC} = \frac{N_c - I}{N} \cdot 100\% = \frac{N - S - D - I}{N} \cdot 100\%, \tag{5.2}$$

where N is the total number of words in the test database, N_c is the number of correctly recognized words, I is the number of word insertions, D is the number of word deletions, and S is the number of word substitutions.

Word accuracy, given by Equation 5.2, is dependent on the relationship between word-internal transition probabilities and between-word transition probabilities. Word-internal transition probabilities are given by the HMM state transition matrices, while between-word transition probabilities are given by the language model. Their relationship is adjusted by modifying the between-word transition probabilities in the following way

$$\log(\hat{P}_{ij}) = p + s\,\log(P_{ij}), \tag{5.3}$$

where $P_{ij} = P(W_j|W_i)$ is the probability for transition from word W_i to word W_j given by the language model, p is the word insertion penalty, and s is the language model scale factor. Parameters s and p can be arbitrarily chosen (options -s and -p in the HTK tool HVite). They regulate the number of deletion and insertion errors, and have a significant effect on the recognition performance. In this study, each feature extraction method was evaluated for several combination of s and p values, and the combination that gave the best overall performance averaged over all background conditions was chosen for each feature extraction method.

5.1.3 Creating Noisy Speech

For the purpose of evaluating the robustness of different speech features in presence of background noise, four different noise types were added to the test data at several different SNRs, namely, white Gaussian noise, factory noise, car noise and background speech. White Gaussian noise was generated using the pseudo-random noise generator provided by the MATLAB program package, while the other three noise types were taken from the NOISEX database [107], where they are referred to as factory1, volvo and babble noise, respectively. The factory noise was recorded near plate-cutting and electrical welding equipment. The car noise was recorded at 120 km/h on an asphalt road in rainy conditions. Finally, the source of the babble noise is 100 people speaking in a canteen. The main observations obtained by examining the spectral characteristics of the different noise types are given in the following:

- White noise has nearly stationary characteristics and essentially flat spectrum.
- Car noise has a very strong spectral peak in the low-frequency region up to approximately 50 Hz, and very little power content in the region above 200 Hz. It has a nearly stationary characteristic.
- Factory noise is highly unstationary. Intervals with relatively flat spectral characteristic alternate with those characterized by strong spectral peaks at different positions, but mainly in the frequency region up to 1 kHz.
- Characteristics of the babble noise also vary with time. However, spectral differences are not as large as for factory noise. Babble noise is characterized by existence of speech-like spectral peaks, and the characteristic speech spectral tilt.

Noisy speech was generated in the following way. For each speech file in the evaluation database, a noise segment of length equal to the length of the speech file was randomly extracted, multiplied by a gain factor, g, and added to the speech file. The gain factor was computed in accordance with the required SNR defined as

$$SNR\,[dB] = 10\log_{10}\left(\frac{p_s^{max}}{g^2\,p_n}\right) \tag{5.4}$$

where p_s^{max} is the maximal frame power of the given speech file, and p_n is the noise power estimated over the entire noise segment. This way of computation makes the SNR independent of the phonetical content of the speech utterance and the length of silence intervals surrounding the speech utterance.

5.2 Optimizing Parameter Values

All feature extraction methods depend on many free parameters. Conventional methods have been evaluated on a large number of different ASR tasks. This resulted in the establishment of standard parameter values for those methods. This section discusses the choice of free parameters in ZCPA and SSCH feature extraction methods. Some of the parameters were set explicitly to the standard values used in conventional feature extraction methods. Other parameters, which were considered to be of particular importance for the ASR performance, were optimized on the small-vocabulary isolated-word task described in Section 5.1.1, both on clean speech and in presence of white Gaussian noise added at various SNRs.

5.2.1 Zero Crossings with Peak Amplitudes

The computation of ZCPA features depends on the choice of the analysis-frame lengths, subband filter bank, and histogram bin allocation. The choice of those parameters is discussed in the following.

5.2.1.1 Analysis-Frame Lengths

In the filter bank approach to spectral analysis it is possible to choose different analysis-frame lengths for different subband signals. This can be advantageous, since shorter frame lengths can be used for rapidly varying high-frequency signals, while longer frame lengths can be used for slowly varying low-frequency signals. Thus, better time resolution can be achieved at higher frequencies, and better frequency resolution at lower frequencies. This is in agreement with speech processing in the human auditory system.

Three different methods for allocating analysis-frame lengths to subband signals were compared for use with the ZCPA feature extraction method. They are described in the following.

Method 1: Analysis-frame lengths of subband signals were chosen in such a way that each frame incorporates approximately the same number of signal periods. Since the number of subband signal periods corresponds roughly to the number of intervals between positive zero crossings, all subband signals contribute to approximately the same number of points in ZCPA histograms. To achieve this, the frame length of k-th subband signal (given in ms) was computed as C/F_{c_k}, where F_{c_k} is the center frequency of the corresponding bandpass filter given in kHz, and C is a constant.

This method was used in the previous studies of EIH and ZCPA methods. For example, $C = 10$ was used in [70]. With filter center frequencies spanning from 200 Hz to 3400 Hz, this choice of parameter C gives frame lengths between 50 ms and 3 ms. Note that, this results in analysis frames at high frequencies being shorter than the average pitch period. This might lead to unreliable frequency estimates at those frequencies. The problem can be solved by increasing the value of the parameter C. However, this gives very long frames at low frequencies, that can cause too low time resolution and obstruction of the stationarity assumption.

Four different values of parameter C were tested in this study: 10, 20, 30 and 40. The results are presented in the first part of Table 5.1.

Method 2: The goal of this method was to increase frame lengths at higher frequencies compared to the previous method without making the frames at low frequencies unreasonably long. Analysis-frame length (given in ms) of k-th subband signal was computed as $C/\sqrt{F_{c_k}}$, where F_{c_k} is the center frequency of the corresponding bandpass filter given in kHz, and C is a constant. Four different values of constant C were tested: 20, 40, 60 and 80. The results are presented in the second part of Table 5.1.

Method 3: In this method, equal analysis-frame lengths were used for all subband signals. Five different choices of frame lengths were tested: 25 ms, 35 ms, 50 ms, 75 ms and 100 ms. The results are presented in the third part of Table 5.1.

In the first method, frame lengths were chosen such that each subband signal contributes to approximately the same number of histogram points. However, in the case of equal frame lengths, high-frequency subband signals contribute to more histogram points than low-frequency subband signals, since they cross the zero axis more often. Thus, in order to avoid histogram biasing toward higher frequencies, histograms were normalized with respect to frequency. The normalization was achieved by dividing each histogram entry by $f_k(i)$, where

Table 5.1: ASR performance of ZCPA features for different choices of analysis frame lengths. F_{c_k} denotes the center frequency of k-th subband given in kHz. The evaluation was done on the ISOLET database, both on clean speech and in presence of additive white Gaussian noise at various SNRs.

Method	Frame lengths [ms]	Word accuracy [%]				
		Clean speech	SNR [dB]			
			25	20	15	10
$10/F_{c_k}$	3–50	78.33	70.51	61.79	55.58	40.64
$20/F_{c_k}$	6–100	81.15	73.78	67.63	61.35	48.01
$30/F_{c_k}$	9–150	80.51	75.51	72.56	65.00	52.24
$40/F_{c_k}$	12–200	79.10	75.90	72.37	65.32	51.79
$20/\sqrt{F_{c_k}}$	11–45	81.09	73.78	68.21	57.82	44.68
$40/\sqrt{F_{c_k}}$	22–89	82.88	77.76	72.69	65.06	51.47
$60/\sqrt{F_{c_k}}$	33–134	82.24	78.08	74.10	68.14	54.74
$80/\sqrt{F_{c_k}}$	43–179	81.22	78.21	74.49	68.08	54.29
equal	25	80.83	71.92	64.87	55.83	41.22
equal	35	81.22	73.78	68.46	59.23	44.74
equal	50	81.35	75.38	71.15	62.12	49.10
equal	75	80.45	75.90	72.37	65.13	52.82
equal	100	80.45	76.86	72.88	66.79	55.58

$f_k(i)$ is given by Equation 4.6. Similarly, when frame lengths were chosen to be inversely proportional to the square root of subband frequencies, the normalization was achieved by dividing each histogram entry by $\sqrt{f_k(i)}$.

In all of the above experiments, the filter bank consisted of 16 filters with bandwidth equal to 2 Bark and center frequencies uniformly spaced on the Bark scale between 200 Hz and 3400 Hz. The histogram consisted of 60 bins uniformly distributed on the Bark scale. The reason for this choice of parameters will become apparent in the next section.

It can be seen from Table 5.1 that the choice of analysis-frame lengths had a large influence on the ASR performance of ZCPA features, especially in noisy conditions. Major observations are summarized in the following:

- Increased analysis-frame lengths, especially in low-frequency subbands, led to a considerable performance improvement in presence of noise, while this did not have large influence on the clean speech performance. The increase in performance is probably due to the fact that larger number of histogram points led to greater emphasis of spectral peaks. This had much larger positive effect on the recognition performance in noisy conditions compared to the negative effect due to violating the stationarity assumption. Another way of emphasizing spectral peaks in the ZCPA method is to increase the weight given to each histogram entry. For example, instead of increasing histogram bin counts by $\ln(p_k)$, they could be increased by $C\ln(p_k)$, where C is a constant. This might give similar performance improvement as increasing the frame lengths C times. However, this investigation is left for future studies.

- Analysis-frame lengths at high frequencies should be carefully chosen in order to obtain optimal performance. If they are too long, the time resolution at high frequencies will be too low, if they are too short they will produce unreliable frequency estimates. Using too long frame lengths at high frequencies had a small negative effect only on the clean speech performance, while using too short frame lengths had a considerable negative effect on the performance in presence of additive white noise.

- The use of frequency-dependent analysis-frame lengths might be advantageous in order to obtain a proper balance between frequency lengths in low-frequency and high-frequency subbands. However, in this study, the performance obtained using sufficiently large equal analysis-frames lengths in all subbands was not significantly lower.

Frame lengths equal to $60/\sqrt{F_{c_k}}$ were used in the comparative study described in Section 5.3.

5.2.1.2 Filter-Bank Design and Histogram Construction

In the development of both ZCPA and EIH methods, a lot of attention was initially paid to the design of filter banks that simulate cochlear filter responses [46, 70]. However, both studies later concluded that simulating cochlear filter shapes was not important for good ASR performance. A comparative study presented in [70] showed that ZCPA features based on FIR filters designed by the windowing method, consistently outperformed the features based on carefully designed cochlear filters. Thus, a similar FIR filter bank was used in this study.

In the windowing method, the impulse response of an FIR filter is designed by multiplying the impulse response of an ideal prototype filter by a window function. The order of the FIR filter is determined by the length of the window. In this study, FIR filters of order 61 were designed using Hamming window. The filters were uniformly spaced on the Bark scale.

Filter bandwidths should ideally be chosen such that each subband contains exactly one dominant spectral peak. In this case, the inverse zero-crossing interval lengths serve as good estimates of spectral peak locations. Too small filter bandwidths result in a number of subbands that do not contain any dominant spectral peak. Thus, ZCPA histograms enhance spurious spectral peaks in those subbands, and are sensitive to random variations in speech spectrum. On the other hand, if filter bandwidths are chosen to be too large, some subbands incorporate more than one dominant frequency. In such situations, the inverse zero-crossing interval lengths are not effective in localizing dominant subband frequencies.

Frequency resolution of the ZCPA parameterization is given by histogram bin widths. In order to accurately locate dominant subband frequencies, bin widths should be small compared to subband bandwidths. On the other hand, if histogram bin widths are too small, ZCPA features can become sensitive to random variations in spectral peak positions. In this study, histogram bins having equal lengths on the Bark scale were used. This provides better frequency resolution at low frequencies than at high frequencies, which is in agreement with human speech perception.

Several experiments were performed in order to determine suitable values for filter bandwidths and the number of histogram bins. The results are shown in Table 5.2. The main observations are listed in the following:

Table 5.2: ASR performance of ZCPA features for different choices of filter bandwidths and number of histogram bins. The evaluation was done on ISOLET database both on clean speech and in presence of additive white Gaussian noise at different SNRs.

Filter bandwidth [Bark]	Number of bins	Word accuracy [%]				
		Clean speech	SNR [dB]			
			25	20	15	10
1	30	75.51	69.81	66.99	60.32	49.36
1	60	74.74	69.49	67.69	60.38	48.08
2	30	81.15	76.15	70.96	63.85	50.45
2	45	80.71	75.13	71.28	63.40	52.18
2	60	80.51	75.51	72.56	65.00	52.24
2	75	79.81	74.68	70.77	63.65	51.28
3	20	82.63	76.79	71.28	61.92	45.00
3	30	82.50	77.37	72.69	62.63	47.37
3	40	81.99	77.18	71.35	62.50	47.88

- The choice of filter bandwidths had a significant effect on the ASR performance of the ZCPA method. The use of filter bandwidths equal to 2–3 Bark gave significantly better performance than the use of narrow filter bandwidths equal to 1 Bark. The best performance at low SNRs was achieved using filter bandwidths equal to 2 Bark. This value was used in the rest of this study.

- The particular choice of number of histogram bins did not have a significant influence on the ASR performance. In the rest of this study, the number of bins was set to 60.

Note that filter bandwidths in Table 5.2 are given in terms of the bandwidths of corresponding ideal prototype filters, rather than 3 dB bandwidths. The corresponding 3 dB bandwidths can be up to 50% larger in the low-frequency subbands. In the above experiments, the number of filters was 16, while frame lengths were equal to $30/F_{c_k}$.

In the initial study of ZCPA [70], the number of filters in FIR filter bank was set to 16 in order to achieve high computational efficiency. No attempt to optimize the number of filters has been reported. We have argued in Section 2.2.2.2 that the basilar membrane can be modeled by 24 bandpass filters uniformly distributed on the Bark scale. The entire length of the basilar membrane corresponds to the whole speech frequency range. When speech bandwidth is limited to 4 kHz, standard MFCC parameterization [22] usually uses 20 subband filters. In order to find out if the use of larger number of subband filters could be beneficial for ZCPA parameterization, the performance of ZCPA features based on 20 subband filters was evaluated. The results are shown in Table 5.3 for two different choices of analysis-frame lengths. It can

Table 5.3: ASR performance of ZCPA features based on different number of subband filters. The evaluation was done on ISOLET database, both on clean speech and in presence of additive white Gaussian noise at different SNRs.

