ABSTRACT

A NEW ROBUST ALGORITHM FOR ISOLATED WORD ENDPOINT DETECTION

Lingyun Gu
Old Dominion University, 2002
Director: Dr. Stephen A. Zahorian

Absolute value energy and Teager energy are two widely used approaches in word endpoint detection. Both have been used recently for locating the endpoints for isolated words. However, each of them has some drawbacks for speech in noisy environments. This work proposes a novel method to combine these two approaches to locate endpoint intervals and yet make a final decision based on energy, which requires far less time than the feature based methods. After introducing the background of isolated word endpoint detection and some important literature from several typical algorithms, the new algorithm description is given in detail. Then an experimental evaluation which compares the automatically determined endpoints with those determined by skilled personnel is presented. It is shown that the accuracy of this algorithm is quite satisfactory and acceptable. However, improvement is still possible to increase the performance of this algorithm. Suggestions for future work are included in this final part of this thesis.

ACKNOWLEDGMENTS

I would like to thank my advisor Dr. Stephen A. Zahorian for his invaluable guidance and support. I feel really fortunate to have such a precious opportunity to work with someone with so many ideas and such a sharp mind. Without his help, my M.S. degree and this thesis would have been impossible. During my years at Old Dominion University, I was always moved not only by his diligence, but also by his magnificent personality. He led me into the realm of speech recognition and let me taste the interests and success of research. A Chinese adag says “One day as my advisor, entire life as my father.” Dr. Zahorian highly qualified deserves this adage.

Another very important person behind the work that led to this thesis is my dear wife, Zhangjing Gu. During the difficult time after I arrived in Florida, she gave me so much support and did everything she could to help me put all my emphasis on this work and thesis writing. Thanks for Bodhisattva letting me have such a considerate and beautiful wife.

A special thank you will also go to my parents. Although they are in China, thousands of miles far away from America, I always feel their true love. The feeling is the same just as if I was still in their bosom when I was a baby.

I also want to thank Mr. Meng and Danielle. They provided me so much useful data and so many materials, which were very useful for my work.

In addition, I would like to thank all my colleagues in the Speech Communication Lab for their help with knowledge, comments and suggestions.

This work was partially supported by NSF grant BES-9977260.

TABLE OF CONTENTS

Page

LIST OF TABLES

Page

LIST OF FIGURES

Page

CHAPTER ONE

INTRODUCTION

1.1 The Importance of Endpoint Detection

Endpoint detection, which aims to distinguish the speech and non-speech segments of a digital speech signal, is considered as one of the key preprocessing steps in automatic speech recognition (ASR) systems. One major factor in overall recognition performance is endpoint accuracy. Proper estimation of the start and end of the speech (versus silence or background noise) avoids the waste of ASR evaluations on preceding or ensuing silence. Conversely, accurate endpoint detection leads to efficient computation and, more importantly, to accurate recognition since proper endpoints will result in good alignment for template comparison [1, 8].

In general, for either isolated word or continuous speech recognition systems, incorrect endpoint detection of an utterance can produce two negative effects [2]:

1) Recognition errors because of the incorrect boundaries;
2) Increased computations "wasted" on the non-speech events in the utterance.

Currently, research on continuous speech recognition is a more focused discipline than research in other areas, such as isolated word recognition. Nevertheless, there are still numerous areas that require the application of the isolated word recognition systems.

Accurate endpoint detection is particularly important for these isolated word recognition systems.

Among various endpoint detection approaches, energy-based methods are the most widely used to solve the problem. The basic idea of using energy to detect endpoints is that, in a clean environment, the energy (or power) of a speech segment is higher than a non-speech one in a speaker's utterance. In these methods, a fixed-length window is defined first. It is then "slid" over the duration of the input utterance, usually with a frame spacing less than the frame length. By continuously monitoring the utterance through the window, the starting point can be found once when the short-time energy of the window is higher than some beginning threshold. Similarly, the ending point can be located when the short-time energy of the window is lower than some ending threshold [3, 7].

In this thesis, our research goal is to present, develop and test a robust algorithm for reliably determining endpoints of isolated words, even in the presence of some noise. This algorithm should also be based on computationally-efficient energy based parameters, rather than features which are a more computationally demanding [1]. The endpoint detection can be viewed as a speech/background-noise classification. For ASR systems, the ideal characteristics for such classification are: reliability, robustness, accuracy, adaptation, simplicity, real-time processing and no a priori knowledge of the noise [8].

One of the specific applications of isolated word recognition is the Visual Speech Articulation Training Aid under development in the Speech Lab of the Electrical and

Computer Engineering Department of Old Dominion University. The process for "training" this aid requires the collection and processing of a large database of isolated words, which must be properly endpointed to enable accurate training.

1.2 Research Objective

Isolated word recognition is one basic class of speech important for satisfactory speech recognition. There are many potential applications, such as remote data entry via voice commands, that would be enabled by very accurate isolated word recognition. However, except for vowels, most isolated words, such as digits and CVCs (consonant-vowel-consonants), are difficult to accurately determine endpoints for, since many consonants are very low energy. However, most practical speech-recognition systems rely heavily on isolated word recognition for these difficult words. For example, for the digit "eight," the final endpoint detection could easily miss the final weak portion ("t"), especially in strong noise background. Therefore, the detection of the weak points of an utterance and the separation from the background noise is the real challenge.

The main objective of this research is to develop a good general endpoint detection method, which will be accurate, or any type of speech utterance. However, since endpoint detection accuracy is most important for isolated words, and because the immediate application of this algorithm is for isolated words, the focus of our work is an algorithm for isolated words. In this work, we wish to emphasize good endpoint detection for sounds which begin, and/or end with weak consonants. The data of particular interest

includes the spoken versions of the 10 digits, 26 letters of the alphabet, 12 CVC sounds (Consonant Vowel Consonant) and 13 vowels. An important goal of this research work is to support the development of the "Visual Speech Articulation Training Aid," a project undertaken by the Speech Communication Lab in the Electrical and Computer Engineering Department of Old Dominion University. The software is able to run on a Windows Multimedia Personal Computer without any specialized hardware. The articulation training aid was developed using neural networks trained to recognize "correct" productions from a large database of speakers. A critical step in the training process is endpoint detection of the training database. The version of endpoint detection used for this database (the method will be discussed in details in chapter two), previous to the routines developed in this thesis, was found to have frequent errors, which provided the motivation for the current work.

This thesis presents a simple and fast algorithm for accurately locating the endpoints of isolated words spoken in an office environment. This algorithm is based on two approaches, Teager Energy and absolute value, which have recently been used for locating the endpoints of an utterance. Since each of them has some drawbacks for speech in noisy environments, our algorithm is a novel method to combine these two approaches to locate endpoint intervals and yet make a final decision based on energy, which requires far less time than the feature based methods. After the algorithm description, an experimental evaluation is also presented, comparing the automatically determined endpoints with those determined by skilled personnel. It is shown that the accuracy of this algorithm is quite satisfactory and acceptable.

A large portion of this research and the most obvious achievement is the improvement of the software for signal processing algorithms for the Visual Speech Articulation Training Aid. The work also includes the improved endpointing of the entire VATA training base (over 10000 files), and the retraining of the VATA with the neural network software. Thus, the final result along these lines is better performance for VATA.

1.3 Overview of Automatic Speech Recognition

Automatic recognition of speech by machine has been a goal of research for more than four decades. However, in spite of the glamour of designing an intelligent machine that can recognize the spoken word and comprehend its meaning, and in spite of the enormous research efforts spent in trying to create such a machine, we are far from achieving the desired goal of a machine that can understand spoken discourse on any subject by all speakers in all environments. Dr. Stephen Zahorian summarized this state of affairs as: "The average 3-year-old kid still understands speech better than the best software program." Because we don't know how to solve the complete challenge of speech recognition, ODU's speech lab's goal is to conduct a series of research tasks on the fundamental principles underlying speech-recognition systems so as to provide a framework from which researchers can expand the frontier. Also, as mentioned before, a specific goal at ODU is to improve the Visual Speech Articulation Training Aid software as much as possible, so that it can really benefit the hearing impaired.

A general model for speech recognition [30], shown in Figure 1, is commonly used. The model begins with a user creating a speech signal to accomplish a given task. The spoken output is first recognized in that the speech signal is decoded into a series of words that are meaningful according to the syntax, semantics and pragmatics of the recognition task. The meaning of the recognized words is obtained by a higher-level processor, which uses a dynamic knowledge representation to modify the syntax, semantics, and pragmatics according to the context of what it has previously recognized. In this manner, things such as nonsequitors are omitted from consideration at the risk of misunderstanding, but at the gain of minimizing errors for sequentially meaningful inputs. The feedback from the higher-level processing box reduces the complexity of the recognition model by limiting the search for valid input sentences from the user. The recognition system responds to the user in the form of a voice output, or equivalently, in the form of the requested action being performed, with the user being prompted for more input.

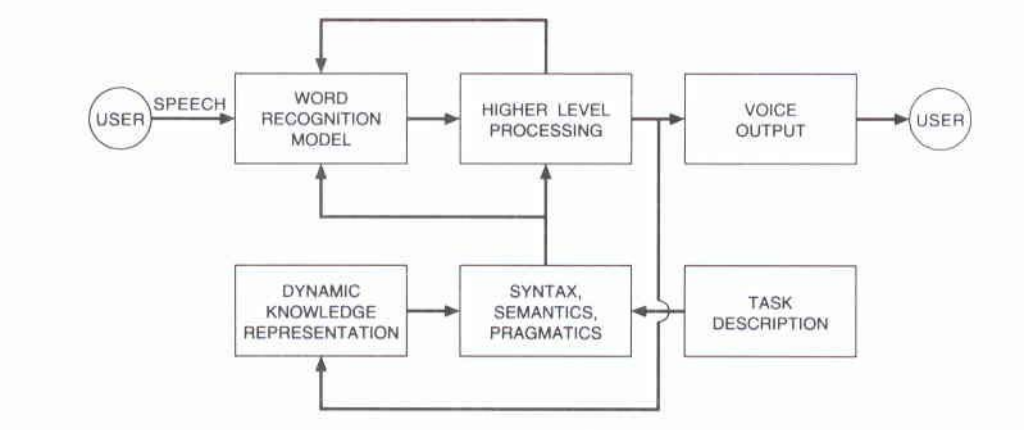

Figure 1: General Block Diagram of a Task-oriented Speech-recognition System [30]

Figure 2 gives a basis overview of a traditional endpoint detection scheme. First, the input utterance is changed from an analog signal to discrete-time data through an A/D converter with a fixed sampling frequency. Then the signal is segmented into overlapping frames. Then endpoints are computed, usually using some empirically determined thresholds, for the beginning and end of the utterance [3].

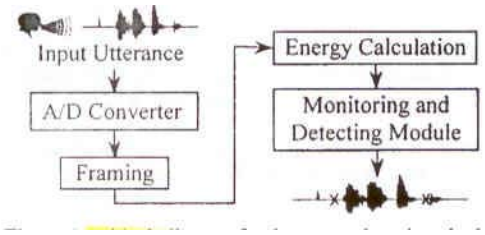

Figure 2: A Block Diagram for the Energy-based methods [3]

1.4 A Brief Introduction to the Development on Speech Recognition

Research in automatic speech recognition by machine has been done for almost four decades. To gain an appreciation for the amount of progress achieved over this period, it is worthwhile to briefly review some research highlights.

The earliest attempts to devise systems for automatic speech recognition by machine were made in the 1950s, when various researchers tried to exploit the fundamental ideas of acoustic-phonetics. In 1952, at Bell Laboratories, Davis, Biddulph, and Balashek built a system for isolated digit recognition for a single speaker. The system relied heavily on measuring spectral resonances during the vowel region of each digit [24].

In the 1960s several fundamental ideas in speech recognition surfaced and were published. The first of these achievements is owed to Martin and his colleagues at RCA Laboratories, who develop realistic solutions to the problems associated with nonuniformity of time scales in speech events. The second happened in USSR, where Vintsyuk proposed the use of dynamic programming methods for time aligning a pair of speech utterances. The final achievement of note in the 1960s was the pioneering research of Reddy in the field of continuous speech recognition by dynamic tracking of phonemes [24].

In the 1970s speech-recognition research resulted in a number of significant milestones. First, isolated word or discrete utterance recognition became a viable and usable technology based on fundamental studies by the researchers in Russia, Japan and U.S.A. The now well-known technique of Linear Predictive Coding (LPC) was first used in that decade, as well as important pattern recognition techniques. At the same time, the research on large vocabulary speech recognition was also in its infancy [24].

Just as isolated word recognition was a key focus of research in the 1970s, the problem of connected word recognition was a focus of research in the 1980s and 1990s. In these two decades, new milestones have been reached one by one. Statistical modeling methods, especially hidden Markov model approach and the new use of neural networks, were the source of most of the improvements. Accuracy of both isolated word recognition and continuous-speech slowly but continuously improved. This whole field of research was

fueled by the dramatic improvements in computer power and reduction in price. Along the way, many new ideas for isolated word recognition research emerged, and this type of ASR also reached a new level [24].

To the particular topic of endpoint detection for isolated words, there have many attempts to "solve" the problem over the past several decades. Computing the energy of speech signal is a computationally simple operation compared to extracting other features, such as LPC derived cepstrum coefficients (LPCC), mel-frequency cepstrum coefficients (MFCC) and so on, which have been found to work well for endpoint detection but are time and computationally intensive. As for energy-based methods, most of the algorithms are based on simple parameters such as energy contours and zero crossings [1]. Thus, a remaining challenge is to retain the computational advantages of energy based methods while preserving and even improving accuracy.

1.5 Overview of the Following Chapters

In this section, we give an overview of the following chapters in this thesis.

The main point of chapter two is a review of the existing endpoint detection algorithms for isolated words as well as the basic concepts of endpoint detection and the problems we need to solve.

In Chapter three, the "heart" of this thesis, the new endpoint detection algorithm, which is based on the Teager energy and absolute value methods, is presented.

Chapter four presents a series of experiments designed to evaluate the effectiveness of the isolated words recognition and the results are evaluated.

In the last chapter, chapter five, conclusions and suggestions for further research are offered.

CHAPTER TWO

TECHNICAL BACKGROUND

2.1 The Process of Speech Production and Perception in Human Beings

Before we proceed with the main portion of this chapter, a brief summary of some of the existing isolated words endpoint detection algorithms, we will discuss the basic process for the production of speech signals and some basic concepts of endpoint detection.

Sound waves are caused when a vibrating body causes a mechanical oscillation that disturbs the surrounding air causing variations in air pressure. Although sound is sometimes defined as the air vibrations audible to man, this definition may be restrictive, especially when dealing with applications for the hearing impaired [30] (Figure 3).

