Multi-lingual (Cantonese, Mandarin and English) Speech Recognition and Voice Response System

by

Li, Nga Ling (Bavy)

Department of Computing
The Hong Kong Polytechnic University
Hung Hom, Hong Kong

October, 2001

A Thesis Submitted
in Partial Fulfillment of
the Requirements for the Degree of
Master of Philosophy

Abstract

Multi-lingual (Cantonese, Mandarin and English) Speech Recognition and Voice Response System

As computer technology increasingly permeates our daily lives, hundreds of speech recognition applications are being implemented and run in business, industry and customer services areas. Hong Kong is a multicultural city, which allows people to use their native tongues to communicate within the same group, to support the three common dialects of Cantonese, Mandarin and English. In this thesis, it was aimed to build an integrated Automatic Speech Recognition (**ASR**) system for the three mentioned dialects without applying any prior knowledge of linguistic information.

For constructing our speech recognition system, (1) *Speech Segmentation*, (2) *Speech Preprocessing*, and (3) *Speech Recognition* are the three essential phases to study in this thesis. The objectives of the thesis include: (1) Finding a segmentation algorithm good for all three different dialects without any prior linguistic knowledge of any of them. (2) Using different existing parametric representations to produce different ranges of improvement on different speech recognition mechanisms for the three dialects. (3) Designing an integrated **ASR** system which would produce better results across the three dialects. In this thesis, the overall performance of our proposed segmentation algorithm and our proposed recognition algorithm were also measured through comparison with some common existing algorithms.

From our experimental results, our proposed Linguistically Free Segmentation (**LFS**) method is shown to be much more stable than the traditional **Zero Crossing** method by considering their standard deviation. It is also shown that different existing parametric representations give varied ranges of improvement on different speech recognition mechanisms for the three dialects. In this thesis, the best performance for recognizing

Cantonese can be achieved by applying Mel-frequency Cepstral Coefficients (**MFCCs**) features into Improved Naïve Bayesian Classification (**INBC**), whereas the best performance for recognizing Mandarin and English can be achieved by applying **MFCCs** features into Hidden Markov Modeling (**HMM**) with Viterbi algorithm. From the results, it is indicated that an integrated **ASR** system (the composition of different algorithms from segmentation, preprocessing, and recognition phases) is needed for constructing a reliable speech understanding system for different kinds of spoken-languages in the society.

Finally, such integrated **ASR** system for the three studied dialects was followed by the use of a Zoological Fortune Telling application. We believe that the development of an integrated **ASR** system can be applied for a Voice Response System, which can provide smart support for millions of business transaction or enquiry customer service everyday. Such system can improve traditional human-computer interactions by permitting users to retrieve or manipulate different forms (speech, text, graphics, or set of actions) of output from applications. This part will be set as the future enhancement of our integrated **ASR** system and will not be emphasized in this thesis.

Acknowledgment

I would like to express my gratitude to my chief supervisor - Dr. James N. K. Liu, for his expert guidance. Without his insight, enthusiasm, and encouragement, this work would not have been possible. I would like to thank Dr. Qin Lu for her reviewing this document so quickly with lots of careful suggestions to improve this work. I would also like to thank both Dr. Jane You and Dr. H. V. Leong for their countless support.

I would also like to thank some of my schoolmates during my research study, namely Filli Cheng, Jimmy Chim and Stanley Yau, for their considerable help and suggestions. Besides, I thank all the research staff and students of the department, particularly those who were studying the similar subjects with me, for providing me a comfortable and warm research environment.

Above all, I thank my dear parents, my brother and sisters for their sacrificial love and endless support.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Speech has evolved over many centuries to achieve today's rich and elaborated form. Today's human communications are dominated by spoken language, whether face-to-face, over the telephone, or through television and radio. However, human-machine (computer) interaction is still largely dependent on keyboard strokes, mouse clicks, or some other mechanical means. As such, this interaction mode demands skills development by individuals, and presents a barrier to widespread use of computer systems. Consequently, the goal of overcoming this barrier by building machines that understand spoken language has attracted our attentions over the past 50 years, and even until today.

A spoken language-understanding interface would be invaluable to this purpose since speech communication is a natural and efficient mode for the human operator [13]. Examples of applications include database access and management (such as airline reservations), automatic dictation (especially for English, Chinese and Japanese languages), voice dialing in a mobile phone system, electronic secretarial assistance products, robots (the well known HAL in "2001" and a Space Odyssey), security control,

aids for the handicapped, and Interactive Voice Response in Call Centers. The market value in these areas is estimated to be in the billions of dollars worldwide.

1.1 Automatic Speech Recognition (ASR)

Automatic speech recognition **(ASR)** is the key issue to achieve a spoken language-understanding interface. An **ASR** system takes the acoustic waveform produced by the speaker as input, and produces a sequence of linguistic words as output corresponding to the input utterance.

The general procedures of **ASR** are listed as below [41]:

- When a person speaks, the vibration of the vocal tract causes disturbances in the surrounding air. Typically, many speech recognition systems utilize some form of microphone, which picks up sound waves and converts them to an **analogue waveform**.
- The analogue waveform is then sampled thousands of times per second so as to convert it to a digital form through **Analog-to-Digital (A/D) converter**. The intensity of the sound wave may be sampled 8,000 or more times per second and the level of the sound at each point may be represented by an 8-bit or 16-bit number.
- In discrete speech recognition, the end point of the speech signal is determined by the period of silence which follows it. In the case of continuous speech recognition, **speech segmentation** techniques have to be applied. However, the end points of words are much more difficult to consider since people tend to run words together.

- After converting the sound wave to a digital form, the recognition system will use a variety of mathematical techniques, called the **speech preprocessing** process, to break it down into a pattern of word templates without any loss of information.
- The system then tries to match the word template to other word templates, which are contained within the pre-defined vocabulary. The system then decides whether or not the input word is close enough to the best match by applying some decision rules. This process is so called **speech recognition**.
- If the system justifies the recognition, it will notify the user through the terminal's display or a synthesis facility. With the forms of text, graphical, and speech in voice response process, such system is *multimodal* and has potential to support more flexible and productive **human-computer interactions**.

1.2 Motivation

Existing **ASR** systems, such as Dragon NaturallySpeaking System [27], MIT SUMMIT System [55], OfficeTALK [21], SPHINX System [26], IBM ViaVoice System [17], must take into account many uncertainties in speech recognition before they become popular. The uncertainty associated with words that have been spoken to a speech recognition system is compounded by the acoustic uncertainty of the different accents (dialects) and speaking styles of individual speakers, by the lexical and grammatical uncertainty of the language the speaker uses, and by the semantic uncertainty of the subject the speaker wishes to talk about.

As shown in Table 1.1 [42], the major components of acoustics uncertainty include the general quality of a speaker's voice, speaking speed and loudness, accent (dialect), and unusual speaking conditions such as illness or stress. In addition, acoustic contaminants such as room noise or competing speakers constitute a problem. Lexical, syntactic and semantic knowledge must then be applied in a manner that permits cooperative interaction among the various levels of acoustic, phonetic and linguistic knowledge in minimizing the uncertainty.

Nature of Input	Isolated Words Connected Speech Continuous Speech
Response Time	Real Time Close to Real Time Offline Time
Accuracy	Error Free (>99.9%) Almost Error Free (>99%) Error with Tolerance (>90%)
Speech Collection Tools	Omni-directional Microphone Uni-directional Microphone Noise-canceling Microphone Telephone Line
Noise Sources	Background (air-conditioning/room) Noise Telephone Noise Reverberation Noise
Speech Features	Spectrum Measurement Number of Formants Zero-crossing Rate LPC, LPCC, MFCC, RASTA
Phonology	Voiced-unvoiced Energy Stress Intonation
Size of Vocabulary	Small (10-100 words) Medium (< 1,000 words) Large (1,000-10,000 words) Very Large (10,000+ words)
Speaker Characteristics	Dialect Sex Age
Speaker Models	Speaker Dependent Model Speaker Independent Model

Table 1.1 Factors affecting Feasibility and Performance of ASR systems

The goal of current **ASR** applications has always been to allow individual speakers to obtain information and perform transactions through machines simply by speaking naturally. Recognition of free-form conversation is not yet a reality, but its technology is now proving itself commercially viable in a number of customer service applications [19]. Thousands of users are using their voices to perform millions of business transaction everyday.

As we live in a multicultural city, Hong Kong is an ideal place to build up an integrated Automatic Speech Recognition (**ASR**) system for its three commonly used dialects (Cantonese, Mandarin and English). It is because we believed that there is no single combination of algorithms from segmentation, preprocessing, to recognition phases that can help to recognize all different spoken-languages with the best performance.

1.3 Suggested Solution

In order to reduce the many uncertainties of our integrated **ASR** system, some principle constraints must be considered as follows:

- It is assumed that speech is recorded under clean and clear conditions. The application environment is also assumed to be in clean and clear environments, or one in which there is relatively little noise in the background.
- In the small vocabulary situation, a word model for each word can be built after feature extraction of speech utterances, rather than to build a recognition system for each phone in the large vocabulary situation.

- In order to enhance the speech quality, it is appropriated that more than one microphone be installed for speech capture. Since noise from the speech can be separated out when captured through multiple microphones, it constitutes another way to perform speech recognition under noisy conditions.

Currently, **ASR** is a matured technology for isolated word (or continuous speech in some restricted domains) for some common spoken-languages in the world with speaker dependent mode working in a clean environment. However, we believe that the best performance for recognizing different languages can be achieved by different combination of algorithms from segmentation, preprocessing, to recognition phases. Our objectives of this thesis include: (1) Finding a segmentation algorithm good for all three different dialects without any prior linguistic knowledge of any of them. (2) Using different existing parametric representations to produce different ranges of improvement on different speech recognition mechanisms for the three dialects. (3) Designing an integrated **ASR** system which would produce better results across the mentioned three dialects. In this thesis, the overall performance of our proposed segmentation algorithm and our proposed recognition algorithm were also measured through comparison with some common existing algorithms.

Three essential phases are studied for constructing our integrated ASR system: (1) *Speech Recognition* is the process of partitioning an entire speech into some isolated sub-words with optimal boundaries. In this thesis, the Linguistically Free Segmentation (**LFS**) method will be proposed and its performance evaluation will be compared with one of the

traditional word-spotting methods, Zero Crossing method. (2) *Speech Preprocessing* is the normalization of the speech signal in order to reduce the variability of those uncertainties as described in Section 1.2. In order to produce different ranges of improvement on different speech recognition mechanisms, the three chosen parametric representations are Weighted Cepstral Coefficients (**WCEP**), Mel-frequency Cepstral Coefficients (**MFCC**), and Relative Spectral Coefficients (**RASTA-PLP**). (3) *Speech Recognition* is the process of matching the input word template to other word templates from the pre-defined vocabulary. In this thesis, the three speech recognition mechanisms are Improved Naïve Bayesian Classification (**INBC**), Hidden Markov Modeling (**HMM**) with Viterbi algorithm, and Multi-Layers Backpropargation Modeling (**ML-BkProp**).