Number of filters	Frame lengths	Word accuracy [%]				
		Clean speech	SNR [dB]			
			25	20	15	10
16	$30/F_{c_k}$	80.51	75.51	72.56	65.00	52.24
20	$30/F_{c_k}$	81.35	75.96	72.55	64.29	51.09
16	$60/\sqrt{F_{c_k}}$	82.24	78.08	74.10	68.14	54.74
20	$60/\sqrt{F_{c_k}}$	83.40	78.21	74.81	66.92	52.05

be seen that the increased number of filters led to some performance improvement at high SNRs, while it caused some performance reduction at low SNRs. However, the differences were not statistically significant.

5.2.2 Subband Spectral Centroid Histograms

The computation of SSCH features depends on the particular choice of analysis-frame lengths, spectral estimate, spectral dynamic range parameter, filter bank parameters, and histogram bin allocation. The choice of those parameters is discussed in the following.

5.2.2.1 Analysis-Frame Lengths

The analysis-frame lengths were set to 25 ms, with 10 ms frame shift between successive frames. For speech sampling frequency equal to 8 kHz, this resulted in 200 speech samples per frame, and 80 samples frame shift. No attempt has been made to optimize the analysis-frame lengths.

5.2.2.2 Power Spectrum Estimation

FFT-based power spectrum estimates were used as basis for centroid computation in this study. The advantage of FFT-based power spectrum estimates over LP-based estimates is in their superior robustness against noise and the simplicity with which centroid computation can be implemented.

Each analysis frame was passed through a first-order preemphasis filter with filter coefficient 0.97, followed by a Hamming window. Next, signal frames were padded with 312 zeros, and FFT of order 512 was performed. Finally, the power spectrum estimates were obtained by squaring the magnitudes of the resulting FFT coefficients.

5.2.2.3 Spectral Dynamic Range

The dynamic range of the power spectrum used in the SSC computation is controlled by the parameter γ in Equation 4.10. If γ is too small (near 0), SSC would approach the centers of their subbands, and thus contain no information. If it is too large (near ∞), SSC would correspond to the locations of the subband peak values of the FFT-based power spectrum, and would thus be noisy estimates. In the previous studies of SSC, described in Section 4.7, the dynamic-range parameter was set to 0.5. No attempt to optimize its value has been reported.

Table 5.4 shows the recognition performance of the SSCH method for different values of parameter γ, evaluated on the ISOLET database both on

Table 5.4: ASR performance of SSCH features for different values of the dynamic range parameter γ. The evaluation was done on the ISOLET database, both on clean speech and in presence of white Gaussian noise at different SNRs.

γ	Word accuracy [%]				
	clean	SNR [dB]			
	speech	25	20	15	10
0.5	86.54	77.88	70.83	58.08	33.97
1	86.15	79.74	73.40	59.87	42.24
2	85.51	76.99	71.51	60.77	44.62
4	83.65	75.32	71.41	60.51	44.62

clean speech and in presence of white Gaussian noise at various SNRs. At high SNRs, the best results were achieved using $\gamma = 0.5$, while the best performance at low SNRs was obtained using larger values of γ. These results are reasonable, since increased γ makes spectral peaks more prominent, and thus reduces the effect of additive noise. In the rest of this study, $\gamma = 1$ was used.

The results shown in Table 5.4 were obtained using a subband filter bank consisting of 48 filters with rectangular frequency response. Filter center frequencies were uniformly distributed on the Bark scale between 100 Hz and 3800 Hz, and filter bandwidths were equal to 3 Bark. The histograms consisted of 38 frequency bins uniformly distributed on the Bark scale between 100 and 3800 Hz. The reason for this choice of parameters will become apparent in the next section. Finally, power estimates were computed over entire subbands.

5.2.2.4 Filter-Bank Design and Histogram Construction

The discussion regarding the choice of the filter bank and histogram bin allocation is, in many aspects, similar to the one for the ZCPA method given in Section 5.2.1.2.

The filter bank used for deriving SSCH features in this study consisted of highly overlapping filters with rectangular frequency responses, and center frequencies uniformly distributed on the Bark scale between 100 Hz and 3800 Hz. The rectangular frequency responses were chosen since any other shape, such as triangular, would favor some frequencies within the subband more than the others, and thus give a biased SSC estimate.

Filter bandwidths should ideally be chosen such that each subband contains exactly one dominant spectral peak. In this case, SSC would serve as good estimates of spectral peak positions. Too small filter bandwidths would result in a number of subbands that do not contain any dominant spectral peak. Centroids of such subbands would be sensitive to random variabilities in speech. On the other hand, if filter bandwidths stretch over several dominant spectral peaks, SSC will no longer represent reasonable estimates of subband peak locations.

If each frequency subband corresponds to a single histogram bin, then SSC do not provide any useful information, and SSCH features reduce to standard MFCC features. In order to capture the information about subband power distribution, the ratio between filter bandwidths and histogram bin widths should be chosen to be sufficiently large. For given filter bandwidths, this is achieved by increasing the total number of frequency bins. Histogram bins should be sufficiently small to provide a good frequency resolution, but not too small to make the resulting speech parameterization sensitive to small fluctuations in spectral peak positions (e.g. due to speaker differences). In this study, histogram bins having equal lengths on the Bark scale were used. This provides better frequency resolution in low-frequency subbands than in high-frequency subbands, which is in agreement with the processing in the human auditory system. Frequencies below 100 Hz and above 3800 Hz were held outside the histogram range, since they were covered by smaller number of filters than the remaining speech frequency range. Thus, the histogram representation in those frequency ranges would be unreliable.

A series of recognition experiments was performed in order to optimize filter bandwidths, the number of histogram bins, and the number of filters in the filter bank. The experiments were performed on the ISOLET database, both on clean speech and in presence of additive white Gaussian noise at different SNRs. The results are presented in Table 5.5. The main observations are summarized in the following:

- The choice of filter bandwidths had a significant effect on the recognition performance, especially at low SNRs. The best results in presence of noise were achieved using filter bandwidths equal to 3 Bark (302 Hz–1927 Hz), while the performance on clean speech was best for filter bandwidths equal to 1 Bark (101 Hz–642 Hz).

- The number of histogram bins had to be chosen large enough to provide good frequency resolution, but the recognition performance was not very sensitive to the particular choice of the number of bins. The values between 30 and 60 gave good results in all test conditions.

- The recognition performance was insensitive to the particular choice of the number of filters.

Table 5.5: ASR performance of SSCH features for different choices of filter bank parameters and histogram bin allocation. The evaluation was done on the ISOLET database, both on clean speech and in presence of additive white Gaussian noise at different SNRs.

Filter			Word accuracy [%]				
bw	$\frac{\text{Filter bw}}{\text{Bin width}}$	Number of	Clean	SNR [dB]			
[Bark]		filters/bins	speech	25	20	15	10
1	4	48/57	88.46	78.08	69.36	55.96	30.83
1	4	143/57	87.95	78.27	70.19	55.38	28.25
1	6	143/86	88.44	77.56	67.50	53.85	27.76
2	2	72/14	84.29	73.53	67.44	54.81	35.90
2	4	72/29	86.28	77.24	70.90	59.55	38.33
2	6	48/43	87.18	78.91	71.67	59.17	38.97
2	6	72/43	86.86	79.17	72.50	58.65	38.21
2	8	72/57	86.47	78.46	71.22	58.85	37.31
3	4	48/19	83.65	76.86	69.74	58.40	40.45
3	6	24/29	85.38	78.65	73.40	60.64	43.97
3	6	48/29	86.47	78.46	72.44	59.81	41.67
3	6	72/29	85.96	78.72	72.37	59.23	41.41
3	8	24/38	86.41	79.87	72.82	59.87	43.21
3	8	48/38	86.15	79.74	73.40	59.87	42.24
4	6	36/21	84.29	75.45	66.60	54.81	33.21
4	8	36/29	86.60	78.08	69.10	56.79	36.22
4	10	36/36	85.19	77.69	68.85	56.22	36.03

In the rest of this study, SSCH features were computed using 48 subband filters with bandwidths equal to 3 Bark. The number of histogram bins was set to 38. The relatively large number of filters was used due to the initial belief that a large number of histogram points would be important for obtaining reliable histograms. However, later experiments have shown that no reduction in performance was observed when the number of filters was reduced to 24.

5.2.2.5 Subband Power Computation

Table 5.6 presents the recognition performance of SSCH features obtained using the two methods of subband power computation described in Section 4.8.1. The evaluation was done both on clean speech and in presence of white Gaus-

Table 5.6: ASR performance of SSCH features for two different methods of subband power computation. The evaluation was done on the ISOLET database, both on clean speech and in presence of additive white Gaussian noise at different SNRs.

Power computation range	Word accuracy [%]				
	Clean speech	SNR [dB]			
		25	20	15	10
whole band	86.15	79.74	73.40	59.87	42.24
1 critical bw	86.35	80.45	74.10	61.60	42.50

sian noise at several SNRs. The first row shows recognition accuracy for the case when subband power estimates were computed over the entire subband, while the second row is for the case when subband power estimates were computed over one critical bandwidth (i.e. 1 Bark) centered around the subband centroid. It can be seen that slightly better results were obtained when subband powers were computed over the narrow bands centered around the centroids, but the difference is not statistically significant.

In the above experiments, filter bandwidths were equal to 3 Bark, and the number of filters and histogram bins was 48 and 38, respectively. In the rest of this study, SSCH features were derived using the subband power estimates computed over one critical bandwidth centered around the corresponding centroid.

5.2.3 Summary of the Main Results

This section presented the results of an experimental study performed in order to optimize a number of free parameters involved in ZCPA and SSCH feature extraction methods. The evaluation was done on the ISOLET database, both on the clean speech and in presence of additive white Gaussian noise. The main results are summarized in the following:

- The ASR performance of ZCPA features in presence of noise was largely increased by using relatively long analysis frames, especially in low-frequency subbands.

- Increase of dynamic spectral range had a positive effect on the ASR performance of SSCH features in presence of noise.

- The choice of filter bandwidths had a significant influence on the ASR performance of both ZCPA and SSCH features.

- The ASR performance of neither ZCPA nor SSCH features was sensitive to the particular choice of the number of histogram bins.
- The ASR performance of neither of the two feature types was sensitive to the particular choice of the number of subband filters.

5.3 Evaluation of the ZCPA Method

This section presents an experimental study performed in order to evaluate the performance of ZCPA features on the two recognition tasks described in Section 5.1 in various background conditions. First, the performance of ZCPA features was compared to that of standard MFCC features in order to verify earlier results, which showed superior performance of ZCPA features in presence of additive noise [70]. Then, three main differences between the two methods were carefully examined in order to determine their relative contribution to the difference in the ASR performance. Finally, the effect of using subband power information in the ZCPA method was investigated.

ZCPA features used in this study differ from the standard MFCC features in three main aspects: they are based on different subband filter banks, they are derived in the time domain rather than frequency domain, and they combine dominant subband frequency information with subband power information, rather than using subband power information alone.

In order to determine the relative contribution of each of the three aspects to the difference in recognition performance between ZCPA and MFCC features, two intermediate feature extraction methods were evaluated. The first one differed from the MFCC method only in the subband filter bank, which was chosen to closely correspond to the filter bank used in the ZCPA method. The resulting speech features are referred to as frequency-domain derived Bark-frequency cepstral coefficients (BFCCF). The second one consisted of deriving cepstral coefficients from the subband power estimates computed in the time domain. The subband filter bank was identical to the one used in the ZCPA method. The resulting features are referred to as time-domain derived Bark-frequency cepstral coefficients (BFCCT).

Then, the effect of using different filter banks in ZCPA and MFCC methods was determined by comparing the performance of BFCCF and MFCC features. Furthermore, the effect of using time-domain processing instead of frequency-domain processing was determined by comparing the performance of BFCCT and BFCCF features. Finally, the effect of incorporating dominant subband frequency information into speech features was determined by comparing ZCPA and BFCCT features.

In order to determine the effect of using subband power information in the ZCPA method, the way of histogram construction was changed. Instead of increasing histogram bin counts by the logarithm of the corresponding peak amplitude, they were increased by one. In this way, subband power information was not used explicitly in the histogram construction. The resulting speech features are referred to as zero-crossing (ZC) features.

Section 5.3.1 describes the implementational details for all the compared feature extraction methods. In Section 5.3.2, the experimental results are presented, followed by a detailed discussion. Finally, a summary of the main results is given in Section 5.3.3.

5.3.1 Implementational Details

General description of all the compared feature extraction methods was given in Chapters 2 and 4. This section summarizes the implementational details used in the experimental study.

5.3.1.1 Mel-Frequency Cepstral Coefficients

The computation of the MFCC features used in this study was done using the HTK tool HCopy. Standard values of the free parameters were used. They are summarized in the following.

The analysis frames were 25 ms long with 10 ms frame shift between successive frames. For speech sampling frequency equal to 8 kHz, this results in 200 speech samples per frame, and 80 samples frame shift. Each frame was passed through a first-order preemphasis filter with filter coefficient 0.97, followed by Hamming window. Each frame was then padded with 56 zeros, followed by a 256-order FFT computation. Note that the magnitudes of the FFT coefficients were used in the subsequent filtering stage, instead of the squared magnitudes shown in Figure 2.2 (this is the default setting in the HCopy tool).

The filter bank consisted of 20 bandpass filters, each having a triangular frequency response, with 50% overlap between neighbouring filters. Filter center frequencies were uniformly distributed on the mel scale between 66 Hz and 3592 Hz (102–2044 mel) with 3 dB bandwidths equal to 102 mel. This corresponds to linear frequency bandwidths between 70 Hz and 389 Hz. The frequency response of the filter bank is shown in the upper part of Figure 5.1.

Twelve cepstral coefficients were derived from the subband power estimates, together with their first and second derivatives, resulting in 36-dimensional feature vectors. The word insertion penalty, p, and the language model scale factor, s, used with the RM recognition task were both equal to 10.