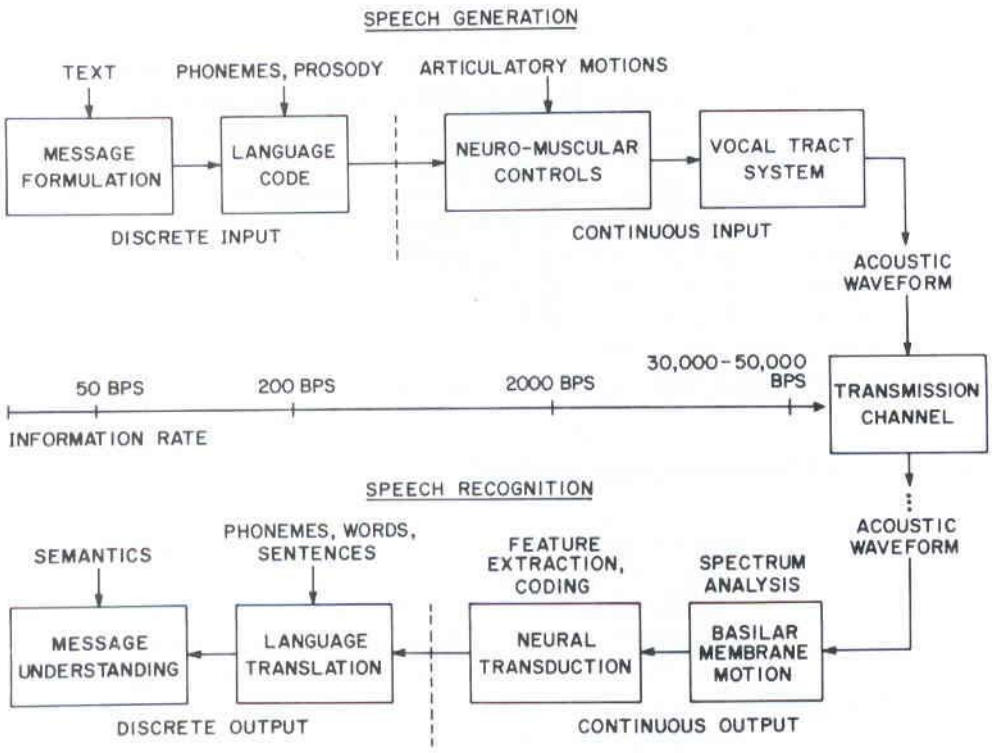

Figure 3: Speech Production and Perception Process [30]

The objective of speech production is to generate meaningful sound combinations. Speech sounds are created by air passing through our sound production organs. The stream of air is modified by movements of our jaw, lips, tongue, pharynx and vocal folds in a way that it becomes audible and meaningful to a listener.

Speech sounds are usually classified as vowels and consonants. Generally vowels are voiced sounds whose spectral characteristics are determined by the size and shape of the vocal tract. Modifying the shape of the vocal tract changes its filtering characteristics, which in turn changes the frequencies at which the speech signal is enhanced or de-emphasized. Consonants are produced by a partial or complete obstruction somewhere

along the vocal tract. The turbulence that is generated causes the sound to be quasi-periodic or aperiodic and noise-like (Gelfand, 1981) [30].

The speech wave generated by the speech organs is transmitted through the air to the ears of listeners. At the ear, it activates the hearing organs to produce nerve impulses, which are transmitted to the listener's brain through the auditory nerve system. The same speech is transmitted to the speaker's ears as well, allowing him to continuously control his vocal organs by receiving his own speech as feedback. The critical importance of this feedback mechanism is clearly apparent for people whose hearing has become disabled for over a year or two (Furui, 1989). It is also evident by the fact that is very hard to speak when our own speech is fed back to our ear with a certain amount of time delay [30].

2.2 Endpoint Detection Problems

The goal of speech detection is to separate acoustic events of interest in a continuously recorded signal from other parts of the signal. A key question in speech recognition is how accurately speech must be detected so as to provide the "best" speech patterns for recognition. The definition of "best" here is the pragmatic one---namely, the patterns that provide highest recognition accuracy.

For speech produced in the most benign circumstances---that is, carefully articulated and spoken in a relatively noise-free environment---accurate detection of speech is a simple

problem. However, this is not usually the case. In practice, one or more problems usually make accurate endpoint detection difficult. One particular class of problems is those attributed to the speaker and to the manner of producing the speech. For example, during articulation, the speaker often produces sound artifacts, including lip smacks, heavy breathing and mouth clicks and pops. Figure 4 and Figure 5 show examples of this type of humanly produced sound artifact. The top part of each figure is the energy contour of the utterance on a logarithmic (dB) scale, and the lower part is the time waveform of the corresponding utterance [30].

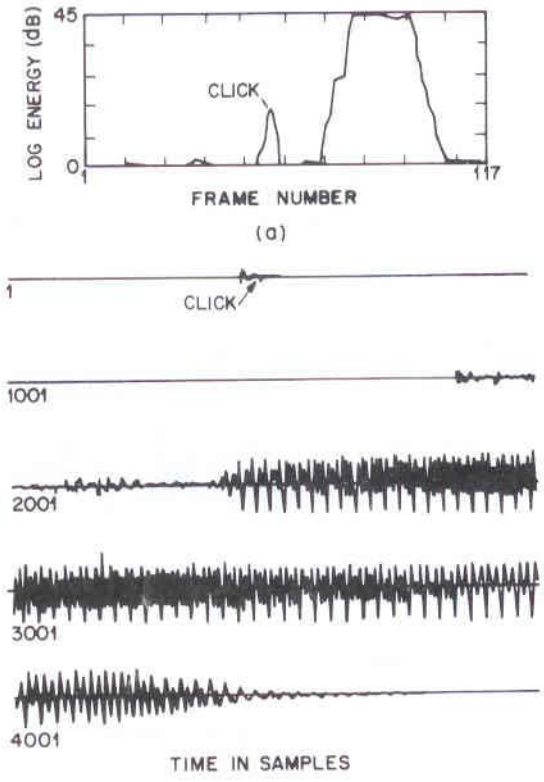

Figure 4: Example of Click Noise [30]

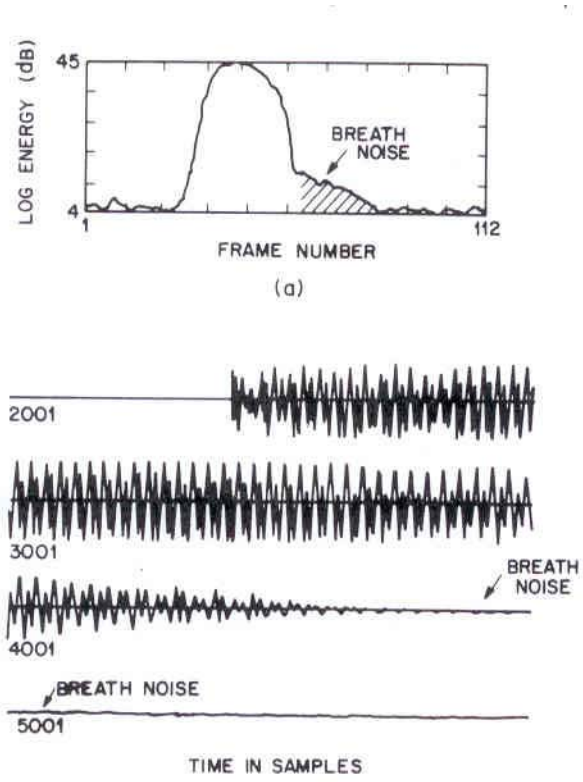

Figure 5: Example of Breath Noise [30]

A second factor making reliable speech endpoint detection difficult is the environmental conditions in which the speech is produced. The ideal environment for talking is a quiet room with on acoustic noise or signal generators other than that produced by the speaker. Such an ideal environment is not always practical; hence, one must consider speech produced in noisy backgrounds, in non-stationary environments, with speech interference, and in hostile circumstances. Some of these interfering signals possess as much speech-like quality as the desired speech signal itself, making accurate endpoint detection quite difficult [17, 30].

A final source of signal degradation is the distortion introduced by the transmission system over which the speech is sent. Factors like cross talk, inter-modulation distortion, and various types of tonal interference arise to various degrees in the communications channel, again adding to the inherent difficulties in reliably detecting speech endpoints [17, 30].

The following portion of this chapter summarizes several robust or milestone algorithms which have relatively successful in combating the above-mentioned endpoint detection problems. The new robust algorithm of this thesis, which will be presented in the third chapter, is based on four of these existing algorithms and combines advantages from them to obtain an even better overall algorithm.

2.3 Algorithms for Endpoint Detection

Many of the published works deal with the problem of voiced-unvoiced-silence detection, which is sometimes dealt with as a part of speech segmentations (Rabiner and Sambur, 1977; Ramamoorthy, 1980; DeSouza, 1983; Mwangi and Xydeas, 1985; Biing-Hwang Juang, 1993; Chin-Hui Lee, 1995; Owen Kimball, 1996; John Hansen, 1999). A lesser number of published works deals with endpoint detection for isolated word recognition (Rabiner and Sambur, 1975; Lamel et al., 1981; Wilpon et al., 1984; Savoji, 1989; Chollet, 1995; Loizou and Spanias, 1996; Rahim and Biing-Hwang Juang, 1997; Kim and Kil, 1999). Voice activation and silence detection have been included in effective coding schemes (Cho and Un, 1982; Don Vito and Schoenherr, 1985; Lynch et al., 1987,

Compernolle, 1998) while some endpoint detectors have been devised based on coding techniques (Das and Tappert, 1978; Tsao and Gray, 1984; Kim and Un, 1995; Womack and Hansen, 1999).

When we search the literature for endpoint detection, the paper most widely referenced is that by L.R. Rabiner and M.R. Sambur in 1975 [17]. In this paper, the author proposes a fairly simple algorithm for locating the beginning and end of an utterance, which can be used in almost any background environment with signal-to-noise ratio of at least 30 dB. The algorithm is based on two measures of speech: short-time energy and the zero crossing rate. The algorithm possesses the feature that it has somewhat self-adapting thresholds for its decision criteria from measurements made directly on the recorded interval. The first step of this algorithm is to define how the energy and zero crossing rate are measured. The speech energy is defined as the sum of the magnitudes of 10 ms of speech centered on the measurement interval. The choice of a 10-ms window for computing the energy and the use of a magnitude function rather than a squared-magnitude function were dictated by the desire to perform the computations in integer arithmetic and, thus, to increase speed of computation. Further, the use of a magnitude de-emphasizes large-amplitude speech variations and produces a smoother energy function. The zero crossing rate of the speech is defined as the number of zero crossings per 10-ms interval. Although the zero crossing rate is highly susceptible to 60-Hz hum, dc offset, etc., in most cases it is a reasonably good measure of the presence or absence of unvoiced speech. Then, it is assumed that during the first 100ms of the recording interval there is no speech present. Thus, during this interval, the statistics of the background

silence are measured. These measurements include the average and standard deviation of the zero crossing rate and the average energy. If any of these measurements are excessive, the algorithm halts and warns the user. Otherwise, a zero crossing threshold, IZCT, is chosen as the minimum of a fixed threshold for unvoiced speech. The values used are IF (25 crossings per 10ms), and the sum of the mean zero crossing rate during silence, IZC, plus twice the standard deviation of the zero crossing rate during silence. Then, the peak energy, IMX, and the silence energy, IMN, are used to set two thresholds, ITL and ITU. A value I1 is set to be at a level which is 3 percent of the peak energy, whereas, I2 is set to be at a level which is four times the silence energy. The lower threshold, ITL, is the minimum of these two conservative energy thresholds, and the upper threshold, ITU, is five times the lower threshold. The algorithm begins by searching from the beginning of the interval until the lower thresholds is exceeded. This point is preliminarily labeled as the beginning of the utterance unless the energy falls below ITL before it rises above ITU. Should this occur, a new beginning point is obtained by finding the first point at which the energy exceeds ITL, and then exceeds ITU before falling below ITL; eventually such a beginning point must exist. A similar algorithm is used to define a preliminary estimate of the endpoint of the utterance. We call these beginning and ending points N_1 and N_2, respectively. By using the zero crossing rate, the algorithm proceeds to examine the interval from N_1 to N_1-25 and counts the number of intervals where the zero crossing rate exceeds the threshold IZCT. If the number of times the threshold was exceeded was three or more, the starting point is set back to the first point (in time) at which the threshold was exceeded. Otherwise, the beginning point is kept at N_1. A similar search procedure is used on the endpoint of the

utterance to determine if there is unvoiced energy in the interval from N_2 to $N_2 + 25$. Then the endpoint is readjusted based on the zero crossing test results in this interval. This algorithm is fast since it requires few computations. However, it is unable to compete with the noise even in benign environments and sometimes it is not reliable to distinguish a weak fricative from background noise.

The reason we give so many details about the above algorithm is because it is the basis of all the algorithms based on energy. Now, we will present three additional algorithms, which are modified versions of the Rabiner/Sambur one.

In 1991, Evangelos S. Dermatas, Nikos D. Fakotakis and George K. Kokkinakis presented a simple and fast algorithm for accurately locating the endpoints of isolated words spoken in an office environment [4]. This algorithm is based on energy measures and threshold logic and is adaptive to almost any low noise acoustic environment. The proposed algorithm consists of four steps. In the first, the speech signal corresponding to a single word is preprocessed and the background noise is estimated which is used to adapt the decision threshold values of the following step. In this step, the input signal is pre-emphasized, to eliminate the d-c component and to emphasize the higher frequency components, using an one-zero filter. Then, from samples taken at the beginning and the ending of the input signal, the background noise is estimated. To this end, the energy levels of two wide-length non-overlapping frames at the beginning and two at the ending of the signal are calculated. The noise level at the front-end of the signal (E_F) is estimated using the first two frames. If the difference between the energy levels of the two frames

is equal or less than twice the value of one frame, the noise level is taken to be equal to the mean value of the two frames, otherwise to be the minimum of the energy levels of the two frames. The noise level at the back-end of the signal (E_B) is estimated in the same way, using the last two frames. Finally, the background noise of the input signal (E_N) is estimated using the noise levels at the front and back ends. If the difference between the two values is equal to or less than twice the value of one level, the background noise is taken to be equal to the mean value of the two noise levels, otherwise the background noise cannot be estimated. In the second case, this algorithm is to locate the first and the last voiced sound. Searching the amplitude-time function from left to right with an 80 msec frame, in a 0.1 msec step, the first frame with V peaks above an amplitude threshold T_A is assumed to lie at the beginning of the first speech sound of the utterance. In the same way, the ending-point can be found. In the third, the focus is on the location of low energy areas. An 80 msec frame is moved backwards, from the front voiced sound starting-point, in a 0.1 msec step, to calculate the energy contour of the signal. This energy function is compared to two thresholds in order to locate the boundaries of the front low energy area. The two experimentally derived energy thresholds are 1.1 E_N and 2.2 E_N, yielding the corresponding points t_{F1} and t_{F2}. The same method is used to find the low energy area at the end, except the two experimentally derived energy thresholds are 3.33 E_N and 3.0 E_N, resulting in corresponding points t_{B2} and t_{B1}. The final step of this algorithm is the endpoint detection. At the front area, from point t_{F1} to t_{F2}, the contour of the energy ratio of two successive non-overlapping 30 ms frames is estimated, in a 0.1 ms step. The actual beginning of the speech signal is chosen as the maximum value of the energy ratio contour. The ending point is obtained by the same method using the interval

from point t_{B2} to t_{B1}. This algorithm is also fast, requiring few computations. However, it must estimate the background noise. The requirement sometimes is not satisfied when there is not "enough" silence at the beginning or ending.