Finally, a Zoological Fortune Telling application is presented to implemented our integrated **ASR** system for the three studied dialects. It is believed that the development of an integrated **ASR** system can be applied for a Voice Response System, which can improve traditional human-computer interactions by permitting users to retrieve or manipulate different forms (speech, text, graphics, or set of actions) of output to support multimodal. This part will be set as the future enhancement of our integrated **ASR** system and will not be emphasized in this thesis.

1.4 Outline of the Thesis

In this chapter, we have briefly looked at **ASR** system and some factors affecting the feasibility and performance of **ASR** system. We have also discussed the motivation, the

objectives, and the suggested solution of this thesis. The rest of the thesis is organized as follows:

- Chapter 2 introduces the three Hong Kong common dialects (Cantonese, Mandarin and English) and their characteristics. It also gives a briefly description of some current approaches of **ASR** system including *Acoustic Phonetic*, *Pattern Recognition*, and *Artificial Intelligence*.
- Chapter 3 gives a description of the Zero Crossing method and our proposed **LFS** method, consisting of *Convex Hull*, *Spectral Variation Function*, *Normal Decomposition*, and *Level Building Dynamic Programming-based Algorithm*. Results for evaluating our proposed **LFS** method are also presented in this chapter with the comparison of the Zero Crossing method.
- Chapter 4 presents a description of speech preprocessing modeling with *Linear Predictive Coding* (**LPC**) analysis. Some existing **LPC**-based parametric representations, **WCEP**, **MFCC**, **RASTA-PLP**, are also presented for further development on different speech recognition mechanisms.
- Chapter 5 gives a description of the three speech recognition mechanisms including our proposed **INBC**, **HMM** with *Viterbi* algorithm, and **ML-BkProp**. Evaluation experiments of our integrated **ASR** system and the results are presented and discussed.
- Chapter 6 briefly describes current Voice Response techniques and a Zoological Fortune Telling application is presented to implement our integrated **ASR** system.
- Chapter 7 concludes the thesis and a summary of further work is also outlined.

Chapter 2

Background

2.1 Introduction

In the past fifty years, all forms of speech and writing have been recorded, stored, manipulated, reproduced or transmitted at will. Everyone is aware of the importance of speech in their lives, and almost everyone has some opinion or preconception about the languages they speak. Languages serve individuals and communities in a variety of ways for social and economic contact, and not just as a means of communication to express and share knowledge and information. Functions of language include thought, communication and self-identification, which are essential and overlapping aspects of finding and maintaining a personal place in the world.

In this chapter, we will introduce the three common dialects (Cantonese, Mandarin and English) and their characteristics in Hong Kong. We will also briefly look at several current approaches of **ASR** system for today's computer technology.

2.2 Languages and Dialects

Upon observation of communities throughout the world different languages are seen have to different pronunciation. All languages have their own inventory of speech sounds that are combined to form syllables and words. At the end of the 20th century, both Mandarin and English languages are widely used and reach approximately one billion speakers (including both mother-tongue and second language speakers) all over the world [50]. Cantonese, one of the so-called "dialects" of Chinese, is in fact related to Mandarin's written form with different pronunciation.

2.2.1 Chinese Language - Mandarin and Cantonese

The Chinese language is one of the oldest human languages known for which usage still continues throughout the world today. Each of the Chinese characters has its own pronunciation and it uniquely contains all three factors (Shape, Sound, and Meaning) [16]. This language has standardized written form but extremely different spoken forms. To consider the status and history of China, there have been many dynastic changes in this nation of two million square miles and a population of one billion. Several major dialects include speech from Beijing (today's Standard Mandarin is heavily influenced by it), Shanghai, Canton (Hong Kong's Cantonese is mainly based on Guangzhou dialect), Szchuan, Amoy, and Harka.

Mandarin was selected and published by the new government of China as the official spoken standard in 1918 [36]. After countless meetings of the scholars at the time, an agreement was finally reached and published by the Ministry of Education. As shown in Table 2.1, the set of pronunciations includes 22 Initials (including the null-initial) and 38 Finals (including the null-final), and 5 tones (Even, Rising, Dipping, Falling and Light). This gives a total of 2055 possible tonal syllables, but only 1335 of them are meaningful.

Cantonese (also known as Yue), a bi-syllabic language, is one of several major dialects in China as well as the target-spoken language in Hong Kong, Large numbers of Cantonese speakers are also found in Malaysia, Vietnam, Macao, Singapore, and Indonesia [50]. As similar as the spoken-Mandarin, each spoken-Cantonese character has two phonemes, initial (consonant) and final (vowel). For the whole set of Cantonese phoneme as shown in Table 2.1, there are 19 initials and 54 finals (including the null-final) for composing 595 syllables [54]. Besides the phonemes, Cantonese is characterized by the nine tones. There are 4115 Chinese syllables occurring with the composition of different initials, finals and tones.

(a) Initials

	Mandarin	**Cantonese**
Aspirated Stops	p t k	p t k ch kw
Non-aspirated Stops	b d g	b d g j gw
Aspirated Affricates	c ch q	-
Non-aspirated Affricates	z zh j	-
Nasals	m n	m n ŋ
Fricative and Continuants	f s sh x h	f l h s
Semi-vowels	l r	y w

(b) Finals

Group	Mandarin Finals	Group	Cantonese Finals
o	o, ou	a	a, ai, au, am, an, aŋ, ap, at, ak
e	e, eh, ei, en, eŋ, er	ɐ	ɐi, ɐu, ɐm, ɐn, ɐŋ, ɐp, ɐt, ɐk
u	u, ua, uo, uai, uei, uan, uen, uaŋ, ueŋ	e	ei
iu	iue, iuan, iun, iuŋ	ɛ	ɛ, ɛŋ, ɛk
i	i, iu, ia, ie, iai, iau, iou, ian, in, iaŋ, iŋ	i	i, iu, im, in, iŋ, ip, it, ik
-	null	o	ou
		X	X, Xi, Xn, Xŋ, Xt, Xk
		œ	œ, œy, œn, œŋ, œt, œk
		u	u, ui, um, un, uŋ, ut, uk
		y	y, yn, yt
		-	m, ŋ

Table 2.1 Characteristics of Mandarin and Cantonese

2.2.2 English Language

Unlike those ideographic spoken-Chinese, English is the alphabetical language with the major development of its recognition strategies and methodologies in the last two decades. Standard phonemes of American English contain 20 consonants, 12 vowels, 4 semi-vowels, and 4 diphthongs with the format of "/phoneme/:ALPABET" for voiced speech **(v)** and for unvoiced speech **(uv)** as shown in Table 2.2 [44]. DragonDictate and SPHINX [25,26] are the two well-known commercial **ASR** systems when dealing with English language study. These two systems have achieved relatively high performance in both speaker-independent model (**SI**, the system can recognize the speech pattern of a large proportion of population without any extra training) and speaker-dependent model (**SD**, the system requires each speaker to train the system individually).

CONSONANTS

Stops	{/b/:B, /d/:D, /g/:G} (v) {/p/:P, /t/:T, /k/:K} (uv)
Fricatives	{/v/:V, /ð/:TH, /z/: Z, /zh/:ZH} (v) {/f/:F, /θ/:THE, /s/:S, /sh/:SH} (uv)
Nasals	/m/:M, /n/:N, /ŋ/:NX
Whisper	/h/:H
Affricates	/j/:JH, /c/:CH

VOWELS

Front	/i/:IY, /I/:IH, /ɛ/:EH, /æ/:AE
Mid	/a/:AA, /ɚ/:ER, /ʌ/:AH, /ɔ/:AX, /X/:AO
Back	/u/:UW, /U/:UH, /o/:OW

SEMIVOWELS

Liquids	/w/:W, /l/:L
Glides	/r/:R, /y/:Y

DIPHTHONGS

/a^{y}/:AY, /X^{y}/:OY, /a^{w}/:AW, /e^{y}/:EY

Table 2.2 Standard Phonemes of American English

2.3 Current Approaches of ASR System

Broadly speaking, there are several basic approaches in **ASR** implementation by computer whereby the computer attempts to decode the speech signal in a sequential manner based on the observed acoustic features of the signal.

2.3.1 Acoustic-Phonetic (AP) Approach

The **AP Approach** is based on the theory of acoustic phonetics that there exist finite, distinctive phonetic units in spoken language (with its *syntax rules*) and such phonetic units are broadly characterized by a set of properties in the speech signal over time. The first step in this approach involves segmentation of the speech signal into discrete regions (in *times*), where the acoustic properties of the signal are representative of one or several

phonetic units, and then attaching one or more labels to each segmented region according to the acoustic properties. The second step attempts to determine a valid word (or *phrase*) from the sequence of phonetic labels produced in the first step, these words should be drawn from a given vocabulary.

Figure 2.1 shows the common modules involved to speech recognition. The first module, *Speech Analysis System* or *Feature Measurement*, provides an appropriate spectral representation of the characteristics of the time-varying speech signal using either the Filter Bank method [48] or the Linear Predictive Coding (**LPC**) method [36]. The second module, *Feature-Detection Stage*, consists of a set of *detectors* that operate in parallel to convert spectral measurements to a set of features that describe the broad acoustic properties of the different phonetic units. Such features include nasality, frication, formant locations, voiced-unvoiced classification, and ratios of high-frequency and low-frequency energy. The third module, *Segmentation-and-Labeling-Phase*, finds stable regions and labels the segmented region according to the individual phonetic units matching. This module is the core of the acoustic-phonetic recognizer and some control strategies need to be applied to limit the range of segmentation points and label possibilities. Therefore, the final output of the recognizer will be either the word or word-sequence of speech with the best matching.

However, there are many problems associated with the **AP approach** to **ASR**,

1. The method requires a prior knowledge of the acoustic properties of phonetic units. However, knowledge of acoustic properties of the phonetic units is often established in a *posterior manner* in the acoustic-phonetic approach.

2. The choice of features is made mostly based on ad hoc consideration, which is often based on intuition and thus not optimal in a well-defined and meaningful sense.
3. The design of sound classifiers is also ad hoc with construction of binary decision trees. Although Classification and Regression Tree (**CART**) methods [43] has been used to make the decision trees more robust, optimal implementation of **CART** methods is seldom achieved.
4. No well-defined of automatic procedure exists for tuning the method on real, labeled speech.