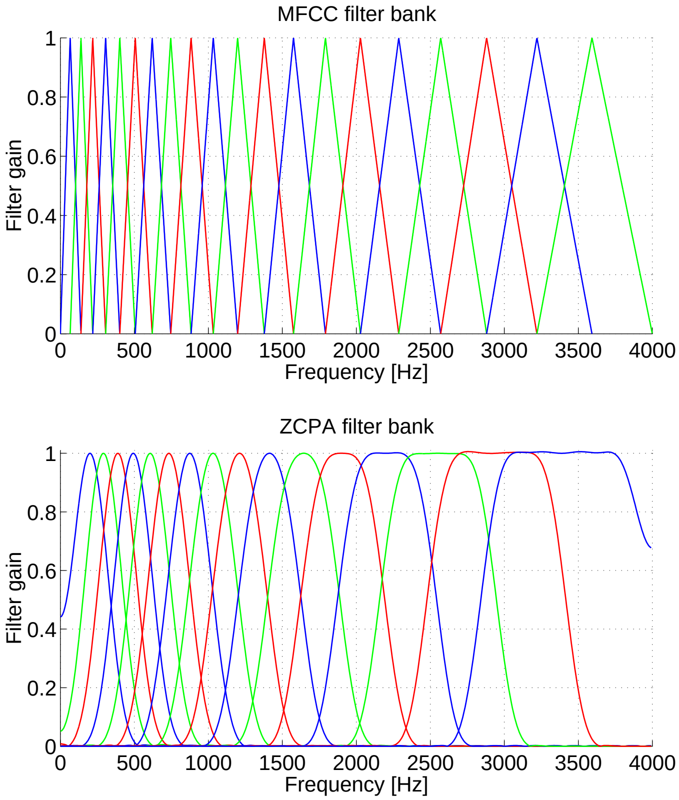

Figure 5.1: Filter banks used in MFCC and ZCPA feature extraction methods

5.3.1.2 Zero Crossings with Peak Amplitudes

Parameters involved in ZCPA computation were chosen in accordance with the results presented in Section 5.2.1, such that the ASR performance at low SNRs was optimized.

The interpolation factors used for the different subband signals ranged from one (i.e. no interpolation) for the first four subband signals to sixteen for the last three subband signals. The average interpolation factor was equal to six. The interpolation was performed using the MATLAB program package. Eight original sample values were used to compute the interpolated values.

The analysis-frame lengths were set to $60/\sqrt{F_{c_k}}$, where F_{c_k} is the center frequency of the k-th subband. This resulted in frame lengths between 33 ms at the high-frequency end and 134 ms at the low frequency end. The centers of all analysis frames were time-aligned. Frame shift was equal to 10 ms for all subband signals. Analysis frames were extracted using rectangular windows.

No preemphasis was used.

The filter bank consisted of 16 Hamming FIR filters designed by the windowing method, with center frequencies uniformly distributed on the Bark scale between 200 Hz and 3400 Hz. The bandwidths of the ideal prototype filters were equal to 2 Bark, while the 3 dB bandwidths ranged from 2.9 Bark to 2.1 Bark for the first nine filters, and were equal to 2 Bark for higher frequency bands. The corresponding linear frequency 3 dB bandwidths ranged from 280 Hz to 1160 Hz. The filter bank is illustrated in the bottom part of Figure 5.1.

Frequency range between 0 and 4000 Hz was partitioned into 60 histogram bins uniformly distributed on the Bark scale. Twelve DCT coefficients were derived from the histograms, as well as their first and second derivatives, resulting in 36-dimensional feature vectors. The word insertion penalty, p, and the language model scale factor, s, used on the RM recognition task were equal to -10 and 13, respectively.

5.3.1.3 Frequency-Domain Bark-Frequency Cepstral Coefficients

BFCCF features were computed in exactly the same manner as MFCC features. The only difference was in the particular choice of the filter bank. The frequency response of the filter bank used in BFCCF computation was identical to that of the FIR filter bank used in ZCPA computation. It was obtained by evaluating the magnitude of the z-transform of the FIR filter impulse responses at required frequency points.

The word insertion penalty, p, and the language model scale factor, s, used on the RM recognition task were both equal to 10.

5.3.1.4 Time-Domain Bark-Frequency Cepstral Coefficients

BFCCT features were derived from subband power estimates computed in the time domain, as described in Section 2.4.2. Analysis-frame lengths were equal to $60/\sqrt{F_{c_k}}$, where F_{c_k} is the center frequency of the k-th subband. The filter bank was chosen identical to the one used in ZCPA computation. Twelve cepstral coefficients were derived from the subband power estimates, as well as their first and second derivatives, resulting in 36-dimensional feature vectors. The word insertion penalty, p, and the language model scale factor, s, used on the RM recognition task were equal to 0 and 10, respectively.

The analysis-frame lengths were chosen to be equal to those used in the ZCPA method. However, it was observed that the benefit due to the use of large frame lengths with the BFCCT method was considerably smaller than for the ZCPA method, especially on the RM task. Detailed recognition results obtained by varying frame lengths in BFCCT and ZCPA feature extraction methods are given in Tables B.1 and B.2 in Appendix B.

5.3.1.5 Zero-Crossing Features

The only difference between computation of ZC features and ZCPA features is in the way histograms were constructed. Instead of increasing the bin counts by the logarithm of peak amplitudes, they were increased by one. Note that this feature extraction method is equivalent to the EIH method described in Section 4.5 with only one level, which is equal to zero.

All free parameters used in computation of ZC features were chosen equal to the corresponding ZCPA parameters. The word insertion penalty, p, and the language model scale factor, s, used on the RM recognition task were equal to 0 and 10, respectively.

5.3.2 Experimental Results

Figures 5.2 and 5.3 illustrate the recognition performance of MFCC, BFCCF, BFCCT and ZCPA features in presence of four different types of additive noise, evaluated on ISOLET and RM databases respectively. Detailed experimental results are given in Tables B.3 and B.4 in Appendix B.

5.3.2.1 Comparing ZCPA and MFCC Performance

It can be seen from Figures 5.2 and 5.3 that the ZCPA performance decreased considerably slower than the MFCC performance with reduced SNR. This was true for all noise types on both databases. This is in agreement with earlier results [70], which showed greater robustness of ZCPA features compared to standard MFCC features. On the other hand, MFCC features outperformed ZCPA features at high SNRs. This indicates superior discriminative properties of MFCC features in matched training and test conditions.

Figure 5.4 shows the absolute difference in ASR performance between ZCPA and MFCC features on ISOLET and RM databases in different background conditions. It can be seen that the advantage of using ZCPA instead of MFCC generally increased with reduced SNR. Furthermore, the advantage was much larger on the ISOLET database than on the RM database. This indicates that the advantage of using ZCPA is reduced with increased task complexity.

The effect of the three main differences between ZCPA and MFCC feature extraction methods on the difference in ASR performance is investigated in the following.

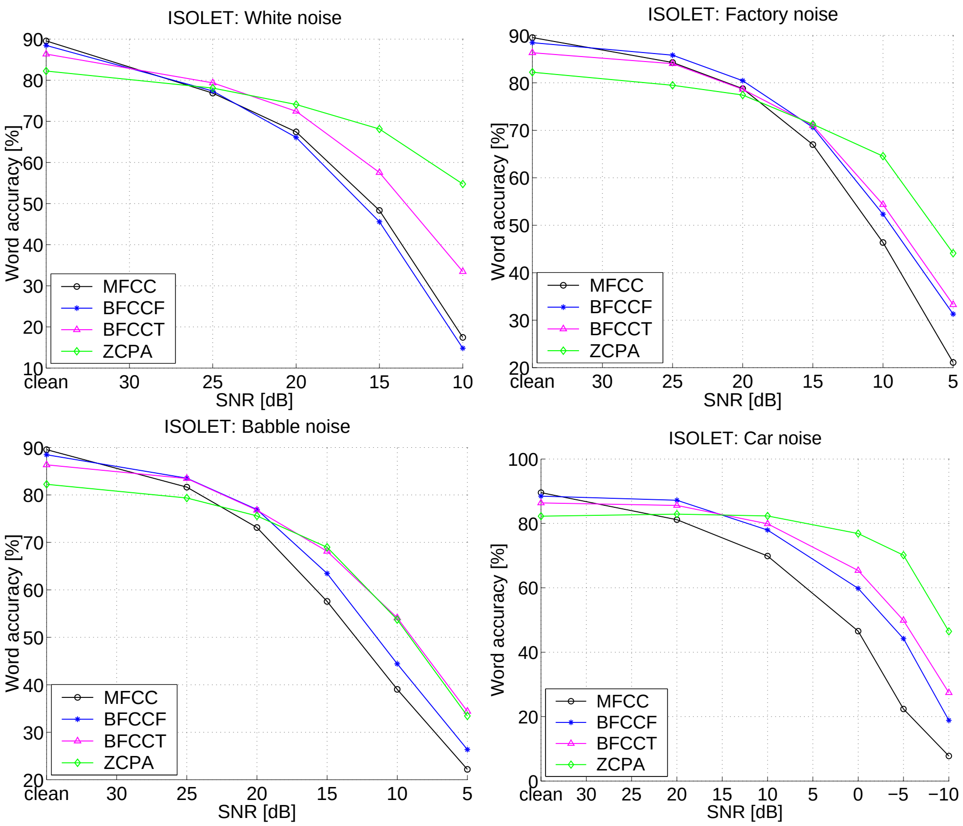

Figure 5.2: Comparison of MFCC, BFCCF, BFCCT and ZCPA features on ISOLET database in presence of different types of additive noise.

5.3.2.2 Effect of Different Filter Banks

The effect of using different filter banks in MFCC and ZCPA feature extraction methods was investigated by comparing the performance of BFCCF and MFCC features in Figures 5.2 and 5.3. The two speech feature types differ only in the subband filter banks used for their computation.

Figure 5.5 shows the absolute difference in the ASR performance between the two feature types on ISOLET and RM databases in different background conditions. The only considerable performance difference between the two methods can be observed in the presence of car noise, where BFCCF features largely outperformed MFCC features on both databases. This can be explained by larger subband bandwidths used in the BFCCF method compared to the MFCC method, and the special narrow-band shape of the car noise.

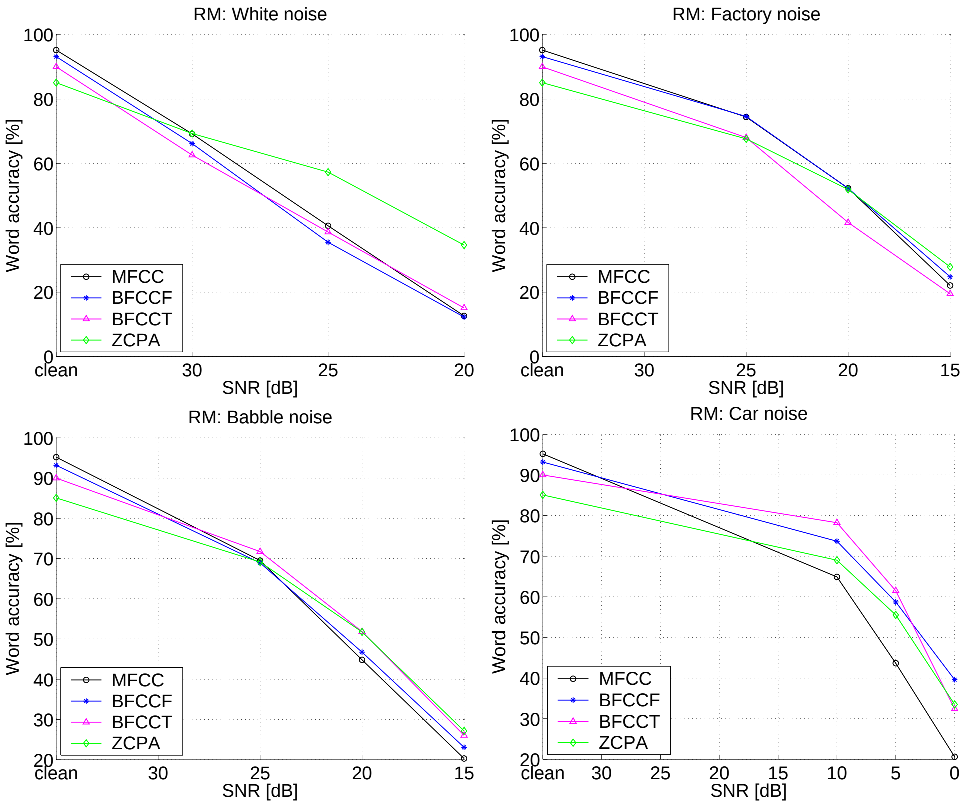

Figure 5.3: Comparison of MFCC, BFCCF, BFCCT and ZCPA features on RM database in presence of different types of additive noise.

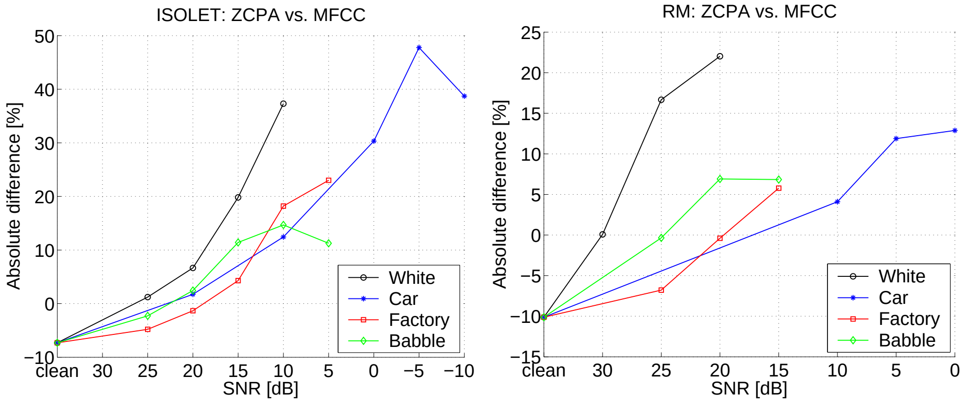

Figure 5.4: Absolute difference in ASR performance between ZCPA and MFCC features on ISOLET and RM databases in presence of four different noise types.

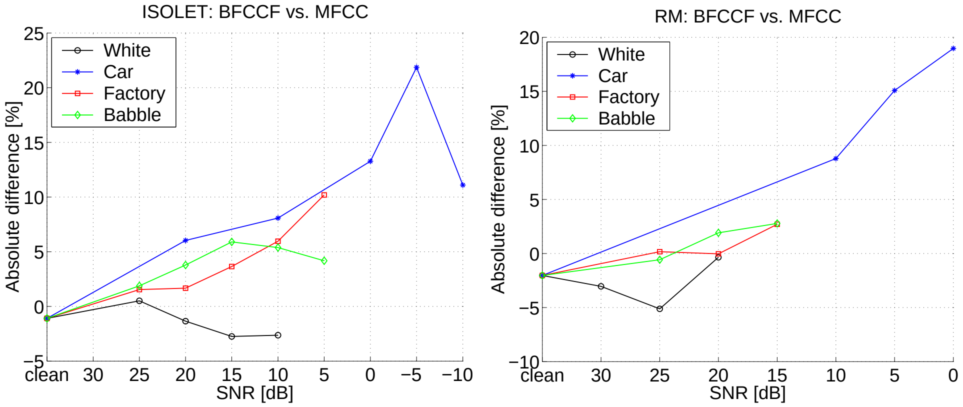

Figure 5.5: Effect of filter bank change: Absolute difference in ASR performance between BFCCF and MFCC features on ISOLET and RM databases in presence of four different noise types.