In 1993, G.S. Ying, C.D. Mitchell and L.H. Jamieson presented a novel algorithm to detect the endpoints by using the Teager Frame Energy [2]. In this paper, the new algorithm is based on a new energy measure, Teager Frame Energy, which replaces the two traditional features, ZCR (Zero Crossing Rate) and energy in an endpoint detection algorithm. Teager showed that the "voiced part" signal energy is not only determined by the amplitude of the signal samples, but also by the oscillation frequency, which is capable of responding rapidly to changes for fricatives and plosives with very low amplitude. The algorithm is quite simple. First, the power spectrum is calculated. Secondly, each sample in the power spectrum is weighted by the square of the frequency. Thirdly, the square root of the sum of the weighted power spectrum is computed. The result after the third step is the Teager Frame Energy, from which a pair of thresholds can be used to make the final decision. This algorithm performs quite well in a mild noise background; however, it still fails in strong noise, especially if the main part is mechanical noise. This algorithm also requires more computations than the previous two, since a power spectrum must be computed.

In 1997, Yiying Zhang, Xiaoyan Zhu, Yu Hao and Yupin Luo [5] extended the algorithm mentioned above by Evangelos S. Dermatas, etc., in 1991. In this paper, they use the same method to estimate the background energy given in [4]. However, the background

ZCR is also estimated as one parameter. Then, they search the energy function from left to right using frame intervals. The first frame whose energy is above an energy threshold T_E is assumed to lie at the beginning of the first voiced sound of the utterance. The point is named P_{F3}. Searching the energy function from right to left, we get the last voiced point P_{B3}. Then, they use the parameter ZCR to relax the endpoints. Searching the ZCR function from point P_{F3} backwards, the reference starting point P_{F2} is obtained with its ZCR greater than a ZCR threshold Z_k. Using the same approach, they find the reference ending point P_{B2}. Finally, they use the Variable Frame Rate (VFR) technique to compute the Euclidean distance D(i, j) between the current frame i and the last retained frame j and to compare this distance to some threshold T. Searching from frame P_{F2} to P_{B2}, if the Euclidean distances between the current frame and the next, the second next and the third next are all greater than a experimental threshold T_D, they find a starting-point P_{F1}. Then, searching from frame P_{B2} to P_{F2}, they find the ending-point P_{B1}. This algorithm is implemented easily. However, this algorithm uses the unreliable parameter ZCR to make interim decision. Therefore, it is not very robust to combat strong background noise.

The last algorithm we will summarize was introduced in 2000 by Liang-sheng Huang and Chung-ho Yang [3]. In this algorithm, they use a novel and powerful parameter, entropy, which is used widely in communications, to help make the final decision. First, the author calculates each frame's energy E_i by the sum of each sample's square, which is very familiar to readers. Secondly, by computing the FFT for each frame, they calculate the spectrum pdf by normalizing each amplitude of frequency component with the sum of the total frequency component's amplitude in each frame. We call this new parameter p_i for

each frame i as the corresponding probability density. Then the negative entropy H_i of frame i can be obtained by taking the sum of product of p_j and $\log(p_j)$ for all j in the frame i. Thirdly, after obtaining the negative entropy H_i, they use the formula:

$$M_i = (E_i - C_E) * (H_i - C_H)$$

$$\text{EE-Feature} = (1 + |M_i|)^{1/2}$$

to define the final parameter EE-Feature. Then, simple decision logic is used with a pair of thresholds to determine the final endpoints. This algorithm combats background noise quite well, especially the mechanical noise. However, it needs to estimate the background noise and energy, which is sometimes not possible, as mentioned above.

In this thesis, the new endpoint detection algorithm is a simple and accurate algorithm for the detection of the endpoints of isolated words spoken in noise environment. This algorithm is based on Teager Frame energy measures and absolute value approach. In our new algorithm, it is sensitive to the fricative and plosive with low amplitude and has a great shape to combat with the background noise. It also achieves satisfactory and acceptable accuracy for all alphabets, digits, vowels and most CVCs.

CHAPTER THREE

ALGORITHM DESPCRIPTION

3.1 Introduction

As mentioned before, each of the existing algorithms has both advantages and some disadvantages. In this chapter, we will present a new algorithm, which combines several merits of the algorithms discussed in chapter two, and which has good performance, such as high sensitivity for the beginnings of fricatives and plosives, good performance to combat noise, etc. However, there are still some shortcomings, which we will discuss and compare with those of the other algorithms.

This algorithm developed in this thesis combines the Teager Energy and absolute energy methods (AETE algorithm). This is a fast, accurate algorithm intended for use in moderate noise environments. In this thesis, we describe our implementation of this algorithm, including several steps not used in previous implementations.

3.2 Basic Structure for AETE Algorithm

In this algorithm, there are three main parts. The first part is the pre-processing and computations used to determine the smoothed absolute value and Teager energy of the

speech signal. The second stage of the algorithm is the computation and decision logic used to estimate the background noise and determine the beginning and ending region. The end result of this second stage of processing consists of four pairs of endpoints (boundaries for beginning region and ending region using the absolute value, and boundaries for beginning regions and ending region using the Teager energy). The "true" endpoints are presumed to be within the regions determined by these parameters. Finally, based on these four pairs of endpoints, the final endpoint decisions are made. In the following subsections, we will develop these steps, combining the discussions for the absolute value and Teager energy, so that we can compare the methods more easily.

3.3 The Algorithm Description

3.3.1 Signal Pre-Emphasis

First, we use a single pole-pair filter to pre-emphasize the speech signal, which has previously been found to be effective for pre-filtering of speech data prior to automatic speech recognition. The center frequency is typically 3000 Hz and the radius of the pole-pair is 0.8. Second, a bandpass filter (FIR filter of order 150) with a low cutoff frequency of 375 Hz and high cutoff frequency of 5000 Hz is applied to the signal, after using the pre-emphasis filter. This band, very similar to the band of telephone lines, is generally considered to contain the most overall speech information. This type of fixed filtering is reasonably effective for improving the signal to noise ratio of speech to non-speech. For example, for most cases of recorded wave files in the speech lab at ODU, many kinds of

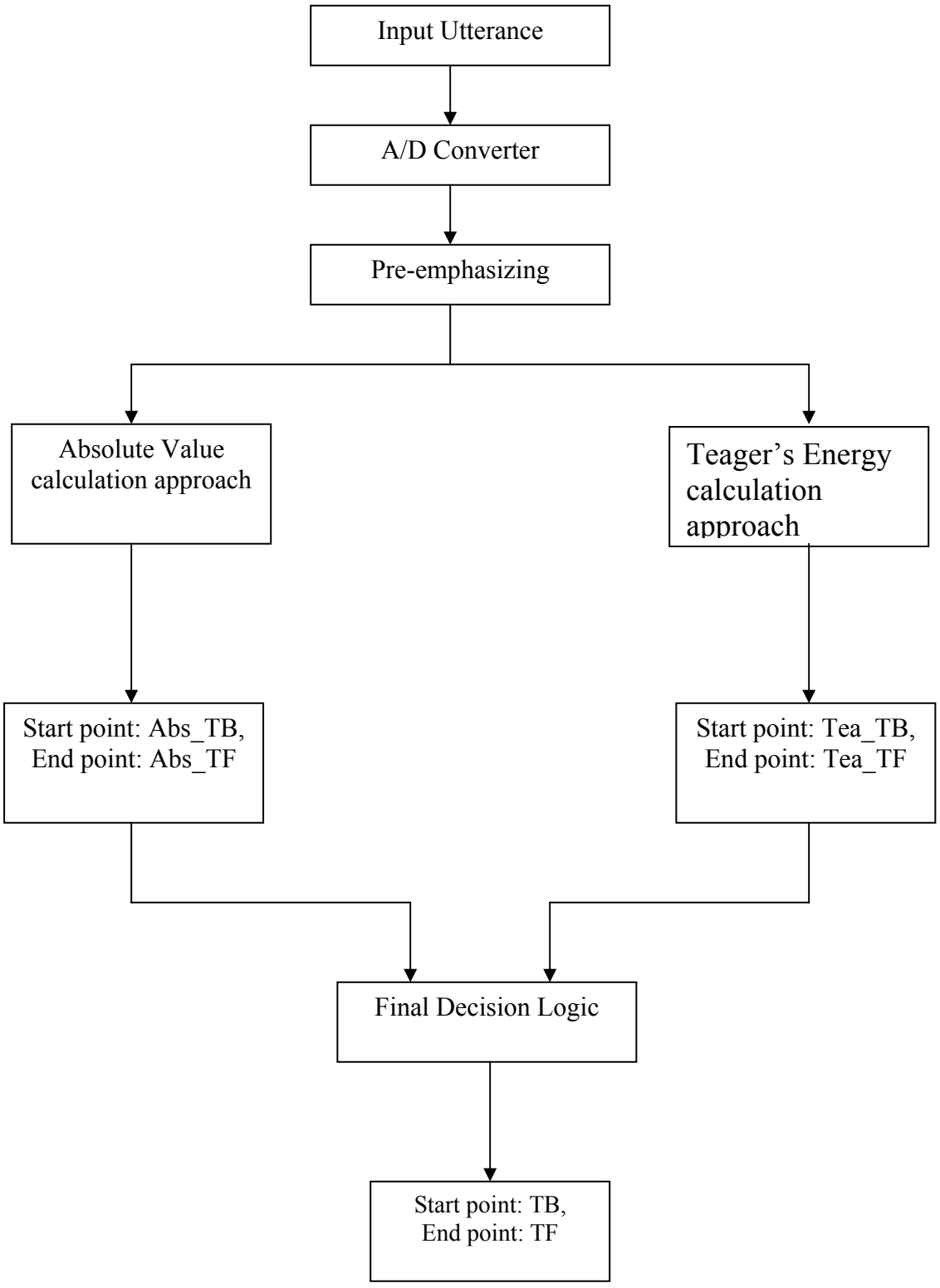

Figure 6: Block diagram of AETE algorithm

noise, such as the heating/cooling system and other machinery noises, tended to be either low frequency noise or high frequency. A big source of unwanted acoustic signals such as breath noises were largely eliminated by this filtering. Thus overall much of the noise from office environments was removed by this filtering approach. This filtering is illustrated for a typical speech signal in Figure 7. Note that the very high amplitude but very low frequency breath noise at the beginning of the signal is eliminated. This filtered signal was used for computing both the absolute value and Teager Energy parameters. Figure 7 depicts an example, which illustrates the benefits of the pre-emphasizing procedure.

3.3.2 Absolute Value and Teager Energy Computation

In this step, we will show the computation of absolute value and Teager energy respectively. We compute the absolute value first for each point of the given signal as

$$E_i = \left| s_i \right|$$

where, s_i is the value for each sampling point. After calculating the absolute value, we lowpass filter it using a 250 point FIR filter with a cutoff frequency of 30 Hz. The resultant signal, after the low-pass filtering, is essentially the envelope of the magnitude of the original speech signal. After this filtering, each speech signal (typically ranging from .5 seconds to 2 seconds in total length for our database of isolated words), was

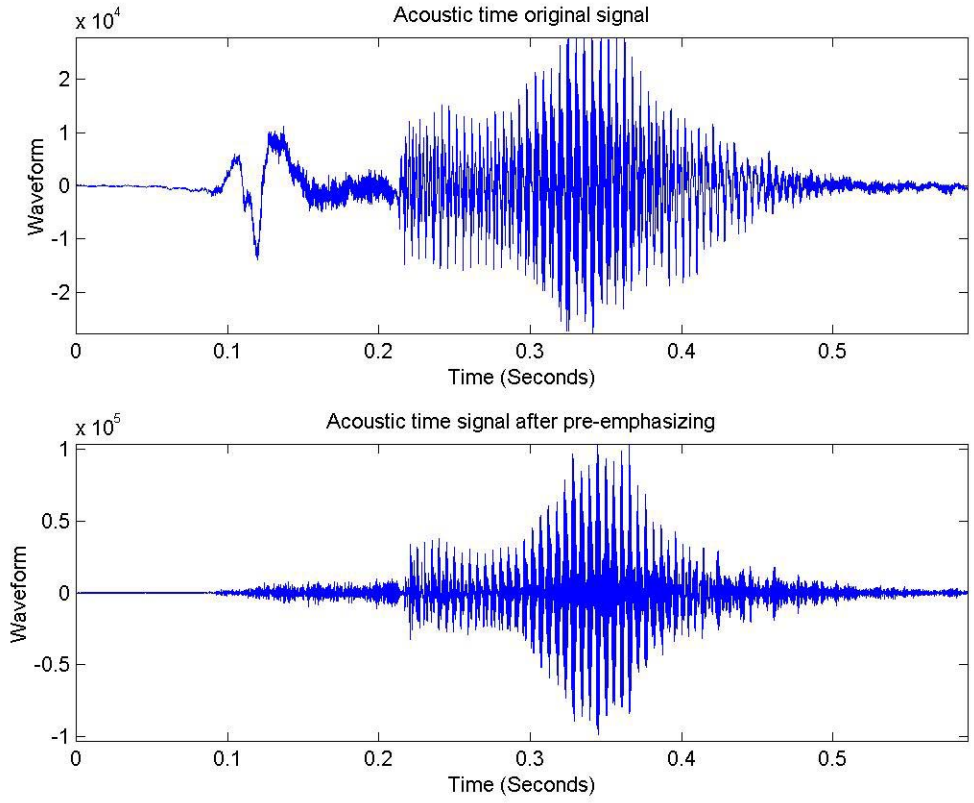

Figure 7: Signal comparison before and after pre-emphasizing and bandpass filtering for Word 'three'

decimated/interpolated to a fixed length of 1000 points. This fixed length is somewhat more convenient for use in the following steps and more than adequate for the required time resolution; it also greatly reduced computations for the following steps. Finally, we also normalize the amplitude of the smoothed absolute value to lie between 0 and 1, using an offset and linear scaling.

