Summer 2001

Minimum Mean Square Error Spectral Peak Envelope Estimation for Automatic Vowel Classification

Jaishree Venugopal

Old Dominion University

MINIMUM MEAN SQUARE ERROR

SPECTRAL PEAK ENVELOPE ESTIMATION

FOR AUTOMATIC VOWEL CLASSIFICATION

by

Jaishree Venugopal
B.E. April 1997, Bharathiar University, India

A Thesis Submitted to the Faculty of
Old Dominion University in Partial Fulfillment
Of the Requirement for the Degree of

MASTER OF SCIENCE

ELECTRICAL ENGINEERING

OLD DOMINION UNIVERSITY
August 2001

Approved by:

Stephen A. Zahorian (Director)

Lee A. Belfore, II (Member)

Amin N. Dharamsi (Member)

ABSTRACT

MINIMUM MEAN SQUARE ERROR SPECTRAL PEAK ENVELOPE ESTIMATION FOR AUTOMATIC VOWEL CLASSIFICATION

Jaishree Venugopal
Old Dominion University, 2001
Director: Dr. Stephen A. Zahorian

Spectral feature computations continue to be a very difficult problem for accurate machine recognition of speech. In this work, which focuses on vowels, a new spectral peak envelope method for vowel classification is developed, based on a missing frequency components model of speech recognition. According to the missing frequency components model, vowel recognition depends only on the spectral (harmonic) peaks. Smoothing and interpolation of the spectra, performed in the standard cepstral analysis method commonly used in automatic speech recognition, actually loses valuable information and results in reduced recognition accuracy. The new method for feature extraction presented in this thesis is based on minimum mean square error curve fitting of cosine-like basis vectors to all peaks in the speech spectrum. A mathematical model for smoothly tracking spectral envelopes using only spectral peak information and ignoring other parts of the spectrum is presented. A software algorithm in Matlab for the model was developed and tested for various speaker types using a neural network classifier. Vowel classification experiments were conducted based on the features derived from the spectral peaks. The classification rates of the peak method under various signal to noise ratios was also studied. The basic conclusion is that the new features perform about the same as cepstral (also referred to as DCTCs, or Discrete Cosine Transform Coefficients) features for clean speech, but have advantages when the signal is degraded by noise.

I dedicate this thesis to my parents

ACKNOWLEDGMENTS

This thesis would not have been possible without the continuous guidance, patience and motivation of my mentor, Dr. Stephen A. Zahorian. I am also grateful to Dr. Amin Dharamsi and Dr. Lee Belfore for graciously agreeing to be on my committee and for their generous assistance and time.

I would like to thank all my family members for their love and support in completing this thesis. I would also like to express my appreciation and gratitude to all my friends for their assistance. Also, thanks to the National Science Foundation for partially funding this work under grant # BES- 9977260.

TABLE OF CONTENTS

LIST OF TABLES

Table Page

CHAPTER I
INTRODUCTION

1.1 Prelude

One of the greatest mysteries that has remained unsolved for centuries is the understanding of the most complex wonder - human speech. Researchers have been working on automatic speech recognition technology for almost fifty years with only partial success. Recent breakthroughs in desktop computing power, improvements in speech algorithms, and advances in signal processing methods have accelerated the developments in speech recognition and processing. The applications are innumerable, ranging from home and education to hospitals and industries.

Speech processing is an interdisciplinary field that involves acoustics, probability theory, linear system theory, physiology, psychology, linear algebra, digital signal processing, computer science, and linguistics [4]. Automatic Speech Recognition (ASR) is defined as the process of interpreting human speech to be used as a communication mechanism. There is an insatiable need of efficient signal processing for speech recognition and perception algorithms to extract the information from speech. Two closely related problem areas in the speech field are classification and recognition. Classification is generally understood to mean automatic determination of a category, given N choices, usually based on segments of speech signals with endpoints already determined, perhaps manually. Recognition is the more open-ended process of converting speech to text, with very few limitations and no manual processing of the data. Neural networks [3] and Hidden Markov Models (HMMs) [3] are typical classifiers and recognizers respectively. The following sections briefly discuss the theory of ASR and also introduce neural networks and HMMs. The objective of the work is explained and related to recent references from the literature. This chapter also gives an overview of the issues covered in further chapters.

1.2 Background

Speech is produced by oscillations of the vocal cords, accompanied by air under pressure in the lungs, which generates pulses or a constant stream of air that resonates through the vocal tract. The repetition rate of the pulses is called the fundamental frequency and is commonly referred to as the pitch. The harmonic-rich pulses of air are then modulated or filtered by the various organs of the vocal tract. The vocal tract, which acts as an acoustical tube, has natural frequencies due to its shape. These natural frequencies or formants are generally considered to be the most important characteristics of the speech signal as it leaves the mouth [3].

The two main subdivisions of the study of speech production and perception are articulatory phonetics and acoustic phonetics [4]. Articulatory phonetics emphasizes the anatomical knowledge of the speech organs for describing and classifying speech sounds. Acoustic phonetics is based on the observable, measurable characteristics in the waveforms of speech sounds--especially those that enable them to be distinguished from one another. It provides a theoretical and experimental background for speech recognition by electronic hardware and computer algorithms.

ASR models are classified into two categories--namely, language models, which are based on articulatory phonetics and acoustic models, which are based on acoustic phonetics. A block diagram of an ASR system is shown in Figure 1. The speech utterance is the input for the signal processor block. Signal processing is used to extract the features or characteristics from the signal that are necessary for differentiating phonemes. Linear prediction [16] and cepstral analysis [3] are two of the most successful analytical techniques currently used in speech processing. Both of these techniques are based primarily on signal processing methods, which have been shown to work well for ASR, but are not fundamentally unique to speech.

Other approaches to ASR are more directly related to the theory of speech science. One such approach is based on distinctive features, which are a set of attributes, which may contribute to form a phoneme—the smallest linguistic unit. Each attribute is an acoustical characteristic, independent of all the others. Thus each phoneme can be regarded as a bundle of distinctive features. Temporal/spectral features describe the acoustic properties of the speech waveform in the time/frequency domain. Hence there is a need for signal processing in time/frequency domain. Examples for temporal features include positive and negative peak amplitudes and positions, peak to previous peak and peak to valley measurements. Nasality, frication, voice-unvoiced classification, and spectral peaks are some of the spectral features [4].

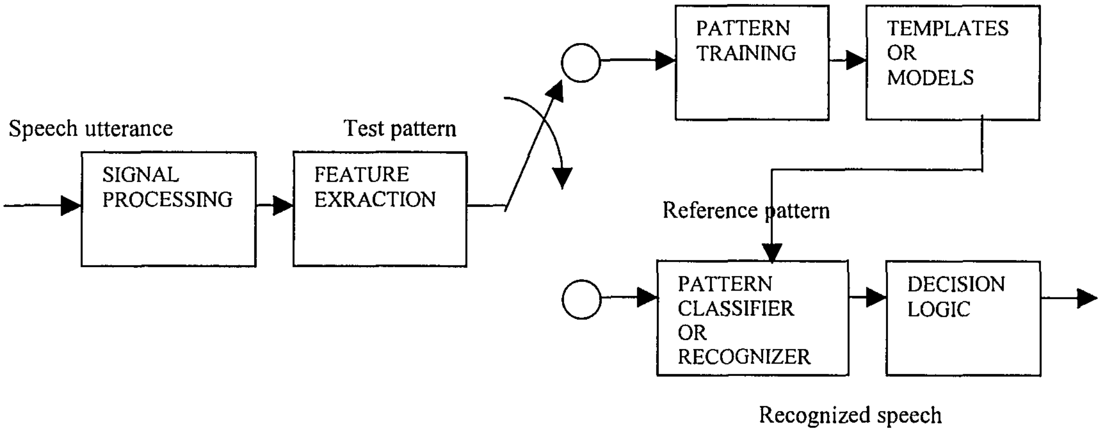


Figure 1. Block diagram of a speech recognizer (After – Fundamentals of Speech Recognition by Lawrence Rabiner and Biing-Hwang Juang) [3]

There are two phases in the recognizer, the training phase and the testing phase. Initially the recognizer is trained to "define" templates or models. These models are used in the testing phase to recognize or classify the unknown phoneme or vowel. Statistical methods are used to construct the models. Based on the similarity distance between the test pattern and reference pattern, the decision logic identifies which reference pattern best matches the test pattern.

ASR systems are classified according to the following considerations:

- Kind of speech: phone, isolated word, connected word, continuous speech

- Vocabulary size: small (up to 100 word), medium (up to 1000 words), large (more than 1000 words)
- Speaker dependence or speaker independence
- Noise: continuous environmental noise, impulsive environmental noise, continuous channel noise, competing speech, dialing noise, cross-talk noise, switching noise
- Channel bandwidth: telephone band width (300 – 3400 Hz), wide-band
- Users: single user, specified group of users, or an unknown population

1.3 Introduction to Neural Networks and HMMs

Neural Networks:

Classification assigns the test pattern to one of a relatively small number of specified reference patterns whereas recognition identifies a test pattern or rejects it as not to belong to any of the reference models. Classification assumes endpoints (points that determine the beginning and end of the acoustic signal) are already known and timing is not a big issue. Neural networks are typical classifiers. Recognition is more open ended since endpoints are not known in advance and timing is an important parameter. The HMM method provides a natural and highly reliable way of recognizing speech for a wide range of applications. A combination of neural networks and HMMs can be used as a pattern recognizer. Recognition is usually applied to whole words or sentences unlike classification, which is typically for a restricted phoneme set such as vowels or stops. Since a recognizer also needs to perform classification, a classifier can be a part of a recognizer.

A neural net, also called a connectionist model or a parallel-distributed processing model, is basically a dense interconnection of simple, nonlinear, computational elements called nodes or neurons. Neurons have N inputs $x_1, x_2 ..... x_N$ which are summed with weights $w_1, w_2 ..... w_N$ thresholded, and nonlinearly compressed [3] to give the output y, defined as

$$y = f\left(\sum_{i=1}^{N} w_i x_i - \phi \right) - - - - - - - - - - - \langle 1 \rangle$$

whereϕ is an internal threshold or offset and f is a nonlinear function [3]. The sigmoid nonlinearities are used most often because they are continuous and differentiable [3].

An Artificial Neural Network (ANN) is an arbitrary connection of simple computational elements as defined by equation $\langle 1 \rangle$. One of the methods of interconnection of simple computational elements (Network Topology) is single/multilayer perceptrons. In a perceptron, the neurons are grouped in layers, an input layer, one or more hidden layers and an output layer. The inputs of the neurons in a layer are the outputs of the neurons from the previous layer, except the input layer. The outputs of neurons in the output layer are the outputs of the neural network. The layers between the input and output layers are the hidden layers, and there may be one or more according to the application.

The neural network can also be viewed as a non-linear mapping of the input to the output space. The selection of inputs to an ANN is directly related to the choice of the features for any pattern classification system. The number of hidden layers and the nodes in the hidden layer affect the accuracy of the network. If the hidden layers are large, training becomes difficult due to the estimation of too many parameters. The network may not be able to accurately classify the input patterns if the hidden layers are too small. The training procedure chooses the values for the interconnecting weights and the offset. The exact choice of the nonlinearity is not very important in terms of the network performance. However, f must be continuous and differentiable for the training algorithm to be applicable.

HMMs:

For about the last two decades, state of art complete ASR systems has been based on Hidden Markov Models (HMMs) [3]. A hidden Markov model is a collection of states connected by a transition matrix. It begins with an initial state. In each time step, a transition is taken into a new state, and an output symbol is then generated in that state. The choice of initial state, transition, and output symbol are randomly governed by probability distributions. A classical example of a hidden Markov process is represented by an urn-and-ball model [17]. In this example, a person takes balls (output symbols) with different colors from different urns (states) randomly. These urns are behind a curtain. An observer can see a sequence of balls with different colors, but she/he cannot tell from which urn this ball was taken out because the process of urn selection was hidden behind the curtain. In a hidden Markov model, the output symbol sequence generated over time can be observed, but the sequence of states visited is hidden from a viewer. The decision is made based on the probability (maximum) of input sequence at the condition of an observed sequence.

1.4 Objectives of this thesis

The general objective of this work is to investigate a novel method for extracting spectral envelope features, which could be used to design and implement a robust vowel classifier. This objective is based on earlier research, which has shown that feature selection, extraction and representation are the keys for better classification [8]. Considerable research has shown that vowel features should be derived from the short time spectrum of the vowels, most likely emphasizing the peaks in the spectrum. However, signal processing for feature extraction has proved to be a most challenging task. There are many unresolved issues, and room for improvement remains.

Ever since the time of Peterson and Barney [18], the first three formants (F1, F2 and F3) have been regarded as the primary source of spectral information. Since then many successful models were developed to implement vowel perception using spectral formants as the fundamental feature set. However, in practical automatic speech recognizers, robust formant tracking is very difficult and thus not often used. Motivated by the idea that vowel information is primarily contained in the spectral peaks, but wanting a more robust method than formant tracking, Douglas B. Paul (1981) [2] tracked the spectral peaks by first computing the fundamental frequency. He then linearly interpolated between these peaks in the frequency domain to derive the spectral envelope.

Very recently, a new model for vowel identification was proposed in Alain de cheveigne & Hideki Kawahara [1]. They argued against smoothing and interpolation of the spectrum since it attempts to guess missing samples based on a predefined model and thus may be misleading. According to them, vowel identification is a process of pattern recognition where matching is restricted to available data, and missing data are ignored using an F0-dependent weighting function that emphasizes regions near harmonics. Their theoretical arguments were based mainly on human perceptual considerations, and they gave no real method for testing their theory in the context of an automatic vowel classifier or recognizer.