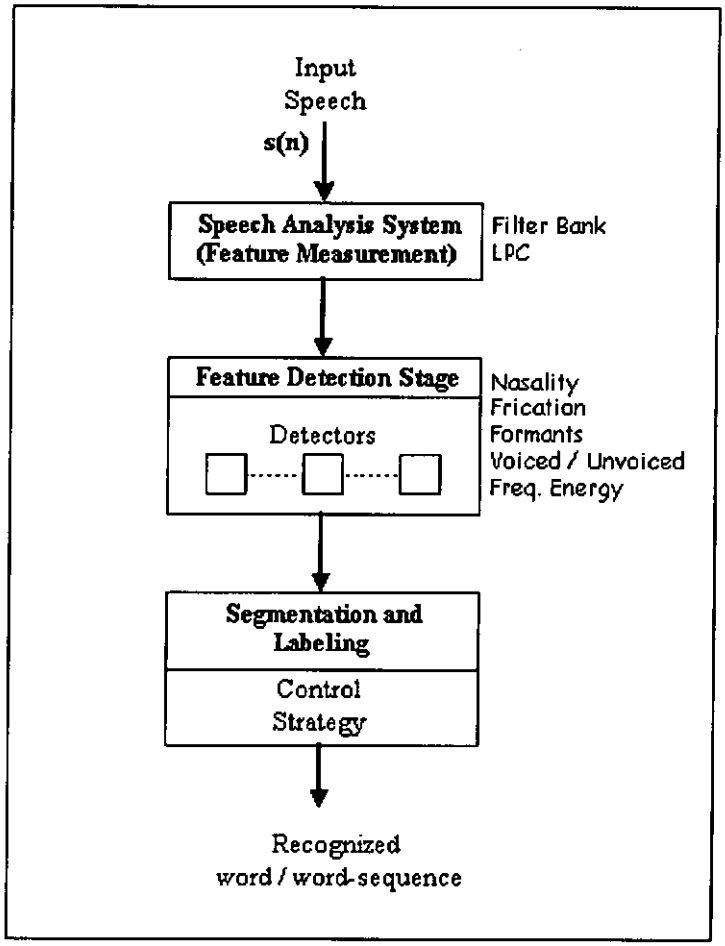


Figure 2.1 Block Diagram of the AP Speech Recognition System

2.3.2 Pattern-Recognition (PR) Approach

The **PR Approach** uses speech patterns directly without explicit feature determination and segmentation. In general, this method consists of *training of speech patterns* (with some existed speech "knowledge") and *recognition of patterns* through their comparison. If enough versions of a speech pattern to be recognized are included in a training set provided to the algorithm, the training procedure should be able to adequately characterize the acoustic properties of the pattern, which is called *Pattern Classification*.

This approach is rich in mathematical and communication theory justification for each training and decoding procedure. Based on robustness and invariance factors, it is easy to understand and simple to use on different speech vocabularies, speakers, and feature sets. A wide range of speech units (words, phrases, and sentences) can be applied on different pattern comparison algorithms and decision rules in order to produce high performance.

In Figure 2.2, *Feature Measurement* proceeds under some types of spectral analysis techniques, such as a filter bank analyzer, a **LPC** analysis, or a *Discrete Fourier Transform* (**DFT**) analysis [6], to define a set of test patterns. The *Pattern Training* of some test patterns, which are corresponding to speech sounds of the same class in order to create a pattern representative of the features of that class. The resulting pattern, a reference pattern, can be a template deriving from some types of averaging techniques, or can be a model characterizing the statistics of the features of the reference pattern. *Pattern Classification* compares the unknown test pattern with each class reference pattern by measuring the similarity between the test pattern and each reference pattern. To compare speech patterns, we require both a *local distance* between two well-defined

spectral vectors, and a global time alignment procedure, *Dynamic Time Warping* (**DTW**) algorithm [39], to adjust different rates of time scales of the two patterns. Finally, *Decision Logic* is achieved by using the reference pattern similarity scores to decide which reference pattern is the best match of the unknown test pattern. However, a new reference pattern is built if the defined decision threshold is exceeded.

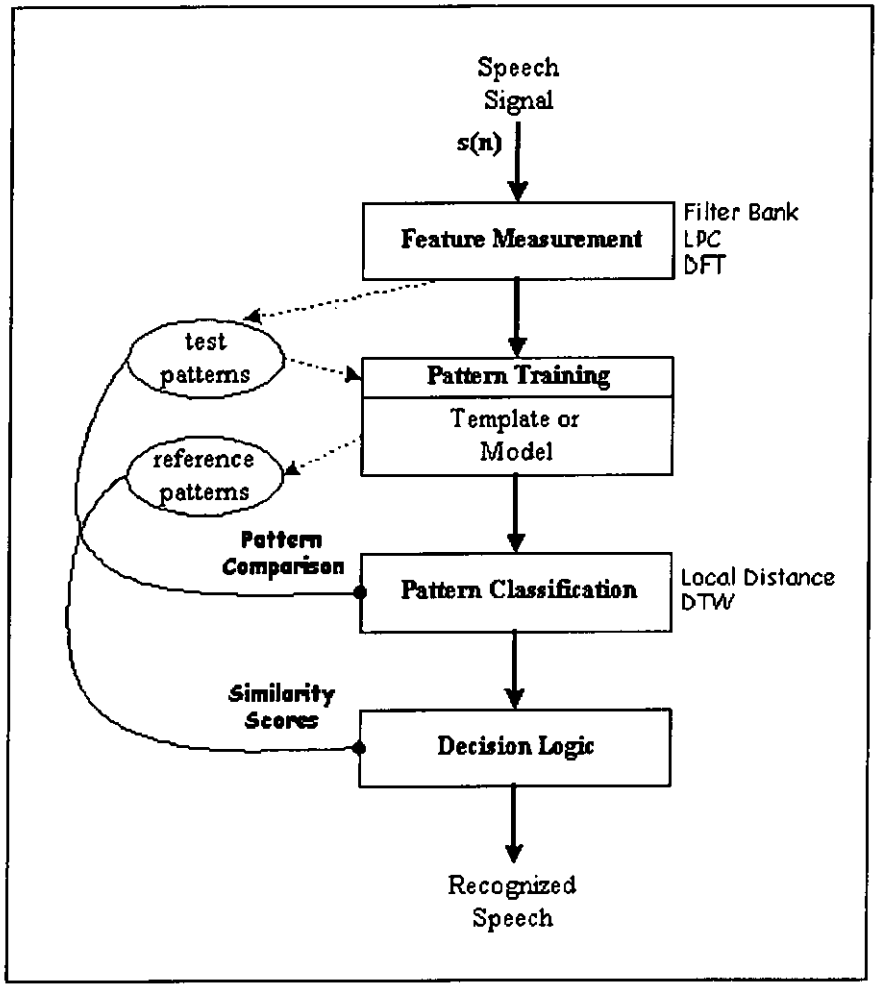


Figure 2.2 Block Diagram of the PR Speech Recognition System

Generally, the performance of the speech system under this approach is affected by the huge amount of training data available for creating sound class reference patterns. The quality of speech reference patterns is affected by the speaking environment and the

speech spectral characteristics' transmission. Moreover, the computation time loads for both pattern training and pattern classification is linearly proportional to the number of patterns being trained or recognized.

2.3.3 Artificial Intelligence (AI) Approach

The **AI Approach** is a *hybrid* of the acoustic-phonetic approach and the pattern-recognition approach by exploiting ideas and concepts of both methods. This approach attempts to mechanize the recognition procedure according to the way a person applies his/her intelligence in "visualizing", "analyzing", and "making a decision" on the measured acoustic features. The techniques applied within this class of methods are the use of an Expert System [44] for segmentation and labeling, learning and adapting over time, and the use of Neural Networks [10] for learning the relationships between phonetic events and all known speech inputs.

The basic idea of the **AI** system of speech recognition is to compile and incorporate knowledge from a variety of Knowledge Sources (**KS**). The overall **KS** include the characteristics of speech sounds (**Phonetics**), variability in pronunciations (**Phonology**), the stress and intonation patterns of speech (**Prosodic**), the sound patterns of words (**Lexicon**), the grammatical structure of language (**Syntax**), the meaning of words and sentences (**Semantics**), and the context of conversation (**Pragmatics**).

To integrate **KS** within a speech recognizer, the two standard approaches are the **Bottom-Up Processor** and the **Top-Down Processor**. For the former approach, the lower-level processes (such as *Feature Detection* or *Phonetic Decoding*) precede the higher-level

processes (such as *Lexical Decoding* or *Language Model*) in a sequential manner, as illustrated in Figure 2.3(a). For the latter approach, Figure 2.3(b) shows that the *Language Model* generates word hypotheses that are matched against the speech signal, and syntactically and semantically meaningful sentences are built on the basis of the word match scores.

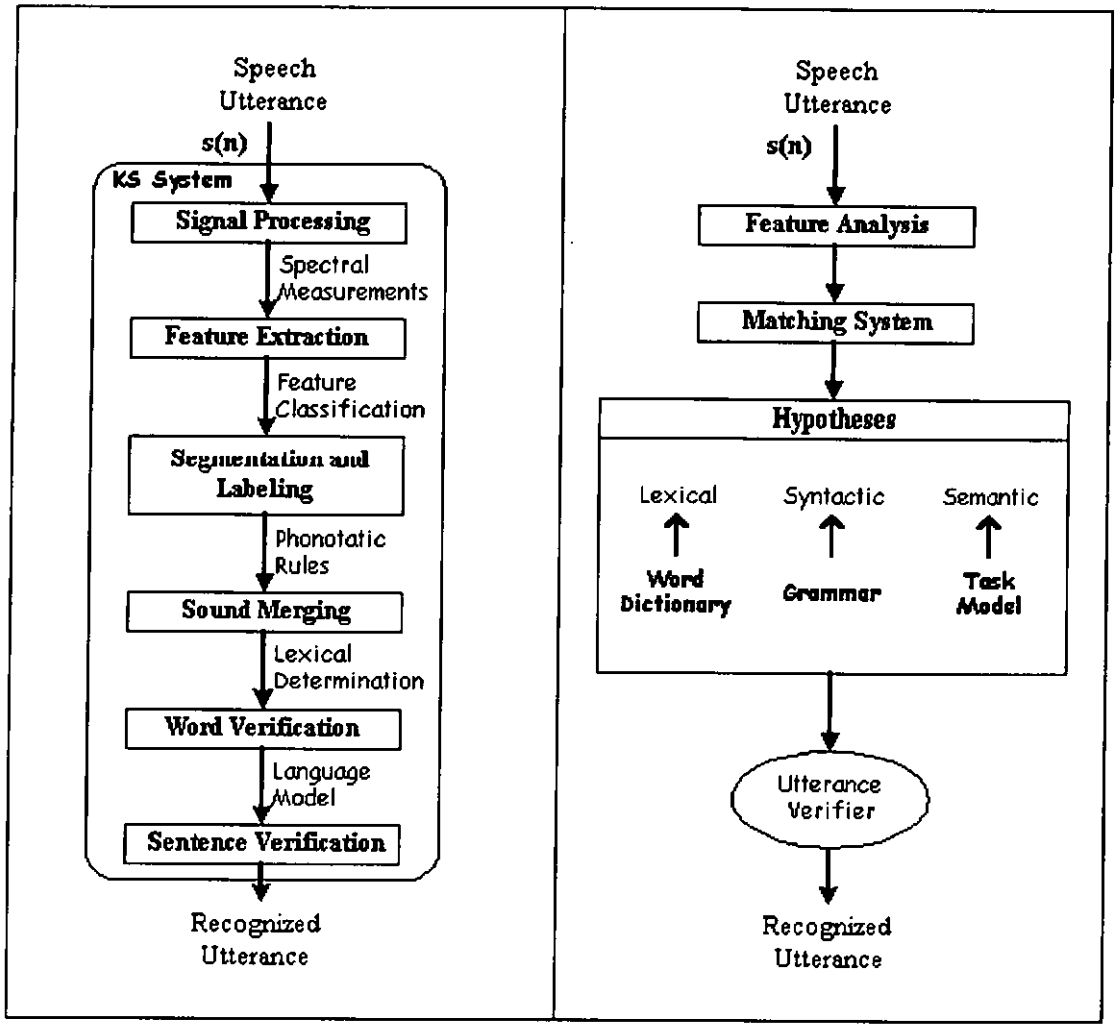


Figure 2.3 (a) A Bottom-Up [Left] and (b) a Top-Down [Right] Approaches to KS Integration

Chapter 3

Segmentation Modeling

3.1 Introduction

Segmentation is the process of partitioning an entire set of elements, such as signal, speech, or image, into finite regions with elements of similar characteristics being found in the same partition. Speech segmentation is the process of partitioning an entire speech into some isolated sub-words with optimal boundaries. Many earlier segmentation techniques were done manually by a trained phonetician using listening and visual cues and those techniques are time-consuming, subjective and error prone. Recently, the development of some **ASR** algorithms can be divided into two significant classes, *hierarchical* and *non-hierarchical*. The *hierarchical speech segmentation* tries to find the best path by applying multi-level tree searching methods [12]. The *non-hierarchical speech segmentation* attempts to locate the optimal segment boundaries by using dynamic programming-based methods [49].