After introducing the absolute value computation, we will give some details about the Teager energy computation. In modeling speech production, Teager developed a new algorithm for computing the energy of a signal. This algorithm has been presented by Kasier as Teager's Energy Algorithm [2]. Given a signal with the motion of an oscillatory body, sample x_i is given below:

$$x_i = A\cos(\Omega i + \Phi)$$

Where A is the amplitude of the oscillation, Ω is the digital frequency, and ϕ is the initial phase. In Teager's algorithm, the instantaneous energy T_i of the sample x_i is:

$$\begin{aligned} T_i &= x_i^2 - x_{i+1}x_{i-1} \\ &= A^2\sin^2(\Omega) \\ &= A^2\Omega^2 \end{aligned}$$

From the above equation, we can see that the output of Teager's algorithm is affected not only by the amplitude of the signal samples, but also by the oscillation frequency. This new measure is therefore capable of responding rapidly to changes in both A and Ω. Therefore, the ability of the measure to track rapid change improves dramatically. In the following sub-sections, we use the Teager energy as another point-based energy measure for "solving" for the endpoint detection problem. From the examples depicted below, we

observe that fricatives and plosives have very low amplitude, but unlike most vowels, these sounds have energy distributed at higher frequencies. A more suitable way to calculate the energy of producing the fricatives is to consider not only the amplitude of the acoustic signal, but also the frequency at which the acoustic energy is located. Therefore, the advantage of the Teager Energy operator appears.

Again using the pre-emphasized sequence as explained in section 3.3.1, we compute the normalized Teager energy. We also nonlinearly compress the Teager energy. The overall equation used for this calculation is:

$$T_i = (x_i^2 - x_{i-1}x_{i+1})^{0.3}$$

This compression was needed due to the very large dynamic range of the Teager energy. In order that the same methods can used to first find beginning and ending intervals, we also normalize the Teager energy contour to lie between 0 and 1, and smooth with the same filter as used to smooth the absolute value energy. Figure 8, Figure 9 and Figure 10 show three tokens with original acoustic signal, filtered acoustic signal, absolute value and Teager energy curves.

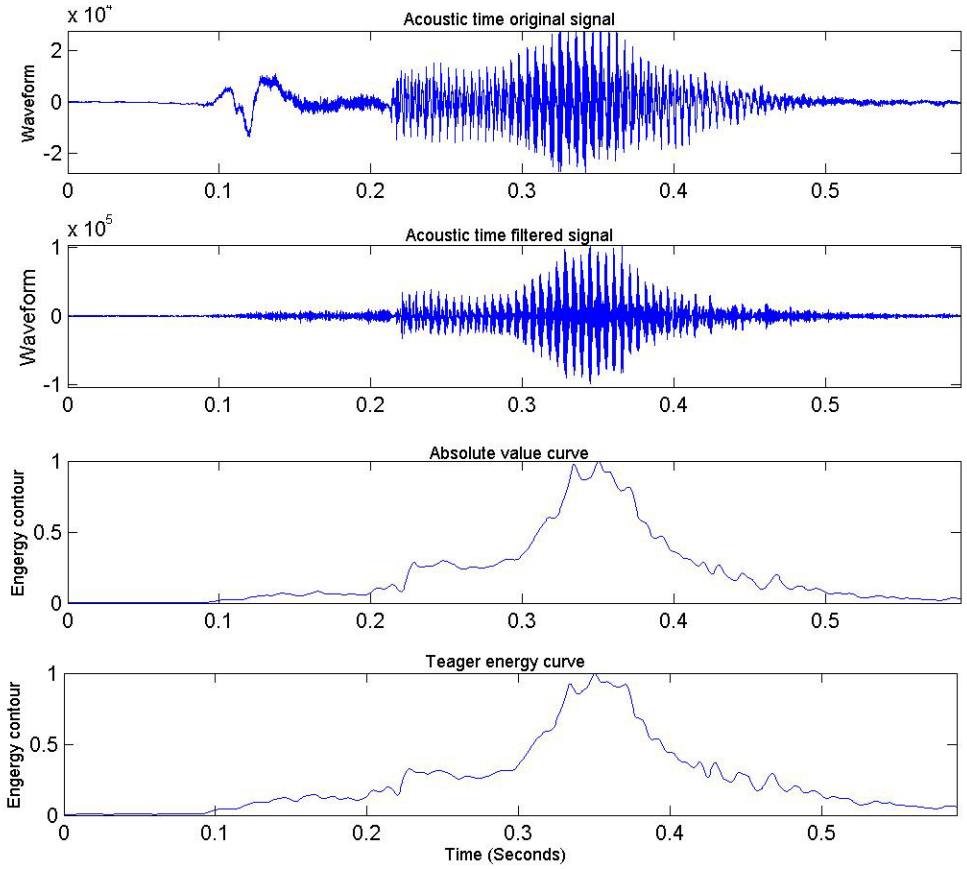

Figure 8: Original acoustic signal, filtered acoustic signal, absolute value and Teager energy curves for digit "three"

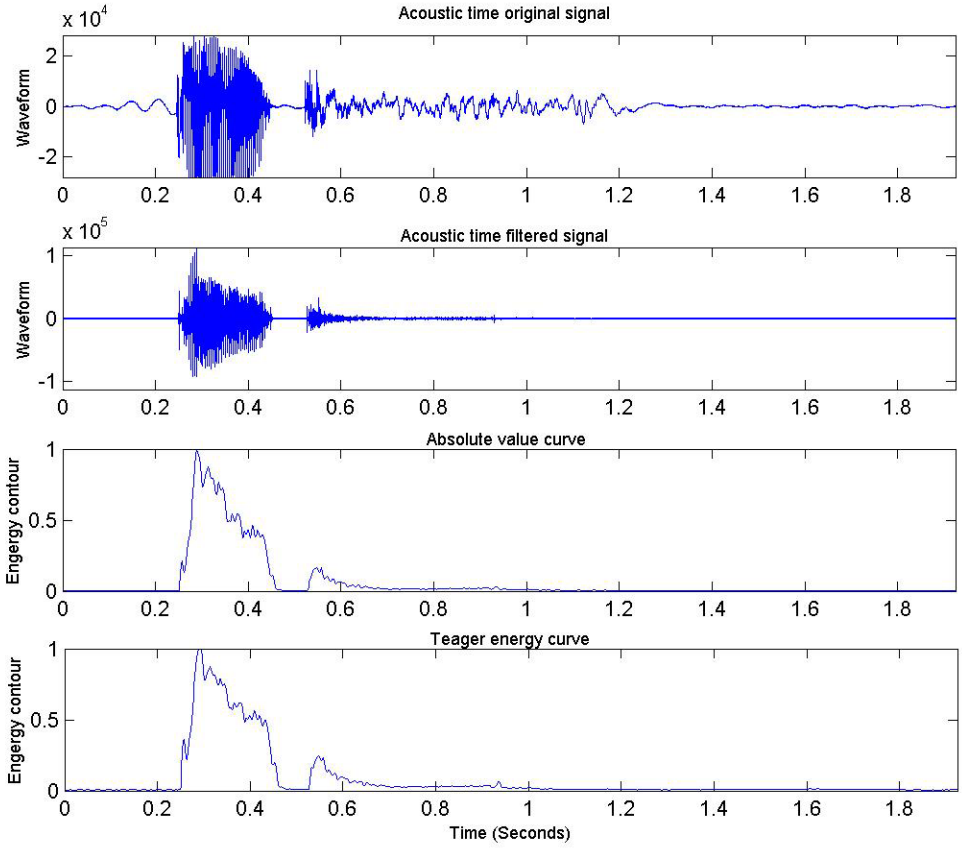

Figure 9: original acoustic signal, filtered acoustic signal, absolute value and Teager energy curves for CVC "beet"

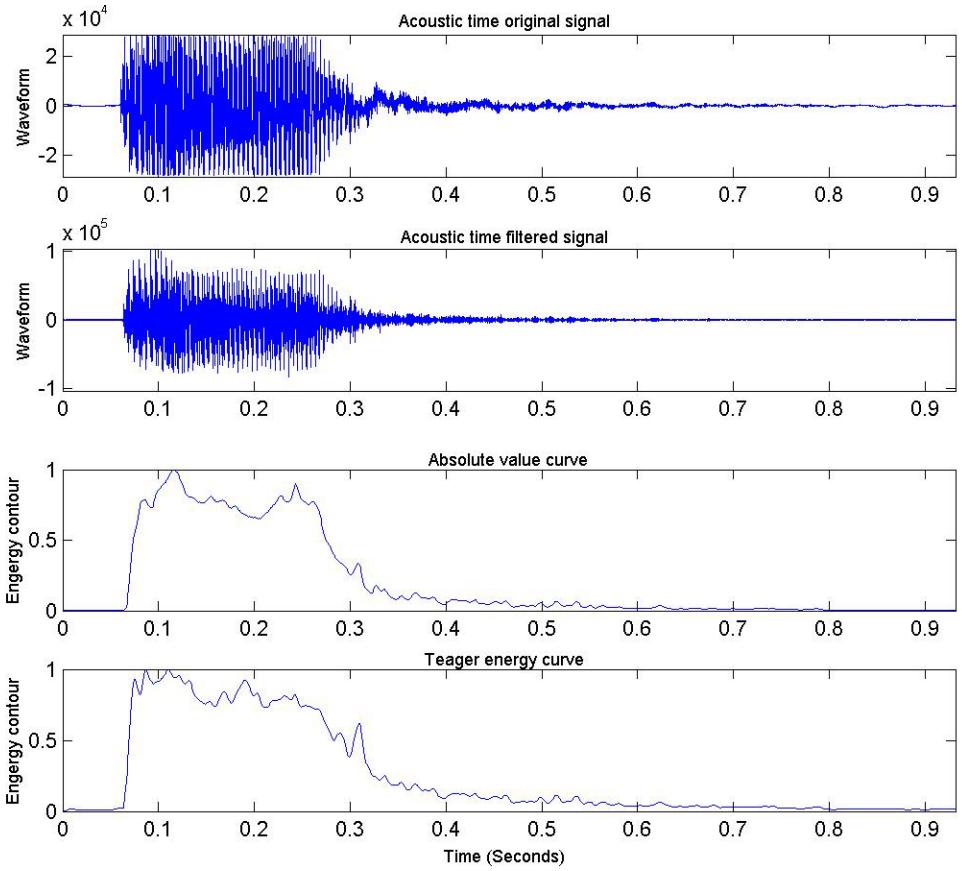

Figure 10: original acoustic signal, filtered acoustic signal, absolute value and Teager energy curves for vowel "UH"

3.3.3 Background Noise Estimation and Computation of Decision Thresholds

Since background noise plays an important role in endpoint detection thresholds, good estimation of background noise is an essential step in endpoint detection algorithms. Based on the normalized smoothed signal energy described above, we first apply yet another smoothing filter (using a 75-point simple running average) on both the absolute energy and Teager energy. The minima of these very smooth signals are used as the

primary estimate of the background noise reference values: *Thres_N* (for absolute energy approach) and *Thres_T* (for Teager energy approach).

With these two noise references, we then define four decision thresholds. Two of these decision thresholds are used to determine an interval of the signal in the beginning region, within which the speech signal is presumed to start. The other two of these decision thresholds are used to determine the interval of the signal in the ending region, within which the speech signal is presumed to end. To obtain the first of these decision thresholds, Thres_B1, let us use the absolute value as an example. The following decision logic is used:

$$Thres\_B11 = \min(c11 \times Thres\_N, c12)$$
$$Thres\_B1 = \max(Thres\_B11, c13)$$

The constants, *c11*, *c12* and *c13* are empirically determined to improve overall starting region endpoint accuracy (Typical values are 1.3, 0.1 and 0.00055 respectively). Note that the basic idea is that the threshold is c11 times the background noise, but also limited to a range determined by c12 and c13. In a similar manner, we determine the ending threshold Thres_B2 of the start region, the beginning threshold Thres_E1 of the end region and the ending threshold Thresh_E2 of the end region, using the following expressions:

$$Thres\_B21 = \min(c21 \times Thres\_N, c22)$$
$$Thres\_B2 = \max(Thres\_B21, c23)$$
$$Thres\_E11 = \min(c31 \times Thres\_N, c32)$$
$$Thres\_E1 = \max(Thres\_E11, c33)$$

$$Thres\_E21 = \min(c41 \times Thres\_N, c42)$$
$$Thres\_E2 = \max(Thres\_E21, c43)$$

The typical values, all empirically determined are *c21=8.0, c22=0.2, c23=0.01, c31=15.0, c32=0.20, c33=0.05, c41=3.0, c42=0.1* and *c43=0.0025*. Note that the same constants were used for both the absolute value and Teager Energy Signals. The basic logic was to choose these parameters is be sure the true endpoints are almost certainly in the interval determined by these thresholds. For the Teager energy approach, the equations are identical, with the same constants, except that we use *Thres_T* instead of *Thres_N* as the different reference threshold.

3.3.4 Locating the Beginning and Ending Regions

As described above, we have four thresholds for each of these two methods, each of which depends on the background noise level. As for the discussion above, we illustrate this method for the absolute value approach as an example. The extension to the Teager energy is straightforward. Here, we use the thresholds to determine starting and ending "regions" for the speech signal. At the beginning of the input signal, we will use the first pair of thresholds (*Thresh_B1, Thesh_B2*) to locate the start region, within which the actual beginning endpoint is presumed to lie. At the end of the signal, we will also use the second pair of thresholds (*Thres_E1, Thresh_E2*) to locate the ending region, within which the actual ending endpoint is presumed to lie.

Using the normalized and smoothed absolute value sequence, we index forwards (from the beginning to the end) point by point. Each energy point E_i is compared to the thresholds to locate the beginning endpoint interval using the following expression.

$$t_{B1} = \arg\max_i (E_i \le Thres\_B1), 1 \le i \le N$$

$$t_{B2} = \arg\max_i (E_i \le Thres\_B2), t_{B1} \le i \le N$$

With the same approach, we index move backwards (from the end to beginning) point by point. Each energy point E_i is compared to the other two thresholds to locate the ending endpoint interval using the following expression.

$$t_{E2} = \arg\min_i (E_i \le Thres\_E2), N \ge i \ge 1$$

$$t_{E1} = \arg\min_i (E_i \le Thres\_E1), t_{E2} \ge i \ge 1$$

An additional refinement on the determination of these intervals was as follows. This refinement was implemented since it was noted, that sometimes, a noise spike early in the recording would cause the beginning time t_{B1} to be much too soon, and sometimes noise bursts late in the recording would cause the final end time t_{E2} to be much too late. These errors could generally be spotted by visual inspection of the filtered waveforms. Generally, if these type errors occurred for the beginning time, there would be a “long”

very low amplitude signal, prior to the main part of the speech signal. Similarly, if these errors occurred for the end time, there would be a "long" interval of very low amplitude signal after the main part of the speech signal, and before the final end time.

Therefore, after the initial beginning time was determined, the number of the low energy points (Energy threshold = .95**Thresh_B1* + .05**Thresh_B2*) between t_{B1} and the point at which the maximum of the signal occurred were counted. If the number of such points exceeded 50 ms, the time t_{B1} was advanced by the number of the low energy points. The time t_{B2} was left unchanged, unless it was less than 50 points ahead of t_{B1}, in which case was changed to be t_{B1}+50.