Increased subband bandwidths lead to increased speech subband power, while the subband power of the narrow-band noise remains approximately the same. Thus, local subband SNRs increase with increased subband bandwidth, which might be the reason for the increased ASR performance.

5.3.2.3 Effect of Time-Domain Processing

The effect of using time-domain processing instead of frequency-domain processing was investigated by comparing the ASR performance of BFCCT and BFCCF features in Figures 5.2 and 5.3. In addition, Figure 5.6 illustrates the absolute difference in ASR performance between the two feature types on ISOLET and RM databases in different background conditions.

It can be seen that on the ISOLET database time-domain processing led to a large performance improvement in presence of white noise compared to frequency-domain processing. The improvement increased with increased noise level. Smaller improvements were also observed for babble and car noise at sufficiently low SNRs, while no significant improvement was observed for factory noise. The improvements due to time-domain processing achieved on the RM database were much smaller, and a performance degradation was observed in the case of factory noise.

One possible reason for the positive effect of time-domain processing is due to the difference in the analysis-frame lengths used in BFCCF and BFCCT methods. The BFCCF method used the frame lengths equal to 25 ms for all subband signal, while the BFCCT method used frequency-dependent frame

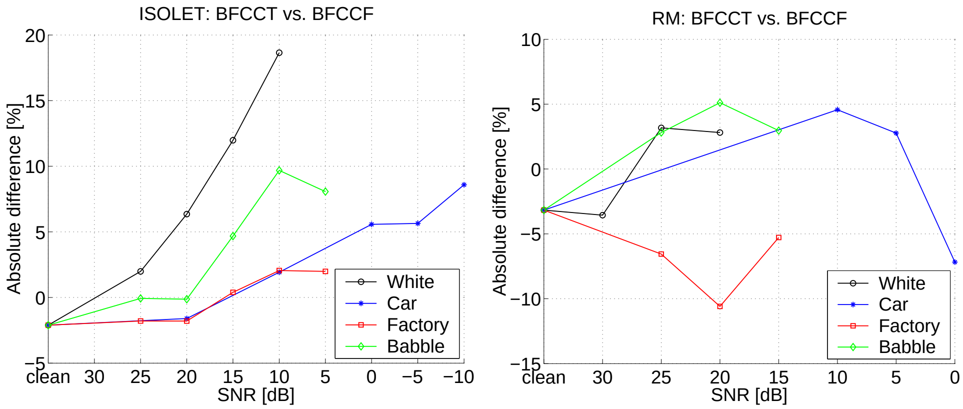

Figure 5.6: Effect of time-domain processing: Absolute difference in ASR performance between BFCCT and BFCCF features on ISOLET and RM databases in presence of four different noise types.

lengths that ranged between 33 ms and 134 ms. Longer frame lengths, especially at low frequencies, lead to more reliable power estimates. On the other hand, too long frames lead to violating the stationarity assumption. This might explain why the use of time-domain processing was least beneficial in the case of highly unstationary factory noise, and why the improvements were reduced when tested on continuous speech, which is characterized by shorter stationary intervals.

5.3.2.4 Effect of Dominant Subband Frequency Information

The effect of using dominant subband frequency information in speech feature extraction was investigated by comparing the ASR performance of ZCPA and BFCCT features in Figures 5.2 and 5.3. Both feature types are derived in the time domain using identical subband filter banks. Thus, the only difference between the two feature types is in the type of information extracted from the subband signals. While BFCCT features are based solely on the subband power estimates, ZCPA features combine dominant subband frequency information with the subband power information.

Figure 5.7 shows the absolute difference in ASR performance between ZCPA and BFCCT features on ISOLET and RM databases in different background conditions. Common observation for both databases is that the use of dominant subband frequency information led to improved ASR performance in presence of white and factory noise at sufficiently low SNRs, while no improvement was achieved in presence of babble noise. The largest improvements were achieved in presence of white noise. This indicates that dominant frequency

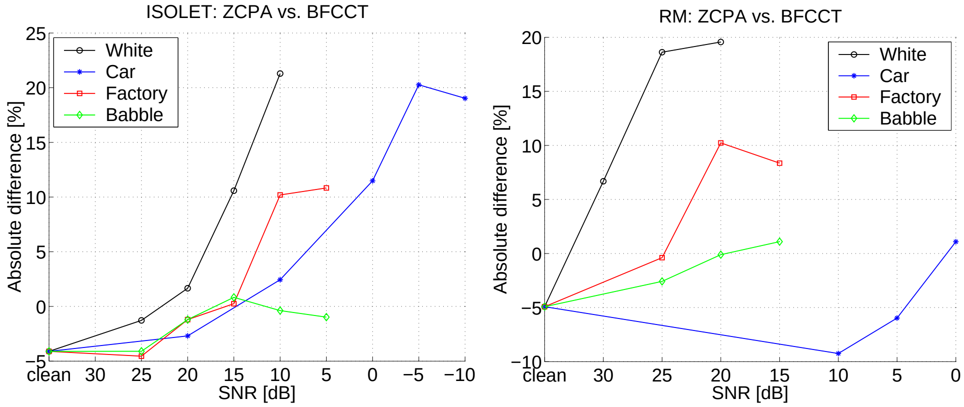

Figure 5.7: Effect of dominant subband frequency information: Absolute difference in ASR performance between ZCPA and BFCCT features on ISOLET and RM databases in presence of four different noise types.

information has the largest positive effect when additive noise has relatively flat spectrum.

At high SNRs, BFCCT features performed better than ZCPA features. This can be explained by the fact that ZCPA features do not provide reliable information about spectral valleys. This information becomes unreliable in presence of additive noise, in which case its exclusion from speech features can be advantageous. However, at high SNRs, this information contributes to better discrimination between different speech units.

The results for car noise were inconsistent. While a large improvement was observed on ISOLET database, some performance reduction was observed on the RM database. The performance reduction on the RM database in presence of car noise can be explained by looking at the results in Table B.1, which show that the choice of long analysis frames in the ZCPA method was not advantageous in presence of car noise. If two times shorter window lengths were used, the improvements would be similar to those observed for white noise. It was argued in Section 5.2.1.1 that longer analysis frames in the ZCPA method lead to greater emphasis of spectral peaks. Since car noise is characterized by a strong spectral peak in the low-frequency region, this peak would also be emphasized when frame lengths are increased. This might be the reason for the reduced performance of the ZCPA method in presence of car noise.

5.3.2.5 Effect of Subband Power Information

At the end, the performance of ZCPA and ZC features was compared, in order to determine the effect of using subband power information in the ZCPA feature extraction method. The two feature types differ only in the way histograms are constructed. While histogram bin counts in the ZCPA method are increased by the logarithm of peak amplitudes, they are increased by one in the ZC method. Thus, no power information is explicitly used in the ZC method.

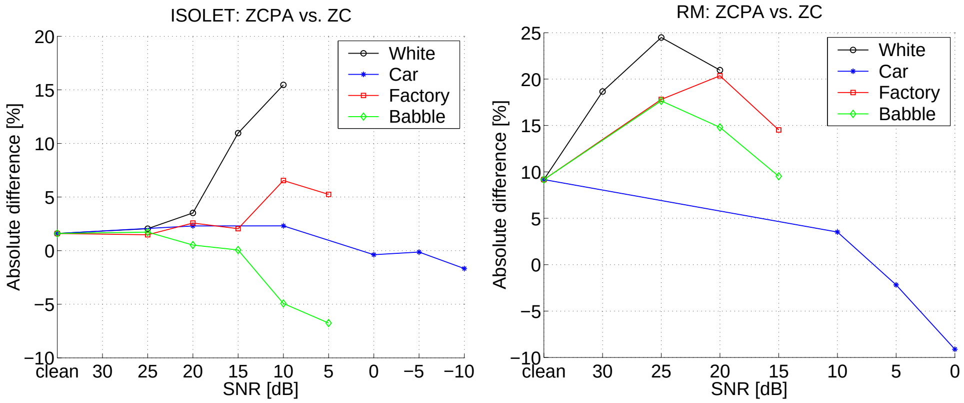

Figure 5.8: Effect of subband power information: Absolute difference in ASR performance between ZCPA and ZC features on ISOLET and RM databases in presence of four different noise types.

Figure 5.8 shows the absolute difference in ASR performance between ZCPA and ZC features on ISOLET and RM databases in different background conditions. On the ISOLET database, the only considerable performance improvement obtained due to the explicit use of subband power information is observed in presence of white noise. In presence of babble noise, on the other hand, ZC features outperform ZCPA feature at low SNRs. This is probably due to the fact that the subband power information becomes highly unreliable when noise subband power becomes relatively large. On the RM database, the positive effect due to the explicit use of subband power information is much more pronounced, and relatively large improvements were observed for all noise types except car noise. The poor ZCPA performance in presence of car noise can again be explained by the use of too long analysis frames.

5.3.3 Summary of the Main Results

In this section the ZCPA feature extraction method was evaluated on two different recognition tasks. The main results are summarized in the following:

- ZCPA features were more robust against additive background noise than MFCC features. The advantage of using ZCPA features generally increased with reduced SNR. Furthermore, it was considerably larger on the isolated-word task, than on the more complex continuous-speech task.

- MFCC features performed somewhat better than ZCPA features at high SNRs. This is probably due to the loss of information in speech spectral valleys in the ZCPA method.

- The performance improvement of the ZCPA features compared to the MFCC features was partly due to the use of dominant subband frequency information, and partly to the use of time-domain processing. Differences in the filter banks used in the two methods had only an effect in presence of car noise, due to its narrow-band spectral shape.

- The use of dominant subband frequency information had the largest positive effect in presence of background noise with a relatively flat spectral characteristic.

- The importance of using subband power information in the ZCPA method was increased with increased task complexity.

5.4 Evaluation of SSC-Based Methods

This section presents an experimental study aimed at evaluating the performance of SSC-based feature extraction methods on the two recognition tasks described in Section 5.1, in presence of various background conditions. First, the effect of augmenting standard MFCC feature vectors by three SSC was investigated, in order to verify the previous results reported on this feature set. The new feature set is referred to as MFCC+SSC. Then, the ASR performance of SSCH features and standard MFCC features was compared in different acoustic background conditions. Finally, the robustness of MFCC, SSCH and SSC features was compared by measuring the distance between feature vector distributions of clean and noisy speech for the three feature types.

This section starts with a description of implementational details for SSCH and MFCC+SSC feature extraction methods in Section 5.4.1. Next, the recognition results are presented in Section 5.4.2, together with a detailed discussion. Finally, the most important results are summarized in Section 5.4.3.

5.4.1 Implementational Details

This section summarizes the implementation details for SSCH and MFCC+SSC feature extraction methods. The general description of the methods was given in Chapter 4.

5.4.1.1 Subband Spectral Centroid Histograms

Parameters involved in SSCH computation were chosen in accordance with the results presented in Section 5.2.2. SSC were computed from the FFT-based spectral estimate, using the dynamic range parameter $\gamma = 1$. The filter bank consisted of 48 subband filters with rectangular frequency responses, and bandwidths equal to 3 Bark. The filter center frequencies were uniformly distributed on the Bark scale between 100 and 3800 Hz. Note that bandwidths of some filters at lowest and highest frequencies had to be reduced in order to fall inside the speech frequency range [0,4000 Hz]. Frequency range between 100 and 3800 Hz was divided into 38 histogram bins uniformly distributed on Bark scale. At the end, 12 DCT coefficients were derived from the histogram representation, as well as their first and second derivatives, resulting in 36-dimensional feature vectors. The word insertion penalty, p, and the language model scale factor, s, used on the RM recognition task were equal to -10 and 10, respectively.

5.4.1.2 SSC as Additional Speech Features

SSC used as additional features to standard MFCC feature vectors were computed in a similar way as those used for SSCH computation. The computation differed only in the choice of filter bank, which in this case consisted of three disjoint rectangular filters uniformly distributed on the linear frequency scale between 0 and 4000 Hz. Thus, the bandwidth of each filter was approximately 1333 Hz. The three SSC were augmented to the 36-dimensional MFCC feature vectors computed as described in Section 5.3.1. This implementation of MFCC+SSC features is very similar to the one reported in [105]. The word insertion penalty, p, and the language model scale factor, s, used on the RM recognition task were equal to 0 and 10, respectively.

5.4.2 Experimental Results

Figures 5.9 and 5.10 illustrate the recognition performance of the standard MFCC features used alone and in combination with three SSC, as well as that of the SSCH features. The evaluation was done on the ISOLET and RM databases, both on clean speech and in presence of four different types of additive noise added at various SNRs. In addition, the performance of a speech representation consisting solely of the three SSC is shown for the ISOLET database. The detailed experimental results are given in Tables B.3 and B.4 in Appendix B.

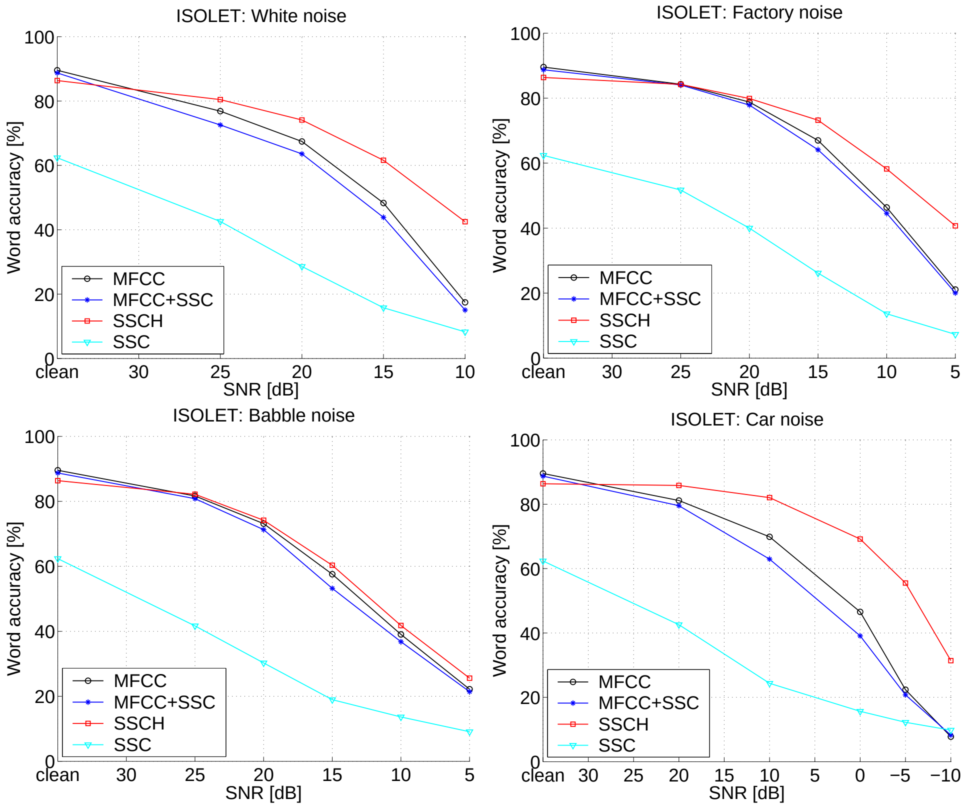

Figure 5.9: Comparison of MFCC, MFCC+SSC, SSCH and SSC features on the ISOLET database in presence of different types of additive noise.