In this chapter, we will propose a new Linguistically Free Segmentation (**LFS**) method [31,32] and compare its performance with the traditional **Zero Crossing** method [44]. Without any prior knowledge of linguistic information, our **LFS** technique not only

determines the optimal number of sub-word units presented in a given speech phrase, but also locates the optimal boundary for each sub-word with a combination of algorithms.

3.2 Zero Crossing

In many real-time applications of speech processing, sampled speech data is presented to the processor as a continuous stream of data, and analysis of the data can be carried out on a sample-by-sample basis. Zero Crossing method is one of the short-time domain analysis and endpoint detection methods. It usually estimates the characteristics of speech as an expectation over a number of sample values using a moving average filter. Zero Crossing rate only gives a reliable estimate in cases where energy level is too low for endpoint detection.

Zero Crossing method (along with energy information) is often used as a gross estimate of the frequency content of a speech signal in making a decision about whether a particular segment of speech is voiced or unvoiced [44]. Typically, this method for segmentation the entire speech into sub-words are based on pitch detection. In the voicing analysis, the zero crossing rate measurement of each frame is considered because the energy of some sounds is low although there is speech. Therefore, the zero crossing rate threshold, normally set as 0.4, must be introduced for indicating the boundaries of those utterances in a speech segment. The implication of the segments is most likely to be voiced when the zero crossing rate is above its threshold value. Traditionally, this word-spotting method is trivial and simple on speech segmentation estimation.

The zero crossing rate of each frame can be formulated as

$$\text{Zero}(f) = \frac{1}{L} \sum_{i=(f-1)\times L+1}^{f\times L-1} \frac{|\, sign(s(i+1)) - sign(s(i)) \,|}{2} \cdot w(f * L - i) \qquad (3.1)$$

where f is the position of the working frame

L is the fixed length of each frame

w is a window function (the hamming window is applied here)

$s(i)$ is the digitized speech signal at position indicator i

Zero Crossing rate is estimated by a stream processing approach if the signal is processed by a single-step quantizer set at zero level. The difference between each quantized sample and its posterior is taken, and half the magnitude of that difference is presented to the window filter as shown in Equation 3.1.

3.3 LFS Method

In general, speech segmentation is an important process in **ASR** system, since the faster training and better results can be achieved with good speech data segments. Our **LFS** method involves estimating the minimum possible number of speech segments (*Convex Hull*) [35]; estimating the maximum possible number of speech segments (*Spectral Variation Function*) [30]; finding the optimal number of segments between the two values (*Normal Decomposition*) [11]; and locating the corresponding speech boundaries (*Level Building Dynamic Programming-based Algorithm*) [40].

3.3.1 Convex Hull

Convex Hull is defined as the smallest convex set containing finite points, thus its computation allows the determination of extreme points of the set [35]. Its *Subjective Loudness Function* (temporally smoothed log-energy of speech) is monotonically non-decreasing from the start of the segment to its point of maximum loudness, and is monotonically non-increasing thereafter. Within the given segment, if the maximal difference between Convex Hull and the Loudness Function exceeds the threshold (normally set as 2 dB), the segment is divided into two sub-segments. Such segmentation procedure is carried out recursively until the maximal difference within the new defined segment does not exceed the threshold. This method eventually finds the number of the syllabic units in a given speech segment, which can logically be treated as the possible minimum number of sub-word segments, S_{min}.

In general definition, the *Convex Hull* of a set X is the smallest convex set containing finite points in X. If $X \subset \Re^n$ and $p \in \text{Hull}\, X$, then $p \in \text{Hull}\, Y$ for some $Y \subset X$ with $\text{card}(Y) \leq n+1$, such as

$$\text{Hull}\, X = \left\{ \sum_{i=1}^{q} \alpha_i p_i, p_i \in X, \sum_{i=1}^{q} \alpha_i = 1 \right\} \tag{3.2}$$

The relationship between Subjective Loudness (SL) and the intensity (I) or average pressure variation (Δp) of a single sound source is approximated as

$$SL = C_1 \sqrt[3]{I} = C_2 \sqrt[3]{(\Delta p)^2} \tag{3.3}$$

where both C_1 and C_2 are parameters that are functions of frequency.

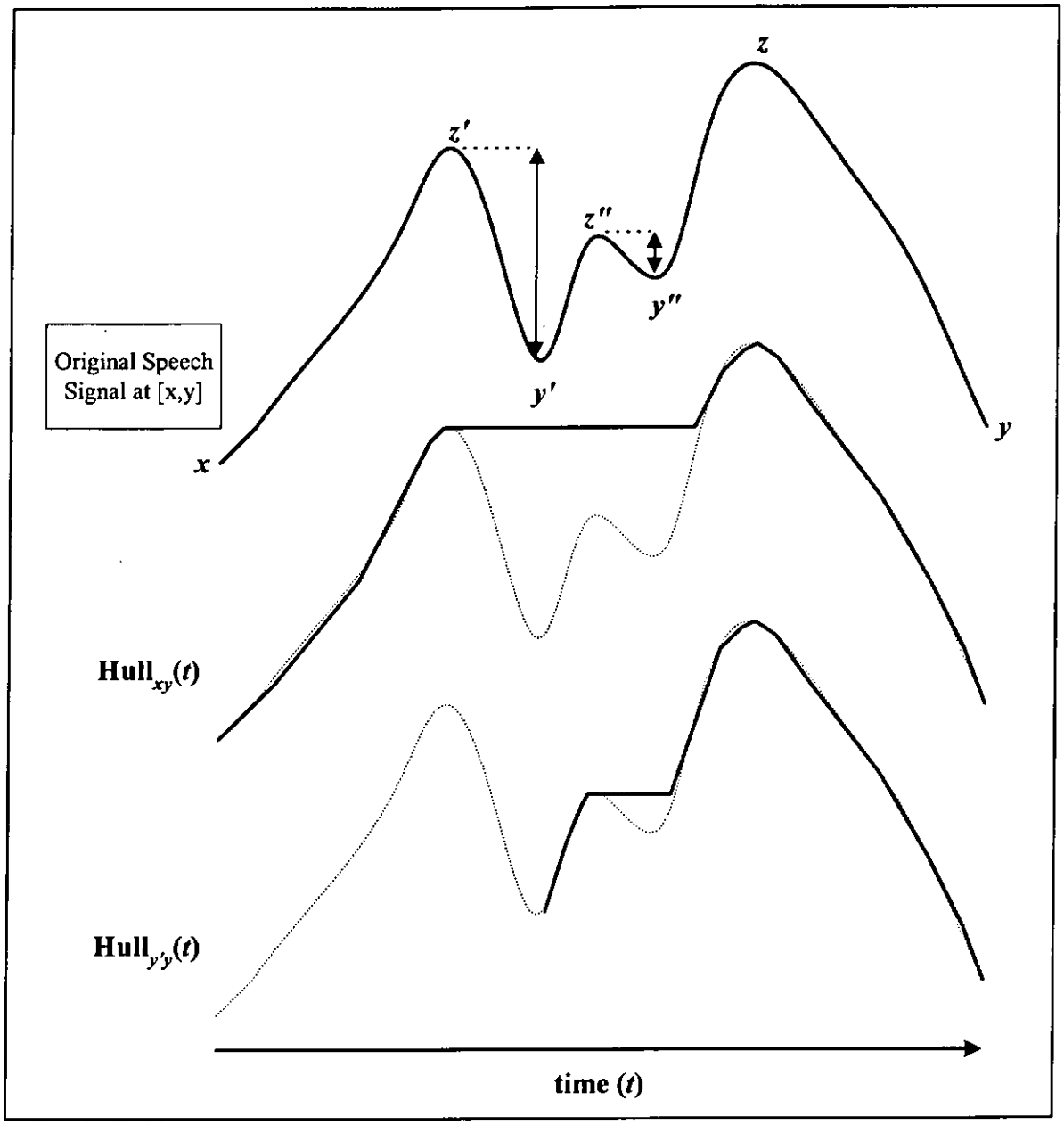


Figure 3.1 Convex Hull for the short-time Speech Interval [x,y]

An original speech segment with at least 200 millisecond time-slices between each sub-word is introduced for demonstrating the mentioned Convex Hull method. As shown in Figure 3.1, the given speech segment over the interval [*x,y*] shows a maximum point at position *z*. After applying Convex Hull for the segment (*x-z-y*), $Hull_{xy}(t)$, we find the

maximum difference is located at position y'. The interval $[x,y]$ has to be divided into two sub-segments, $(x\text{-}z'\text{-}y')$ and $(y'\text{-}z\text{-}y)$, only if the maximum difference exceeds the threshold. Convex Hull application for the segment $(x\text{-}z'\text{-}y')$, $Hull_{xy'}(t)$, produces a point of maximum output at position z' and results in a zero-like difference. Hence, the segment $(x\text{-}z'\text{-}y')$ is not segmented further. Convex Hull for the segment $(y'\text{-}z\text{-}y)$, $Hull_{y'y}(t)$, we find the maximum difference at position y''. If we assume that such difference does not exceed the threshold, and therefore, the segment $(y'\text{-}z\text{-}y)$ is not segmented further. It is clear that the above segmentation procedure is carried out recursively to each sub-segment until no further segmentation is performed. In the above case, the number of syllabic units (S_{min}) in the speech interval $[x,y]$ can be estimated to 2.

3.3.2 Spectral Variation Function (SVF)

The frame-based **SVF** is computed by applying the Euclidean norm of the delta cepstral coefficients set for estimating the maximum number of sub-word segments [30], S_{max},

$$SVF(f) = \sqrt{\sum_{m=1}^{p} \left(\Delta C_f(m)\right)^2} \tag{3.4}$$

where p is the order of a cepstral coefficients' vector

$\Delta C_f(m)$ is the m^{th} delta cepstral coefficient at frame f

In Equation 3.4, the coefficients represent the log of the time-varying and short time spectrum of speech, which can be derived from the well-known **LPC** parameters set with the residual energy [44]. Although **SVF**(f) exhibits peaks at boundaries between phonemes (speech sounds) over time, it may also locate too many insignificant (small and

spurious) peaks to the input speech. This function estimates the value of S_{max} by considering the maximum round up integer from the whole set of SVF(f) values.