A very similar procedure was used to check, and possibly modify t_{E1} and t_{E2}. For this case, the time threshold for low energy intervals prior to $t_{F2}$was set to 200 ms, and t_{E1} was made to be at least 75 points earlier than t_{E1}. Figure 11, Figure 12 and Figure 13 show the beginning and ending region obtained by the absolute value and Teager energy separately.

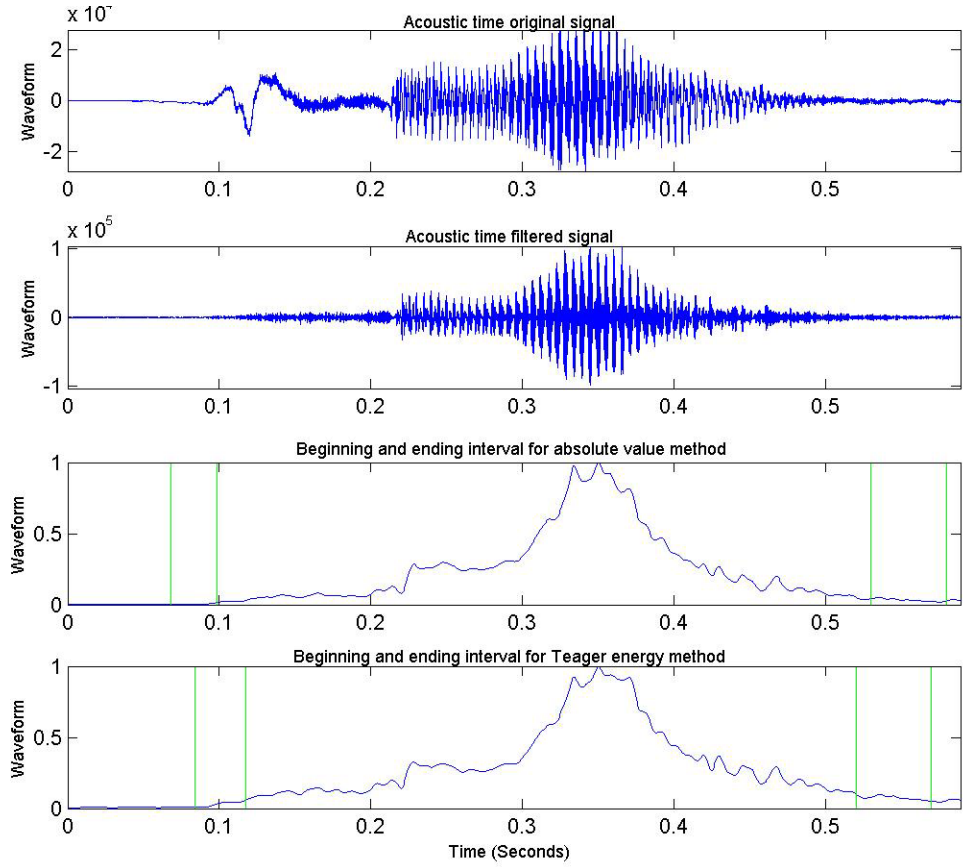

Figure 11: Beginning and ending region using absolute value and Teager energy for digit "three"

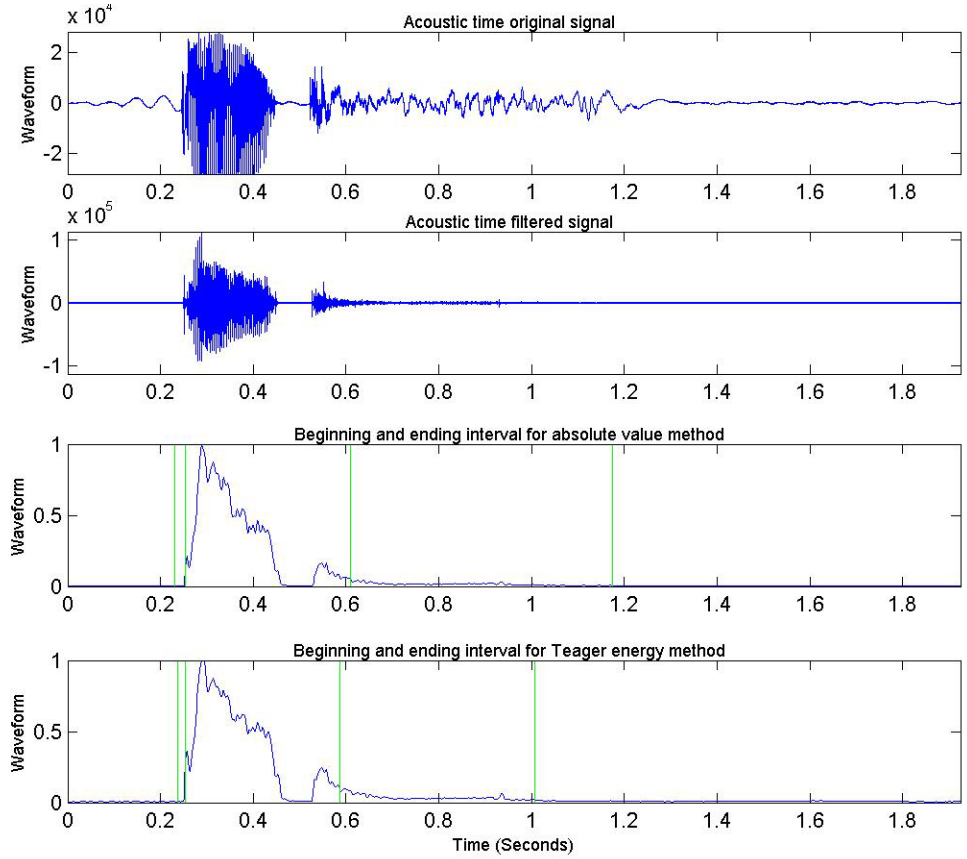

Figure 12: Beginning and ending region using by absolute value and Teager energy for CVC "beet"

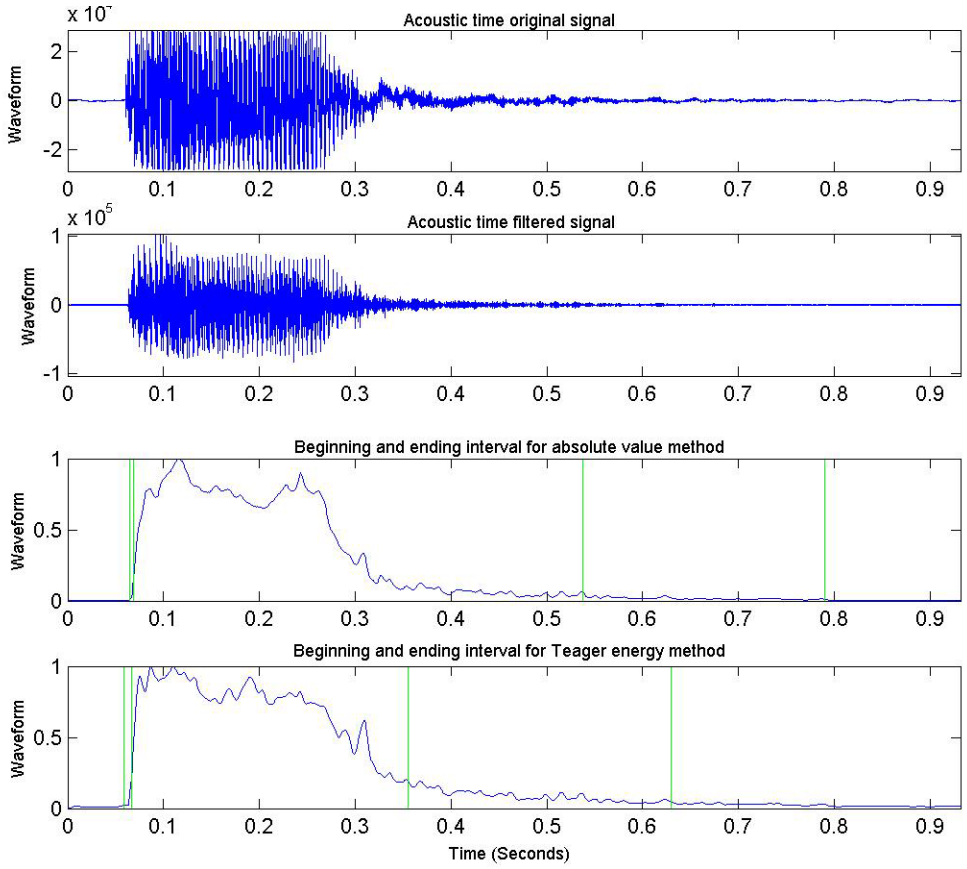

Figure 13: Beginning and ending region using by absolute value and Teager energy for vowel "UH"

3.3.5 Final Endpoints for Absolute Value and Teager Energy Approaches

From the literature [4, 17], we know that the boundary between non-speech and speech is usually accompanied by a large change in some energy-based parameter. Therefore, in the low area of the beginning, from the point t_{B1} to t_{B2}, we calculate the first order difference of the given energy between the two adjacent points one by one forwards. The actual beginning of the speech signal corresponds to the maximum value of this energy difference.

In the same way, in the ending region described above, from point t_{E2} to t_{E1}, the end of the speech signal is determined as the minimum value of the energy difference of adjacent points in the energy contour.

By applying the exactly same logic on both absolute value approach and Teager energy approach, we have the final pair of endpoints, respectively.

3.3.6 Final Decision Logic

So far, by using the absolute value and Teager energy parameters, we determine two pairs of endpoints individually. One we call *Abs_TB* and *Abs_TF* (for the ones obtained with the absolute value signal), and the other pair is called *Tea_TB* and *Tea_TF* (for the ones obtained from the Teager energy signal). From these two pairs, we choose the averages for both estimates using the following simple equations.

$$TB = (Abs\_TB + Tea\_TB)/2$$

$$TF = (Abs\_TF + Tea\_TF)/2$$

Figure 14, Figure 15 and Figure 16 illustrates the endpoints obtained with the absolute value energy, Teager energy and final logic. The corresponding original signal and filtered signals were already shown above. The first panel depicts the endpoints as

determined from the smoothed absolute value. The second panel depicts endpoints obtained from the smoothed Teager energy. The third panel depicts the endpoints, after combining the results of the first two panels.

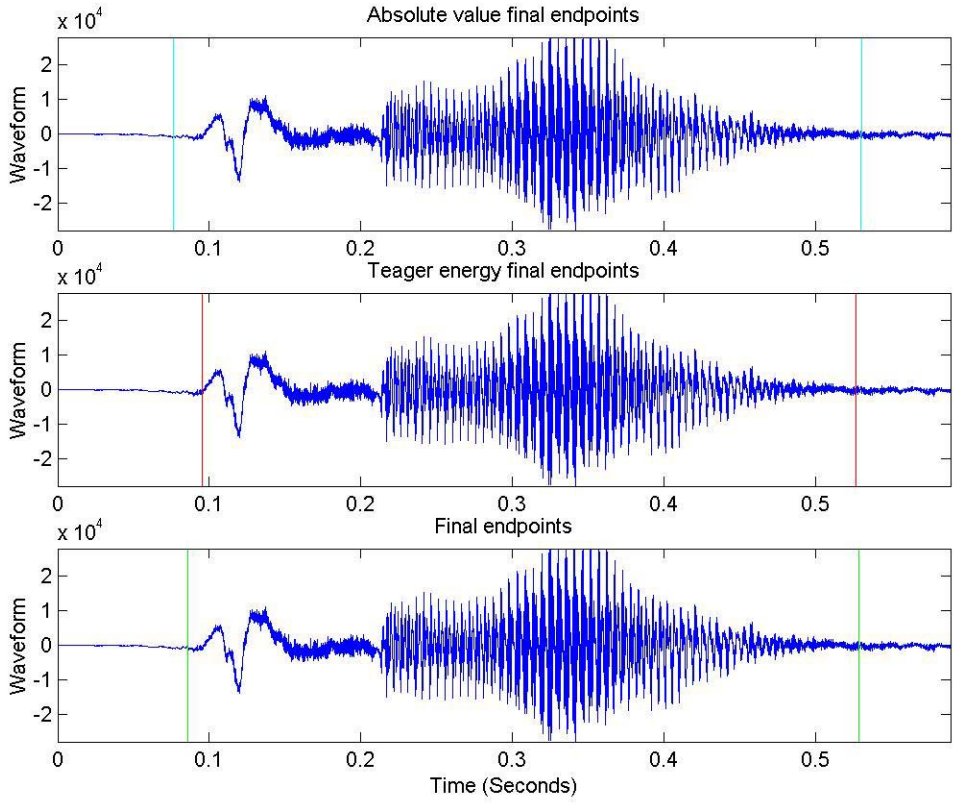

Figure 14: Endpoints from absolute value, Teager energy and final logic for digit "three"

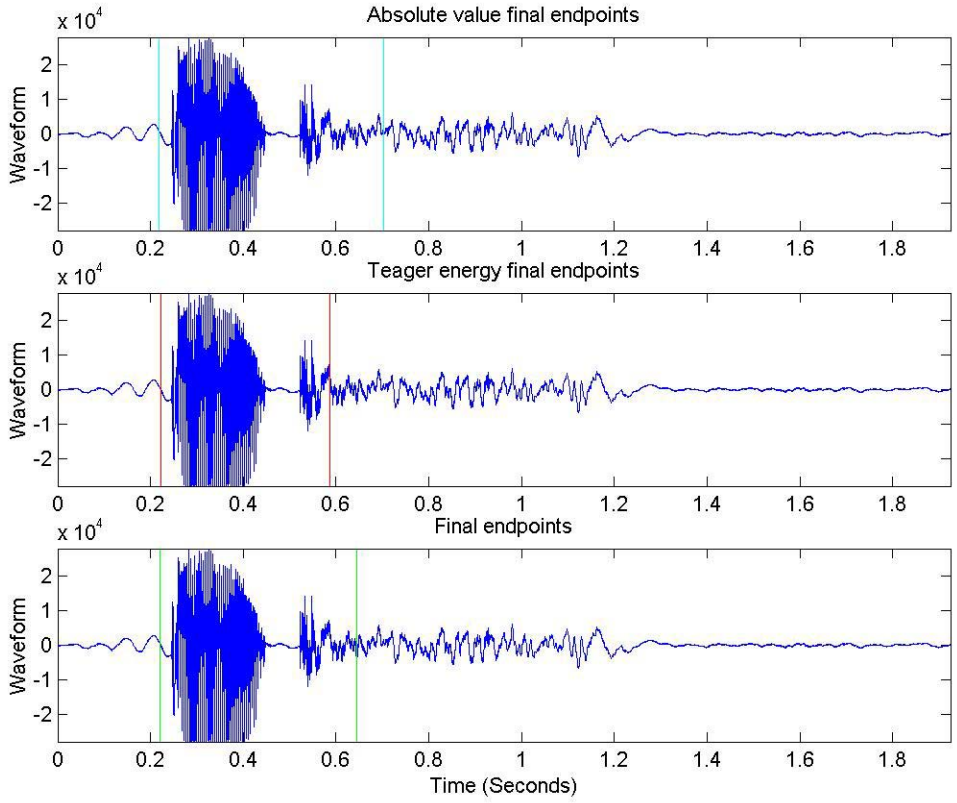

Figure 15: Endpoints from absolute value, Teager energy and final logic for CVC "beet"