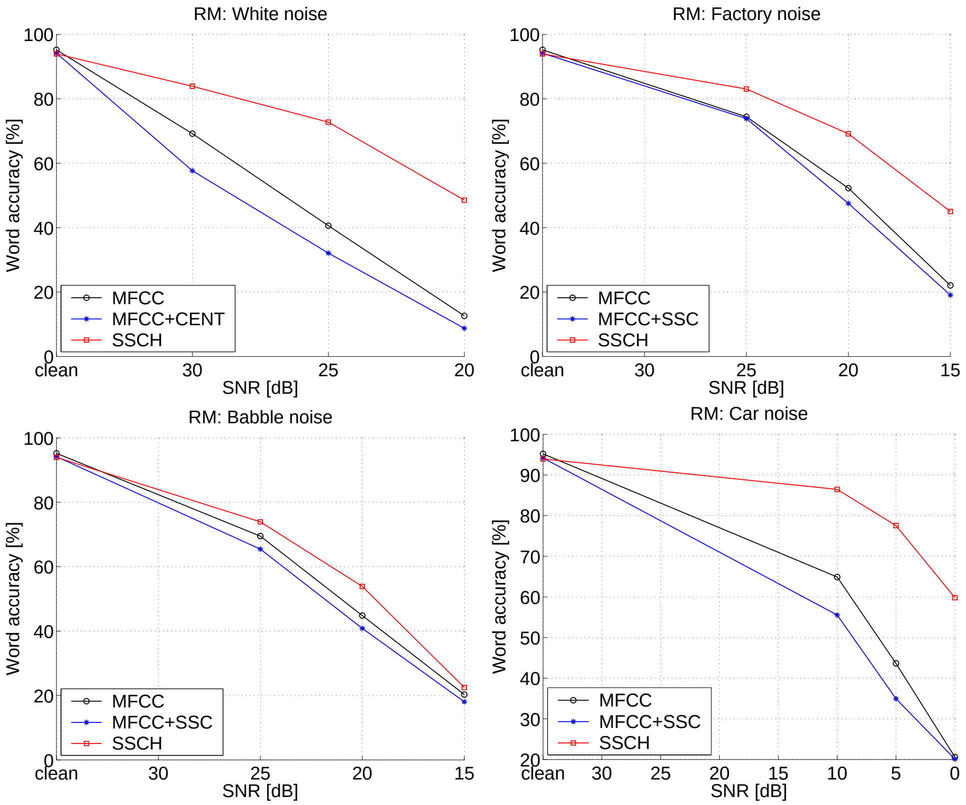

Figure 5.10: Comparison of MFCC, MFCC+SSC and SSCH features on the RM database in presence of different types of additive noise.

5.4.2.1 SSC as Additional Speech Features

A comparison of the recognition performance of MFCC and MFCC+SSC features in Figures 5.9 and 5.10 shows a consistent small performance reduction caused by augmenting three SSC to the standard MFCC feature vectors. This result differs from the previous studies described in Section 4.7, which all reported some positive effect of augmenting SSC to MFCC feature vectors. However, the previous results were somewhat inconsistent. While some researchers observed a positive effect of adding SSC only in clean speech, others reported increased positive effect with reduced SNR.

In order to explain the negative effect of augmenting MFCC feature vectors by SSC, the performance of feature vectors consisting solely of the three SSC was evaluated on the ISOLET database. The results are illustrated in Figure 5.9. It can be seen that the three SSC used alone provided relatively good discrimination on clean speech, but the performance deteriorated rapidly with increased background noise level. The poor robustness of the SSC features can be explained by the fact that SSC serve as reasonable estimates of speech spectral peak positions only in the subbands that contain a single speech spectral peak. However, since spectral peak positions vary with the particular speech sound, it is not possible to design a filter bank that would produce suitable subband locations for all speech sounds.

In order to examine the effect of using different filter banks in SSC computation, two additional recognition experiments were performed. In the first one, the passbands of the three subband filters were changed to correspond to the frequency ranges of the first three speech formants (i.e. 0–1175 Hz, 315–2860 Hz and 1175–4000 Hz), as suggested in [23]. In the second experiment, 13 disjoint subband filters with bandwidths equal to 300 Hz were used. The two SSC implementations are referred to as SSC-formant and SSC-300Hz, respectively. The recognition performance on the ISOLET database in presence of different levels of additive white Gaussian noise is shown in Table 5.7. It can be seen that some performance improvement can be achieved by choosing the subbands that comply better with spectral peak positions in speech signals. On the other hand, a large performance reduction was experienced after reducing filter bandwidths. This can be explained by the existence of a number of subbands that do not contain any dominant speech spectral peak. The centroids of such subbands are seriously affected by noise.

Table 5.7: ASR performance of SSC features for different choices of the subband filter bank. The evaluation was done on the ISOLET database, on clean speech and in presence of additive white Gaussian noise at different SNRs.

Method	Word accuracy [%]				
	Clean	SNR [dB]			
	speech	25	20	15	10
SSC	62.37	42.56	28.59	15.77	8.27
SSC-formant	69.55	43.65	30.19	18.72	12.12
SSC-300Hz	59.10	20.83	14.10	10.83	8.97

5.4.2.2 Comparing SSCH and MFCC Performance

The performance of SSCH and MFCC features is compared next. It can be seen from Figures 5.9 and 5.10 that SSCH features outperformed MFCC features in presence of additive noise, while MFCC features performed slightly better on clean speech. Figure 5.11 shows the absolute difference in ASR performance between SSCH and MFCC features on the ISOLET and RM databases in various background conditions. The advantage of using SSCH features in place of MFCC features was largest in car noise, followed by white and factory noise, while only a small improvement was observed in babble noise. The large improvement in presence of car noise is partly due to the exclusion of the frequency range below 100 Hz from the histogram representation, since most of the noise power is concentrated in that frequency region. A very limited improvement in presence of babble noise can be explained by the presence of prominent spectral peaks in this noise type, which makes dominant subband frequency information less reliable. This problem is less pronounced in presence of factory noise where intervals characterized by prominent spectral peaks interchange with those characterized by relatively flat spectrum. Another interesting observation drown from Figure 5.11 is that the maximal improvement achieved by using SSCH features instead of MFCC features was considerably larger on the more complex continuous-speech recognition task than on the isolated-word task. This result was consistent for all noise types.

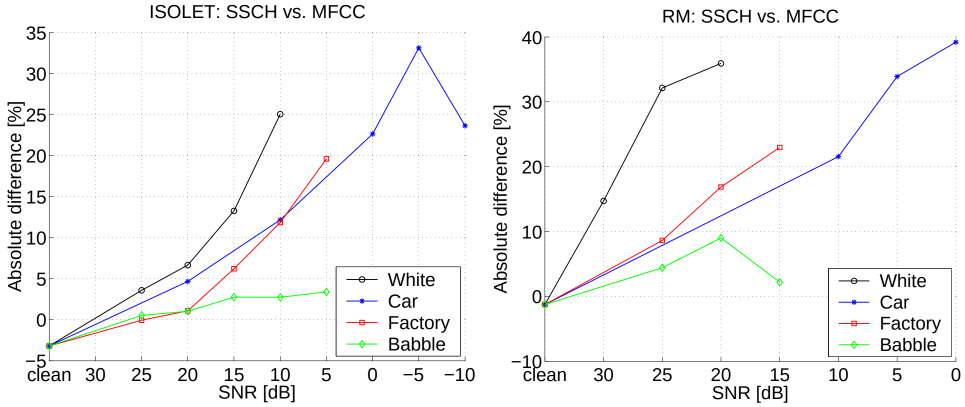

Figure 5.11: Absolute difference in ASR performance between SSCH and MFCC features on ISOLET and RM databases in presence of four different noise types.

In order to determine the influence of using different filter banks in the MFCC and SSCH methods on the difference in their robustness, MFCC features derived using a filter bank identical to that used in the SSCH method were evaluated. The evaluation was done on the ISOLET database, both on clean speech and in presence of additive white Gaussian noise at different

SNRs. The performance of the modified MFCC features is presented in Table 5.8, together with that of standard MFCC features and SSCH features. It can be seen that the performance of the modified MFCC features followed closely that of standard MFCC features, with a small degradation at high SNRs, and a small improvement at low SNRs. This indicates that the superior robustness of the SSCH method compared to the MFCC method was mainly due to the use of dominant subband frequency information provided by SSC.

Table 5.8: Performance comparison between standard MFCC features, SSCH features, and modified MFCC features derived using the same filter bank as in the SSCH method. The evaluation was done on the ISOLET database, both on clean speech and in presence of additive white Gaussian noise at different SNRs.

Feature type	Word accuracy [%]				
	Clean speech	SNR [dB]			
		25	20	15	10
MFCC standard	89.55	76.86	67.44	48.33	17.44
MFCC modified	86.09	74.36	64.42	48.59	21.15
SSCH	86.35	80.45	74.10	61.60	42.50

The problem of lacking robustness of SSC features is efficiently circumvented in the SSCH method. The subbands containing speech spectral peaks have considerably larger power than those containing no spectral peaks, and will thus lead to larger histogram contribution. Furthermore, the centroids of the subbands that contain a speech spectral peak are much less affected by additive noise than the centroids of the subbands that contain no speech spectral peaks. Thus, the SSCH method incorporates an efficient weighting scheme, which assigns larger weights to reliable SSC. However, this is true only if the noise does not contain prominent spectral peaks.

5.4.2.3 Noise Effect on MFCC, SSC and SSCH Features

Deterioration of ASR performance in presence of noise is due to the fact that the distributions of noisy-speech feature vectors differ from those of clean-speech feature vectors used in model training. By measuring the distance between corresponding feature vector distributions for clean and noisy speech, it is possible to get an indication of the noise robustness of a particular feature type. In this study, the distance was measured for MFCC, SSC and SSCH features. The clean-speech feature vectors were obtained from the original, clean-speech test set of the ISOLET database. Noisy-speech feature vectors were obtained from the same test set with white Gaussian noise added at SNR=15 dB. The procedure for measuring the distance between feature vector distributions for clean and noisy speech is described in the following:

1. The maximum-likelihood state sequence was found for each clean-speech utterance using a set of acoustic speech models. In this way, each feature vector was assigned to a particular model state. Furthermore, each feature vector was assigned to the Gaussian mixture component within the corresponding state that had the highest probability of having generated the vector. Thus, a number of feature vectors was assigned to each Gaussian mixture components in the model set.

2. For each Gaussian mixture component in the model set, the mean vector, $\{\mu_{cl}(k)\}_{k=1}^{K}$, and variance vector, $\{\sigma_{cl}^2(k)\}_{k=1}^{K}$ were computed over the set of clean-speech feature vectors assigned to the mixture component, $\{\boldsymbol{o_n}\}_{l=1}^{L}$, by

$$\mu_{cl}(k) = \frac{1}{L}\sum_{l=1}^{L} o_l(k) \tag{5.5}$$

$$\sigma_{cl}^2(k) = \frac{1}{L}\sum_{l=1}^{L}[o_l(k) - \mu_{cl}(k)]^2, \quad \text{for } k = 1, \ldots, K \tag{5.6}$$

where L is the total number of feature vectors assigned to the mixture component, and K is the feature vectors dimension. Similarly, noisy-speech mean and variance vectors, $\{\mu_n(k)\}_{k=1}^{K}$ and $\{\sigma_n^2(k)\}_{k=1}^{K}$, were computed by averaging over corresponding noisy-speech feature vectors.

3. The distance between feature vector distributions for clean and noisy speech corresponding to a Gaussian mixture component was computed using the following two distance measures:

$$d^{mean} = \sum_{k=1}^{K} \frac{[\mu_n(k) - \mu_{cl}(k)]^2}{\sigma_{cl}^2(k)} \tag{5.7}$$

$$d^{var} = \sum_{k=1}^{K} \frac{\sigma_n^2(k)}{\sigma_{cl}^2(k)}. \tag{5.8}$$

Ideally, the two measures should be equal to zero and one, respectively.

4. Finally, the weighted average of the distance measures given by Equations 5.7 and 5.8 was computed over all the mixture components in the model set in the following way

$$D^{mean} = \sum_{i=1}^{I} \sum_{j=1}^{J_i} s_{ij} \sum_{m=1}^{M_{ij}} c_{ijm} \, d_{ijm}^{mean} \tag{5.9}$$

$$D^{var} = \sum_{i=1}^{I} \sum_{j=1}^{J_i} s_{ij} \sum_{m=1}^{M_{ij}} c_{ijm} \, d_{ijm}^{var}. \tag{5.10}$$

   Indexes i, j and m refer to a particular model, model state, and state mixture components, respectively. Furthermore, I, J_i, and M_{ij} denote the number of models, model states, and state mixture components, respectively. Finally, c_{ijm} denote mixture component weights, and s_{ij} denote model state weights. The mixture component weights were given by the acoustic speech models, while the model state weights were computed as the ratio between the number of feature vectors associated to a model state and the total number of feature vectors associated to the corresponding model. The particular weighting scheme ensures that larger weights are given to the more likely mixture components.

Table 5.9 shows the distance measures given by Equations 5.9 and 5.10 computed for MFCC, SSCH and SSC features. In the case of MFCC and SSCH feature vectors, the distance measures were computed separately for the static, Δ and $\Delta\Delta$ features. The first and last model states were excluded from the averaging process, since they correspond mainly to the background events that surround speech utterances. Furthermore, in order to obtain a more reliable segmentation of the speech utterances in the case of SSC features, the segmentation was done on the MFCC+SSC feature vectors, while the distance measures were computed over the three SSC only.