3.3.3 LBDP-based Algorithm

A Level Building Dynamic Programming (**LBDP**)-based algorithm obtains the best path (the optimal boundaries set) by minimizing distortion metric [40]. This algorithm can be used to segment the given speech phrase into S^* sub-words' segments (where $S^* \in [S_{min}, S_{max}]$) by minimizing the overall internal segment distortion.

Assume that the original speech signal, *s(i)*, is blocked into a set of M frames with the fixed frame-length L of each frame. Local Distance from frame a to frame b in segment counter, S, can be formulated as

$$LD_S(a,b) = \sum_{i=a}^{b}\sum_{j=1}^{L}\left(x_j^i - \bar{x}_j^{ab}\right)^2 \tag{3.5}$$

where x_j^i denotes the speech sample point at position j and the i^{th} frame

$\bar{x}_j^{ab}$ denotes the speech sample point at position j from the mean vector of frames a through b.

Based on the Local Distance definition [24], the local Minimum Accumulated Distance for all possible segments combination can be computed for producing the optimal sub-words boundaries in the original speech signal. The calculation for the segments' set is presented as follows:

For Segment $S = 1$:

$$\begin{aligned} AD_S(i) &= LD_S(1,i) \qquad \forall i = 2 \text{ to } (M - 2(S^* - 1)) \\ BP_S(i) &= 1 \end{aligned} \tag{3.6.1}$$

For Segment $S = 2$ to $(S^*\text{-}1)$:

$$\begin{aligned} AD_S(i) &= \min_j \{LD_S(j+1,i) + AD_{S-1}(j)\} \\ BP_S(i) &= \arg\min_{j+1} \{LD_S(j+1,i) + AD_{S-1}(j)\} \\ &\forall i = 2S \text{ to } (M - 2(S^* - S)) \quad \text{and} \quad \forall j = i - 2 \end{aligned} \tag{3.6.2}$$

For Segment S = S^*:

$$\begin{aligned} AD_S(i) &= \min_j \{LD_S(j+1,M) + AD_{S-1}(j)\} \\ BP_S(i) &= \arg\min_{j+1} \{LD_S(j+1,M) + AD_{S-1}(j)\} \\ &\forall i = 2S^* \text{ to } M \quad \text{and} \quad \forall j = i - 2 \end{aligned} \tag{3.6.3}$$

where S is the counter of the segments set

S^* is total number of segments in the set

$AD_S(i)$ is the minimum accumulated distance for the segment S to frame i

$BP_S(i)$ is the best path for locating the segment S with ending boundary at frame i

3.3.4 Normal Decomposition

Normal Decomposition [11] is used to estimate the optimal number of sub-word segments, S_{opt}, for the given set of M frames of speech, $\{X^1, X^2, \ldots, X^M\}$. This technique assumes that the overall distribution of the given speech can be approximated by the summation of the normal-like distributions of its boundaries' set. Using the sub-words' boundaries obtained from the **LBDP**-based algorithm in Section 3.3.3, the distribution of the given speech of S sub-words' segments can be expressed as

$$p(X) = \sum_{i=1}^{S} P_i p_i(X) \tag{3.7}$$

where $X = \{X^1, X^2, \ldots, X^M\}$ is a vector set of M frames of speech

P_i is the prior probability

$p_i(X)$ is normal with the expected vector E_i and covariance matrix Σ_i

Now, we can apply the Maximum Likelihood (**ML**) estimation to estimate P_i, E_i and Σ_i from M available vectors, $X^1, X^2, \ldots, X^M$, under the constraint $\sum_{i=1}^{S} P_i = 1$. The **ML** estimation formulas are shown as follow,

$$\begin{aligned} P_i &= \frac{1}{M}\sum_{j=1}^{M} q_i(X^j) \qquad\qquad \forall\, i = 1 \text{ to } S \\ E_i &= \frac{1}{M \cdot P_i}\sum_{j=1}^{M} q_i(X^j) X^j \\ \Sigma_i &= \frac{1}{M \cdot P_i}\sum_{j=1}^{M} q_i(X^j)(X^j - E_i)(X^j - E_i)^T \end{aligned} \tag{3.8}$$

where $q_i(X)$ is the posteriori probability of i^{th} segment and can also be expressed as $P_i p_i(X)/p(X)$.

To compute the Equation 3.8, we have to set some initial values of the parameters P_i, E_i and Σ_i. The original speech data vectors belonging to the i^{th} segment are suitable to estimate those initial parameters of the i^{th} normal distribution. In each segment i ($\forall i = 1$ to S), a set of λ values are found for each covariance matrix, Σ_i, by use of Lower and Upper Triangular (**LU**) Decomposition. Each speech data value of $p_i(X)$ in Equation 3.7 can be expressed as

$$\frac{1}{\sqrt{2\pi\lambda^2}}\exp\left[\frac{-(x-e)^2}{2\lambda^2}\right] \tag{3.9}$$

where e is one of the data values in E_i of the segment i. Substituting the Equations 3.9 back into the Equations 3.5, the optimal number of sub-word segments (S_{opt}) can be finally adjusted by maximizing the log likelihood criterion, Q_S,

$$Q_S = \sum_{j=1}^{M} \ln p\left(X^j\right) \tag{3.10}$$

Equation 3.10 is calculated individually for various possible values of S (segments) until Q_S reaches a flat plateau at the optimal number, S_{opt}, or even decreases beyond S_{opt} due to estimation errors [30,31]. Once the optimal number of sub-words segments is set, we can reuse the **LBDP**-based algorithm as introduced in Section 3.3.3 to locate the optimal boundaries for the fixed value of S_{opt}.

3.4 Experimental Environment

The experimental setting of speech segmentation process is outlined before we compare the performance of the Zero Crossing method and our **LFS** method. In our experiment, five repetitions of digits (from five 0s to five 9s) of the three common dialects (*Cantonese*, *Mandarin*, and *English*) were recorded from 5 male and 5 female speakers. Each speaker recorded two sets of such five-times-repeated-digits for each dialect through the PC Sound Recorder software. We collected the speech data with Pulse Code Modulation (**PCM**) format, which is sampled at a rate of 8000Hz. The sampling frequency is set at 8-bits mono, which ensures the original signal can be reconstructed afterward.

	In Male Voice		
	CANTONESE	MANDARIN	ENGLISH
	$[S_{min}, S_{max}]$	$[S_{min}, S_{max}]$	$[S_{min}, S_{max}]$
Data Set I			
0	[4,7][5,6][4,8][5,6][4,6]	[3,6][4,6][3,6][5,7][5,7]	[3,8][5,7][3,8][3,9][4,7]
1	[5,6][3,7][3,6][5,7][3,7]	[4,6][4,8][4,7][4,6][4,6]	[5,7][4,7][5,7][4,7][4,7]
2	[4,8][4,7][5,6][4,6][3,7]	[3,6][3,6][5,7][5,7][4,8]	[3,8][3,8][3,9][4,7][4,7]
3	[5,6][4,6][3,7][3,7][5,7]	[5,7][5,7][4,8][5,7][4,6]	[3,9][4,7][4,7][3,6][4,7]
4	[4,6][5,6][4,8][5,6][4,7]	[5,7][5,7][3,6][4,6][3,6]	[4,7][3,9][3,8][5,7][3,8]
5	[3,7][4,7][5,6][4,8][5,6]	[4,8][3,6][4,6][3,6][5,7]	[4,7][3,8][5,7][3,8][3,9]
6	[3,7][3,6][5,7][3,7][4,6]	[5,7][4,7][4,6][4,6][5,7]	[3,6][5,7][4,7][4,7][4,7]
7	[3,6][4,8][5,6][4,6][3,7]	[4,7][3,6][5,7][5,7][4,8]	[5,7][3,8][3,9][4,7][4,7]
8	[5,7][3,7][4,6][4,8][4,7]	[4,6][5,7][5,7][3,6][3,6]	[4,7][3,6][4,7][3,8][3,8]
9	[3,7][3,6][3,7][5,6][5,6]	[4,6][4,7][4,8][5,7][4,6]	[4,7][5,7][4,7][3,9][5,7]
Data Set II			
0	[3,7][4,6][5,6][4,8][5,6]	[4,8][5,7][5,7][3,6][4,6]	[4,7][4,7][3,9][3,8][5,7]
1	[3,7][5,7][3,6][3,7][3,7]	[4,6][4,6][4,7][5,7][4,8]	[4,7][4,7][5,7][3,6][4,7]
2	[5,6][4,6][3,7][3,7][3,6]	[5,7][5,7][4,8][5,7][4,7]	[3,9][4,7][4,7][3,6][5,7]
3	[5,6][4,8][5,6][3,7][4,6]	[4,6][3,6][5,7][4,8][5,7]	[5,7][3,8][3,9][4,7][4,7]
4	[3,7][5,6][3,7][3,6][5,7]	[4,6][5,7][5,7][4,7][4,6]	[4,7][3,9][3,6][5,7][4,7]
5	[5,6][5,6][3,7][3,6][3,7]	[4,6][5,7][4,8][4,7][4,6]	[5,7][3,9][4,7][5,7][4,7]
6	[4,7][4,8][4,6][3,7][5,7]	[3,6][3,6][5,7][5,7][4,6]	[3,8][3,8][4,7][3,6][4,7]
7	[3,6][5,7][3,7][4,7][5,6]	[4,7][4,6][4,6][3,6][4,6]	[5,7][4,7][4,7][3,8][5,7]
8	[4,8][5,6][4,6][4,7][3,7]	[3,6][5,7][5,7][3,6][4,6]	[3,8][3,9][4,7][3,8][4,7]
9	[3,7][5,7][3,6][3,7][3,7]	[4,6][4,6][4,7][5,7][4,8]	[4,7][4,7][5,7][3,6][4,7]