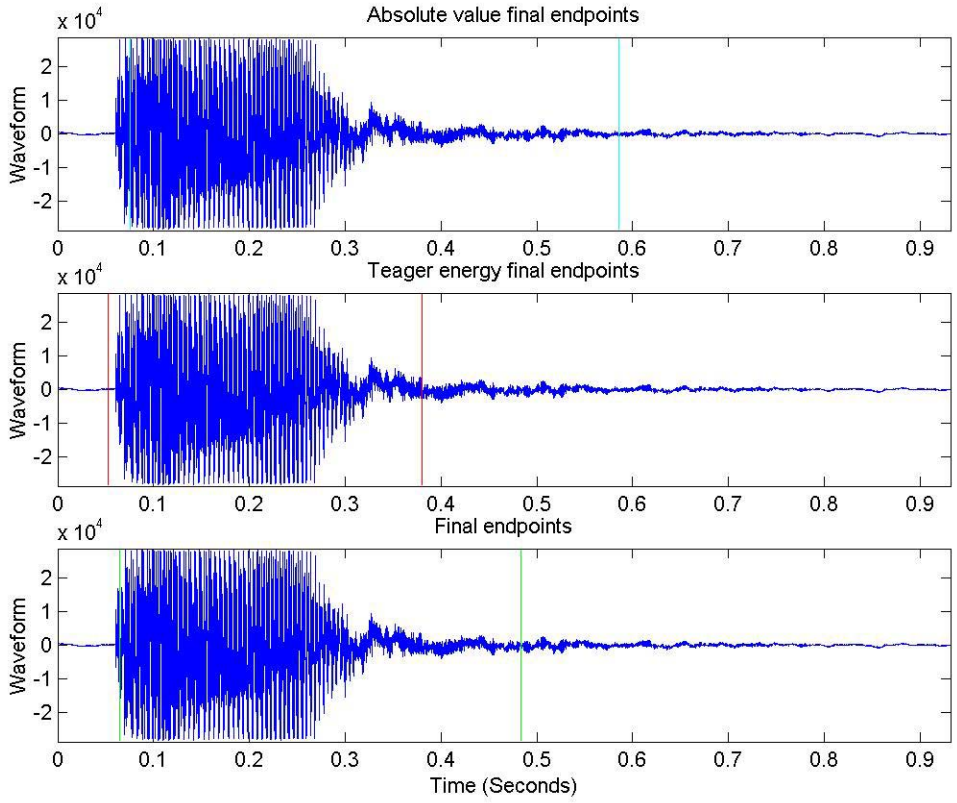

Figure 16: Endpoints from absolute value, Teager energy and final logic for vowel "UH"

From the figures above, this algorithm appears to perform very well to eliminate noise. In Figure 14, we can see the final endpoints are still correct even though the plosive part is very weak. From Figure 15, we find that after 0.63 s, the noise is successfully removed while the beginning and ending plosives are successfully kept.

In the following chapter we will compare this AETE algorithm with another algorithm presented in 1991 and give a thorough analysis of the errors made.

CHAPTER FOUR

EXPERIMENTAL VALIDATION

4.1 Brief Description of Test Database

The endpoint algorithm has been tested with a database that was collected by the Speech Communication Lab in the Electrical and Computer Engineering Department at Old Dominion University. In this database, we have several kinds of word recorded, including vowels, consonant-vowel-consonants (CVCs), spoken digits and the spoken letters of the alphabet. From the database, we chose two male speakers, two female speakers and two children speakers to test 10 digits, 26 alphabet letters, 13 CVCs and 13 vowels for a total of 512 wave files. Table 1 is a list and count of all the words we used for testing. A sampling rate of 22.05 KHz was used for all data. All recordings were made in a "normal" computer lab environment; for some of the recordings there was significant background and/or breath noise. We used this database to evaluate the algorithm presented in this thesis, and to compare its performance with the PE algorithm [4] discussed in chapter 2, where "PE" denotes pure energy algorithm. We also use the database to evaluate the two portions of the AETE algorithm—one based on the absolute value parameter only (AE), and the other based on the Teager Energy parameter only (TE). This additional evaluation is useful to help assess the relative usefulness of the AE and TE parameters for endpoint determination.

Alphabet	Number	CVC	Number	Vowel	Number	Digit	Number
A	7	bag	10	ae	10	one	7
B	7	bed	10	ah	10	two	7
C	7	beet	10	aw	10	three	7
D	7	bird	10	ay	10	four	7
E	7	boat	10	ee	10	five	7
F	7	book	10	eh	10	six	7
G	7	boot	10	ih	10	seven	7
H	7	boyd	10	oh	10	eight	7
I	7	cake	10	oo	10	nine	7
J	7	cot	10	oy	10	ten	7
K	7	cup	10	ue	10		
L	7	dog	10	uh	10		
M	7	pig	10	ur	10		
N	7						
O	7						
P	7						
Q	7						
R	7						
S	7						
T	7						
U	7						
V	7						
W	7						
X	7						
Y	7						
Z	7						

Table 1: List of test words and number of occurrences of each. Database was used to evaluate the endpoint detection algorithm

4.2 General Comments

In order to compare the performance of the two algorithms, we performed the test in the following steps. First, each algorithm was used for all of these test words to compute

both the beginning and ending endpoints. The endpoints results obtained from algorithmic methods were then compared with the endpoints determined manually using visual and auditory inspections by skilled personnel. The manual endpoints were determined by a gradual segmentation of the speech signal and careful listening and visual inspection of the waveforms to locate the endpoints. The errors were also separated into several different categories, such as beginning and ending errors of various lengths, and also the "signs" of errors (identified start or end point before or after actual start or end point). Errors were also analyzed in terms of vocabulary type: digit, alphabet, CVC and vowel. More details are presented in the following sections.

4.3 Method of Manual Endpoints Detection

In this section, we will summarize the method used to determine the endpoints manually. For the entire test, Cool Edit 2000 (from Syntrillium Software Corporation) software was used to help determine the final endpoints manually. Cool Edit 2000 has more than twenty audio effects and tools such as Echo, Flange, Compression, Amplify, Noise Reduction, Reverb, Time/Pitch stretch, and so on. More importantly, for the present application, it also has powerful analysis tools. For example, the frequency components and other details about our audio files could be examined with the Frequency Analysis, Statistics and Spectral View features.

For each test word, after using Cool Edit to open it, the total duration (length) is given and the entire file can be viewed as an acoustic waveform. Then, we used two

independent ways to examine each word and to determine the endpoints. The first way was by acoustical (listening) examination. For this approach, we chose the waveform view from Cool Edit and listened carefully to the whole word. Sometimes, we also used the mouse to highlight parts of the given word to check whether it was speech or not. By carefully performing this step, we could determine accurate endpoints for both the beginning and end. To further verify that the reference endpoints obtained by this manual method were reliable, we also used another way to check the each word. This second approach was by examination of the spectrogram of each word by using the spectral view from Cool Edit. Generally the transitions from background noise to speech, and from the end of the speech to background noise were quite obvious from the spectrogram. Generally speaking, the speech part of the signal was red, whereas the non-speech part was blue or green. Therefore, by verifying the results with the spectral view, and correcting as necessary, we were reasonably sure that the manually determined endpoints formed a good reference. As yet one additional check, manual endpoints were independently determined by two people for one set of 512 waveforms, and the results compared. The average difference in results for the two people were less than 10 ms for the starting point, and less than 20 ms for the ending point. Thus these manually determined "reference" endpoints were used to evaluate the endpoint algorithm presented in this thesis, and to compare them with another control automatic method.

4.4 Algorithm Performance Comparison and Analysis

In this section, we will first define several kinds of errors and arrange them in a table to evaluate the new algorithm and compare it to an existing algorithm

4.4.1 Mean and Variance

One way to systematically evaluate and describe the performance of an endpoint detection algorithm is to compute and show the mean and standard deviation of errors of detected endpoints as compared to reference values. That is, we first compute the differences between computed and reference endpoints, and then calculate the mean and standard deviation of these error values. This results in four useful variables: mean for beginning endpoints, standard deviation for beginning endpoints, mean for final endpoints and standard deviation for final endpoints. In this chapter we refer to the algorithm we presented in chapter two as the "PE," as mentioned above. With the notation mentioned in the beginning of the chapter, and also using the error analysis discussed above, the following overall performance results are obtained, as shown in Table 2.

	AE only (ms)	TE only (ms)	AETE Algorithm (ms)	PE Algorithm (ms)
Mean for beginning	5.3	4.6	0.0	-4.270
Standard deviation for beginning	26.6	30.2	26.8	43.07
Mean for end	34.9	40.4	0.0	31.66
Standard deviation for end	60.7	70.7	58.1	109.1

Table 2: Mean and standard deviation for two algorithms

From Table 2, we can see all of the four performance measures for the AETE algorithm are much better than for the PE algorithm. Both the mean values for the beginning and end obtained from the AETE algorithm are essentially zero. However, the mean error is not really significant since the mean can always be "adjusted" to zero by adding or subtracting from the originally obtained endpoint, using the original mean from a large

amount of data as the "correction" factor. It is the standard deviation, or variability in the error which is much more significant, since it cannot easily be reduced. In this regard, the standard deviations, at both the beginning and the end, are approximately half the values obtained from the PE algorithm. We also note that both the AE method and TE method (each based on one parameter) are better than the PE algorithm, but do not perform as well as the combination AETE algorithm. However, if only one parameter were to be used, it appears that the absolute value energy is a better choice than the Teager Energy.

4.4.2 Beginning Errors

Here, we will show in more detail six different kinds of error that occurred at the beginning of an utterance. They are: (1) more than 50 ms and less than 100 ms (too soon), (2) more than 100 ms and less than 200 (too soon), (3) more than 200 ms (too soon), (4) more than 50 ms and less than 100 ms (too late), (5) more than 100 ms and less than 200 ms (too late) and (6) more than 200 ms (too late). Here, "too soon" means the endpoints detected by the algorithm are earlier than the reference endpoints. Similarly, "too late" means the endpoints detected by the algorithms are later than the reference endpoints. Through the following table, we can find the performance of the AETE and PE algorithms, as well as the routines working only with one parameter absolute algorithm-AE and Teager algorithm-TE. In general, we consider a "significant" endpoint error has occurred if the beginning difference is greater in magnitude than 50 ms or the end difference is greater than 100 ms in magnitude.

	AE Beginning	TE Beginning	AETE Beginning	PE algorithm Beginning
More than 50 ms (too soon)	13	15	12	5
More than 100 ms (too soon)	7	9	7	8
More than 200 ms (too soon)	1	1	1	4
More than 50 ms (too late)	9	11	12	0
More than 100 ms (too late)	2	3	2	0
More than 200 ms (too late)	0	0	0	0
Total errors	32	39	34	17
Total corrects (less than 50 ms)	478	471	476	493

Table 3: Distribution of beginning errors for four methods

From Table 3, we can see the number of errors obtained from the PE algorithm is slightly less than the number obtained from the AETE algorithm. However, for the error case of beginning too soon, the number of serious errors, that is errors s greater than 100 ms, the AETE algorithm's performance is much better than for the PE algorithm. For the error of beginning too late, the PE algorithm is better than the AETE algorithm.

4.4.3 Ending Errors

In this section, we will show six different kinds of error that occurred at the end of an utterance, using the same type of analysis as for the beginning error. These error categories (also as mentioned in section 4.4.2) are 1) more than 50 ms and less than 100 ms (too soon), 2) more than 100 ms and less than 100 ms (too soon), 3) more than 200 ms (too soon), 4) more than 50 ms and less than 100 ms (too late), 5) more than 100 ms and less than 200 ms (too late) and 6) more than 200 ms (too late). Using the following table, we can examine the performance and observe the differences between these two algorithms, including the two "partial" algorithms (absolute algorithm-AE and Teager algorithm-TE).

	AE End	TE End	AETE End	PE algorithm End
More than 50 ms (too soon)	102	123	49	20
More than 100 ms (too soon)	49	65	18	7
More than 200 ms (too soon)	9	9	2	2
More than 50 ms (too late)	14	5	47	70
More than 100 ms (too late)	6	3	9	42
More than 200 ms (too late)	0	2	1	12
Total errors	180	207	126	153
Significant errors (more than 100 ms)	64	79	30	63
Total corrects (less than 50 ms)	330	303	384	357

Table 4: End error analysis

From Table 4, we can see both of algorithms have their advantages and disadvantages. However, the overall performance of the AETE algorithm is much better than for that of the PE algorithm. For the error case of beginning too soon, the PE algorithm performance is better than for the AETE algorithm. The total number of errors is 29 for the PE algorithm as compared to 69 for the AETE algorithm. However, for the error case of ending too late, the AETE algorithm performance is much better than that of the PE algorithm. The total number of errors is 57 for the AETE algorithm as compared to 124 for the PE algorithm. We can also see that there are many more serious errors (greater than 100 ms in either direction) for the PE algorithm than for the AETE algorithm. As pointed out from the data in Table 4, in the overall evaluation of the ending errors, including both detecting too soon and detecting too late, the AETE algorithm is much better than the PE since the standard deviation is 58 ms for AETE compared with 109.1 ms for PE. Also, as shown in Table 4 above, there are 126 total errors for AETE versus 153 for PE. However, the results from AE or PE purely are 180 and 207, both of which are inferior to PE and AETE.

4.4.4 Errors according to word type

In this section, we present a table which sorts the errors according to different types of words, for both beginning and ending errors. This gives another view of the relative performance of the AETE and PE algorithm.

	AETE (beginning)		PE Algorithm (beginning)	
	Vowel	Word	Vowel	Word
More than 50 ms (too soon)	3	9	1	4
More than 100 ms (too soon)	4	3	3	5
More than 200 ms (too soon)	1	0	2	2
More than 50 ms (too late)	0	12	0	0
More than 100 ms (too late)	0	2	0	0
More than 200 ms (too late)	0	0	0	0
Total errors	8	26	6	11
Total corrects (less than 50 ms)	122	354	124	369

Table 5: Beginning error analysis for different kinds of words

From Table 5, we can see that for detecting vowels, the performance of PE algorithm and the performance of AETE algorithm are almost same. However, for detecting words, the performance of the PE algorithm is better than the performance of the AETE algorithm.