Table 5.9: Average distance between feature vector distributions for clean and noisy speech measured for different feature types.

Distance	MFCC			SSCH			SSC
measure	static	Δ	$\Delta\Delta$	static	Δ	$\Delta\Delta$	static
D^{mean}	1.19	0.16	0.13	0.90	0.11	0.11	5.07
D^{var}	0.72	1.00	1.13	0.96	1.17	1.29	1.97

It can be observed from Table 5.9 that additive noise had the most serious effect on the SSC features. Furthermore, SSCH features were somewhat less affected by noise than MFCC features. Those results agree with the ones obtained by evaluating the recognition performance. Furthermore, the dynamic features exhibited very good robustness against additive noise compared to the static features. This is due to the fact that the effect of stationary additive noise is practically eliminated by subtracting the spectral feature vectors that are closely spaced in time.

5.4.3 Summary of the Main Results

This section presented the results of an experimental study performed in order to evaluate the recognition performance of SSC-based feature extraction methods in various background conditions. The main results are summarized in the following:

- Using SSC as additional features to standard MFCC feature vectors had a small negative effect on the MFCC performance. This can be explained by poor robustness of the SSC features if subband positions are not chosen appropriately.

- SSCH features outperformed standard MFCC features in presence of additive noise. The advantage of using SSCH features generally increased with reduced SNR. It was largest in presence of noise types with relatively flat spectral characteristic.

- The advantage of using SSCH features compared to MFCC features was mainly due to the use of dominant subband frequency information in the SSCH method.

- The distance between feature vector distributions for clean and noisy speech was smallest for SSCH features, followed by MFCC features, while it is by far the largest for SSC features. This result confirms the poor robustness of the SSC features, as well as the potential advantage of SSCH features compared to MFCC features in presence of additive noise.

5.5 Comparing ZCPA and SSCH Performance

It has been shown that both ZCPA and SSCH feature extraction methods were capable of improving the ASR performance in presence of additive noise compared to the standard MFCC feature extraction method. In the following, the performance of the two methods is compared.

Figure 5.12 illustrates the absolute difference in ASR performance of SSCH and ZCPA features on the ISOLET and RM databases in various background conditions. It can be seen that, on the ISOLET database, SSCH features performed slightly better than ZCPA features at high SNRs, while ZCPA features performed better at low SNRs. On the RM database, on the other hand, SSCH features performed considerably better than ZCPA features in all testing conditions. The only exception is in the case of babble noise, where the difference in performance between the two methods was small.

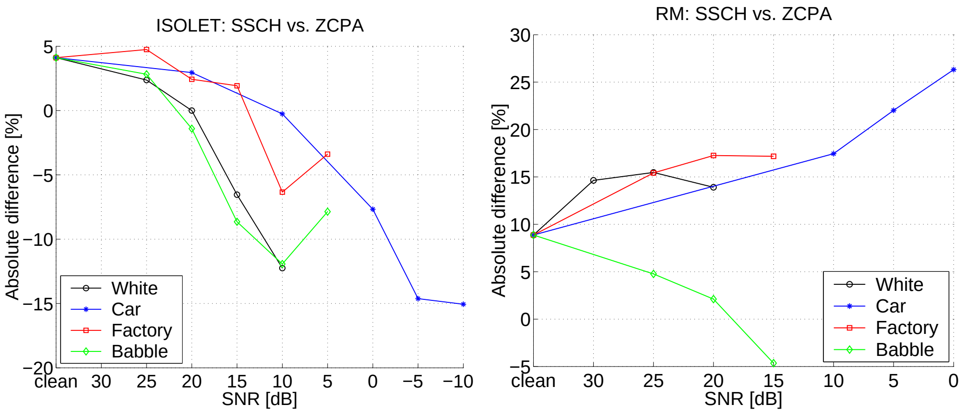

Figure 5.12: Absolute difference in ASR performance between SSCH and ZCPA features on ISOLET and RM databases in presence of four different noise types.

It has been shown that the robustness of ZCPA features was partly due to the advantages of time-domain processing, and partly to the use of dominant subband frequency information. On the other hand, the robustness of SSCH features was mainly due to the use of dominant subband frequency information. In order to compare the effect of using dominant subband frequency information in the ZCPA and SSCH feature extraction methods, Figures 5.7 and 5.11 are compared. It can be seen that the improvement of ASR performance due to the use of dominant frequency information is considerably larger for SSCH features, especially on the RM database. Furthermore, the advantage of using ZCPA features at low SNRs on the ISOLET database is mainly due to the positive effect of time-domain processing. Finally, the computational complexity of the SSCH method is much lower than that of ZCPA method. Thus, the SSCH feature extraction method represents a more attractive way of reducing the noise robustness problem in automatic speech recognition compared to the ZCPA method.

Chapter 6

Conclusions

This thesis presented a study of alternative speech feature extraction methods aimed at increasing ASR robustness against additive background noise. The main objective of the study was to investigate the effect of incorporating dominant subband frequency information into speech feature vectors. If frequency subbands are properly chosen, dominant subband frequencies correspond closely to spectral peak positions, which remain practically unchanged in presence of additive noise. Consequently, it was expected that the incorporation of dominant subband frequencies into speech feature vectors would improve ASR robustness in presence of additive noise.

Two earlier proposed feature extraction methods that combine dominant subband frequency information with subband power information were carefully studied. The first method, referred to as ZCPA, estimates dominant subband frequencies from zero-crossing statistics of the subband signals, while peak signal values between subsequent positive zero crossings serve as subband power estimates. The two types of information are then combined into a histogram representation. This method has earlier shown promising results in presence of additive noise. The second method, referred to as MFCC+SSC, estimates dominant subband frequencies as subband spectral centroids, and uses them as additional features to standard MFCC feature vectors. Also this method was earlier shown to have a positive effect on ASR performance, although the results were somewhat inconsistent.

In this study, a new method for incorporating dominant subband frequencies into speech feature vectors was proposed. The dominant subband frequencies were estimated as subband spectral centroids, and combined with subband power estimates into a histogram representation. The method is referred to as SSCH.

An experimental study was performed in order to optimize the free parameters involved in computation of ZCPA and SSCH features. This was followed by a comparison of ZCPA, SSCH, MFCC+SSC and standard MFCC features on two different recognition tasks in various background conditions. The major results are summarized in the following:

- It was shown that the use of dominant subband frequencies in speech feature extraction led to a considerable improvement in ASR performance in presence of additive noise. Largest improvements were achieved in presence of noise with relatively flat spectrum. Both ZCPA and SSCH features exhibited greater robustness compared to standard MFCC features in such conditions. However, the improvement due to the use of dominant subband frequency information was considerably larger for SSCH features, especially on the more complex continuous-speech recognition task. MFCC+SSC features, on the other hand, led to a small performance reduction compared to standard MFCC features. This was explained by poor robustness of subband spectral centroids when they are used directly as ASR features, which is due to their large dependence on the particular choice of frequency subbands.

- The computational complexity of the SSCH method is two orders of magnitude lower than that of the ZCPA method, and of the same order of magnitude as the MFCC method.

- The results of the optimization of the analysis frame lengths used in the ZCPA method indicate that the use of relatively long analysis frames is advantageous in presence of noise, while this does not have a significant negative effect on clean speech performance. The increase in performance is probably due to the fact that larger number of histogram points led to greater emphasis of spectral peaks in ZCPA feature vectors.

The major limitation of the SSCH method lies in the fact that it is designed to deal with additive noise only. Furthermore, it is implicitly assumed that spectral peaks belong to speech. Thus, the method is not expected to be effective in presence of additive background noise characterized by strong spectral peaks.

An advantage of robust feature extraction methods compared to most other methods for increasing noise robustness in ASR is the fact that they do not require any knowledge of the target environment. However, in the situations where such knowledge is available, or easy to obtain, a better recognition performance might be obtained by utilizing this knowledge. Thus, an important extension of the work presented in this thesis would be to investigate whether the use of SSCH features can be effectively combined with some of the methods for increasing noise robustness described in Chapter 3. Note that such a combined approach could also circumvent the limitation of the SSCH method to additive noise.

All the methods for incorporating dominant-frequency information into speech feature vectors studied in this work were dependent on the particular choice of frequency subbands. The problem with such approaches is that it is not possible to find a single subband allocation that would be optimal for all speech sounds. This problem was partly circumvented by the special way of histogram construction used in both ZCPA and SSCH methods. However, it is expected that the benefit of using dominant-frequency information in speech feature extraction would be considerably larger if this information could be estimated in a more reliable way. Furthermore, there might exist more effective ways of combining the dominant subband frequencies and subband power estimates. Those issues are left for future studies.

At the end, it should be remembered that all feature extraction methods depend on a large number of free parameters. Although an attempt was made in this study to optimize the most important parameters in ZCPA and SSCH methods, it was not feasible to investigate all parameter combinations in all the methods. In addition, the parameter optimization was done on the small-vocabulary isolated-word task, and only in presence of white noise. Thus, the results reported in this study will not necessarily extend to other noise types and other databases. Furthermore, although the evaluation of the different feature extraction methods was done on two different recognition tasks, and the major results were consistent on both tasks, different behavior might be observed with different choices of free parameters and evaluation tasks. Finally, it should be noted that, in spite of the improvements reported in this study, the obtained recognition performance at low SNRs is still far too low for most practical applications. Thus, this work represents only a small contribution to solving the difficult problem of increasing noise robustness in automatic speech recognition.

Appendix A

Statistical Considerations

ASR systems are usually evaluated by measuring word error rate on a given test database. Evaluation on different databases gives different estimates of the word error rate. Thus, it is important to asses some knowledge about quality of the estimates. This is usually done by finding a confidence interval (c_1, c_2) such that

$$P(c_1 < e < c_2) = 1 - \alpha, \tag{A.1}$$

where e is the true error rate of the system, and α is a constant that determines the significance level. Parameter α is typically set to 0.05 or 0.01, giving rise to 95% and 99% confidence intervals respectively.

A procedure for estimating confidence intervals is described in [55]. Under the assumptions that recognition of each word is done independently, and that the probability of erroneous recognition of each word is the same, the confidence interval is given by

$$c_1 = \frac{N\hat{e} - \frac{1}{2} + \frac{C(\alpha)^2}{2} - C(\alpha)\sqrt{\frac{C(\alpha)^2}{4} + (N\hat{e} - \frac{1}{2})(1 - \hat{e} + \frac{1}{2N})}}{N + C(\alpha)^2} \tag{A.2}$$

$$c_2 = \frac{N\hat{e} + \frac{1}{2} + \frac{C(\alpha)^2}{2} + C(\alpha)\sqrt{\frac{C(\alpha)^2}{4} + (N\hat{e} + \frac{1}{2})(1 - \hat{e} - \frac{1}{2N})}}{N + C(\alpha)^2}, \tag{A.3}$$

where $\hat{e}$ is the measured word error rate, N is the total number of words in the database, and $C(\alpha)$ is a constant dependent on the chosen significant level. Constant $C(\alpha)$ is computed as

$$C(\alpha) = \Phi^{-1}(1 - \alpha/2), \tag{A.4}$$

where $\Phi(\cdot)$ is standard normal probability distribution whose values can be obtained from the statistical tables.

Figure A.1 illustrates confidence interval lengths, $c_2 - c_1$, estimated for ISOLET and RM databases for different values of measured recognition rate. The confidence level was equal to 95%. Smaller confidence intervals imply greater statistical significant of the recognition results. Confidence intervals can be reduced by increasing the size of the test set. Indeed, it follows from Equations A.2 and A.3 that

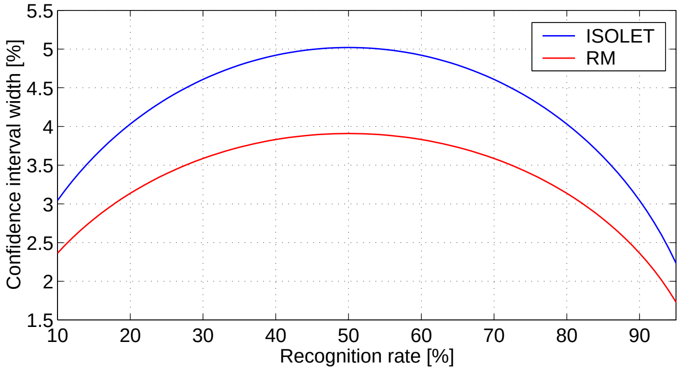

Figure A.1: Confidence interval lengths as a function of recognition rate on ISOLET and RM databases. The confidence interval was 95%.

$$\lim_{N\to\infty} c_1 = \lim_{N\to\infty} c_2 = \hat{e}, \tag{A.5}$$

thus reducing the confidence interval to a single point.

Confidence intervals provide information about reliability of a single recognition experiment. Since the objective of this study was to compare the performance of different feature extraction methods, there is a need to determine whether the difference in word error rates obtained by two different methods is statistically significant. This is basically a hypothesis testing task, where the zero and alternative hypotheses are given by

$$H_0 : \text{Word error rates of the two ASR systems are equal} \qquad e_1 = e_2 \tag{A.6}$$

$$H_1 : \text{Word error rates of the two ASR systems are different} \qquad e_1 \neq e_2 \tag{A.7}$$

In order to accept the zero hypothesis with significance level $1-\alpha$, the following condition must be fulfilled

$$P(|\hat{e_1} - \hat{e_2}| < c) > 1 - \alpha, \tag{A.8}$$

where $\hat{e_1}$ and $\hat{e_2}$ are measured word error rates for the two different ASR systems, and parameter c represents the minimal absolute difference in the recognition performance needed to accept the alternative hypothesis. Parameter c can be estimated using a simple two-tailed significance test described in [50]. The resulting estimate is given by

$$c = C(\alpha)\sqrt{\frac{2\hat{e}(1-\hat{e})}{N}}, \tag{A.9}$$

where $C(\alpha)$ is given by Equation A.4 and $\hat{e} = (\hat{e_1}+\hat{e_2})/2$. Figure A.2 illustrates the minimal significant difference in word error rate between two ASR systems as a function of the average recognition rate for on ISOLET and RM databases. The confidence level was 95%.