The specific objectives of this thesis are to:

1. Present a mathematical model for smoothly tracking spectral envelopes using only spectral peak information, and ignoring other parts of the spectrum.
2. Present and illustrate the mathematical model from 1, in several variations, using speech signals.
3. Conduct vowel classification experiments based on the features derived from the spectral peak features for assessing the suitability of these features for vowel classification.
4. To support or refute the theory of Alain de cheveigne & Hideki Kawahara, within the context of automatic vowel classification.

1.5 Overview of the following chapters

This chapter briefly discussed the basics of speech recognition and its implementation. The objectives of the thesis were presented. Its implementation is discussed in detail in the succeeding chapters.

Chapter two discusses the motivation for the spectral envelope estimation method using spectral peaks. It also covers the theory of standard spectral estimation method (standard DCTC method) and the theory of spectral peak envelope estimation method with references. It also justifies the use of formants as spectral features and the missing data model of vowel identification with references.

Chapter three deals with the algorithm, its software implementation, practical problems faced, and their solutions. Chapter four summarizes the testing of the algorithm with a speech database and relates the results to the objectives of the thesis. Chapter five gives the conclusion and mentions several topics for further expansion of the work.

CHAPTER II
BACKGROUND

2.1 Introduction

Chapter 1 gave an introduction to automatic speech recognition, classification and also discussed classification and recognition methods. The objective of this work is based on the missing data model of vowel identification. This chapter begins with a summary of the standard spectral shape method for speech feature extraction, including implementation algorithms, and some limitations. The next section is a discussion of the theory of the missing data model. The focus is the frequency-domain version of the model. According to this theory, the spectral peaks, which have high energy, have most of the information required for recognition. This chapter also summarizes another spectral peak envelope estimation method from the literature, with specific reference to a paper by Douglas B. Paul [2]. An introduction to Matlab along with its features is also given in this chapter.

2.2 Spectral Shape Method & its Limitations

The application of cepstral analysis to process speech signals, (Oppenheim 1969b; Schafer and Rabiner 1970) was a turning point for speech processing. In a related work, series of experiments were performed on spectral shape representation of vowel spectra by Plomp et al., 1967, Pols et al., 1969 and Klein et al., 1970. It defined a principal-components spectral shape representation of vowel spectra and demonstrated that vowels could be classified automatically as accurately from a principal-components representation as from a formant representation [9]. Both cepstral methods and principal-components analysis result in a complete spectral shape representation.

To have a quantitative description of a signal, it is necessary to represent the signal in terms of explicit functions whose numerical values are exactly defined. For mathematical convenience, a signal is represented as a linear combination of a set of elementary basis functions. The basis functions may be based on time or frequency. The property required to select the weighting coefficients in the linear combination of basis functions is called the Finality of Coefficients [7]. To achieve this, the basis functions should be orthogonal or orthonormal over the interval for which the representation is to be valid. Sinusoidal functions are extremely useful in such analysis since they remain sinusoidal after various mathematical operations such as the sum, difference, derivative or integration are performed. A periodic signal can be decomposed in to a sum of sinusoids ($\cos(\omega_0 t + \phi)$) that are harmonically related and aperiodic signals such as the speech signal can be decomposed in to a continuum of sinusoids having infinitesimal amplitudes [7]. The Visual Speech Display (VSD) system developed in Old Dominion University's speech lab uses DCT basis vectors over frequency (see Figure. 2) as the feature set for vowel classification. One reason for using the Discrete Cosine Transform (DCT) is because of its high information packing ability and less computational complexity compared to other transforms.

Studies in our laboratory (Nossair and Zahorian, 1991; Zahorian and Jagharghi, 1993 [9]; Correal, 1994 [12]; Nossair et. al., 1995) showed that the DCTC method could be implemented with a computationally efficient FFT-based signal processing method, and still function similarly to the human auditory system. It also gave classification results comparable to those obtained with complex auditory models.

The Discrete Cosine Transform Coefficients (DCTCs) of the log magnitude spectrum are one type of global spectral shape features. Zahorian and Gordy (1983) [13] showed that a series cosine basis vector representation is very similar to a principal-components representation. Zahorian and Jagharghi (1993) [9] experimentally compared vowel classification test results between formants and DCTCs, and found that the DCTC results were significantly higher (3.5%) than for the formant case, provided enough DCTCs (ten or more) were used.

Earlier research of Zahorian and Jagharghi, 1990 [15] shows that a logarithmic amplitude scaling of the spectrum envelope without any frequency scaling was found to be similar to the traditional cepstral coefficients. The log magnitude spectrum was also found to show the lower magnitude details clearly. A smoothed magnitude spectrum can be obtained by reconstructing the Discrete Cosine Transform Coefficients (DCTCs) with the degree of smoothing varying with the number of DCTCs. The spectrum envelope is constructed using DCTCs computed over the entire frequency range. Note that the global representation of the magnitude spectrum of speech de-emphasizes the spectral peaks by smoothing the peaks and valleys of the magnitude spectrum.

The DCTC speech analysis method used in ODU's speech lab is implemented as follows:

1. In frame-level processing, the speech signal is divided into overlapping frames. A typical frame size is 30 ms, with an overlap of 15 ms.
2. A window is applied to each frame to reduce the effects caused due to segmentation. The window is usually a smoothly tapered window, such as a Hamming or Hanning window.
3. For each frame of speech, the log-magnitude of the FFT is computed.
4. The underlying basis vectors used to represent the log magnitude spectrum are cosine basis vectors over frequency, specifically integer multiples of a half-cycle of a cosine, defined as

$$\phi_k(n) = \cos\left(\frac{\pi\,(n-0.5)(k-1)}{N} \right) \text{-----------------} \langle 2 \rangle$$

$$0 \le k \le N-1$$

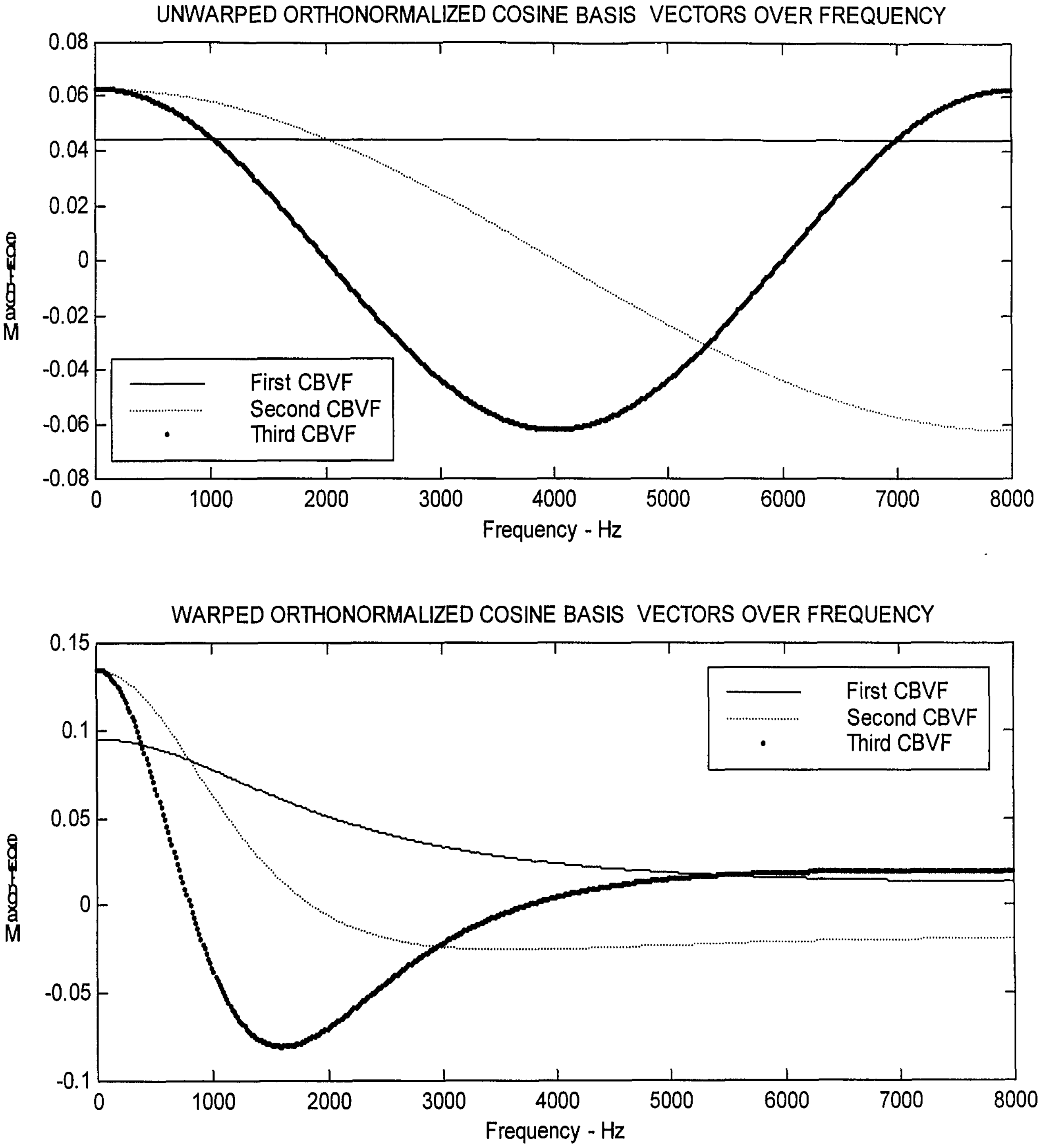


Figure 2. a. (Top) Plot of Unwarped & b. (Bottom) Warped Orthonormalized CBV's over frequency

Non-uniform resolution in representations was found to improve recognition accuracy for automatic speech recognition [8]. This effect is related to the properties of human hearing, which is approximated by a nonlinear frequency scale, with a bilinear frequency function of the form

$$f' = f + \frac{1}{\pi} \tan^{-1}\left(\frac{\alpha \sin(2\pi f)}{1 - \alpha \cos(2\pi f)} \right) ---------- \langle 3 \rangle$$

Equation$\langle 3 \rangle$ is the warping function over frequency. The parameter, α called the warping factor, controls the degree of warping. α of 0.45 has been found to approximate the frequency scale for human hearing, and thereby increasing classification accuracy. Zahorian and Jagharghi (1990) have shown that this frequency "warping" can be efficiently realized by modifying the basis vectors. Figure 2.a. shows the first three basis vectors without warping and 2.b. shows the first three basis vectors modified to accommodate the non-uniform frequency resolution. These are the forms of the basis vectors used in this work. Warping is incorporated into the basis vectors for easy computation.

5. The DCTCs, or spectral shape features, are computed as dot products of each basis vector in step 4, with log-magnitude spectral vector from step 3.

$$DCTC(n) = \sum_{k=0}^{N-1} X(k)\, \phi_k(n) ---------------- \langle 4 \rangle$$

Where $X(k)$ is the log magnitude spectral array

6. These DCTCs are then the parameters, or features, used for recognition. Typically 10-15 terms are used for each speech frame.

The main potential limitation or drawback is that all sections of the spectrum are weighted equally. Thus valleys in the spectrum are just as important as peaks. This ignores evidence that peaks should be more highly weighted. One of the consequences of this more "complete" representation is that a much higher dimensionality feature space is needed relative to those representations, which focus on spectral peaks, such as the formant method summarized in the next section.

2.3 Formants as acoustic features for vowels

As mentioned earlier, the resonant frequencies of the vocal tract are the formants of the vowel during the production of that particular vowel. They are bands of high energy in the frequency domain representation. They can be seen as dark bands in the spectrogram illustrated in Figure 11. Formants represent the most immediate source of articulatory information. Peterson and Barney (1952) [18] first introduced formants as primary features in speech recognition.

Many researchers used formant synthesis to examine and determine the role of formant frequencies in the perception of vowels. In the study by Carlson, Fant and Granstorm (1975), subjects were able to successfully achieve matching of Swedish vowels by a two-formant approximation. Miller (1989) developed the auditory-perceptual theory, which was based on formant-ratio theory, and demonstrated that the preliminary target zones in formant space could be used to classify a proprietary database of American English vowels with up to 93% accuracy.

Although the formant representation of speech spectra is used widely, it has major drawbacks. Bladon (1982) argued against the formant representation of speech primarily with regard to three aspects: first, formant representation is an incomplete spectral description; second, there is a great difficulty in locating the formants in many cases; third, a formant representation does not provide a good prediction when the spectral peaks are widely spaced.

F1, F2 & F3 (formant 1, formant 2 & formant 3 respectively) are the first three major peaks in the spectral envelope, corresponding to the first three resonances of the vocal track. These formants are critical to the perception of the speech sound and contribute to the identification of the phonemic category to which the sound belongs. F3 is used in the perception of labial, alveolar and velar stops as well as the allophonic qualities of the phoneme. The formant frequencies vary depending on the adjacent phonemes in continuously spoken utterances [4].

The identity of a vowel depends on the shape of the spectral envelope, especially the position of the first two or three formants. The harmonic structure of the spectral representation interferes with the determination of the spectral envelope. A later section discusses a vowel identification model, which uses the spectral peaks for estimating the spectral envelope.

2.4 Missing data vowel identification model

According to the frequency-domain version of the missing-data vowel identification model, mentioned in chapter 1, important spectral information is lost due to. smoothing and interpolation. It also leads to F0-dependent distortion. This can be avoided by applying a nonuniform weighting function to the unsmoothed representation derived from the waveform. The following steps involve the working of the frequency-domain model: Estimate the short-term spectra and calculate the spectral weighting function that emphasizes spectral peaks. The short-term spectrum is then compared to all vowel templates using the weighting function. The template that yields the smallest distance determines the vowel that is identified. The weighting function $W(f)$ and the spectral distance $D(T, T_i)$ from the target to template T might be defined as T_i

$$D(T, T_i) = \int (T(f) - T_i(f))^2 \, W(f) \, df ----------- \langle 5 \rangle$$

where $T(f)$ is the short-term spectrum and $T_i(f)$ is the spectral envelope of the ith vowel. The nonuniform weighting function should be 1.0 at the peak frequencies and 0.0 elsewhere. A specific algorithm for feature calculations, which makes use of $\langle 5 \rangle$ is discussed in chapter 3.