Table 3.1(a) Estimated Segment Range for Male Speakers

	In Female Voice		
	CANTONESE	MANDARIN	ENGLISH
	$[S_{min}, S_{max}]$	$[S_{min}, S_{max}]$	$[S_{min}, S_{max}]$
Data Set I			
0	[3,6][4,6][3,7][4,6][4,7]	[4,8][3,7][5,7][4,7][3,7]	[5,9][4,7][4,7][4,9][4,6]
1	[4,6][3,8][5,8][4,6][4,7]	[3,7][5,9][4,8][4,7][5,6]	[4,7][3,6][5,6][5,6][5,7]
2	[3,7][3,6][4,6][4,7][3,8]	[5,7][4,8][4,7][3,7][5,9]	[4,7][5,9][4,9][4,6][3,6]
3	[4,6][4,7][3,8][4,6][4,6]	[4,7][3,7][5,9][3,7][4,7]	[4,9][4,6][3,6][4,6][5,6]
4	[4,7][4,6][3,7][4,6][3,6]	[3,7][4,7][5,7][3,7][4,8]	[4,6][4,9][4,7][4,7][5,9]
5	[3,8][3,6][4,6][3,7][4,6]	[5,9][4,8][3,7][5,7][4,7]	[3,6][5,9][4,7][4,7][4,9]
6	[4,6][5,8][4,6][4,7][4,7]	[3,7][4,8][4,7][5,6][3,7]	[4,6][5,6][5,6][5,7][4,6]
7	[5,8][3,7][4,6][4,7][3,8]	[4,8][5,7][4,7][3,7][5,9]	[5,6][4,7][4,9][4,6][3,6]
8	[4,6][4,6][4,7][3,7][3,6]	[4,7][3,7][3,7][5,7][4,8]	[5,6][4,6][4,6][4,7][5,9]
9	[4,7][5,8][3,8][4,6][4,6]	[5,6][4,8][5,9][4,7][3,7]	[5,7][5,6][3,6][4,9][4,7]
Data Set II			
0	[3,8][4,7][4,6][3,7][4,6]	[5,9][3,7][4,7][5,7][3,7]	[3,6][4,6][4,9][4,7][4,7]
1	[4,6][4,7][4,6][3,8][5,8]	[5,6][4,7][4,8][3,7][5,9]	[5,6][5,7][4,7][3,6][5,6]
2	[4,6][4,7][3,8][4,6][5,8]	[4,7][3,7][5,9][3,7][4,8]	[4,9][4,6][3,6][4,6][5,6]
3	[4,6][3,7][4,6][3,8][4,7]	[3,7][5,7][4,7][5,9][3,7]	[4,7][4,7][4,9][3,6][4,6]
4	[4,7][4,6][4,6][5,8][4,6]	[5,6][4,7][3,7][4,8][4,7]	[5,7][4,9][4,6][5,6][5,6]
5	[4,6][4,6][3,8][5,8][4,7]	[3,7][4,7][5,9][4,8][5,6]	[4,7][4,9][3,6][5,6][5,7]
6	[3,6][3,7][4,7][4,6][4,6]	[4,8][5,7][3,7][3,7][4,7]	[5,9][4,7][4,6][4,6][5,6]
7	[5,8][4,6][4,7][3,6][4,6]	[4,8][4,7][5,6][4,8][3,7]	[5,6][5,6][5,7][5,9][4,7]
8	[3,7][4,6][4,7][3,6][4,7]	[5,7][4,7][3,7][4,8][5,6]	[4,7][4,9][4,6][5,9][5,7]
9	[4,7][4,6][5,8][4,6][3,8]	[5,6][4,7][4,8][3,7][5,9]	[5,7][5,6][5,6][4,6][3,6]

Table 3.1(b) Estimated Segment Range for Female Speakers

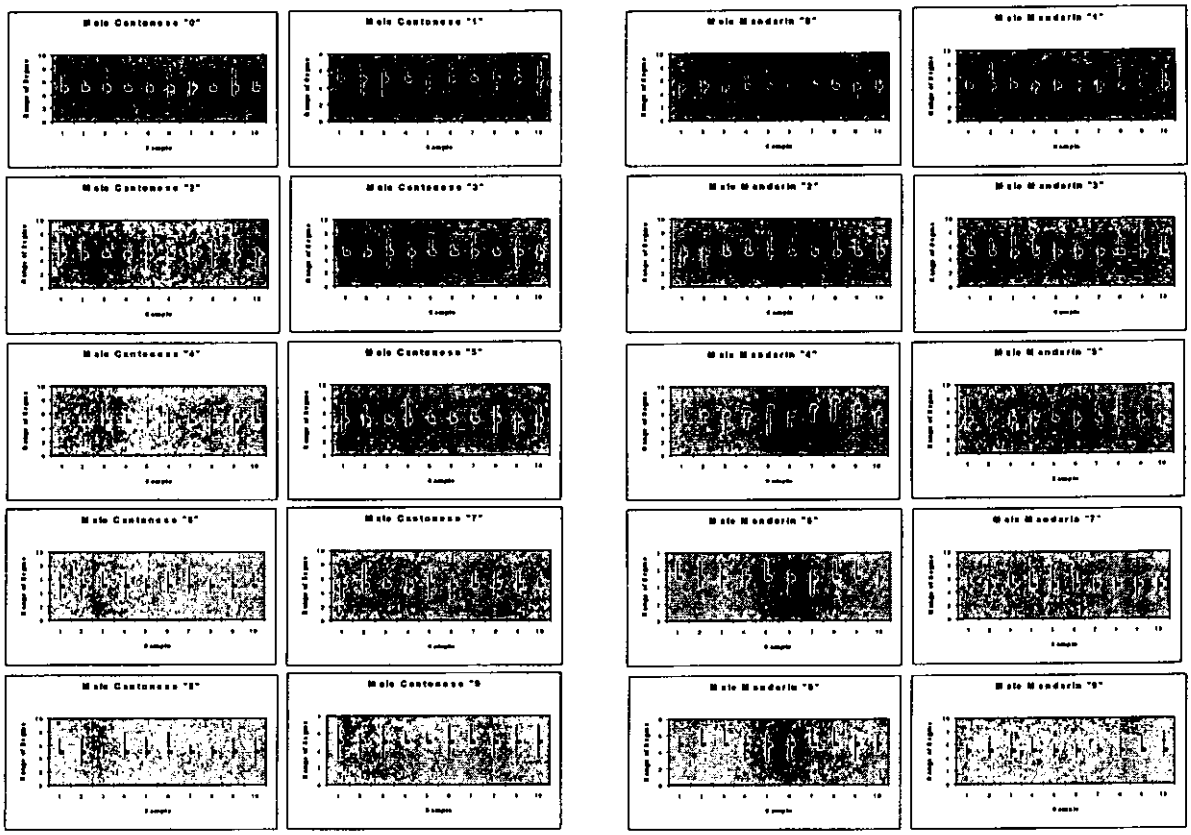
In Tables 3.1(a) and 3.1(b), the estimated range of sub-words segments $[S_{min}, S_{max}]$ of the three dialects set for the 5 male and 5 female speakers by applying Convex Hull and **SVF** from our **LFS** method is shown. Both tables contain 2 sets of data (I and II) of the ten speakers of each dialect.

3.5 Performance Evaluation

As shown in Tables 3.1(a) and 3.1(b), each speaker recorded 20 phrases of the five-times-repeated-digits for each dialect for segment range estimation. If the segment range had been found not less than 5 times from the 20 phrases for each speaker, it is summarized in Table 3.2 (except for female 2 and female 3 for English dialect). In the table, the format "$[3,7]^7[5,6]^5$" denotes that the range "[3,7]" occurred 7 times and the range "[5,6]" occurred 5 times for the particular speaker.

	CANTONESE	MANDARIN	ENGLISH
Male 1	$[3,7]^7[5,6]^5$	$[4,6]^8$	$[4,7]^8[5,7]^5$
Male 2	$[5,6]^5$	$[5,7]^8[3,6]^5$	$[4,7]^7[3,8]^5$
Male 3	$[3,7]^6[5,6]^5$	$[5,7]^8$	$[4,7]^9$
Male 4	$[3,7]^7$	$[5,7]^9[3,6]^5$	$[3,6]^5[3,8]^5[4,7]^5$
Male 5	$[3,7]^7$	$[4,6]^9$	$[4,7]^{13}$
Female 1	$[4,6]^8$	$[3,7]^5$	$[4,7]^5$
Female 2	$[4,6]^8$	$[4,7]^7[3,7]^5$	$\{[4,6]^4[4,7]^4[4,9]^4[5,6]^4\}$
Female 3	$[4,6]^8$	$[3,7]^5[4,7]^5$	$\{[3,6]^4[4,6]^4[4,7]^4[4,9]^4\}$
Female 4	$[4,6]^8$	$[3,7]^8$	$[4,6]^6$
Female 5	$[4,6]^7[4,7]^6$	$[3,7]^6$	$[5,6]^5$

Table 3.2 Common Segmented Range for Speakers



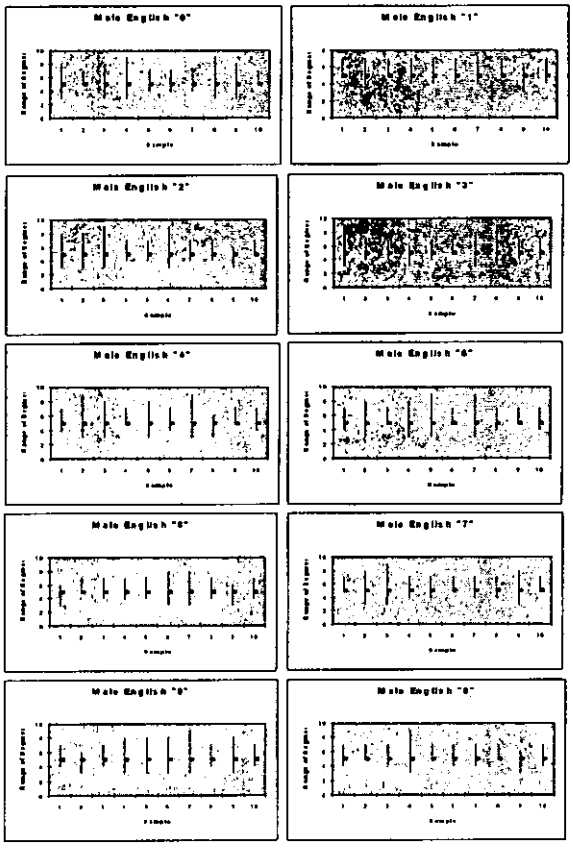
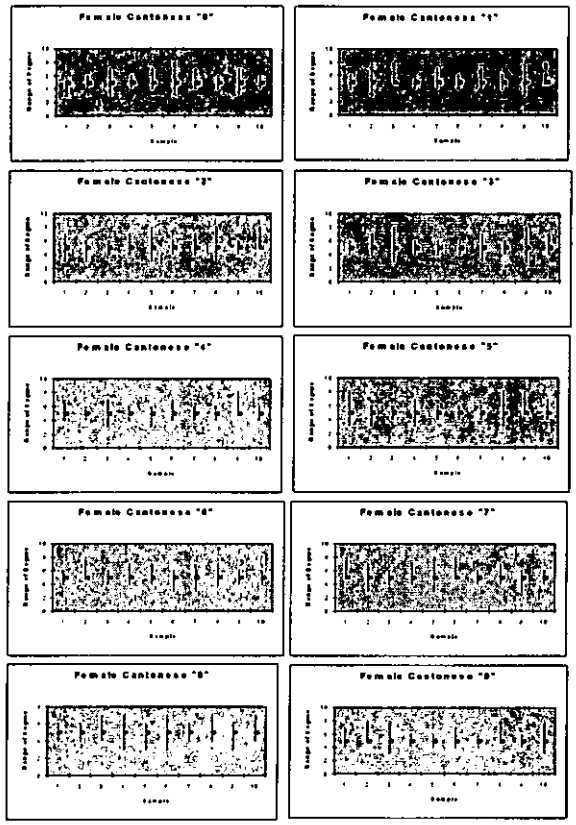
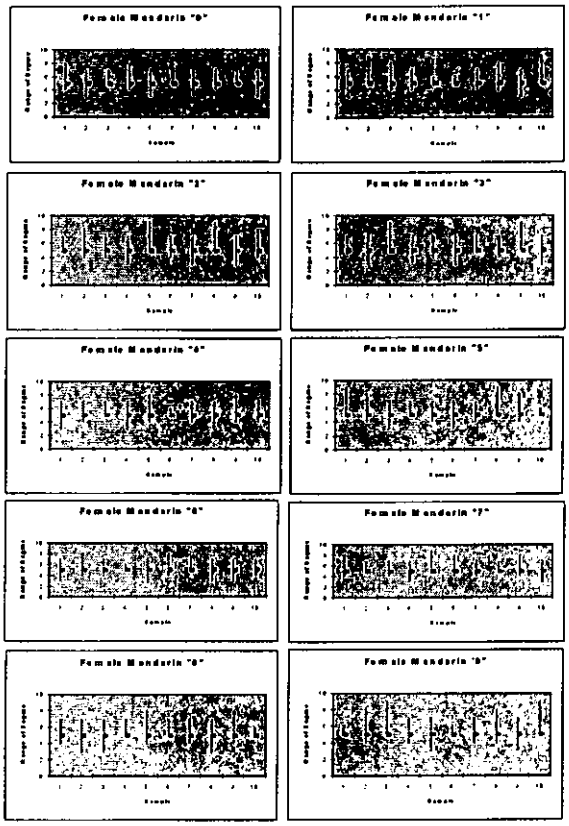