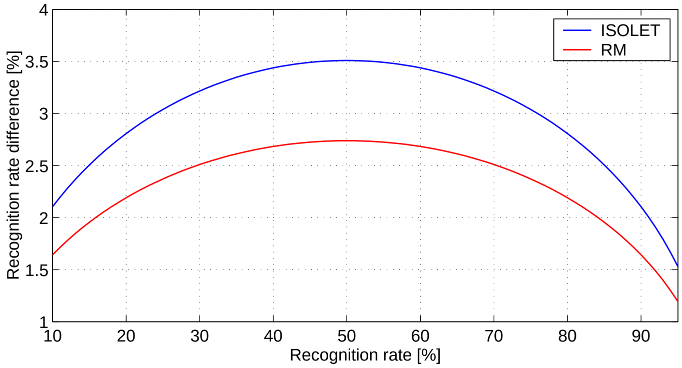

Figure A.2: Minimal significant difference in word error rate between two ASR systems as a function of average recognition rate on ISOLET and RM databases. The confidence interval was 95%.

The above test assumes that the word error rates of the two compared ASR systems are independent, in addition to the assumptions used in computation of the confidence intervals. However, this assumption is erroneous when the two ASR systems are tested on the same database. Thus, Equation A.9 provides only a rough estimate of the minimum significant difference. Two better tests are described in [50].

Appendix B

Detailed Experimental Results

Table B.1: ASR performance of BFCCT and ZCPA features on the RM database for two different choices of analysis frame lengths, $30/\sqrt{F_{c_k}}$ and $60/\sqrt{F_{c_k}}$, where F_{c_k} is the center frequency of k-th bandpass filter given in kHz.

Noise type	SNR [dB]	Word accuracy [%]			
		BFCCT $30/\sqrt{F_{c_k}}$	BFCCT $60/\sqrt{F_{c_k}}$	ZCPA $30/\sqrt{F_{c_k}}$	ZCPA $60/\sqrt{F_{c_k}}$
No noise	-	93.21	90.00	85.12	85.08
White	30	68.72	62.59	63.49	69.27
White	25	43.93	38.66	43.46	57.28
White	20	18.74	15.07	18.27	34.63
Factory	25	73.33	68.02	62.67	67.63
Factory	20	49.20	41.66	43.42	51.89
Factory	15	20.50	19.48	17.88	27.84
Bobble	25	73.80	71.73	64.94	69.15
Bobble	20	50.57	51.85	47.99	51.74
Bobble	15	29.83	26.04	26.12	27.14
Car	10	76.61	78.25	75.52	69.00
Car	5	61.19	61.50	66.85	55.53
Car	0	33.93	32.41	49.98	33.50

Table B.2: ASR performance of BFCCT features on the ISOLET database for two different choices of analysis frame lengths, $30/\sqrt{F_{c_k}}$ and $60/\sqrt{F_{c_k}}$, where F_{c_k} is the center frequency of k-th bandpass filter given in kHz.

Noise type	SNR [dB]	Word accuracy [%] BFCCT $30/\sqrt{F_{c_k}}$	 BFCCT $60/\sqrt{F_{c_k}}$
No noise	-	87.12	86.35
White	25	75.51	79.36
White	20	67.37	72.44
White	15	56.79	57.56
White	10	26.92	33.46
Factory	25	80.77	84.04
Factory	20	74.36	78.65
Factory	15	71.15	71.03
Factory	10	55.13	54.36
Factory	5	32.18	33.27
Bobble	25	78.91	83.46
Bobble	20	72.44	76.79
Bobble	15	61.54	68.14
Bobble	10	49.62	54.10
Bobble	5	26.03	34.42
Car	20	85.90	85.58
Car	10	74.94	79.87
Car	0	65.51	65.38
Car	-5	43.85	49.87
Car	-10	26.60	27.44

Table B.3: Comparison of different speech features on ISOLET database in various background conditions.

Noise type	SNR [dB]	Word accuracy [%]						
		MFCC	BFCCF	BFCCT	ZCPA	ZC	SSCH	MFCC +SSC
No noise	-	89.55	88.46	86.35	82.24	80.64	86.35	88.72
White	25	76.86	77.37	79.36	78.08	76.03	80.45	72.56
White	20	67.44	66.09	72.44	74.10	70.58	74.10	63.59
White	15	48.33	45.58	57.56	68.14	57.18	61.60	43.85
White	10	17.44	14.81	33.46	54.75	39.29	42.50	15.06
Factory	25	84.29	85.83	84.04	79.49	78.01	84.23	84.04
Factory	20	78.78	80.45	78.65	77.44	74.87	79.87	77.88
Factory	15	66.99	70.64	71.03	71.28	69.23	73.21	64.10
Factory	10	46.35	52.31	54.36	64.55	58.01	58.21	44.55
Factory	5	21.09	31.28	33.27	44.10	38.85	40.71	20.06
Bobble	25	81.65	83.53	83.46	79.36	77.63	82.18	80.83
Bobble	20	73.14	76.92	76.79	75.58	75.06	74.17	71.28
Bobble	15	57.56	63.46	68.14	68.97	68.91	60.32	53.21
Bobble	10	39.04	44.42	54.10	53.72	58.65	41.79	36.79
Bobble	5	22.18	26.35	34.42	33.44	40.19	25.58	21.41
Car	20	81.15	87.18	85.58	82.88	80.58	85.83	79.55
Car	10	69.87	77.95	79.87	82.31	80.00	82.05	62.95
Car	0	46.54	59.81	65.38	76.86	77.24	69.19	39.10
Car	-5	22.37	44.23	49.87	70.13	70.26	55.51	20.77
Car	-10	7.76	18.85	27.44	46.47	48.14	31.41	8.27

Table B.4: Comparison of different speech features on RM database in various background conditions.

Noise type	SNR [dB]	Word accuracy [%]						
		MFCC	BFCCF	BFCCT	ZCPA	ZC	SSCH	MFCC +SSC
No noise	-	95.20	93.17	90.00	85.08	75.91	93.95	94.18
White	30	69.19	66.15	62.59	69.27	50.61	83.91	57.67
White	25	40.61	35.49	38.66	57.28	32.80	72.75	32.10
White	20	12.61	12.26	15.07	34.63	13.67	48.54	8.75
Factory	25	74.41	74.58	68.02	67.63	49.82	83.05	73.84
Factory	20	52.28	52.25	41.66	51.89	31.53	69.15	47.52
Factory	15	22.06	24.76	19.48	27.84	13.32	45.02	19.06
Bobble	25	69.50	68.92	71.73	69.15	51.50	73.92	65.44
Bobble	20	44.83	46.74	51.85	51.74	36.94	53.85	40.84
Bobble	15	20.30	23.08	26.04	27.14	17.61	22.49	18.04
Car	10	64.90	73.68	78.25	69.00	65.48	86.45	55.56
Car	5	43.65	58.73	61.50	55.53	57.71	77.55	34.95
Car	0	20.62	39.59	32.41	33.50	42.62	59.82	20.11

Bibliography

[1] A. Acero, C. Crespo, C. de la Torre, and J. C. Torrecilla, "Robust HMM-based endpoint detector," in *Proc. Eur. Conf. on Speech Commun. and Technol. (EUROSPEECH)*, (Berlin, Germany), pp. 1551–1554, Sept. 1993.

[2] S. M. Ahadi and P. C. Woodland, "Combined bayesian and predictive techniques for rapid speaker adaptation of continuous density hidden Markov models," *Computer Speech and Language*, vol. 11, pp. 187–206, 1997.

[3] W. A. Ainsworth, G. C. M. Fant, O. Fujimura, H. Fujisaki, W. J. Hess, J. N. Holmes, F. Itakura, M. R. Schroeder, and H. W. Strube, "Speech processing by man and machine," in *Recognition of Complex Acoustic Signals*, Life Science Research Report 5, pp. 307–351, 1977.

[4] D. Albesano, R. De Mori, R. Gemello, and F. Mana, "A study of the effect of adding new dimensions to trajectories in the acoustic space," in *Proc. Eur. Conf. on Speech Commun. and Technol. (EUROSPEECH)*, vol. 4, (Budapest, Hungary), pp. 1503–1506, Sept. 1999.

[5] J. B. Allen, "Cochlear modeling," *IEEE ASSP Mag.*, vol. 2, pp. 3–29, Jan. 1985.

[6] J. B. Allen, "How do humans process and recognize speech," *IEEE Trans. Speech Audio Processing*, vol. 2, pp. 567–577, Oct. 1994.

[7] A. Andreou, T. Kamm, and J. Cohen, "Experiments in vocal tract normalization," in *Proc. CIAP Workshop: Frontiers in Speech Recognition II*, 1994.

[8] B. S. Atal, "Automatic speaker recognition based on pitch contours," *J. Acoust. Soc. Am.*, vol. 52, no. 6, pp. 1687–1697, 1972.

[9] B. S. Atal, “Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification,” *J. Acoust. Soc. Am.*, vol. 55, pp. 1304–1312, June 1974.

[10] B. S. Atal and L. S. Hanauer, “Speech analysis and synthesis by linear prediction of the speech wave,” *J. Acoust. Soc. Am.*, vol. 50, pp. 637–655, 1971.

[11] B. S. Atal and M. R. Schroeder, “Adaptive predictive coding of speech signals,” *Bell Syst. Tech. J.*, vol. 49, pp. 1973–1986, 1970.

[12] M. Berouti, R. Schwartz, and J. Makhoul, “Enchancement of speech corrupted by acoustic noise,” in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Processing (ICASSP)*, pp. 913–916, 1979.

[13] S. F. Boll, “Suppression of acoustic noise in speech using spectral subtraction,” *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 27, pp. 113–120, Apr. 1979.

[14] H. Boulard and S. Dupont, “Subband-based speech recognition,” in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Processing (ICASSP)*, (Munich, Germany), pp. 1251–1254, Apr. 1997.

[15] H. Boulard, S. Dupont, H. Hermansky, and N. Morgan, “Towards subband-based speech recognition,” in *Proc. Eur. Signal Processing Conf. (EUSIPCO)*, vol. 3, (Trieste, Italy), pp. 1579–1582, Sept. 1996.

[16] C. Cerisara, J.-P. Haton, J.-F. Mari, and D. Fohr, “Multi-band continuous speech recognition,” in *Proc. Eur. Conf. on Speech Commun. and Technol. (EUROSPEECH)*, vol. 3, (Rhodos, Greece), pp. 1235–1238, Sept. 1997.

[17] C. Cerisara, J.-P. Haton, J.-F. Mari, and D. Fohr, “A recombination model for multi-band speech recognition,” in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Processing (ICASSP)*, vol. 2, (Seattle, USA), pp. 717–720, May 1998.

[18] C. Chesta, O. Siohan, and C. H. Lee, “Maximum a posteriori linear regression for hidden Markov model adaptation,” in *Proc. Eur. Conf. on Speech Commun. and Technol. (EUROSPEECH)*, (Budapest, Hungary), pp. 211–214, Sept. 1999.

[19] W. Chou, “Maximum a posteriori linear regression with elliptically symmetric matrix priors,” in *Proc. Eur. Conf. on Speech Commun. and Technol. (EUROSPEECH)*, (Budapest, Hungary), pp. 1–4, Sept. 1999.

[20] R. A. Cole, Y. K. Muthusamy, and M. Fanty, "The ISOLET spoken letter database," Technical report CSE 90-004, Oregon Graduate Institute of Science and Technology, Beverton, OR, USA, Mar. 1990.

[21] D. V. Compernolle, "Noise adaptation in a hidden Markov model speech recognition system," *Computer Speech and Language*, vol. 3, no. 2, pp. 151–167, 1989.

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

[23] R. De Mori, D. Albesano, R. Gemello, and F. Mana, "Ear-model derived features for automatic speech recognition," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Processing (ICASSP)*, vol. 3, (Istambul, Turkey), pp. 1603–1606, 2000.

[24] J. R. Deller, J. G. Proakis, and J. H. L. Hansen, *Discrete-Time Processing of Speech Signals.* Upper Saddle River, New Jersey: Prentice Hall, 1993.

[25] P. Duchnowski, *A New Structure for Automatic Speech Recognition.* PhD thesis, Massachusetts Institute of Technology, Sept. 1993.

[26] R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis.* New York: Wiley-Interscience, 1973.

[27] S. Dupont and H. Boulard, "Using multiple time scales in a multi-stream speech recognition system," in *Proc. Eur. Conf. on Speech Commun. and Technol. (EUROSPEECH)*, (Rhodos, Greece), pp. 3–6, Sept. 1997.

[28] S. Dupont, H. Boulard, and C. Ris, "Robust speech recognition based on multi-stream features," in *Proc. ESCA-NATO Tutorial and Res. Workshop on Robust Speech Recognition for Unknown Commun. Channels*, (Pont-à-Mousson, France), pp. 95–98, Apr. 1997.

[29] E. Eide and H. Gish, "A parametric approach to vocal tract length normalization," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Processing (ICASSP)*, vol. 1, (Atlanta, USA), pp. 346–348, May 1996.

[30] Étienne Bauche, B. Gajić, Y. Minami, T. Matsuoka, and S. Furui, "Connected digit recognition in spontaneous speech," in *Proc. Eur. Conf. on Speech Commun. and Technol. (EUROSPEECH)*, (Rhodos, Greece), pp. 923–926, Sept. 1997.

[31] C. G. M. Fant, *Speech Sounds and Features.* Cambridge, Mass.: M.I.T. Press, 1973.

[32] J. Flanagan, *Speech Analysis Synthesis and Perception, 2nd ed.* Springer-Verlag, 1972.

[33] H. Fletcher, *Speech and Hearing in Communication.* New York: D. Van Nostrand Company, Inc., 1953.

[34] S. Furui, "Speaker-independent isolated word recognition using dynamic features of speech spectrum," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 34, pp. 52–59, Feb. 1986.

[35] B. Gajić, "Automatic speech detection for speech recognition under adverse conditions," Diploma thesis, Norwegian Institute of Technology, Trondheim, Norway, Apr. 1996.

[36] B. Gajić and K. K. Paliwal, "Robust feature extraction using subband spectral centroid histograms," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Processing (ICASSP)*, vol. 1, (Salt Lake City, USA), pp. 85–88, May 2001.

[37] B. Gajić and K. K. Paliwal, "Robust parameters for speech recognition based on subband spectral centroid histograms," in *Proc. Eur. Conf. on Speech Commun. and Technol. (EUROSPEECH)*, vol. 1, (Aalborg, Danmark), pp. 591–594, Sept. 2001.

[38] B. Gajić and R. C. Rose, "Hidden Markov model environmental compensation for automatic speech recognition on hand-held mobile devices," in *Proc. Int. Conf. on Spoken Language Processing (ICSLP)*, vol. 1, (Beijin, China), pp. 405–408, Oct. 2000.

[39] M. J. F. Gales, "Maximum likelihood linear transformations for HMM-based speech recognition," *Computer Speech and Language*, vol. 12, pp. 75–98, 1998.