The authors of [1] discussed the effects of fundamental frequency on the model. They conclude that the effects are small but an orderly relation exists between F0 and vowel quality. Complete insensitivity to F0 is not desirable in vowel perception models. They also argue that an increase in F0 decreases the intelligibility of the spectrum. Schemes based on individual harmonics or their weighted sums give an incomplete solution to harmonic structure problems. The model is based on spectral samples that may not coincide with formant peaks. It forms a pattern, which has an F0 sampled spectrum and this pattern is to be compared with incoming patterns. Unlike Bladon's (1982) "whole spectrum" model, which de-emphasizes the fact that spectral peaks carry a stronger weight than spectral valleys, the missing-data model puts a strong weight on the peaks that are emphasized in the square-magnitude spectrum. Avoiding spectral smoothing and restricting pattern matching to available samples can eliminate aliasing. They also claim that the model can be implemented in the spectral domain using a harmonic sieve based on an estimate of F0. The model ensures F0-independent pattern matching and does not account for the loss of information due to sampling.

2.5 Spectral Envelope Estimation Using Spectral Peaks

The spectral peak envelope method is based on the peaks of the log magnitude spectrum. One of the reasons for selecting peaks is when the noise level increases the peaks are the last parts of the spectrum to be submerged. The other reason is that the peaks have high energy and also, they may be the harmonics of the fundamental frequency.

Douglas B. Paul [2] showed that spectral peaks could be used to estimate the spectral envelope. According to his model, the sampled magnitude spectrum of the speech waveform will yield a spectral envelope estimate using interpolation, provided the sampling is dense enough or is reasonably smooth. Further, this spectral envelope estimate will be an estimate of the vocal tract filter. He discussed two problems: one is the location of the samples of magnitude spectrum; two, reconstruction of the spectral envelope from the measured samples. He proposed a heuristic, which locates the samples at multiples of T (pitch period) without accurate knowledge of the pitch. It also incorporates the shifting of the peaks due to the nonstationarity of the signal. The peak-finding heuristic requires a parameter, the average fundamental frequency F0. The procedure for locating the peaks follows:

1. Initialize

$k = 1, f_0 = 0$

2. Search the spectrum

$$f_{k-1} + \frac{1}{2}\overline{F_0} \ \mathit{to} \ f_{k-1} + \frac{3}{2}\overline{F_0} \ ----------------\langle 6 \rangle$$

for every f_k such that the spectrum is maximized

3. $k = k + 1$

Repeat equation 6 until the entire spectrum is covered.

The second problem is solved by a third order spline interpolation of the samples of the magnitude spectrum to yield the spectral envelope. For voiced phonemes, the spectral envelope estimator performed a good pitch-spectrum separation. On aperiodic speech, the estimator approximated the spectral envelope with sufficient accuracy for perceptually good reproduction.

2.6 Introduction to Matlab

Matlab is an integrated technical computing environment that combines numeric computation, advanced graphics and visualization, and a high-level programming language[11]. It includes hundreds of functions for data analysis, numeric computation, engineering graphics, programming, GUI design, etc. Matlab is used in a variety of application areas including signal and image processing research. Its architecture makes it easy to explore data and create custom tools that provide early insights and competitive advantages.

Matlab has a family of application specific toolboxes and has capabilities to create standalone C / C++ code from Matlab language programs. Matlab's data representation is matrix-based. Matlab provides an excellent prototyping environment. It can also be linked to external software. Matlab code and data files are platform independent.

2.7 Conclusion

This chapter discussed the standard spectral shape method and its limitations, formants as features, theory of missing-data model of vowel identification and its implementation, and a brief discussion on Matlab. The next chapter derives the algorithm for implementing the missing-data model.

CHAPTER III
ALGORITHM

3.1Introduction

Chapter 2 (Section 2.2) summarized the "standard" spectral shape method for representing speech spectra, using a Discrete Cosine Transform of the log magnitude spectrum. Although this DCT method, also commonly referred to as cepstral analysis is widely used in automatic speech recognition, the missing frequency components model implies that the "missing" portions of the speech spectra should not be used for extracting features from speech spectra. Rather, the features should only be based on the prominent spectral components actually contained in the speech signal. In this work, we consider the problem of computing features, similar in nature to DCT terms, but computed only from the spectral peaks. In particular, the focus of this chapter is the derivation of the mathematical formulae associated with the peak envelope estimation model described in this thesis.

A summary of this chapter is as follows. First we describe general properties of the frame level processing (short-time spectral analysis), typically performed in speech processing for automatic speech recognition. The goal of the frame-level processing is to "transform" the speech signal to a compact feature set that conveys most of the phonetic information in speech. Next, we derive the algorithm used to compute features based on spectral peaks. We go on to describe the techniques used for peak picking, an essential signal processing step for the peak envelope estimation and illustrate several practical problems. Throughout the chapter, the algorithms and practical problems are illustrated with time domain and frequency domain plots.

Matlab's signal processing functions and visualization functions are used extensively for algorithm implementation and testing. The testing is done using the isolated vowel database from the speech lab at Old Dominion University.

3.2 Frame Level Signal Processing and Implementation

Since the speech signal has short-time spectral characteristics that signify the phonetic content, there is a need for frame-level processing to extract these characteristics. Generally, the speech signal is segmented into overlapping frames, with a typical frame length of 30ms and a frame spacing of approximately 10 ms. As in many signal processing applications of this type, a window is applied to each frame to reduce the signal discontinuities created by segmenting the signal into frames. In speech analysis, Hamming and Hanning windows are the most commonly used tapered windows.

Figure 3 is a flow diagram of "generic" frame-level processing for speech analysis. The first major step of signal processing is generally to compute the log magnitude spectrum. The next step is usually to estimate the spectral envelope, using a method similar to that described in chapter 2 using Cosine Basis Vectors (CBVs). Note that these spectral analysis/feature calculations must then be repeated for the entire speech utterance to be analyzed. There may be several other processing steps before recognition is performed. However, in this chapter, the focus is on the calculation of features based on the short-time spectra (i.e., frame based analysis). In the remainder of this chapter, we describe a new method for computing the short-term speech features.

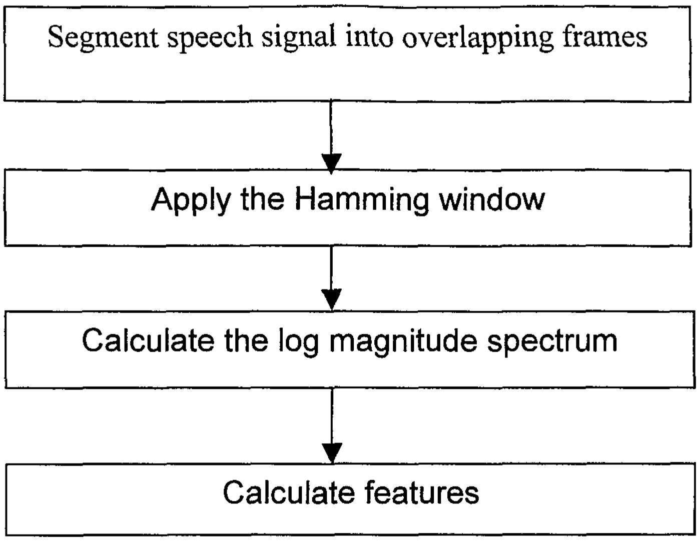

Segment speech signal into overlapping frames
Apply the Hamming window
Calculate the log magnitude spectrum
Calculate features

Figure 3 Flow Diagram of Frame Level Signal Processing

Figure 4 shows a plot of the 30ms acoustic signal segment (that is, one frame) and its log magnitude spectrum. Features are then computed from the log magnitude spectrum. About 10-15 features are generally used to represent the overall shape of the spectrum. According to models of speech production, these features represent the vocal tract configuration used to produce the sounds. Such features are also closely related to the phonetic content of the sounds.

3.2.1 Envelope Estimation

The goal of this section is to describe an algorithm, which can be used to approximate the envelope of the speech spectrum as a weighted sum of orthonomal basis vectors. The weighting coefficients will then be considered as the "features," which represent the spectrum. These coefficients can also be used to compute a smoothed version of the envelope spectrum.

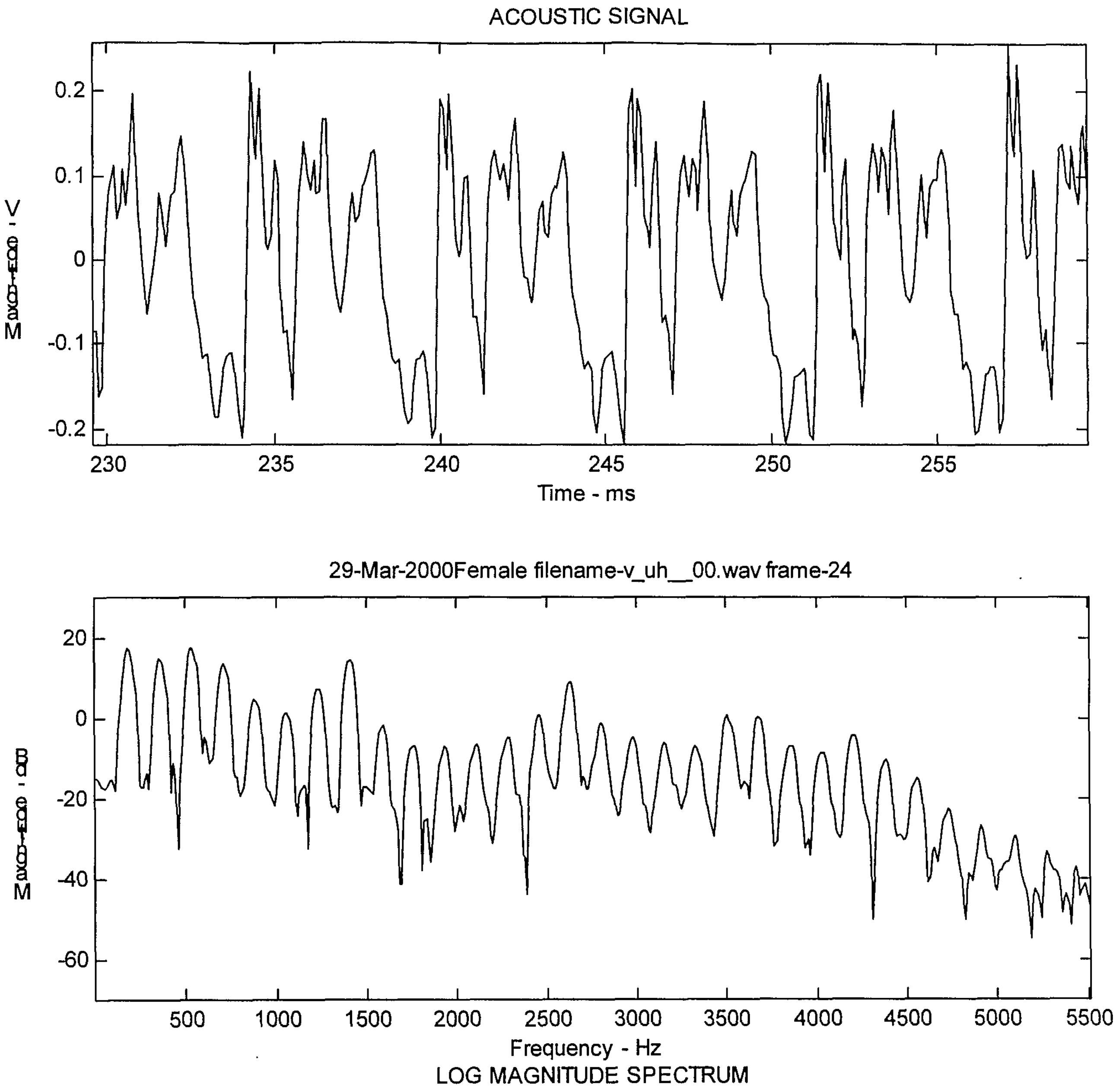


Figure 4 a. Acoustic Signal for one frame b. Its Log Magnitude Spectrum

In this section, we assume that the peaks in the spectrum, which the envelope spectrum ideally should track in a smoothed way, are available. The basic method used to select spectral peaks, along with several refinements used to overcome some practical problems both with peak picking and using peaks for the envelope curve fitting, are described in a later section of this chapter. The section continues with the mathematics underlying the curve-fitting problem used for envelope estimation.

The curve-fitting problem is posed in terms of forming a linear combination of a set of elementary basis functions. As for most basis vector expansions, the basis functions should be orthogonal or orthonormal over the interval for which the representation is to be valid. For example, Fourier series analysis is based on sinusoidal basis functions used to approximate periodic time functions. In this work, the basis vectors are derived from real multiples of a half cycle of a cosine (same as DCTC analysis described in chapter 2). However, the coefficients are derived to minimize mean square error between a smoothed curve and the signal to be approximated for selected samples. There is no requirement that the samples be uniformly spaced. Thus the method is well suited to approximate the spectral envelope, with missing frequency components.

The method can be derived as follows.

Consider first the general curve-fitting problem. Let

$x(n)$ -- be the signal to be represented

$\phi_k(n)$ -- be the basis functions for the representation, where

$$1 \le k \le P$$

$x(n)$ is then to be approximated as a linear combination of the basis functions using

$$\hat{x}(n) = \sum_{k=1}^{P} a_k \, \phi_k(n) ----------------------\langle 7 \rangle$$

where a_k is the set of coefficients which are to be determined. For exact representations, the value of P could be quite high. However, for most signals of interest, the value of a_k tends to become small as k becomes large. Since it is not desirable to use too many terms in most situations, the series is terminated after some small number of terms, and the resulting expression is an approximation of $x(n)$

To be more specific, in our work

$x(n)$ -- is an array of log spectral magnitudes $1 \le n \le N$

Typically N is one half of the fft length used for spectral estimation.