Figure 3.2 Segmented Range Estimation for Male Speakers for Cantonese (Top Left), Mandarin (Top Right), and English (Bottom Left)





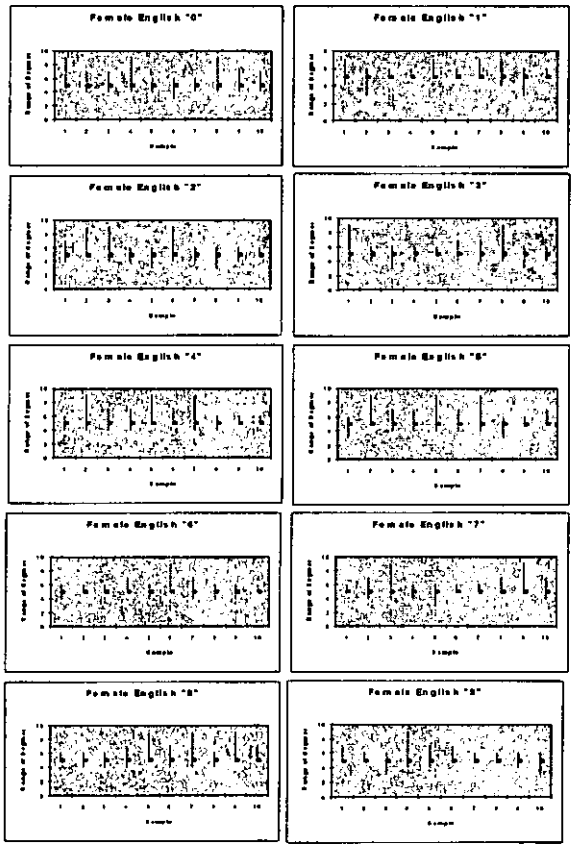


Figure 3.3 Segmented Range Estimation for Female Speakers for Cantonese (Top Left), Mandarin (Top Right), and English (Bottom Left)

From Tables 3.1 and 3.2, it is observed that the 5 male speakers have common segmented ranges at {29% for "[3,7]", 22% for "[5,6]"} when spoken in continued *Cantonese* digits. For both *Mandarin* digits and *English* digits, the common segmented ranges are {33% for "[5,7]", 27% for "[4,6]"} and {42% for "[4,7]", 18% for "[3,8]"}. The common segmented ranges of the 5 female speakers in the three sets of digits (*Cantonese, Mandarin*, and *English*) are represented as {39% for "[4,6]"}, {29% for "[3,7]"} and {20% for "[4,7]", 19% for "[4,6]"}, respectively.

In Figures 3.2 and 3.3, it shows that the segmented ranges estimation for both male and female speakers for the three observed dialects. For male speakers' cases, the overall best range estimation is shown in Mandarin digits set and the worse is shown in English digits set. Over 50% of the collected Mandarin digits have not more than 2-units range difference and some good estimation results can be found at digit 0, 1, 3, 6, 8 and 9. Over 30% of the collected English digits have at least 5-units range difference, thus longer time has to be spent on finding the optimal number of such segments and their corresponding boundaries by using LBDP-based algorithm and Normal Decomposition. The better-segmented ranges estimation in the Cantonese set is located at digit 0, 3, and 4. For female speakers' cases, the overall performance on range estimation in Cantonese set is relatively good results are shown at digit 3, 4, and 6. About 60% of the collected Mandarin digits have 4-units range difference and the results are the worst among the three dialects. In the English digits set, the good ranges estimation is located at digit 1, 6, 7 and 9.

In order to compare our proposed **LFS** method with the traditional **Zero Crossing** method, the optimal number of segments, S_{opt}, for each five-times-repeated-digits speech phrase for the three dialects has to be considered. Table 3.3 shows the accuracy of the S_{opt} count of the three dialects. As we mentioned before, 100 five-times-repeated-digits phrases were collected from 5 male speakers and 5 female speakers separately for each of the three dialects.

	LFS (in %)						*Zero Crossing (in %)*					
	Cantonese		*Mandarin*		*English*		*Cantonese*		*Mandarin*		*English*	
	M	**F**	**M**	**F**	**M**	**F**	**M**	**F**	**M**	**F**	**M**	**F**
0	88	85	87	83	77	83	82	70	88	70	95	92
1	84	84	87	84	86	90	72	71	77	71	78	83
2	86	83	86	82	78	84	93	81	79	70	78	92
3	87	84	87	83	79	85	72	71	78	83	94	93
4	87	87	88	84	78	85	71	82	77	83	67	82
5	87	84	86	84	79	83	81	92	95	93	66	82
6	85	87	88	84	83	90	81	71	77	83	67	70
7	86	85	87	84	81	87	70	94	88	82	68	83
8	85	86	88	85	77	86	83	71	78	71	68	82
9	85	84	87	84	84	88	81	82	69	71	79	82
Std	1.247	1.37	0.738	0.823	3.155	2.601	7.26	9.132	7.501	8.097	11.02	6.855

Table 3.3 Overall Accuracy of S_{opt} of the Three Dialects

From the results in Table 3.3, it is shown that the traditional word spotting method (**Zero Crossing**) is not as stable as the **LFS** method with extremely high standard deviation. Our proposed **LFS** method has an average of 85% accuracy in overall performance. Even for the worst case (the 5 male speakers in English digits set), an average of 80% accuracy is obtained in this set. On the other hand, the performance of the **Zero Crossing** method has range between 66% accuracy to 95% accuracy. For the male speakers' case, some extremely good results were found at digit 2 in *Cantonese* set, digit 5 in *Mandarin* set,

and digit 0 and 3 in *English* set. For the female speakers' case, some extremely good results were found at digit 5 and 7 in *Cantonese* set, digit 5 in *Mandarin* set, and digit 0, 2 and 3 in *English* set.

3.6 Conclusion

The general problem with segmentation is that we cannot segment perfectly. There is no simple computation method to judge the exact position of an expected boundary under the condition of continuous variation. It is observed that there a few missing sub-words boundaries, some extra boundaries, and some cases in which the location of the boundary is shifted when doing segmentation. Poor speech segmentation techniques are not only prolonging the time on training and recognizing, but also ruining the entire recognition system.

Amplitude (or energy), the single most important measure in segmentation, may be used to detect many of the speech boundaries. This is also the main concept for constructing the traditional **Zero Crossing** method. However, due to the broad frequency spectrum of most speech sounds, the interpretation of this method for speech is much less precise. Different kinds of speech noises also have disastrous effects on the measurement. Our proposed **LFS** method can automatically segment any continuous speech without the knowledge of any linguistic information. From our experimental results, it is shown that the **LFS** method is much more stable than the traditional word-spotting method.

Chapter 4

Parametric Representation

4.1 Introduction

The major decision to be made in the design of an **ASR** system is how to digitize and represent speech in the computer system. The first step of the system is the division of the continuous speech phrases into a sequence of segmented words (**speech segmentation**) as discussed in the last Chapter. The second step is the normalization of the speech signal (**speech preprocessing**) [18] in order to reduce the variability of those uncertainties as introduced in Chapter 1. In carrying out the normalization of the speech, using different existing parametric representations was believed to produce different ranges of improvement on different speech recognition mechanisms for the three studied dialects.

By taking the digitized speech input and converting it into some feature vectors for further computation and recognizing in our **ASR** system, some existing Linear Predictive Coding (**LPC**)-based parametric representations which include *Weighted Cepstral Coefficients* (**WCEP**), *Mel-frequency Cepstral Coefficients* (**MFCC**), and *Relative Spectral Coefficients* (**RASTA-PLP**) are presented.

4.2 Noise Normalization

Currently, **ASR** systems tend to be reliable as long as the sources of speech noise are kept invariant through some noise normalization techniques [51]. Such sources of noises include: background and channel noise from different speaking environments; electrical noise from different types of microphones, telephones and other sound recording devices; and different speaking styles and conditions of individual speakers.

Background Noises usually appear in real-life environments, such as sounds of air-conditioning systems, electric fans, fluorescent lamps, computer systems, background conversation, opening and closing doors, and so on. These types of noises are usually in steady state with the noise levels varying from **60 dB** to **90 dB** [47]. A head-mounted *noise canceling close speaking microphone* is one of the common tools to minimize the effect of background noises, since a slight movement of a microphone during sound recording may cause large fluctuation between the original and the recorded speech signals. Some normalization techniques are applied to reduce the effects of background noise and distortion, such as using Inverse Filtering on the long-time spectrum of speech and Spectrum Weighting to eliminate those parts of the spectrum with low signal-to-noise ratio.

The restriction of the bandwidth in telephone input system is usually from **300 Hz** to **3000 Hz**, which may introduce problems on burst noise, distortion, echo, crosstalk, frequency translation, envelope delay, clipping and so on. The use of different microphone headsets connected to the telephone system can improve its overall accuracy.

Moreover, techniques suggested for background noise normalization would also be applicable in telephone input normalization.

4.3 LPC Analysis

The changes in air pressure caused by the speech are sampled and digitized by a computer using an Analog-to-Digital (**A/D**) converter and the simplest digital representation of speech is Pulse Code Modulation (**PCM**). In *Sound Recorder* program under any PC window platforms, speech is sampled from 8-to-40 thousand times a second and is quantized at 8 to 16 bits per sample. It is assumed that parameters of speech remain unchanged over a short-time spectrum period and can be calculated by using the **LPC** analysis.

LPC [44], a mathematical method on speech preprocessing, provides an efficient way of finding a short-term spectral envelope estimate that has many desirable properties for the representation of speech, in particular the emphasis on the peak spectral values that characterize voiced sounds. The basic idea behind **LPC** is that a speech sample can be expressed as a linear function of a certain number of the preceding speech samples. The coefficients of the linear function are determined by least square error fitting of the short-time speech signal from which other parametric representations of speech, such as spectrum, formants, and vocal tract shape can be derived.