[40] M. J. F. Gales and P. C. Woodland, "Mean and variance adaptation within the MLLR framework," *Computer Speech and Language*, vol. 10, pp. 249–264, 1996.

[41] M. J. F. Gales and S. J. Young, "Cepstral parameter compensation for HMM recognition in noise," *Speech Commun.*, vol. 12, pp. 231–239, 1993.

[42] M. J. F. Gales and S. J. Young, "A fast and flexible implementation of parallel model combination," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Processing (ICASSP)*, (Detroit, USA), pp. 133–136, 1995.

[43] M. J. F. Gales and S. J. Young, "Robust speech recognition in additive and convolutional noise using parallel model combination," *Computer Speech and Language*, vol. 9, pp. 289–307, 1995.

[44] M. J. F. Gales and S. J. Young, "Robust continuous speech recognition using parallel model combination," *IEEE Trans. Speech Audio Processing*, vol. 4, pp. 352–359, Sept. 1996.

[45] J.-L. Gauvain and C.-H. Lee, "Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains," *IEEE Trans. Speech Audio Processing*, vol. 2, pp. 291–298, Apr. 1994.

[46] O. Ghitza, "Robustness against noise: The role of timing-synchrony measurement," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Processing (ICASSP)*, (Dallas, USA), pp. 2372–2375, Apr. 1987.

[47] 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. 55–76, Jan. 1988.

[48] O. Ghitza, "Auditory nerve representation as a basis for speech processing," in *Advances in Speech Signal Processing* (S. Furui and M. Sondhi, eds.), ch. 15, pp. 453–485, Marcel Dekker, Inc., 1992.

[49] O. Ghitza, "Auditory models and human performance in tasks related to speech coding and speech recognition," *IEEE Trans. Speech Audio Processing*, vol. 2, pp. 115–132, Jan. 1994.

[50] L. Gillick and S. J. Cox, "Some statistical issues in the comparison of speech recognition algorithms," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Processing (ICASSP)*, (Glasgow, Scotland), pp. 532–535, May 1989.

[51] S. Greenberg (editor) *J. Phonetics*, vol. 16, Jan. 1988. (theme issue "Representation of Speech in the Auditory Periphery").

[52] R. Haeb-Umbach, D. Geller, and H. Ney, "Improvements in connected digit recognition using linear discriminant analysis and mixture densities," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Processing (ICASSP)*, pp. 239–242, 1993.

[53] R. Haeb-Umbach and H. Ney, “Linear discriminant analysis for improved large vocabulary continuous speech recognition,” in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Processing (ICASSP)*, (San Francisco, USA), pp. 13–16, Mar. 1992.

[54] J. A. Haigh and J. S. Mason, “Robust voice activity detection using cepstral features,” in *Proc. IEEE Region 10 Conf. on Digital Signal Processing (TENCON)*, pp. 321–324, 1993.

[55] E. Harborg, *Hidden Markov Models Applied to Automatic Speech Recognition.* PhD thesis, Norwegian Institute of Technology, Trondheim, Norway, Aug. 1990.

[56] H. Hermansky, “Perceptual linear predictive (PLP) analysis of speech,” *J. Acoust. Soc. Am.*, vol. 87, pp. 1738–1752, Apr. 1990.

[57] H. Hermansky, S. Tibrewala, and M. Pavel, “Towards ASR on partially corrupted speech,” in *Proc. Int. Conf. on Spoken Language Processing (ICSLP)*, (Philadelphia, USA), pp. 462–465, Oct. 1996.

[58] X. Huang, A. Acero, and H.-W. Hon, *Spoken Language Processing: a Guide to Theory, Algorithm, and System Development.* Upper Saddle River, New Jersey: Prentice Hall PTR, 2001.

[59] M. Hunt, S. Richardson, D. Bateman, and A. Piau, “An investigation of PLP and IMELDA acoustic representations and of their potential for combination,” in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Processing (ICASSP)*, (Toronto, Canada), pp. 881–884, May 1991.

[60] M. J. Hunt and C. Lefèbvre, “A comparison of several acoustic representations for speech recognition with degraded and undegraded speech,” in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Processing (ICASSP)*, (Glasgow, Scotland), pp. 262–265, May 1989.

[61] C. Jankowski, “A comparison of auditory models for automatic speech recognition,” Master’s thesis, Massachusetts Institute of Technology, May 1992.

[62] J.-C. Junqua, *Toward Robustness in Isolated-Word Automatic Speech Recognition.* PhD thesis, Univ. Nancy I, Nancy, France, 1989.

[63] J.-C. Junqua and J.-P. Haton, *Robustness in Automatic Speech Recognition – Fundamentals and Applications.* Kluwer Academic Publishers, 1996.

[64] J.-C. Junqua, B. Mak, and B. Reaves, "A robust algorithm for word boundary detection in the presence of noise," *IEEE Trans. Speech Audio Processing*, vol. 2, pp. 406–412, July 1994.

[65] J.-C. Junqua, H. Wakita, and H. Hermansky, "Evaluation and optimization of perceptually-based ASR front-end," *IEEE Trans. Speech Audio Processing*, vol. 1, no. 1, pp. 39–48, 1993.

[66] S. Kajita and F. Itakura, "Subband-autocorrelation analysis and its application for speech recognition," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Processing (ICASSP)*, vol. 2, pp. 193–196, 1994.

[67] S. Kajita and F. Itakura, "Robust speech feature extraction using SBCOR analysis," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Processing (ICASSP)*, vol. 1, (Detroit, USA), pp. 421–424, May 1995.

[68] S. M. Kay, *Modern Spectral Estimation: Theory and Application.* Englewood Cliffs, New Jersey: Prentice Hall, 1988.

[69] B. Kedem, "Spectral analysis and discrimination by zero-crossings," *Proc. IEEE*, vol. 74, pp. 1477–1493, Nov. 1986.

[70] 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. Speech Audio Processing*, vol. 7, pp. 55–69, Jan. 1999.

[71] L. F. Lamel, R. H. Kassel, and S. Seneff, "Speech database development: Design and analysis of the acoustic-phonetic corpus," in *Proc. DARPA Speech Recogn. Workshop*, pp. 100–109, Feb. 1986.

[72] L. F. Lamel, L. R. Rabiner, A. E. Rosenberg, and J. G. Wilpon, "An improved endpoint detector for isolated word recognition," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 29, pp. 777–785, Aug. 1981.

[73] L. Lee and R. C. Rose, "Speaker normalization using efficient frequency warping procedures," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Processing (ICASSP)*, vol. 1, (Atlanta, USA), pp. 353–356, May 1996.

[74] L. Lee and R. C. Rose, "A frequency warping approach to speaker normalization," *IEEE Trans. Speech Audio Processing*, vol. 6, pp. 49–60, Jan. 1998.

[75] C. J. Leggetter and P. C. Woodland, "Flexible speaker adaptation for large vocabulary speech recognition," in *Proc. Eur. Conf. on Speech Commun. and Technol. (EUROSPEECH)*, (Madrid, Spain), pp. 1155–1158, Sept. 1995.

[76] C. J. Leggetter and P. C. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models," *Computer Speech and Language*, vol. 9, pp. 171–185, Apr. 1995.

[77] P. Lockwood and J. Boudy, "Experiments with a nonlinear spectral subtraction (NSS), hidden Markov models and projection, for robust speech recognition in cars," *Speech Commun.*, vol. 11, no. 2–3, pp. 215–228, 1992.

[78] J. Makhoul, "Linear prediction: A tutorial review," *Proc. IEEE*, vol. 63, pp. 561–580, Apr. 1975.

[79] J. D. Markel and A. H. Gray Jr., *Linear Prediction of Speech.* Springer-Verlag, 1976.

[80] S. L. Marple, *Digital Spectral Analysis with Applications.* Englewood Cliffs, New Jersey: Prentice Hall, 1987.

[81] Y. Minami and S. Furui, "Adaptation method based on HMM composition and EM algorithm," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Processing (ICASSP)*, (Atlanta, USA), pp. 327–330, May 1996.

[82] S. Molau, S. Kanthak, and H. Ney, "Efficient vocal tract normalization in automatic speech recognition," in *Proc. ESSV*, (Cottbus, Germany), pp. 209–216, Sept. 2000.

[83] D. O'Shaughnessy, *Speech Communication: Human and Machine.* Addison-Wesley, 1987.

[84] K. K. Paliwal, "Spectral subband centroid features for speech recognition," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Processing (ICASSP)*, vol. 2, (Seattle, USA), pp. 617–620, May 1998.

[85] J. W. Picone, "Signal modelling techniques in speech recognition," *Proc. IEEE*, vol. 81, pp. 1214–1247, Sept. 1993.

[86] P. Price, W. M. Fisher, J. Bernstein, and D. S. Pallett, "The DARPA 1000-word resource management database for continuous speech recognition," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Processing (ICASSP)*, vol. 1, (New York, USA), pp. 651–654, Apr. 1988.

[87] D. Pye and P. C. Woodland, "Experiments in speaker normalization and adaptation for large vocabulary speech recognition," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Processing (ICASSP)*, vol. 2, (Munich, Germany), pp. 1047–1050, Apr. 1997.

[88] L. R. Rabiner and M. R. Sambur, "An algorithm for determining the endpoints of isolated utterances," *Bell Syst. Tech. J.*, vol. 54, pp. 297–315, Feb. 1975.

[89] L. Rabiner and B.-H. Juang, *Fundamentals of Speech Recognition.* Englewood Cliffs, New Jersey: Prentice Hall, 1993.

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

[91] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proc. IEEE*, vol. 77, pp. 257–286, Sept. 1989.

[92] A. E. Rosenberg, C.-H. Lee, and F. K. Soong, "Cepstral channel normalization techniques for HMM-based speaker verification," in *Proc. Int. Conf. on Spoken Language Processing (ICSLP)*, (Yokohama, Japan), pp. 1835–1838, 1994.

[93] H. W. Ruehl, S. Dobler, J. Weith, P. Meyer, A. Noll, H. H. Hamer, and H. Piotrowski, "Speech recognition in the noisy car environment," *Speech Commun.*, vol. 10, no. 1, pp. 11–22, 1991.

[94] S. Sandhu and O. Ghitza, "A comparative study of mel cepstra and EIH for phone classification under adverse conditions," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Processing (ICASSP)*, (Detroit, USA), pp. 409–412, May 1995.

[95] M. H. Savoji, "A robust algorithm for accurate endpointing of speech signals," *Speech Commun.*, vol. 8, pp. 45–60, Mar. 1989.

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

[97] K. Shinoda and C. H. Lee, "Structural MAP speaker adaptation using hierarchical priors," in *Proc. IEEE Workshop on Automatic Speech Recognition and Understanding*, (Santa Barbara, USA), pp. 381–388, Dec. 1997.

[98] O. Siohan, "On the robustness of linear discriminant analysis as a preprocessing step for noisy speech recognition," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Processing (ICASSP)*, (Detroit, USA), pp. 125–128, May 1995.

[99] O. Siohan, T. A. Myrvoll, and C. H. Lee, "Maximum a posteriori regression for fast HMM adaptation," in *Proc. ISCA Tutorial and Res. Workshop ASR2000*, (Paris, France), pp. 120–127, 2000.

[100] S. S. Stevens, "On the psychophysical law," *Psychol. Rev.*, vol. 64, pp. 153–181, 1957.

[101] S. S. Stevens and J. Volkmann, "The relation of pitch of frequency: A revised scale," *Am. J. Psychol.*, vol. 53, pp. 329–353, 1940.

[102] S. Theodoridis and K. Koutroumbas, *Pattern Recognition.* Academic Press, 1999.

[103] S. Tibrewala and H. Hermansky, "Multi-band and adaptation approaches to robust speech recognition," in *Proc. Eur. Conf. on Speech Commun. and Technol. (EUROSPEECH)*, (Rhodos, Greece), pp. 2619–2622, Sept. 1997.

[104] S. Tibrewala and H. Hermansky, "Sub-band based recognition of noisy speech," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Processing (ICASSP)*, (Munich, Germany), pp. 1255–1258, Apr. 1997.

[105] S. Tsuge, T. Fukada, and H. Singer, "Speaker normalized spectral subband parameters for noise robust speech recognition," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Processing (ICASSP)*, (Phoenix, USA), May 1999.

[106] R. Tucker, "Voice activity detection using a periodicity measure," *IEE Proceedings*, vol. 139, pp. 377–380, Aug. 1992.

[107] A. Varga and H. Steeneken, "Assessment for automatic speech recognition: II. NOISEX-92: A database and an experiment to study the effect of additive noise on speech recognition systems," *Speech Commun.*, vol. 12, pp. 247–251, 1993.

[108] G. von Békésy, *Experiments in Hearing.* McGraw-Hill Book Company, Inc., 1960.

[109] H. Wakita, “Normalization of vowels by vocal-tract length and its application to vowel identification,” *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 25, pp. 183–192, Apr. 1977.

[110] C. J. Wellekens, J. Kangasharju, and C. Milesi, “The use of meta-HMM in multistream HMM training for automatic speech recognition,” in *Proc. Int. Conf. on Spoken Language Processing (ICSLP)*, (Sidney, Australia), pp. 2991–2995, Dec. 1998.

[111] J. G. Wilpon and L. R. Rabiner, “Application of hidden Markov models to automatic speech endpoint detection,” *Computer Speech and Language*, vol. 2, pp. 321–341, 1987.

[112] J. G. Wilpon, L. R. Rabiner, and T. Martin, “An improved word-detection algorithm for telephone-quality speech incorporating both syntactic and semantic constraints,” *Bell Syst. Tech. J.*, vol. 63, pp. 479–497, Mar. 1984.

[113] P. C. Woodland, M. J. F. Gales, and D. Pye, “Improving environmental robustness in large vocabulary speech recognition,” in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Processing (ICASSP)*, vol. 1, (Atlanta, USA), pp. 65–68, May 1996.

[114] S. Young, D. Kershaw, J. Odell, D. Ollason, V. Valtchev, and P. Woodland, *The HTK Book*. Microsoft Corporation, 2000.

[115] P. Zhan and M. Westphal, “Speaker normalization based on frequency warping,” in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Processing (ICASSP)*, vol. 2, (Munich, Germany), pp. 1039–1042, Apr. 1997.

[116] E. Zwicker and E. Terhart, “Analytical expressions for critical-band rate and critical bandwidth as a function of frequency,” *J. Acoust. Soc. Am.*, vol. 68, pp. 1523–1525, 1980.