	AETE (end)		PE Algorithm (end)	
	Vowel	Word	Vowel	Word
More than 50 ms (too soon)	8	41	2	18
More than 100 ms (too soon)	2	16	0	7
More than 200 ms (too soon)	0	2	0	2
More than 50 ms (too late)	15	32	31	39
More than 100 ms (too late)	3	6	23	19
More than 200 ms (too late)	1	0	6	6
Total errors	29	97	62	91
Significant errors (more than 100 ms)	6	24	29	34
Total corrects (less than 50 ms)	101	283	68	289

Table 6: End error analysis for different kinds of words

From Table 6, we can see that AETE algorithm performance is much better in determining endpoints for vowels than is the PE algorithm. The two algorithms perform approximately the same for determining endpoints for words. Overall, there are fewer errors for the AETE algorithm than for the PE algorithm.

4.5 Error Examples

In this section, we will show several figures to illustrate some correct endpoint determinations and some error endpoints. Figure 17 shows an example of detecting too soon at the beginning. From the intermediate experimental results, we find that both the AE and TE algorithms make some errors due to beginning breath noise. Therefore, the final results contain too much silence.

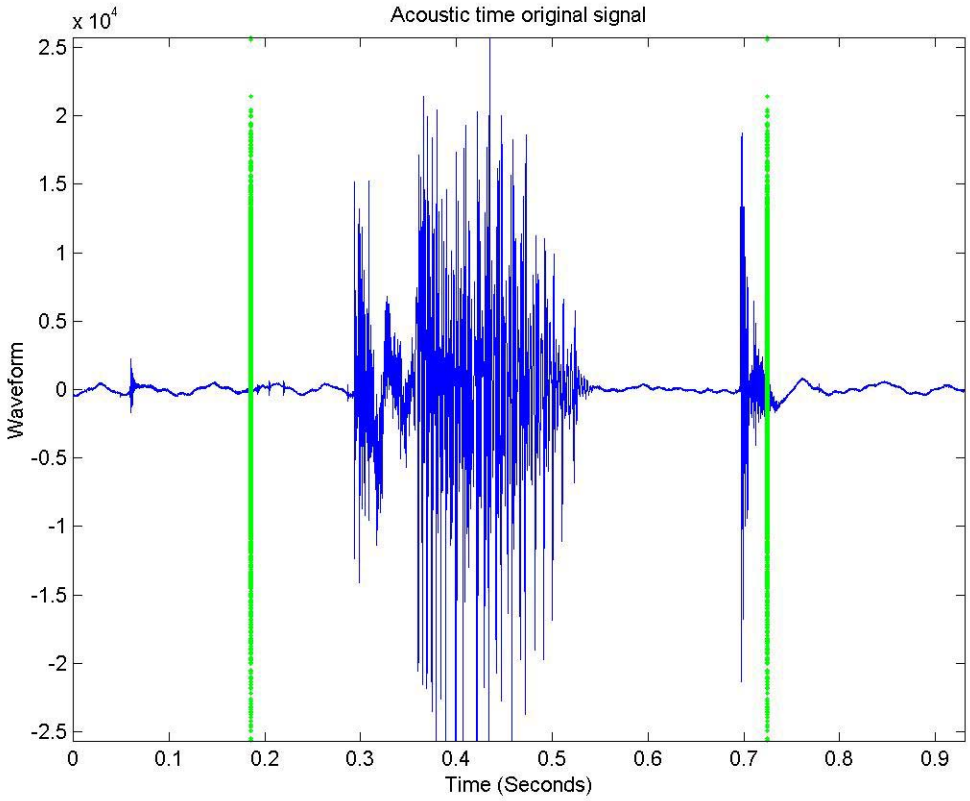

Figure 17: Example of detecting too soon at beginning for word "cake"

Figure 18 shows an example of detecting too late at the beginning. In this case, TE algorithm gives the correct reference points, however, the AE algorithm is in error by nearly 100 ms. Therefore, based on the final decision logic, it also gives us a wrong decision.

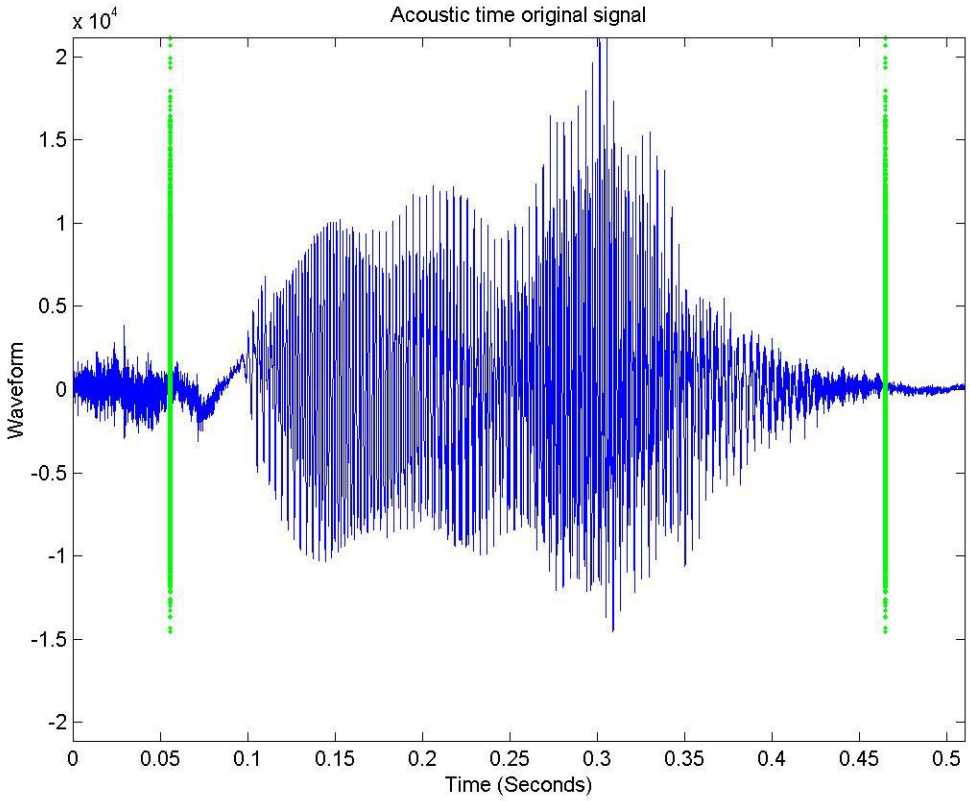

Figure 18: Example of detecting too late at beginning for word "four"

Figure 19 shows an example of detecting too soon at the end. At the end, it loses the part "v" of word "five." By examining the intermediate experimental results, we find both the AE and TE endpoints contain major errors. The reason is likely that the fricative part "v" of "five" is very weak and considered as noise by the detecting algorithm.

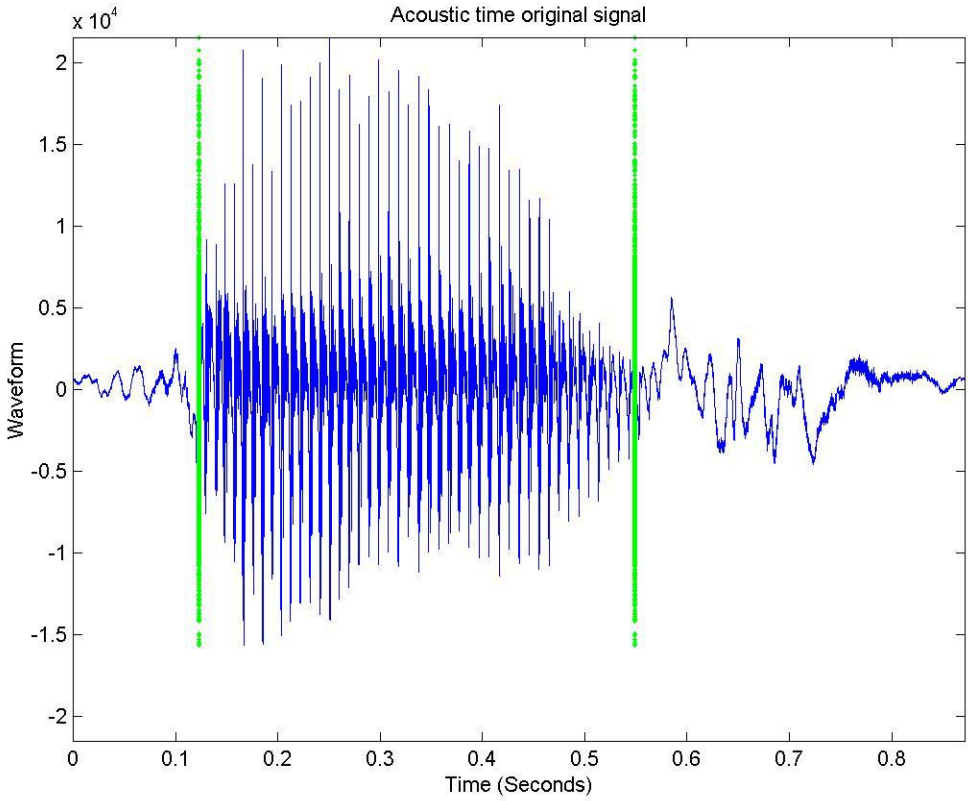

Figure 19: Example of detecting too soon at the end for word "five"

Figure 20 shows an example of detecting too late at the end. From this case, we see that the final part contains strong noise with high frequency components. Therefore, both the AE and TE algorithms fail. Part of the noise is considered as speech.

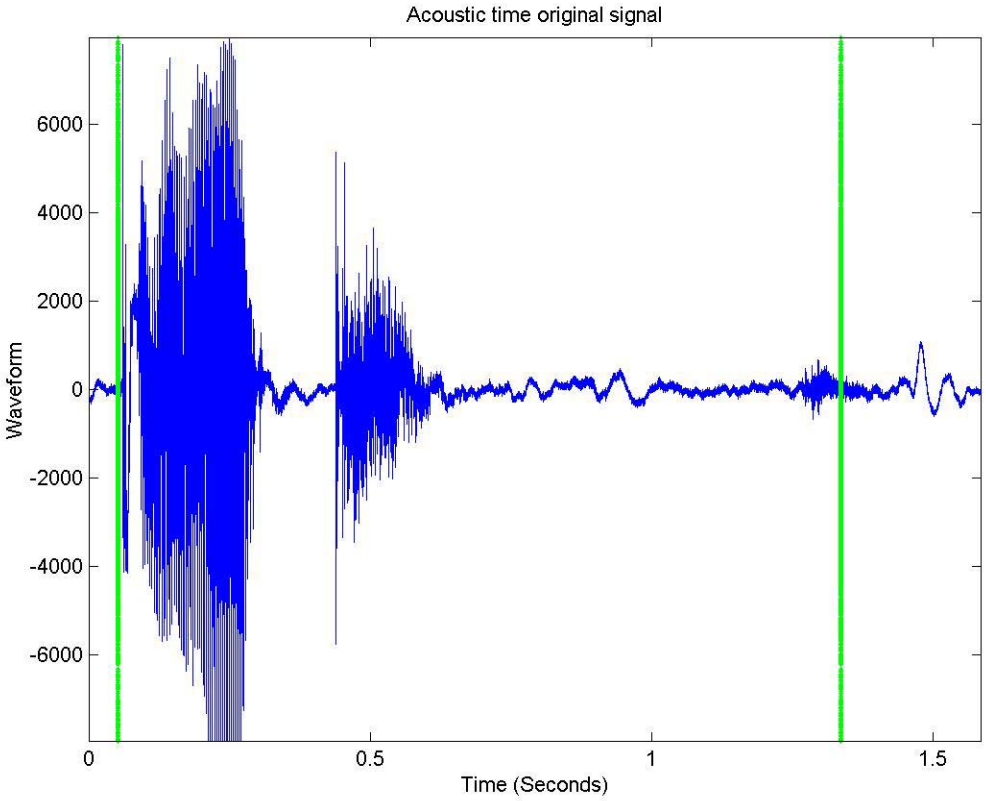

Figure 20: Example of detecting too late at the end for word "book"

Figure 21 shows a correct detection for word "boat," which successfully keeps the weak voiced part and rejects the noise.

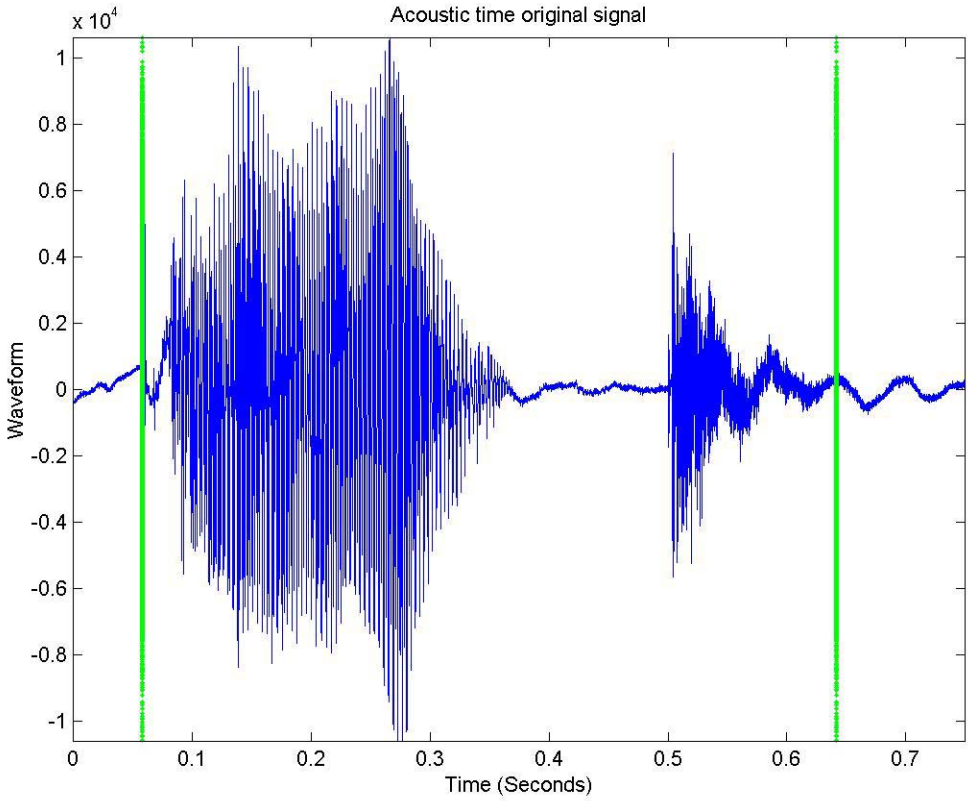

Figure 21: A correct detection for word "boat"

Figure 22 shows a correct detection for vowel "oo." Although the vowel is easily endpoint detected compared with other kinds of words, our algorithm still successfully rejected the moderate background noise in this example.