$\phi_k(n)$ -- is a matrix of Cosine Basis Vectors (CBV's) orthonormalized over frequency with N rows and P columns where

$1 \leq n \leq N$

$1 \leq k \leq P$

P - the number of CBV's. Typically between 10 and 15.

The CBV's can be represented as an N by P matrix, with each column a basis vector. Due to the orthonormal property, the dot product of each column with itself is 1, and the dot product between differing columns is 0. Note these basis vectors are the same as those used for a discrete cosine transform of an even function, but with calculations based on only one "half" of the function. (Only "half" of the DCT function is considered.)

Continuing with the derivation, the approximation of $x(n)$, $\hat{x}(n)$ is given by

$$\hat{x}(n) = \sum_{k=1}^{P} c_k \phi_k(n) - - - - - - - - - - - - - - - \langle 8 \rangle$$

Selection of the coefficients c_k (which we call Discrete Cosine Transform Coefficients -DCTCs, as for the "standard" spectral shape method) is based on minimizing the error between the original and the approximation. The Weighted Mean Squared Error E between $\hat{x}(n)$ and $x(n)$ is

$$E = \sum_{n=1}^{N} [x(n) - \hat{x}(n)]^2 \, index \, (n) - - - - - - - - - \langle 9 \rangle$$

where $index \; (n)$ is the factor used to select peaks in the spectrum ($x(n)$).
In particular,

$index\ (n)$ -- vector of 0's and 1's

0 if $x(n)$ is not a peak

1 if $x(n)$ is a peak

Thus, the use of $index \quad (n)$ vector determines E based on only the peaks of $x(n)$, as shown in equation 9 . If $x(n)$ is not a peak, then the contribution to E is zero. The use of this index term is what differentiates this new method from the method described in chapter 2. As mentioned previously, we address the peak picking problem in a later section.

Our goal is to find the coefficients c_k such that E is minimized. Differentiating E with respect to each of the coefficients and setting these derivatives equal to zero is used to solve this problem.

Substituting for $\langle 8 \rangle$ in $\langle 9 \rangle$ we obtain

$$E = \sum_{n=1}^{N} \left[x(n) - \sum_{k=1}^{P} c_k \, \phi_k(n) \right]^2 index(n) ------------------ \langle 10 \rangle$$

Differentiating $\langle 10 \rangle$with respect to the coefficients c_m

$$\partial E / \partial c_m = -\sum_{n=1}^{N} 2 \left\{ x(n) - \sum_{k=1}^{P} c_k \, \phi_k(n) \right\} \phi_m(n) \, index(n) -------- \langle 11 \rangle$$

for $1 \le m \le P$

From $\langle 11 \rangle$ we obtain

$$-\sum_{n=1}^{N} 2 \left\{ x(n) - \sum_{k=1}^{P} c_k \, \phi_k(n) \right\} \phi_m(n) \, index(n) = 0 ------ \langle 12 \rangle$$

Expanding $\langle 12 \rangle$

$$\sum_{n=1}^{N} x(n) \phi_m(n) \, index \, (n) = \sum_{n=1}^{N} \sum_{k=1}^{P} c_k \, \phi_k(n) \phi_m(n) index \, (n) -- \langle 13 \rangle$$

Rearranging terms

$$\sum_{n=1}^{N} x(n)\phi_m(n)index(n) = \sum_{n=1}^{N} c_k \sum_{k=1}^{P} \phi_k(n)\phi_m(n)index(n) ------\langle 14\rangle$$

$\langle 14\rangle$ is equivalent to the matrix equation

$$\begin{bmatrix} B_1 \\ B_2 \\ .. \\ .. \\ B_p \end{bmatrix} = \begin{bmatrix} c_1 \\ c_2 \\ .. \\ .. \\ c_p \end{bmatrix} \begin{bmatrix} A_{11} & A_{12} & .... & A_{1k} \\ A_{21} & .... & .... & A_{2k} \\ & & & \\ & & & \\ A_{p1} & .... & ..... & A_{pp} \end{bmatrix} --------------- \langle 15 \rangle$$

where

$$A_{ij} = \sum_{n=1}^{N} \phi_j(n)\phi_i(n)index\ (n)$$

$$B_{i=} \sum_{n=1}^{N} x(n)\phi_i(n)index\ (n)$$

for

$1 \le i \le P$

$1 \le j \le P$

Solving for c_k in $\langle 15\rangle$

$$\begin{bmatrix} c_1 \\ c_2 \\ .. \\ .. \\ c_P \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} & .... & A_{1k} \\ A_{21} & .... & .... & A_{2k} \\ & & & \\ & & & \\ A_{P1} & .... & ..... & A_{PP} \end{bmatrix}^{-1} \begin{bmatrix} B_1 \\ B_2 \\ .. \\ .. \\ B_P \end{bmatrix} ----------- \langle 16 \rangle$$

Thus$\langle 16 \rangle$ solves for those coefficients that best approximate the spectral envelope, as a curve fitting to spectral peaks, using an underlying set of basis functions. Note that the solution is equivalent to the "standard" basis vector representation of spectra, if every point is considered to be a peak. This can be seen as follows:

When all the points are chosen as peaks, (i.e., $index(n) = 1 \; \forall \; n$), the coefficient matrices are equivalent to:

$$A_{ij} = \sum_{n=1}^{N} \phi_j(n)\phi_i(n)$$

Since $\phi_j(n)$ & $\phi_i(n)$ are orthonormal, A is an identity matrix.

$$A_{ij} = \begin{Bmatrix} 1, & i = j \\ 0, & i \neq j \end{Bmatrix}$$

Hence A^{-1} is also an identity matrix, and the solution for the coefficients simplifies to

$$B_{i=} \sum_{n=1}^{N} x(n)\phi_i(n)$$

$1 \leq i \leq P$

$1 \leq j \leq P$

Also, using (16),

$$\begin{bmatrix} c_1 \\ c_2 \\ .. \\ .. \\ c_p \end{bmatrix} = \begin{bmatrix} B_1 \\ B_2 \\ .. \\ .. \\ B_p \end{bmatrix} \quad ----------------------------\langle 17 \rangle$$

This shows that, as expected, the envelope estimation method is equivalent to the standard spectral shape method when all the points are selected as peaks.

However, for the case of interest in this section (equation 16), a matrix inverse is required, and it is possible for problems to arise due to an ill conditioned matrix. Although a complete treatment of the matrix inverse stability issues is likely to be quite complex and is not addressed in this thesis, the essence of the problem can be described as follows. As mentioned previously, this overall estimation solution is a form of curve fitting using cosine functions. The problems encountered are very similar to those in polynomial curve fitting. It is well known from polynomial interpolation theory that when a small number of points on an interval are made to fit a polynomial of high degree, the curve fit may be very poor between sampling points. In general, the number of interpolation points must be greater than the order of the polynomials used for fitting. Similar restrictions apply to function approximation with neural networks. These kinds of problems are typically referred to as "over fitting." Thus, for the problem at hand, we could expect problems if too few peaks are selected relative to the number of basis vectors used.

3.2.2 Matlab Implementation

The implementation of the estimation of the spectrum envelope using spectral peaks and frequency warped orthonormalized CBVs was done using MATLAB. The code is quite straightforward to implement, since Matlab is very well suited to matrix operations. Some attention was paid to decrease time and computational complexity in calculating the matrix A, the column vectors B & C. It is observed from Equation 16, that the elements of matrix A and column vector B have common factors, i.e, the CBVs and the index vector. Hence, the resultant matrix of the array multiplication of each column of the CBVs with the index vector is used to calculate the A matrix and the B vector. The column vector, C, is the product of the B vector and the inverse of the A matrix. Matrix multiplication of the C vector and the CBVs gives the estimated peak envelope. Appendix gives the actual Matlab code used to implement the envelope routine (Pkreest.m) --- approximately 20 lines of code.

3.3 Peak Picking & Processing

Two methods were tried for peak picking. In the first method, all local spectral peaks were first found. This was followed by steps to remove spurious peaks (peaks "close" to other much larger peaks) and peak broadening. Although this algorithm appeared to perform reasonably well, as judged by inspection of spectral plots, it still often appeared to miss valid peaks and/or include unwanted peaks. Therefore, a second method, described below, and motivated by the work of Douglas B. Paul [2] was implemented and used for the experimental results reported in this thesis.

The peak picking method can be described algorithmically as follows:

1. Consider a spectral vector X(k), $1 \leq k \leq N$, as the speech spectrum for which the peak regions are to be identified. k is considered as the frequency index.
2. Define a frequency dependent window width for finding maxima in X(k). This function is referred as W (k).
3. For each frequency index k, determine the maximum of X(k) over the width determined by W (k). That is, compute

   $Y(k) = \max (X(j), k - W(k)/2 \quad j \leq k \nleq W(k)/2)$,

   for $W(1)/2 < k < W(N)/2$

   For k outside the range given above, define Y(k) = X(k).
4. Next compare X (k) and Y (k), for $1 \quad k \leq N$, using delta as a parameter to compare closeness.

If (Y(k) - X(k)), < delta, then X(k) is close to a peak, and index(k) = 1.0
If (Y(k) - X(k)), > delta, then it is assumed that X(k) is not close to a peak,
and index(k) = 0.0

This entire method was implemented using three control variables. They are,

1. Freq_kernel_min --- used to specify the minimum width of the frequency window (typical value of 150 Hz).
2. Freq_kernel_max --- used to specify the maximum width of the frequency window (typical value of 300 Hz).
3. Half_length --- parameter equivalent to delta above (typical value of 2.0 using natural log scaling of spectrum).

Note that the width of the frequency window was then linearly interpolated between the minimum and maximum values, as the frequency index ranged from the minimum to the maximum values. The basic idea was to search for harmonic peaks in the spectrum, but use a wider search range at higher frequencies to match the property of the reduced frequency resolution of the human ear at higher frequencies. The Half_length parameter was used to determine the width of the region retained for each peak.

This method is illustrated with figures. Figure 5.a gives a spectral plot, and Figure 5 b, shows the maximum spectra, (Y (k) given above). Figures 6 & 7 show the spectra with peaks indicated for a half-length of 1 & 3 respectively.

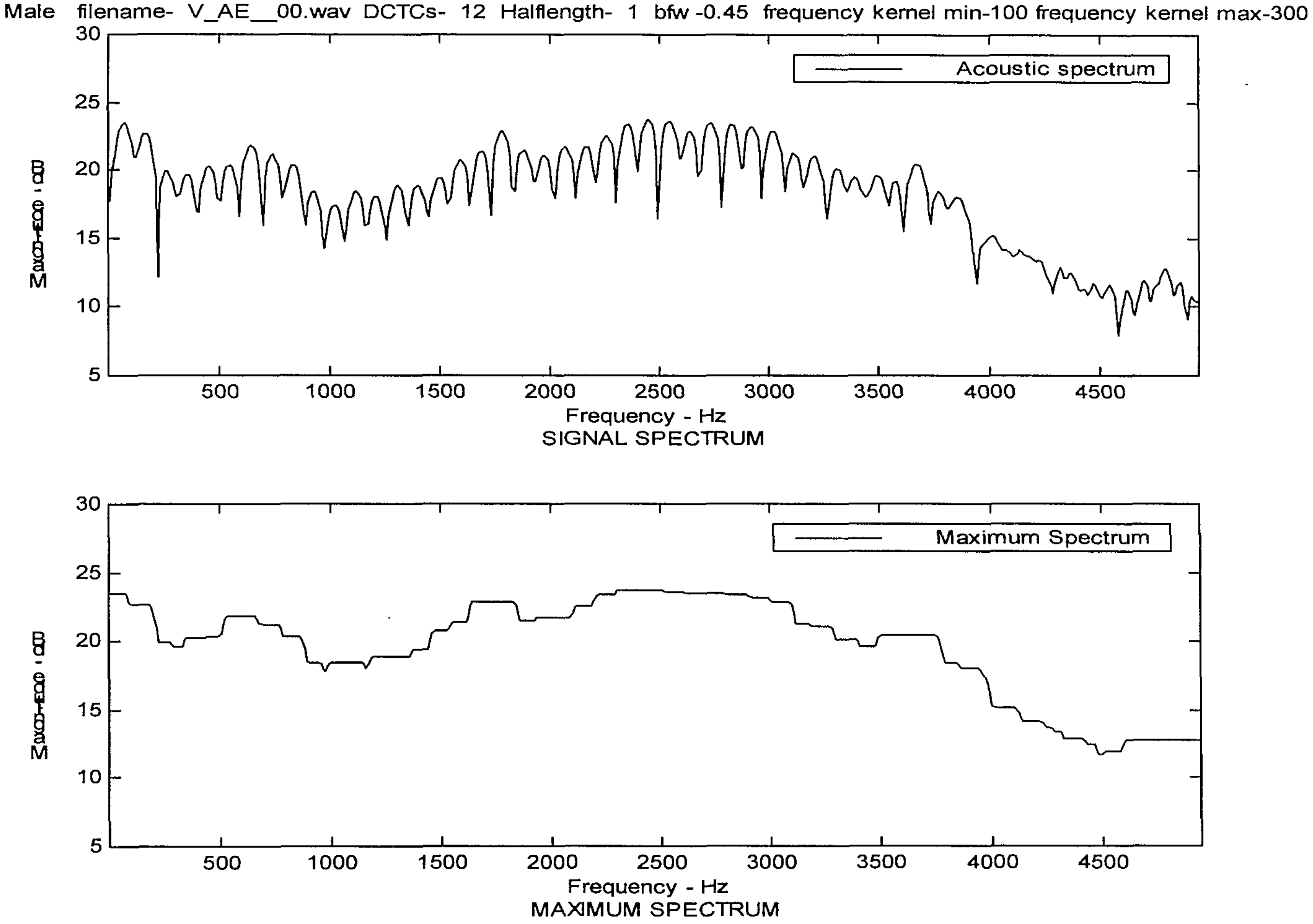


Figure 5.a gives a spectral plot, and Figure 5 b, shows the maximum spectra

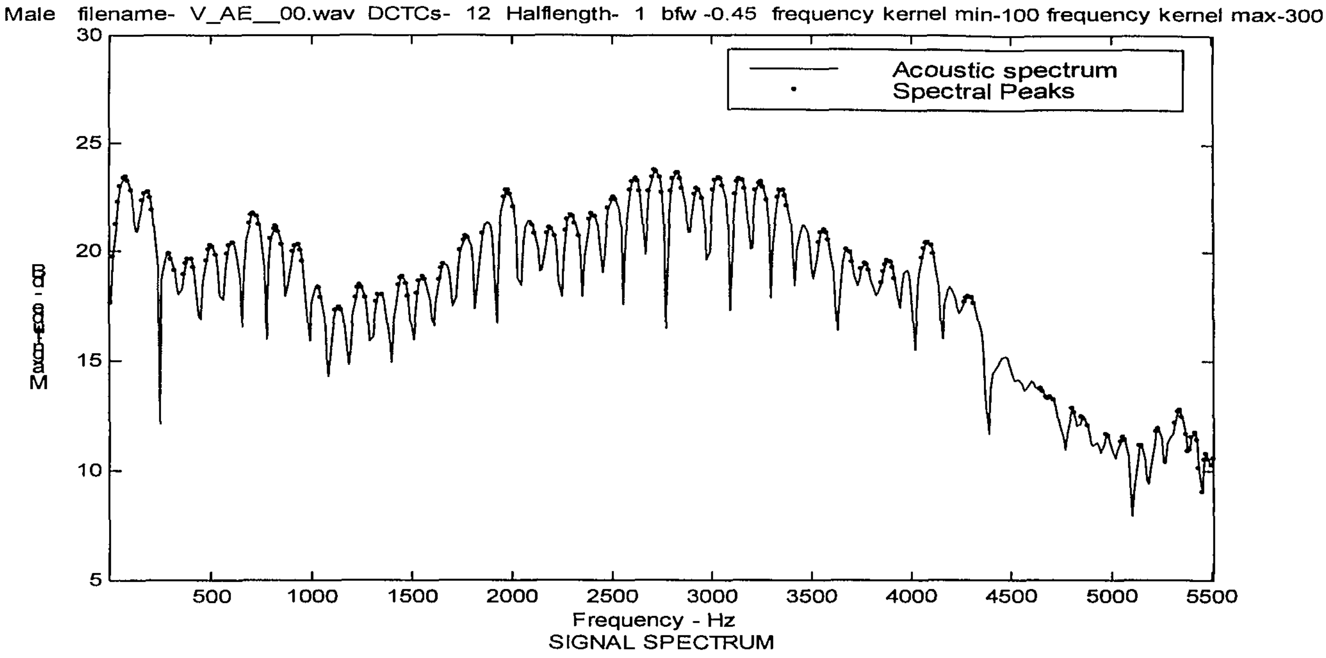


Figure 6. Plot of the Acoustic Spectrum and Spectral Peaks for a Half_length of 1

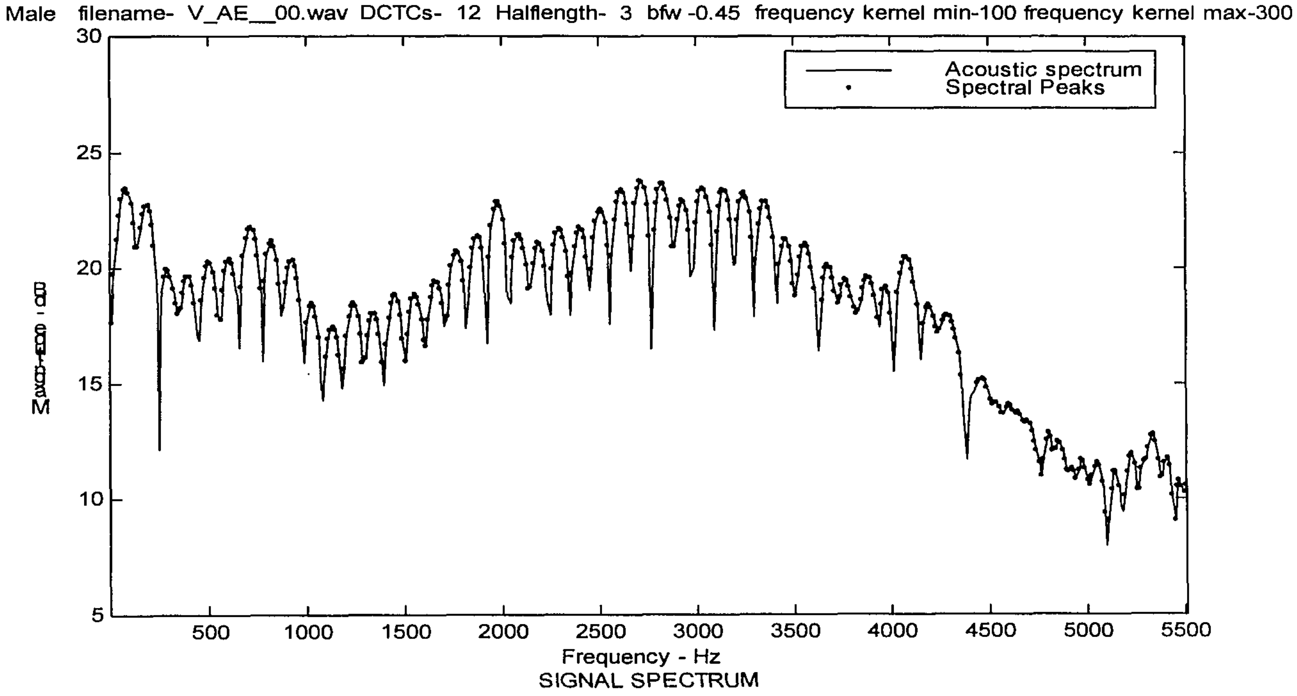


Figure 7. Plot of the Acoustic Spectrum and Spectral Peaks for a Half_length of 3

3.4 Examples

In this section, the peak envelope method discussed in the earlier chapters is illustrated with examples. The standard units of measurement used in the following examples are frequency in hertz, time in ms and magnitude in volts or decibels. Since the spectrum is even, only one half of the frequency range is used for analysis. An FFT length of 1024 and a sampling rate of typically, 16KHz were used for the examples. The graphs and spectrograms are plotted using 2D and 3D MATLAB commands.

The peak envelope method was tested on vowels from the speech lab's database. Figures 8 & 9 show the signal spectrum, spectral peaks, the peak envelope, and the standard DCTC envelope computed from the entire spectrum for male vowel and sentence respectively. The results are shown for a Half_length of 1, using 12 DCTCs with a warping factor of 0.45 and a frequency window width of 100/300. It is also seen that the frame is rich in harmonics and hence has many peaks. It may also be observed that the peak envelope and the standard envelope are not significantly different except for their magnitude levels. It is also noted that the peaks in the envelopes indicate the formant frequencies.

Figure 10 shows a spectrogram for the male vowel with the color axis indicating the intensity levels. A single function in Matlab, pcolor() plots the peak envelope matrix, which has the estimated peak envelope values of the entire utterance with the given frequency and time indices.

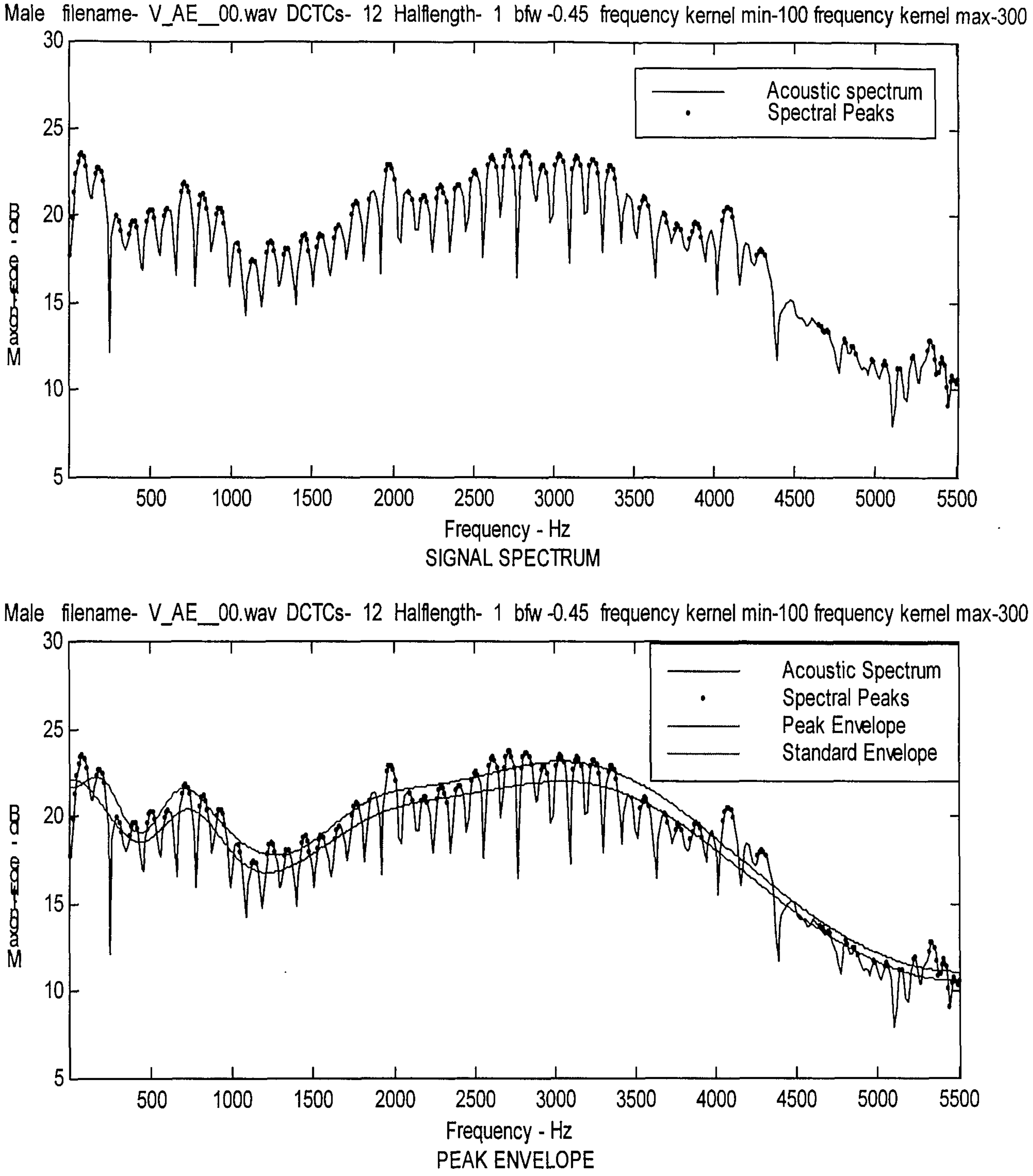


Figure 8 a. (Top) Log Magnitude Acoustic Spectrum with the peaks b. (Bottom) Acoustic Spectrum, Peak Envelope, Standard Envelope

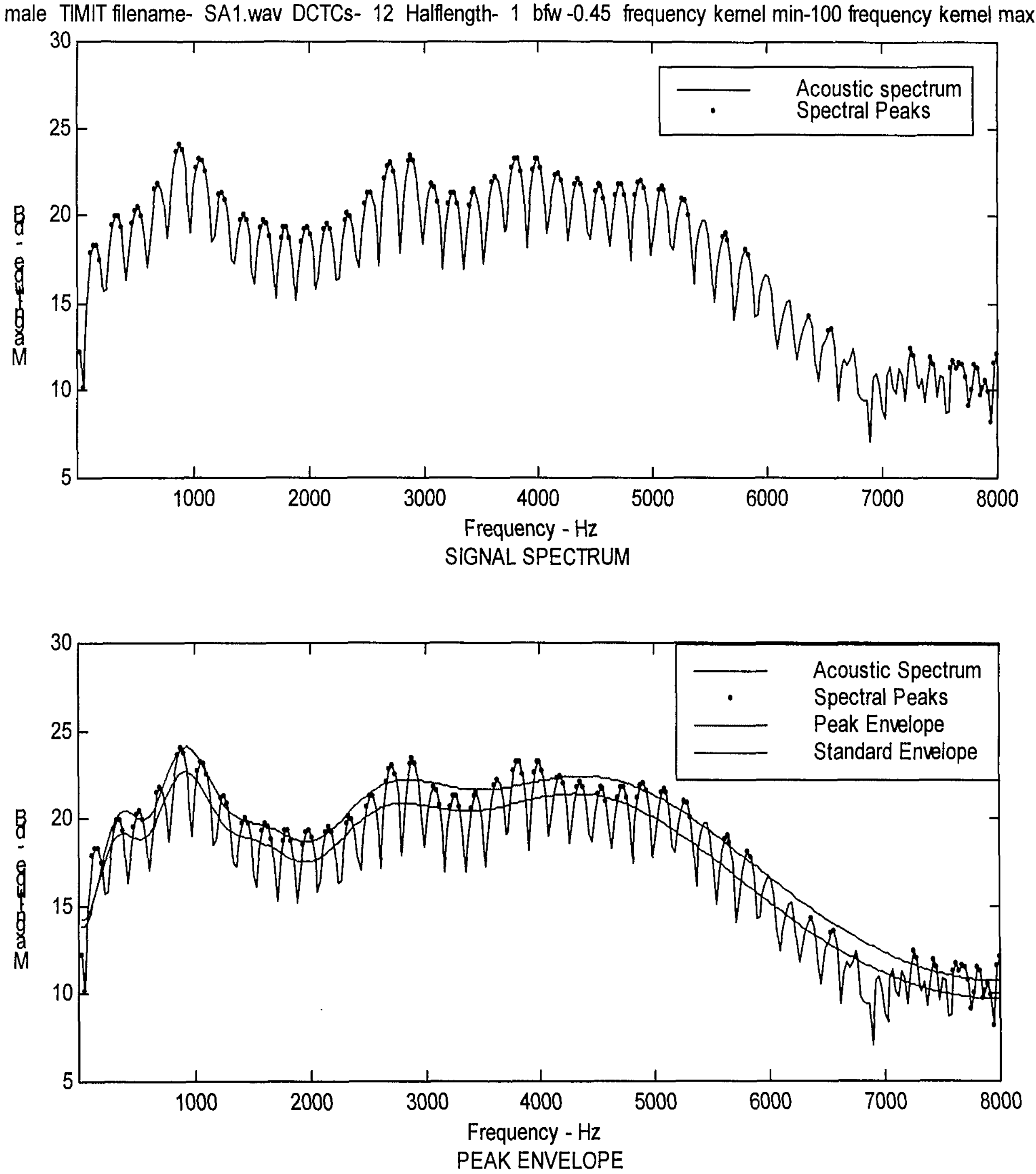


Figure 9 a. Acoustic Signal Spectrum b. Plot of the Spectrum, Spectral peaks, Peak Envelope and the spectral envelope computed from entire spectrum (harmonic spectrum)