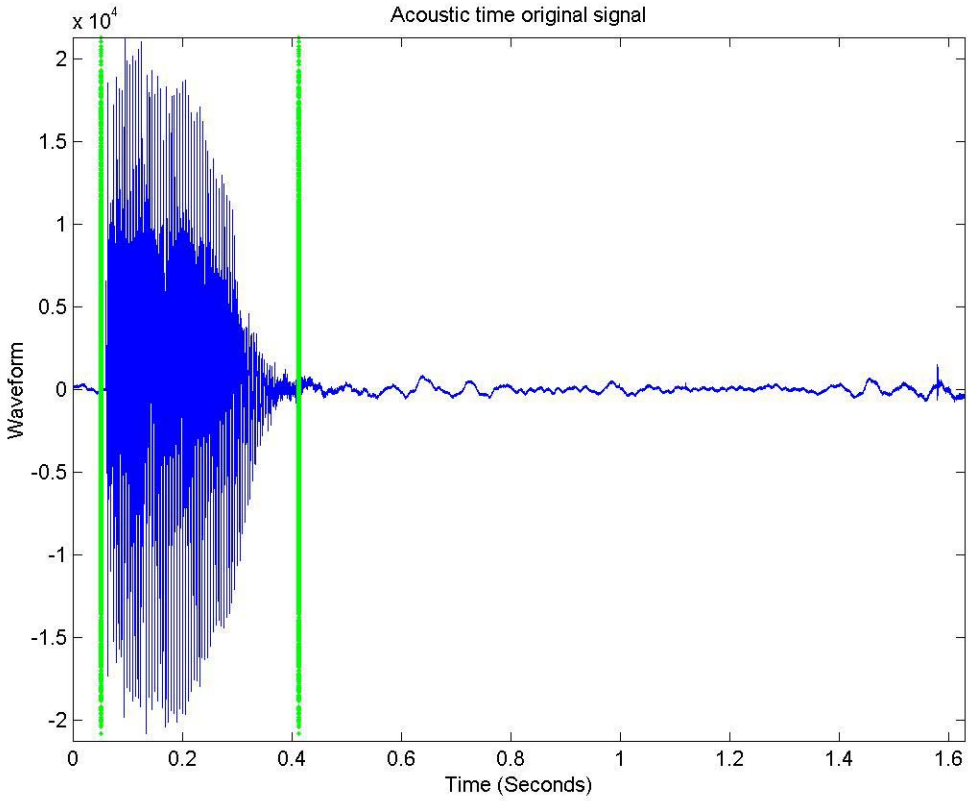

Figure 22: A correct detection for vowel "oo"

4.6 Discussion of Remaining Problems

From the tables and examples we presented above, we can see the error rate, error mean and error standard deviation of end detection are all much higher or bigger than the ones for beginning detection. That means it is much easier to detect the beginning endpoint than the final endpoint. The reason is that ending fricatives or plosives which occur at the end of words, are generally much weaker than the same phonemes at the beginning of words. This kind of fricative or plosive has a low energy level and sometimes is lower than the background noise. Often these phonemes also reduce in amplitude very

gradually, rather than abruptly. In some case, the Teager energy approach can solve such a problem due to presence of high frequency components. However, in some cases, the background noise also has high frequency and energy, and both the absolute value method and Teager energy method will fail. All of these problems remain as challenges for future endpoint detection algorithms.

CHAPTER FIVE

CONCLUSIONS AND FUTURE WORK

5.1 Conclusions

The general contribution of this thesis is the introduction of a new algorithm to endpoint isolated words. It is a combination method using the smoothed contours of absolute value and Teager energy as two separate parameters as indicators of the endpoints. In this combination approach, we perform several important steps, which are not applied in other algorithms.

1) In general, the new algorithm discussed in this thesis is fast, accurate and easily implemental for isolated words endpoint detection. Two separate approaches are employed in this new algorithm: they are absolute value approach and Teager energy approach. For the absolute value approach, the biggest advantages are the smoothed absolute value is reasonably measure of short time speech energy, and therefore this method has been widely used since the end of 1970s. However, the method does have problems in combating background noise, and does have problems with weak fricatives and plosives at the beginning and ends of utterances. For the Teager energy method, the most distinguishing property is the sensitivity to fricatives and plosives with high frequency components. Both

methods are computationally efficient. By combining these two algorithms properly, we can achieve satisfactory results for isolated words endpoint detection, with a low computational burden.

2) The performance of this new algorithm is also very promising. Comparing with the PE algorithm, discussed in detail in chapter 2, the accuracy of our new algorithm is 87.3% in a moderate noise environment, while the accuracy of the PE algorithm is 84.3% in quite office environment. These numbers are based on the percentage of tokens with endpoint errors below a certain threshold, as discussed in chapter 4. At the same time, not only the overall performance, but also the detection performance for both beginning and end are improved. For the PE algorithm, the standard deviation of the error for the beginning is 43.07 ms, and the ending error has a standard deviation of 109.12 ms. Compared with the above performance, our new algorithm errors have a standard deviation of 26.3 ms and 53.3 ms for the beginning and ending respectively. Thus the average errors are about the magnitude of those for the PE method.

3) The signal pre-emphasis is different from other approaches. We apply two different filters for the signal pre-processing. First is a single pole filter controlled by two parameters (in our algorithm, the typical value is 3000 Hz for the center frequency and 0.8 for the radius) to emphasize part of the bandwidth in frequency domain. This filter is followed by 150-point band pass FIR from 375 Hz to 5000

Hz. From the experimental results, we find the system performance for discriminating speech from office noise is improved by using these filters.

4) In our algorithm, the biggest difference from previous algorithms is that all processing is based on individual points, rather than frames. This enables better resolution, since decisions are made based these individual points. Note, however, that smoothing low pass filter at 30 Hz was used to smooth both the absolute value and Teager energy, prior to decision making.

5) The final decision is made by combing the results of two individual decisions rather than using one method only. The experimental results did show that the combination method was better than using either method alone.

From a performance point of view, the method introduced in this thesis did give significantly better performance for identifying both the start point and end point of isolated utterances, at least in terms of the mean and standard deviation of the error.

5.2 Future Work

However, this algorithm is still far from perfect. There are several suggestions and possible ways to achieve the desired performance and accuracy.

1) Better knowledge and modeling of typical beginning and ending patterns could be used to determine better detection schemes. As we discussed in chapter 3, background noise is sometimes strong enough to cause the algorithm to make the wrong start point, sooner than the true starting point, or cause the algorithm to make the wrong end point, that is later than the true end point. In our solution, we calculate the number of sample points between the endpoints algorithm made and the peak value. If this number is too big, it suggests that that the endpoints are triggered by background noise. Therefore, we move start point forwards and move end point backwards by the experimental values. However, sometimes, this approach is not exactly correct. Hence, a better method to summarize some patterns to describe the beginning and ending speech "mode" will improve the decision making between background noise and the onset of speech.

2) In the signal pre-emphasizing step, we can still try different parameters for the lowpass and signal pole filter to improve the performance. Theoretical speaking, we can increase the cutoff frequency of the lowpass filter a little bit as long as it does not degrade the useful signals severely.

3) For the step both of absolute value method and Teager energy method, we can still try other values for the parameters which determine the threshold for the low energy area, where the true endpoints are supposed to be. To modify this step, the most important criteria we must follow is to be sure that the low energy area

always contains the true endpoints, while simultaneously being as short as possible.

4) We might change the final logic a little bit to make the detection performance much better. Since sometimes one set of the final endpoints from one original approach is already correct, and the other set is incorrect, thus making the final endpoints based on mean value incorrect. Although this kind of case doesn't occur very often, we can improve the system performance greatly if we can solve such a problem.

In summary the endpoint detection algorithm introduced in this thesis is fast and accurate, and compares favorable with the best reported algorithms from the literature. However, the method can be improved even more by taking into account the points mentioned above.

BIBLIOGRAPHY

[1] Lingyun Gu, Stephen A. Zahorian, *"A New Robust Algorithm for Isolated Word Endpoint Detection." Submitted to Proc.* IEEE ICASSP-02, 2002

[2] G.S. Ying, C.D. Mitchell, L.H. Jamieson, *"Endpoint Detection of Isolated Utterances Based on A Modified Teager Energy Measurement."* In Proc. IEEE ICASSP-92, pp.732-pp.735, 1992

[3] Liang-Sheng Huang, Chung-Ho Yang, *"A Novel Approach to Robust Speech Endpoint Detection in Car Environment."* In Proc. IEEE ICASSP-00, pp.1751-pp.1754, 2000

[4] Evangelos S. Dermatas, Nikos D. Fakotakis, George K. Kokkinakis, *"Fast Endpoint Detection Algorithm for Isolated Word Recognition in Office Environment."* In Proc. IEEE ICASSP-91, pp.733-pp.736, 1991

[5] Yiying Zhang, Xiaoyan Zhu, Yu Hao, Yupin Luo, *"A Robust and Fast Endpoint Detection Algorithm for Isolated Word Recognition."* IEEE ICIPS-97, pp.1819-pp.1822, 1997

[6] Jean-Claude Junqua, Brian Mark, Ben Reaves, *"A Robust Algorithm for Word Boundary Detection in the Presence of Noise."* IEEE Transactions on Speech and Audio Processing, Vol.2. No.3. July 1994, pp.406-412, 1994

[7] Montri Karnjanadecha, Stephen A. Zahorian, "Signal Modeling for High-Performance Robust Isolated Word Recognition." IEEE Transactions on Speech and Audio Processing, Vol.9 No.6, pp.647-654, 2001

[8] Montri Karnjanadecha, Stephen A. Zahorian, "Signal Modeling for Isolated Word Recognition." In Proc. ICASSP-99, pp.293-296, 1999

[9] James F. Kaiser, *"On a simple algorithm to calculate the 'energy' of a signal."* In Proc. IEEE ICASSP-90, pp.381-pp.384, 1990

[10] Wang Fan, Zheng Fang, Wu Wenhu, *"A Self-adapting Endpoint Detection Algorithm for Speech Recognition in Noisy Environments Based on 1/F Process."* ISCPLP-00, pp.327-330, 2000

[11] He Qiang, Zhang Youwei, *"On Prefiltering and Endpoint Detection of Speech Signal"*, In Proc. ICSP-98, pp.749-752, 1998

[12] John Huang, Ben-Dau Tseng, *"A Walsh Transform Based Endpoint Detection of Isolated Uterances."* Conference Record of the Twenty-Fifth Asilomar Conference-91, pp.335-338, 1991

[13] Minsoo Hahn, Chan Kyung Park, *"An Improved Speech Detection Algorithm for Isolated Korean Utterances."* In Proc. IEEE ICASSP-92, pp.525-528, 1992

[14] J. Taboada, S. Feijoo, R. Balsa, C. Hernandez, *"Explicit Estimation of speech Boundaries."* IEE Proceedings-Science, Measurement and Technology, pp.153-159, 1994

[15] D. Wu, M. Tanaka, R. Chen, L.Olorenshaw, M. Amador, X. Menendez-Pidal, *"A Robust Speech Detection Algorithm for Speech Activated Hands-free Applications."* In Proc. IEEE ICASSP-99, pp.2407-2410, 1999

[16] A. Ganapathiraju, L. Webster, J. Trimble, K. Bush, P. Kornman, *"Comparison of Energy-Based Endpoint Detectors for Speech Signal Processing."* In Proc. IEEE ICASSP-96, pp.500-503, 1996

[17] L.R. Rabiner, M.R. Sambur, *"An Algorithm for Determining the Endpoints of Isolated Utterances."* Bell Syst. Tech. J., Vol.54, pp.297-315, 1975

[18] L.F. Lamel et. Al., *"An Improved Endpoint Detector for Isolated Word Recognition."* IEEE Transactions on Acoustic, Speech, Signal Processing, Vol.29 pp.777-785, 1981

[19] Aini Hussain, Salina Abdul Samad, Liew Ban Fah. *"Endpoint Detection of Speech Signal Using Neural Network."* TENCON 2000. Proceedings, Vol.1 pp.271-274, 2000

[20] J. Navarro-Mesa, A. Moreno-Bilbao, E. Lleida-Solano. *"An Improved Speech Endpoint Detection System in Noisy Environments by Means of Third-Order Spectra."* IEEE Signal Processing Letters, Vol.6 No.9. pp.224-226, 1999

[21] Brian Mak, Jean-claude Junqua, Ben Reaves, *"A Robust Speech/Non-speech Detection Algorithm Using Time and Frequency-Based Features."* In Proc. IEEE ICASSP-92, pp.269-272, 1992

[22] Qifeng zhu, Abeer Alwan, *"On the Use of Variable Frame Rate Analysis in Speech Recognition."* In Proc. IEEE ICASSP-00, pp.1783-1786, 2000

[23] Philippe Le Cerf, Dirk Van Compernolle, *"A New Variable Frame Rate Analysis Method for Speech Recognition."* IEEE Signal Processing Letters, Vol.1 No.12 pp.185-187, 1994

[24] M.H. Savoji, *"A Robust Algorithm for Accurate Endpointing of Speech Signals."* Speech Communication-89, pp.45-60, 1989

[25] M.Bilginer Gulmezoglu, Vakif Dzhafarov, Mustafa Keskin, Atalay Barkana, "A Novel Approach to Isolated Word Recognition." IEEE Transactions on Speech and Audio Processing, Vol.7 No.6, pp.620-628, 1999

[26] Jianing Dai, "Isolated Word Recognition Using Markov Chain Models." IEEE Transactions on Speech and Audio Processing, Vol.3 No.6, pp.458-463, 1995

[27] Charles R. Jankowski Jr, Hoang-Doan H. Vo, Richard P. Lippmann, "A Comparison of Signal Processing Front Ends for Automatic Word Recognition." IEEE Transactions on Speech and Audio Processing, Vol.3 No.4, pp.286-293, 1995

[28] Stephen A. Zahorian, Zaki B. Nossair, "A Partitioned Neural Network Approach for Vowel Classification Using Smoothed Time/Frequency Features." IEEE Transactions on Speech and Audio Processing, Vol.7 No.4, pp.414-424, 1999

[29] Stephen A. Zahorian, Danming Qian, Amir J. Jagharghi, "Acoustic-phonetic Transformation for Improved Speaker-independent Isolated Word Recognition." In Proc. ICASSP-91, pp.561-564, 1991

[30] Lawrence Rabiner, Biing-Hwang Juang, "Fundamentals of Speech Recognition." Prentice Hall, Englewood Cliffs, New Jersey, 1993

CURRICULUM VITA
for
LINGYUN GU

DEGREES:

Master of Science (Electrical Engineering), Old Dominion University, Norfolk,Virginia, May 2002

Bachelor of Science (Electrical Engineering), University of Electronic Science and Technology of China, Chengdu, Sichuan, P.R.China, July 1998

SCIENTIFIC AND PROFESSIONAL SOCIETIES MEMBERSHIP:

IEEE Student Member, 2001 - Present

SCHOLARLY ACTIVITIES COMPLETED:

"A New Robust Algorithm for Isolated Word Endpoint Detection" (student paper), Orlando, ICASSP 2002

"Toolbox for Fundamental Frequency Estimation" (student paper), Salt Lake City, ICASSP 2001