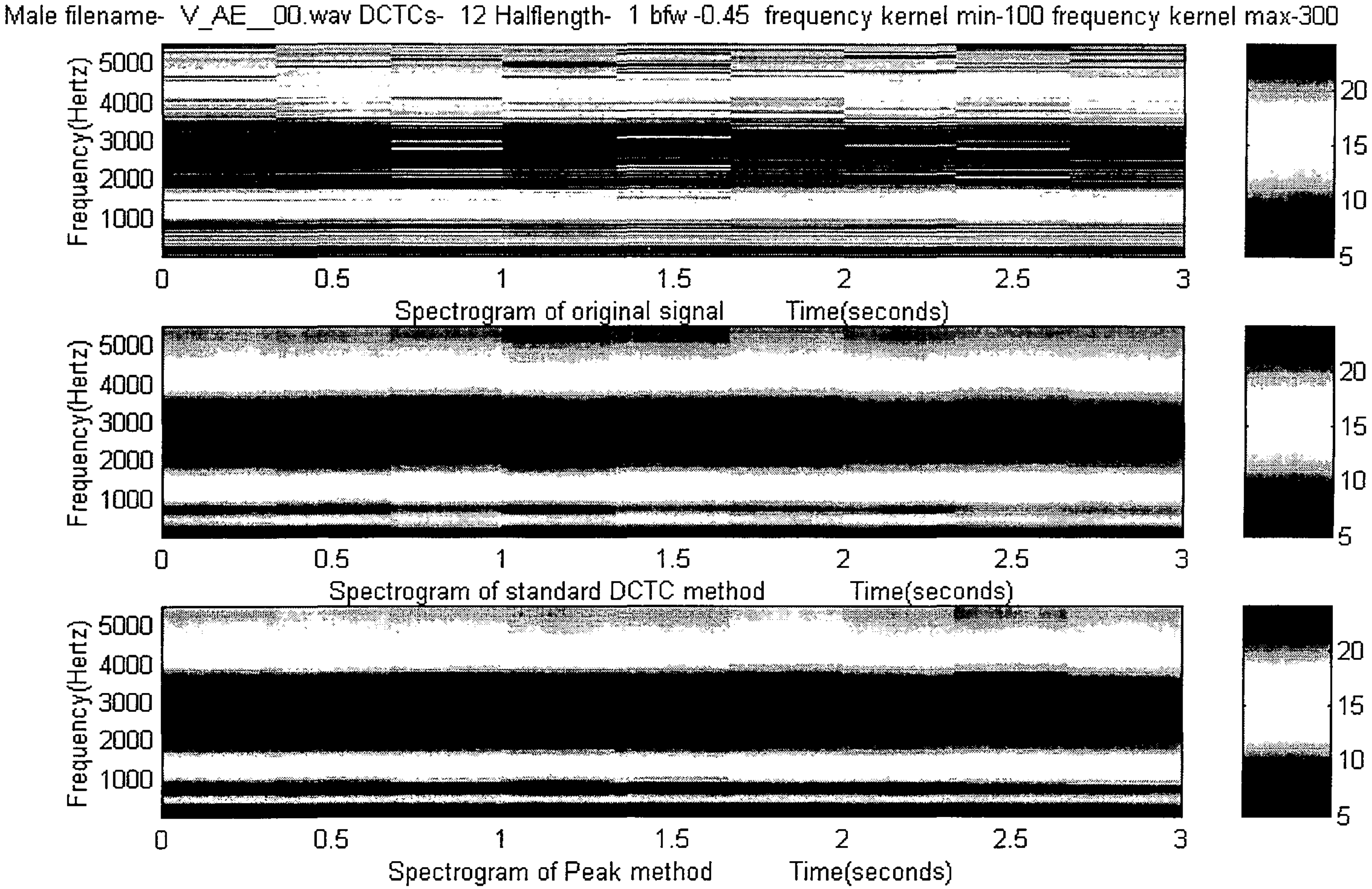


Figure 10. Spectrogram of Acoustic Signal [Top], Spectrum estimated using Standard DCTC method [center], Spectrum estimated using Peak Envelope Method [Botttom]

3.5 Conclusions

This chapter detailed the derivation and implementation of the peak envelope method. Example plots in the previous section show the functionality of the peak method. The following conclusions can be made from the experiments and the plots:

The peak spectral method basically works for harmonic spectra due to the abundance of peaks. Instability due to over-fitting of the peaks was not found due to the peak picking method used. In particular, a large number of peaks were found, and peak broadening was used, so that overall the spectrum was still quite well sampled.

The spectral envelope and the entire spectral shape are observed to differ in magnitude levels, but be quite similar in shape. However, there are some cases (some frames of speech) where the differences appear to be large without obvious reasons. This method should probably be used only for voiced frames of speech, and the "regular" method used for unvoiced speech. These spectral peak features are investigated in detail in chapter 4, with vowel classification experiments.

CHAPTER IV
EXPERIMENTAL VERIFICATION

4.1 Introduction

The preceding chapters discussed the background of ASR, classifiers, and recognizers. The missing data model of vowel identification for spectral envelope estimation was also presented. As mentioned previously, signal processing for speech recognition has two main steps -- frame level and block level processing. The concepts algorithm and implementation of frame level processing for the peak spectral envelope representation were explained in chapter 3. The main goal of this chapter is to present an experimental verification for the algorithm using an isolated vowel database. Some additional block level processing steps are described. Preprocessing for speech signal processing is also needed for effective classification. This is performed by the Tfrontm function, which is explained in detail. A series of experiments were conducted on the peak envelope method with and without block level time smoothing and also with a varying signal to noise ratio.

4.2 Processing Overview

The steps performed in the vowel classification system used to evaluate the peak spectral envelope features consisted of the creation of feature files, computing the mean and standard deviations for each feature, normalizing the feature files, and classifying the test files. Feedforward neural networks with one hidden layer were used for classification. The functions used were

1. Tfrontm( )
2. Scale( )
3. Transfor( )
4. Neural( )

The second, third, and fourth functions listed were previously developed in the speech laboratory at Old Dominion University and are regularly used for speech classification experiments. An overview of these three functions is given later in this chapter. The Tfrontm program, developed during the course of this research using Matlab, was used to implement the peak envelope implementation described in chapter 3.

4.2.1 Tfrontm

The front end processing program, Tfrontm is functionally similar to a previously developed C function, tfrontc, used to implement the DCTC spectral shape analysis method. A flow diagram of tfrontm() is given in figure 11. Initialization, preprocessing, frame-level processing, block-level processing and feature extraction are the various stages in Tfrontm. In the initialization stage, parameters are initialized by reading the setup files, tfront.dat, sentence.dat, phone.dat and feature.ini. During the pre-processing stage, a pre-emphasis filter is applied to enhance the high frequencies around 2 kHz, to match the sensitivity of the human ear. Noise may be added to simulate real-world conditions. The frame-level signal processing is the same as that presented in chapter 3. The spectral segments are processed by the peak envelope method and the features are written into respective feature files (one file for each phoneme).

In block level processing, blocks, which are the combination of approximately 5 to 6 frames, are the functional units. Block level processing is motivated by the observation that speech signals in general are non-stationary, and some of the very short non-stationary events need to be represented for better classification. Block level processing uses a technique that combines frequency and time domain information. In particular the feature vector components for each frame are represented by a cosine expansion over time. This second basis vector expansion, which compactly represents the temporal history of each frame-level

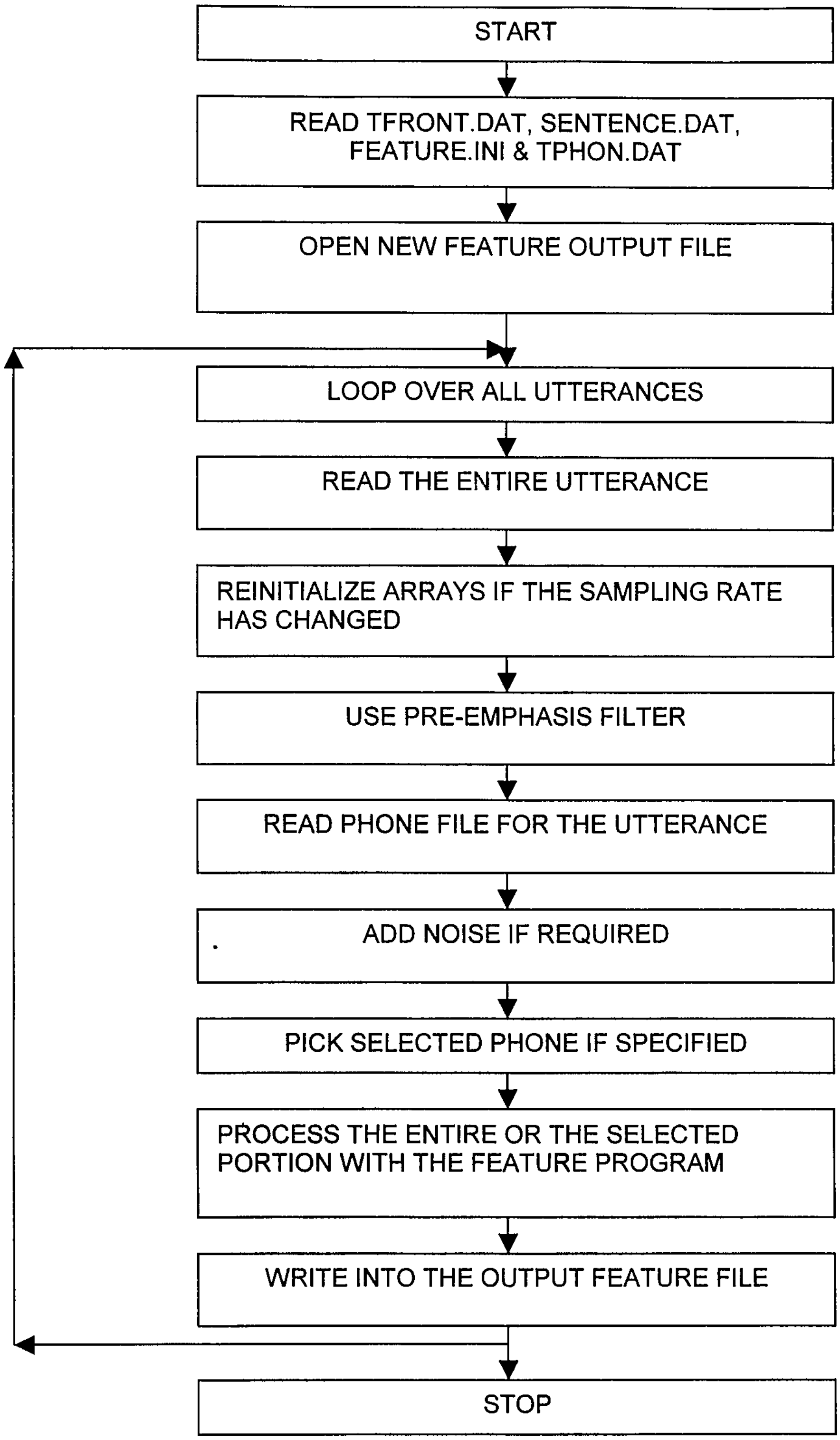

START
READ TFRONT.DAT, SENTENCE.DAT, FEATURE.INI & TPHON.DAT
OPEN NEW FEATURE OUTPUT FILE
LOOP OVER ALL UTTERANCES
READ THE ENTIRE UTTERANCE
REINITIALIZE ARRAYS IF THE SAMPLING RATE HAS CHANGED
USE PRE-EMPHASIS FILTER
READ PHONE FILE FOR THE UTTERANCE
ADD NOISE IF REQUIRED
PICK SELECTED PHONE IF SPECIFIED
PROCESS THE ENTIRE OR THE SELECTED PORTION WITH THE FEATURE PROGRAM
WRITE INTO THE OUTPUT FEATURE FILE
STOP

Figure 11. Flow Diagram of Tfrontm()

spectral feature, is called a Discrete Cosine Series (DCS) representation[8]. This frame and block level processing result in computationally efficient "temporal / spectral" features that have been demonstrated to be effective for speech classification (Zahorian and Jagharghi, 1993; Wang et. al., 1996) .

The block level processing results in a set of DCS terms for each DCTC. For some phonetic classification applications (for example Zahorian and Jagharghi, 1993), better results are obtained with a subset of these terms. These DCS terms over the DCTCs are the final features representing the speech signal for each block. In the experiments reported in this chapter, the DCS method was used, but only in the basic sense of using the first DCS term to perform an averaging of frame level terms over a block of five frames. For steady vowels, spoken in isolation, this additional averaging can increase classification accuracy.

The first step in the training procedure is to collect training data. For the work presented in this thesis, training data was previously collected as an isolated vowel database at ODU. A certain group of speakers can be selected, for example, male speakers to train a specialized neural network. The training files are then read for all these speakers. For each file the final features are computed using the signal processing functions described in chapter 3. The features are stored in files using a flexible format developed in the speech lab at ODU for speech processing applications. One feature file is created for each vowel. Each file contains all selected training tokens for that vowel, thus encompassing a broad range of pronunciations of the vowel independently of a particular speaker.

The features for one particular phoneme vary from speaker to speaker and even with different utterances spoken by the same speaker. Therefore the mean and standard deviation can be computed for each feature. It has been shown that (for example, Haykin, 1994) neural networks work best if the features are scaled to have zero mean and a standard deviation of about 0.2. If a Gaussian distribution of the features is assumed, which is often valid due to the Central Limit theorem, the scaled values will then be between –1 and +1 in more than 99% of all cases. The scale( ) reads the feature files and computes the mean and standard deviation of each feature over all phonemes. These statistics, which are used as scale factors, are written to a "scale" file. The actual scaling is done with the Transfor( ). Transfor( ) reads the scale file and creates a new set of feature files which now contain the scaled version of the original features. After the features are scaled, the neural network can be trained. Neural( ) is the neural network function that classifies the test utterances. It reads in the feature files for each vowel and uses these to train the network and classifies the test sentences using the back propagation iterative procedure. The classifier structure used was single large neural network, with one hidden layer with 25 nodes, and 10 outputs (1 for each vowel). The next section discusses the experiments and their results.

4.3 Basic Tests

The isolated vowel database recorded in the speech lab at Old Dominion University was used for all the vowel classification tests. It has over 300 speakers with each speaker speaking the 10 vowel sounds (ae, ah, aw, ee, oo, uh, eh, ih, ue, ur) three times. They were recorded in isolation, in response to a computer visual prompt. The speaker typically held each vowel sound for about one second. The acoustic signals were automatically endpointed and saved to a binary file, using the TIMIT NIST header format commonly used for speech ASR databases. The data files were labeled uniquely and stored in a certain directory structure organized according to gender and speaker. The filename specifies which vowel is contained in the file. In addition, a secondary file containing labeling information was created for each waveform file. The sampling rates were either 11.025 kHz or 22.050 kHz. The analysis software is flexible enough to accommodate different sampling rates for each file, provided the frequency range selected for analysis is less than half the sampling frequency for lowest sampling rate.

The specific amounts of data used for the experiments reported in this thesis are as follows:

Men --- 90 training speakers, 24 tests speakers

Women --- 100 training speakers, 24 test speakers

Children --- 34 training speakers, 24 test speakers
Thus the data based was comprised of a total of 296 speakers.

Note that data for each training and test set were processed identically. Neural network training was based only on the training data. Results are reported only for the test speakers, since this is a much better indication of the potential performance of automatic classifier.

The basic analysis parameters used in experiments were:
Sampling rate: 16000 Hz (This was later to changed to match actual sampling rate for each file)
Segment time: 20000 ms (Used only to set array sizes in Tfrontm)
Frame length: 30 ms
Frame space: 10 ms
FFT length: 1024
Kaiser Window Beta: 6
Number of DCTCs: 12
Frequency Warping: 0.45
Basis vector orthonormalization: 1
Frequency range: 50 to 5000 Hz
Prefilter center frequency: 3200 Hz
Spectral range: 90dB

The block processing parameters:
Block length minimum: 1 frame (5 frames for time smoothing)
Block length maximum: 1 frame (5 frames for time smoothing)
Block jump: 1 frame

The frequency based parameters:
Frequency window width minimum: 150 Hz (varies according to speaker type)
Frequency window width maximum: 300 Hz (varies according to speaker type)

The peak method parameters:
Halflength: 3 (varies according to speaker type)

As discussed in the section 4.2, the parameters of each feature were scaled for a mean of 0.0 and a standard deviation of 0.2 by the scale( ) and transfor( ) programs. The neural network used for classification had one hidden layer, 12 input nodes, 25 hidden nodes, and 10 output nodes, and was trained with back propagation using 125,000 updates.

As a control, the "standard" DCTC method was performed as explained in chapter two. The method under investigation was the peak DCTC method described in chapter three.

Experiment 1:

The objectives of the first experiment were to evaluate some basic control and test conditions. Specifically, classification rates were obtained for the standard DCTC method without time smoothing, standard DCTC method with time smoothing, peak method without time smoothing, peak method with time smoothing, peak method tracking the valleys of the spectrum without and with time smoothing. Time smoothing was done by averaging over five frames for each token. Only test results are given.

The settings for the peak method were

- Frequency window width minimum: 150 Hz
- Frequency window width maximum: 300 Hz
- Halflength: 3 (-3 to track valleys in the spectrum)

The testing was done for male, female, children, male+female vowels, and the results are presented as a bar graph. The coordinate axis has the classification rates and the abscissa (x) axis has the different methods. The graphs were plotted using Microsoft Excel.

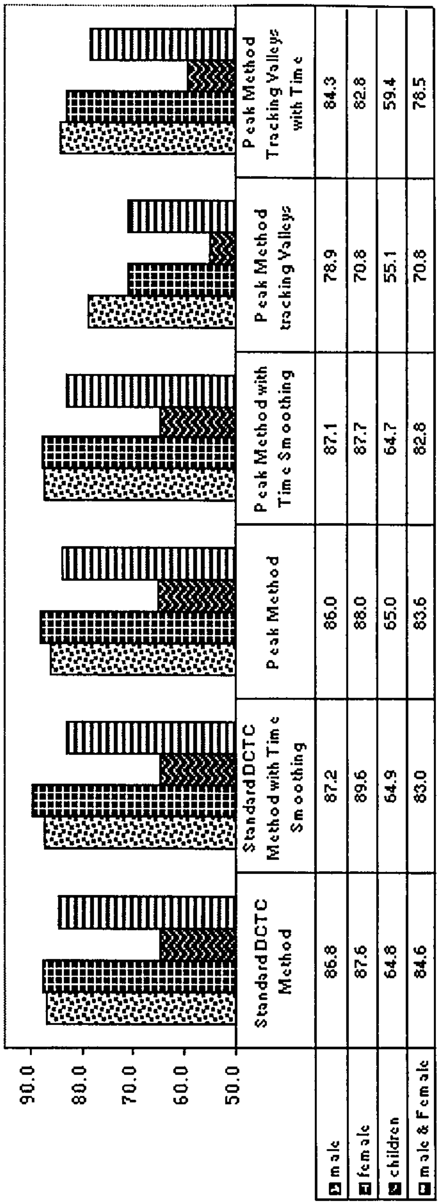
From the bar graph of figure 13, it can be concluded that the peak method performs comparably to the standard DCTC method for automatic vowel classification. However classification rates obtained from valley envelope parameters, without time smoothing, were distinctly worse. Time smoothing of valley envelope information considerably improved classification rates based on this information. Since time smoothing was found to generally improve classification rates, it is used in all further experiments.

Experiment 2:

The objective of the second experiment was the optimization of frequency window width and halflength for male, female, and child speakers using an SNR of 5 dB. The frequency window width minimum values were 75 Hz, 100 Hz, 150 Hz and 200 Hz, and maximum values are 200 Hz and 300 Hz respectively. A total of eight frequency combinations for halflengths of 1, 2, 3, 5, 9 were tested for the three speaker types. Table 1 shows the classification rates for all the combinations.

On a first inspection, the classification rates are quite similar for all parameter setting tried for the male and female speakers. For the case of the

Experiment 1 : Vowel Classification Rates for Various Spectral Processing Methods for different Speaker Types



	Standard DCTC Method	Standard DCTC Method with Time Smoothing	Peak Method	Peak Method with Time Smoothing	Peak Method tracking Valleys	Peak Method Tracking Valleys with Time
▤ male	86.8	87.2	86.0	87.1	78.9	84.3
▦ female	87.6	89.6	88.0	87.7	70.8	82.8
▨ children	64.8	64.9	65.0	64.7	55.1	59.4
▩ male & Female	84.6	83.0	83.6	82.8	70.8	78.5

Figure 12

children, there is more variability in classification rates as a function of the parameter settings. For the male speakers, the maximum classification rate was found to be for frequency widths of 75/100 and 100/300, both with a half_length of 1. For female speakers, the classification rate was the highest of all the speakers --- 84.5% for a frequency width of 200/300 with a halflength of 1. For children, the highest classification rate was 57.7% for a frequency width of 200/200 and a halflength of 3.

It is also observed that the classification rate is low for a halflength of 9 independent of the frequency width. Note that a half width of 9 implies that nearly all points in the spectrum are used, so this Half_length corresponds to a standard DCTC method. For the high pitched voices of women and children, it can be seen that larger window widths work better.

Experiment 3:

The objective of the third experiment was to examine performance as a function of signal to noise ratio for two control methods and the peak envelope method. The "best" parameters settings obtained in experiment two were used. The signal to noise ratio used was varied in steps of 5 dB, from 25 dB to –10 dB, for each speaker type. The 25 dB case represents clean speech since the noise level is still very low. This was done both for the standard DCTC method and the peak method with time smoothing, for male, female and child speakers. The half length's and frequency window widths for each speaker type are the values corresponding to the highest classification rate from experiment two.

The best parameter settings found were:

- Male --- halflength of 1, frequency widths of 100/300
- Female --- halflength of 1, frequency widths of 200/300
- Children --- halflength of 3, frequency widths of 200/200

One control method was the standard DCTC method, as described in chapter 2. The second control method was to compute DCTCs using the entire spectrum, but after first envelope tracking using a method similar to that described by Paul in chapter 2.

The plots of test classification rates versus SNR are shown in figures 13, 14 and 15. It is observed that the peak method is superior to the standard DCTC method, particularly for low SNR values. However the DCTC method applied to the entire spectrum, but preceded by envelope tracking, is quite similar to the peak method. The classification rates were found to decrease as noise was increased. For males all the three methods performed similarly. However, for both females and children, the peak method

EXPERIMENT 2 :OPTIMIZATION OF PARAMETERS FOR THE SPECTRAL PEAK ENVELOPE METHOD

SPEAKER TYPE	FREQ. WIDTH WINDOW MIN/MAX	75/100	75/300	100/200	100/300	150/200	150/300	200/200	200/300
	HALFLENGTH								
MALE	1	82.8	**83.7**	82.8	**83.7**	81.4	82.8	79.4	79.1
	2	81.9	82.6	82.3	83.6	82.2	82.8	82.8	81.6
	3	81.8	82.3	82.8	82.1	81.9	82.8	82.6	82.9
	5	81.9	82.3	82.5	82.9	82.9	82.8	82.8	83.0
	9	82.2	82.1	81.9	81.9	81.9	82.1	82.1	82.1
FEMALE	1	81.2	81.8	82.6	82.5	83.2	83.2	83.5	**84.5**
	2	82.2	81.5	81.9	82.9	83.2	83.3	83.9	83.5
	3	82.2	82.2	81.8	82.1	82.6	81.9	82.4	82.2
	5	80.8	81.9	81.2	81.1	81.1	82.2	82.1	82.8
	9	80.3	80.1	80.8	80.4	80.0	80.1	80.1	80.8
CHILD	1	45.5	45.4	50.7	50.4	55.2	53.9	54.2	52.6
	2	47.7	48.9	47.9	47.4	52.4	54.3	54.6	54.5
	3	48.6	48.9	46.0	48.0	50.4	53.0	**57.7**	56.4
	5	46.0	45.7	46.4	47.3	49.3	51.5	52.6	52.1
	9	44.6	43.9	43.6	42.7	44.8	45.5	45.4	45.4

TABLE 1

with time smoothing and the DCTC method preceded by envelope tracking are superior to the standard DCTC method.

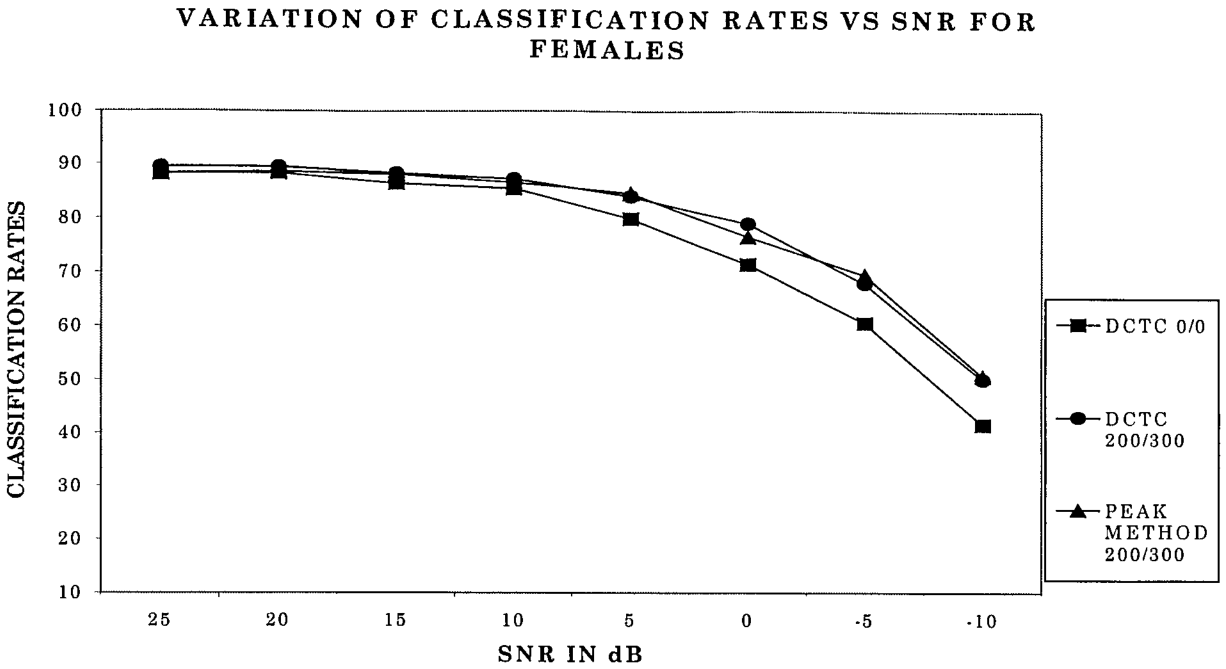


Figure 13. Performance variation of the classification rates of the standard and the peak method under varying SNR for males

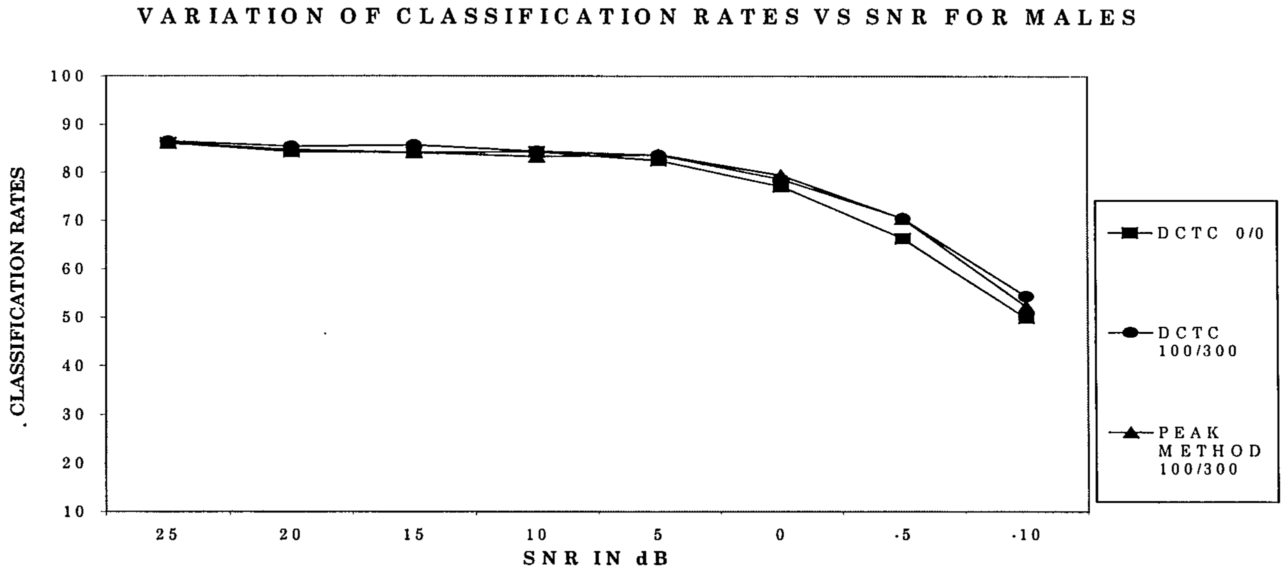


Figure 14. Performance variation of the classification rates of the standard and the peak method under varying SNR for females

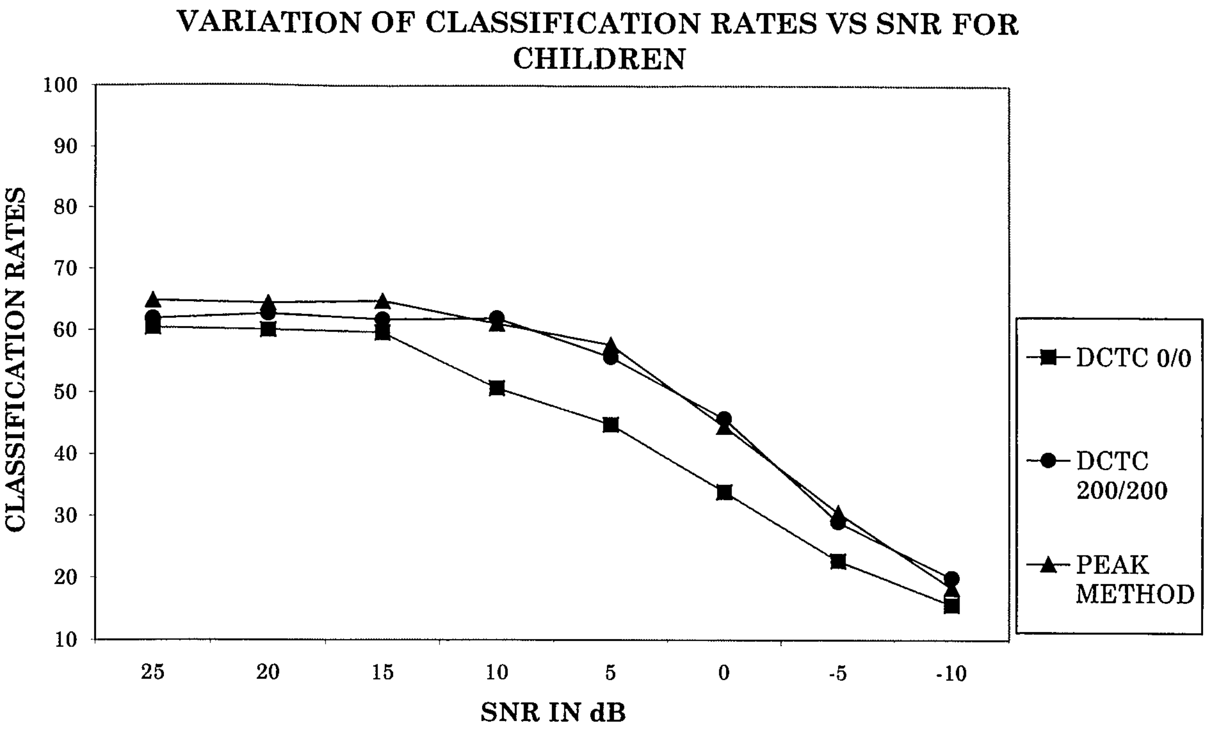


Figure 15. Performance variation of the classification rates of the standard and the peak method under varying SNR for children

4.4 Concluding comments

In this chapter, experiments were described and results presented to evaluate the peak envelope spectral feature method and compare it to the standard DCTC method. Experiments were performed using vowel data obtained from 300 speakers, using a neural network classifier. For the case of clean speech, the new method and the DCTC method are comparable. However, for the case of noisy speech, there do appear to be advantages to using spectral envelope features which are derived primarily from the peaks in the spectrum.

CHAPTER V
CONCLUSIONS AND FUTURE IMPROVEMENTS

5.1 Overview

In this thesis, a mathematical model was designed, implemented and tested to extract spectral envelope features for vowel classification. This model opens a new area of research using peaks as spectral features for automatic vowel recognition. This chapter discusses the achievements of the work and makes suggestions for further research.

5.2 Conclusions & Future work

The following conclusions can be made based on results of several experiments. This research showed that the new features derived from peaks performed similar to the DCTC features for clean speech. They were effective for speech signals degraded by noise. This observation is consistent with basic theoretical considerations, since the peaks are the last parts of the spectrum to be submerged by noise.

The spectral envelopes for each speech frame were tracked smoothly using only the spectral peak information and ignoring other parts of the spectrum. As discussed in section 4.3, the classification rates for the peak method and the standard method are close. Furthermore, the spectral envelopes of the spectral valleys result in significantly lowered classification rates. This leads to the conclusion that spectral peaks carry most speech information and definitely more information than spectral valleys.

This research partially supports the theory proposed by Alain de cheveigne and Hideki Kawahara. According to them, the harmonic peaks contain the information needed for identification, and this can be seen in our results. They also concluded that the vowel identification model should be done using a harmonic sieve based on an estimate of the fundamental frequency. This aspect of their theory was not really tested, since no fundamental frequency information was used in the methods presented in this thesis.

Further research could be done on the time domain version of the model, which is based on the autocorrelation function of the waveform. Both versions of the model emphasize F0-independent pattern matching. This is yet to be proved experimentally.

Suggestions for further research are an estimate of pitch may be used for picking peaks near the multiples of the fundamental frequency, since, according to the missing data model theory, it is these peaks that should have the most valuable information. This method may also be developed for continuous sentence and isolated words. It may also be tested with large databases. White gaussian noise was added to simulate the real time conditions. This may also be extended to study the effects of other types of noises. Speaker dependence may also be investigated.

Another point that should be noted is that the implementation presented does not require orthogonal basis vectors. This is the benefit of using the matrix inverse procedure to solve for coefficients. Thus another avenue for research is to explore the use of non-orthogonal basis vectors, particularly selected to emphasize peak representations for use with automatic speech recognition.

REFERENCES

1. Alain de cheveigne and Hideki Kawahara (1999) " Missing-data model of vowel identification", in Journal of the Acoustic society of America 105 (6), June 1999, pp. 3497-3508

2. Douglas B. Paul (August 1981) " The Spectral Envelope Estimation Vocoder", in IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol.29, No.4, pp. 786-793

3. Lawrence Rabiner and Biing-Hwang Juang (1993) " Fundamentals of Speech Recognition" (Prentice Hall, Englewood Cliffs, New Jersey).

4. Thomas W. Parsons (1987) "Voice and speech Processing" (McGraw-Hill Book Company, New York).

5. Alan V. Oppenheim and Ronald W. Schafer (1989) " Discrete-Time Signal Processing" (Prentice Hall, Englewood Cliffs, New Jersey).

6. Peter Lancaster and Kestutis Salkauskas (1986) "Curve and Surface Fitting An Introduction" (Academic Press Ltd., Oval Road, London).

7. Clare D. McGillem and George R. Cooper (1984) "Representation in Terms of Basis functions" in Continuous and Discrete Signal and System Analysis, Second Edition

8. Stefan Auberg (1996) "Speech Feature Computation for Visual Speech Articulation Training ", Masters Thesis, Old Dominion University.

9. Zahorian S., and Jagharghi, A., (1993) "Spectral-shape features versus formants as acoustic correlates for vowels", J. Acoust. Soc. Amer. Vol.94, No.4, pp. 1966-1982

10. Zahorian, S., Nossair, Z., and Norton, C., (1993) "A partitioned neural network approach for vowel classification using smoothed time/frequency features", Eurospeech-93, pp. II:1225-1228

11. www.mathworks.com

12. Correal, N., (1994), "Real-time Visual Speech Articulation Training Aid", Masters Thesis, Old Dominion University.

13. Zahorian, S. A. and Gordy, P.E. (1983) “Finite Impulse Response (FIR) filters for speech analysis and synthesis”, ICASSP-83, pp.808-811

14. Zahorian, S. A. and Rudasi, L. (1993) “Frequency warping with modified cosine transform basis functions”.

15. Jagharaghi, A. J. (1990) “Automatic speaker-identification of American English vowels based on spectral shape factors versus spectral peaks”, Ph.D. Dissertation, Old Dominion University.

16. J. D. Markel & A. H. Gray, Jr., (1976) “Linear Prediction of Speech”, Springer-Verlag.

17. L.R. Rabiner,“A Tutorial on Hidden Markov Models & selected Applications in speech recognition”, Proceeding of the IEEE, 77(2):257-286, February 1989.

18. G. E. Peterson & H. L. Barney, “Control methods used in the study of the vowels “, Journal of the Acoustic Society of America, 24(2):175-194, March 1952.

APPENDIX

```
function[dctc_pk]= pkreest1(X_magn,ncosbv,bvo,Index)

% fuction to reestimate the signal using the cos bv's and peaks
%
% function[dctc_pk]= pkreest1(X_magn,pk_inds,ncosbv,bvo,i,N_addvls)
%
% Where the Input arguments are
% X_magn    - is the fft array which is to be reestimated
% Index     - vector of 0 and 1 to indicate presence or absence of spectral peaks
% ncosbv    - number of cosine basis vectors
% bvo       - orthonormalized cosine basis vectors

% Output arguments are
% dctc_pk  -  dctc coefficients for single frame of data

% version 0.03  - calls the adjpeak_vales function which includes
%               additional values to make the reestimated signal stable
%
% Programmer  -  Jaishree.V
% version     -  0.04
% Date        -  04/11/2000
%
% version 0.04 - modified the function to return only the dctc coeffcients


% scalar multiplication of the index array with the orthonormalized basis vectors

for k =1:ncosbv
  bvi(:,k)=bvo(:,k).* Index;
end;

% calculates the B matrix

b=bvi'* X_magn;

% calculates the A matrix(symmetric) and singular

a= bvo' * bvi;

% Calculate the dctc coefficients

dctc_pk = inv(a) * b;
```

CURRICULUM VITA
for
Jaishree Venugopal

DEGREES:

Bachelor of Science (Electronics and Communication Engineering), Bharathiar University, Coimbatore, Tamil Nadu, India, April 1997

PROFESSIONAL CHRONOLOGY:

Rndsoftech Private Limited, Coimbatore, Tamil Nadu, India
Software Professional, June 2001 - Present
Enterprise Telesys Limited, Coimbatore, Tamil Nadu, India
Software Trainee, May 1998 - July 1998

CONSULTING/PART TIME EMPLOYMENT:

Department of Electrical Engineering, Old Dominion University, Norfolk, Virginia
Research Assistant, October 1998 - May 2000

Department of Electrical Engineering, Old Dominion University, Norfolk, Virginia
Teaching Assistant, September 1999 - December 1999

COURSES TAUGHT DURING LAST FIVE YEARS:

Department of Electrical Engineering, Old Dominion University, Norfolk, Virginia
ECE 284 Digital Design Lab, Fall 1999

PUBLICATION:

Venugopal J., Zahorian S. A., and Karnjanadecha M., "Minimum Mean Square Error Spectral Peak Envelope Estimation for Automatic Vowel Classification," Proc. ICSLP 2000, vol. 2, pp. 1081-1084, Beijing, China, Oct 16-20, 2000