4.3.1 Pre-emphasizer

The digitized speech signal with time n, s_n, is put through a low-order digital system, to spectrally flatten it and make the signal less susceptible to finite precision effects during subsequent signal processing. The digital system used in the pre-emphasizer is either fixed or slowly adaptive by averaging transmission conditions, noise backgrounds, and so on. The most widely used emphasis network is presented,

$$\tilde{s}_n = s_n - p s_{n-1} \tag{4.1}$$

where the output of the emphasis network, $\tilde{s}_n$, is shown as the difference equation of the input to the network, s_n. It is often used to boost the higher frequencies. Typically, $0.96 \le p \le 0.99$ is used for fixed-point implementation. In a first-order adaptive pre-emphasizer, p is replaced by p_n and such value changes with time n according to the chosen adaptation criterion.

4.3.2 Blocking and Windowing

The pre-emphasized speech signal, $\tilde{s}_n$, is blocked into a set of frames of N samples, with M samples shifting for each of their adjacent frames. For instance, the first frame consists of the first N speech samples, then the second frame overlaps it by $N - M$ samples. This process will continue until all the speech is accounted for within one or more frames.

Similarly, if we denote the ℓ^{th} frame of speech by x_n^{ℓ}, and the entire speech signal contains L frames, we have

$$x_n^{\ell} = \tilde{s}_{M\ell+n} \qquad \begin{cases} \forall\, n = 0,1,\ldots,N-1 \\ \forall\, \ell = 0,1,\ldots,L-1 \end{cases} \tag{4.2}$$

In general, the suggested values for N and M are **300** and **100** with the frame rate **6670 Hz**, which corresponds to **45 msec** frames and **15 msec** frame shift. This is because when M is smaller than N, the resulting **LPC** spectral estimates will be correlated from frame to frame. When M is much smaller than N, the resulting **LPC** spectral estimates will be quite smooth from frame to frame. However, when M is larger than N, there will be no overlap between adjacent frames and some of the speech information will totally be lost.

Multiplying the speech signal by a window function, w_n, can minimize its discontinuities at the beginning and end of each individual frame for $0 \le n \le N-1$ as shown in Equation 4.3. Hamming Window [22] is one of the common window functions to be used for the auto-correlation method of **LPC** in the analysis.

$$\tilde{x}_n^{\ell} = x_n^{\ell} \cdot w_n \qquad 0 \le n \le N-1 \tag{4.3}$$

where

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

4.3.3 Gain Computation

Gain Computation is an optional step for computing the gain of each speech frame, which indicates the amplitude of the speech signal. It is measured by the RMS of the ℓ^{th} frame of speech signal [11] as shown in Figure 4.4. If the value of Gain^{ℓ} is less than the pre-defined threshold, it is concluded that no data is extracted from the ℓ^{th} frame.

$$\text{Gain}^{\ell} = \sqrt{\frac{\sum_{n=0}^{N-1}\left(\tilde{x}_n^{\ell} - \bar{\tilde{x}}_n^{\ell}\right)^2}{N}} \tag{4.4}$$

where $\bar{\tilde{x}}_n^{\ell}$ denotes the mean of $\tilde{x}_n^{\ell}$ for $0 \le n \le N-1$.

4.3.4 Cepstral Analysis

The source filter model of speech production decomposes the speech signal, s_n, into an excitation, e_n, and a linear filter, $H(e^{i\theta})$. In the frequency domain, this can be written as

$$S(e^{i\theta}) = H(e^{i\theta}) \cdot E(e^{i\theta}) \tag{4.5}$$

where $H(e^{i\theta})$ can be defined as the envelope of the speech power spectra and $E(e^{i\theta})$ can be defined as the fine detail of the excitation.

For most speech processing applications, we require only the amplitude spectra (with a suitable definition of the log of a complex number), hence the equation is

$$\log\left(\left|S(e^{i\theta})\right|\right) = \log\left(\left|H(e^{i\theta})\right|\right) + \log\left(\left|E(e^{i\theta})\right|\right) \tag{4.6}$$

The slowly varying components of $\log\left|S(e^{i\theta})\right|$ are represented by the low frequencies and the fine detail by the high frequencies. Hence, we can easily remove $\log\left|E(e^{i\theta})\right|$ by taking the Fourier Transform and retaining only the low frequency terms. This produces the cepstral analysis, as shown diagrammatically in Figure 4.1.

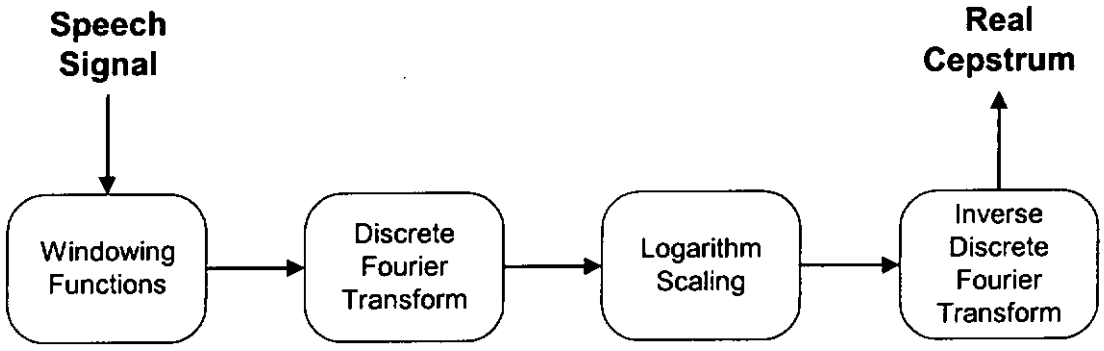


Figure 4.1 Cepstral Analysis

In the early 19th century, Fourier proposed to represent periodic signals by linearly combinations of harmonically related complex exponential [18]. The Discrete Fourier Transform (**DFT**) is exactly the output of Fourier Transform of an aperiodic sequence at some particular frequencies. The **DFT** followed by the Inverse-**DFT** results in an exact reconstruction of the original speech signal and provides an easy way to implement a digital filter. Let's consider that for $\widetilde{N}$ equally spaced frequencies of each individual window of the Fourier spectrum, the short-term **DFT** can be defined as

$$\begin{aligned}\widetilde{X}^{\ell}(\omega) &= \sum_{n=0}^{\widetilde{N}-1} \widetilde{x}_n^{\ell} \cdot e^{-i\omega n} \\ &= \sum_{n=0}^{\widetilde{N}-1} \widetilde{x}_n^{\ell} \cdot e^{-i\frac{2\pi k}{\widetilde{N}}n} \\ &= \sum_{n=0}^{\widetilde{N}-1} \widetilde{x}_n^{\ell} \cdot \left(\cos\left(\frac{2\pi kn}{\widetilde{N}}\right) - i\sin\left(\frac{2\pi kn}{\widetilde{N}}\right)\right) \qquad 0 \le k \le \widetilde{N}\end{aligned} \tag{4.7}$$

where $\tilde{N} \geq N$ denotes the new window size with the pattern 2^d and d is an integer, thus zero pads the windowed signal if $\tilde{N} > N$. Note that zero padding of the input is needed to ensure that there are no wrap-around effects for windows overlapping.

Equation 4.7 defines an algorithm that takes an array of $\tilde{N}$ real numbers and $\tilde{N}$ imaginary numbers and returns an array of $\tilde{N}$ complex numbers. The short-term Inverse-**DFT** can be similarly defined as

$$\begin{aligned}\tilde{x}_n^{\ell} &= \frac{1}{\tilde{N}} \sum_{k=0}^{\tilde{N}-1} \tilde{X}^{\ell}\left(\frac{2\pi k}{\tilde{N}}\right) \cdot e^{i\frac{2\pi k}{\tilde{N}}n} \\ &= \frac{1}{\tilde{N}} \sum \tilde{X}^{\ell}(\omega) \cdot e^{i\omega n} \\ &= \frac{1}{\tilde{N}} \sum \tilde{X}^{\ell}(\omega) \cdot (\cos(\omega n) + i\sin(\omega n)) \qquad 0 \leq n \leq \tilde{N}\end{aligned} \tag{4.8}$$

where ω is equivalent to $\frac{2\pi k}{\tilde{N}}$.

4.3.5 Autocorrelation

When dealing with windowed speech we need to take into account the boundary effects in order to avoid large prediction errors at the edges. Autocorrelation can be applied for each frame of windowed signal [11],

$$r_m^{\ell} = \sum_{n=0}^{\tilde{N}-1-m} \tilde{x}_n^{\ell} \cdot \tilde{x}_{n+m}^{\ell} \qquad \text{for } m = 0,1,\ldots,q \tag{4.9}$$

where the highest autocorrelation value, q, is the order of the **LPC** coefficients set with range from **8 to 16**. At the ℓ^{th} frame, Equation 4.9 can be represented in matrix form, which is shown as follows:

$$\begin{pmatrix} r_1^\ell \\ r_2^\ell \\ r_3^\ell \\ \cdots \\ r_q^\ell \end{pmatrix} = \begin{pmatrix} r_0^\ell & r_1^\ell & r_2^\ell & \cdots & r_{q-1}^\ell \\ r_1^\ell & r_0^\ell & r_1^\ell & \cdots & r_{q-2}^\ell \\ r_2^\ell & r_1^\ell & r_0^\ell & \cdots & r_{q-3}^\ell \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ r_{q-1}^\ell & r_{q-2}^\ell & r_{q-3}^\ell & \cdots & r_0^\ell \end{pmatrix} \begin{pmatrix} a_1^\ell \\ a_2^\ell \\ a_3^\ell \\ \cdots \\ a_q^\ell \end{pmatrix} \tag{4.10}$$

Equation 4.10 converts the ℓ^{th} frame of windowed signal into a set of **LPC** coefficients, a_m^ℓ, for $m = 1,2,\ldots,q$. The matrix can be solved by using Durbin's algorithm, thus the values of a **LPC** parameters set at iteration i are denoted by $a_m^{(i)}$ and the related residual energy is denoted by $E^{(i)}$ (with initial residual energy, $E^{(0)} = r_0$)

By reason of convenience, we will omit the superscript ℓ on the following parameters,

$$\begin{aligned} K_i &= \left(r_i - \sum_{j=1}^{i-1} a_j^{(i-1)} \cdot r_{|i-j|} \right) \Big/ E^{(i-1)} \\ a_i^{(i)} &= K_i \\ a_j^{(i)} &= a_j^{(i-1)} - K_i \cdot a_{i-j}^{(i-1)} \qquad\qquad i > j \geq 1 \\ E^{(i)} &= (1 - K_i \cdot K_i) \cdot E^{(i-1)} \end{aligned} \tag{4.11}$$

The set of formulae in Equation 4.11 are solved recursively for $i = 1,2,\ldots,q$, with the reflection parameters K_i. The value of the residual energy, $E^{(i)}$, under the squared prediction can decrease or remain constant at each iteration.

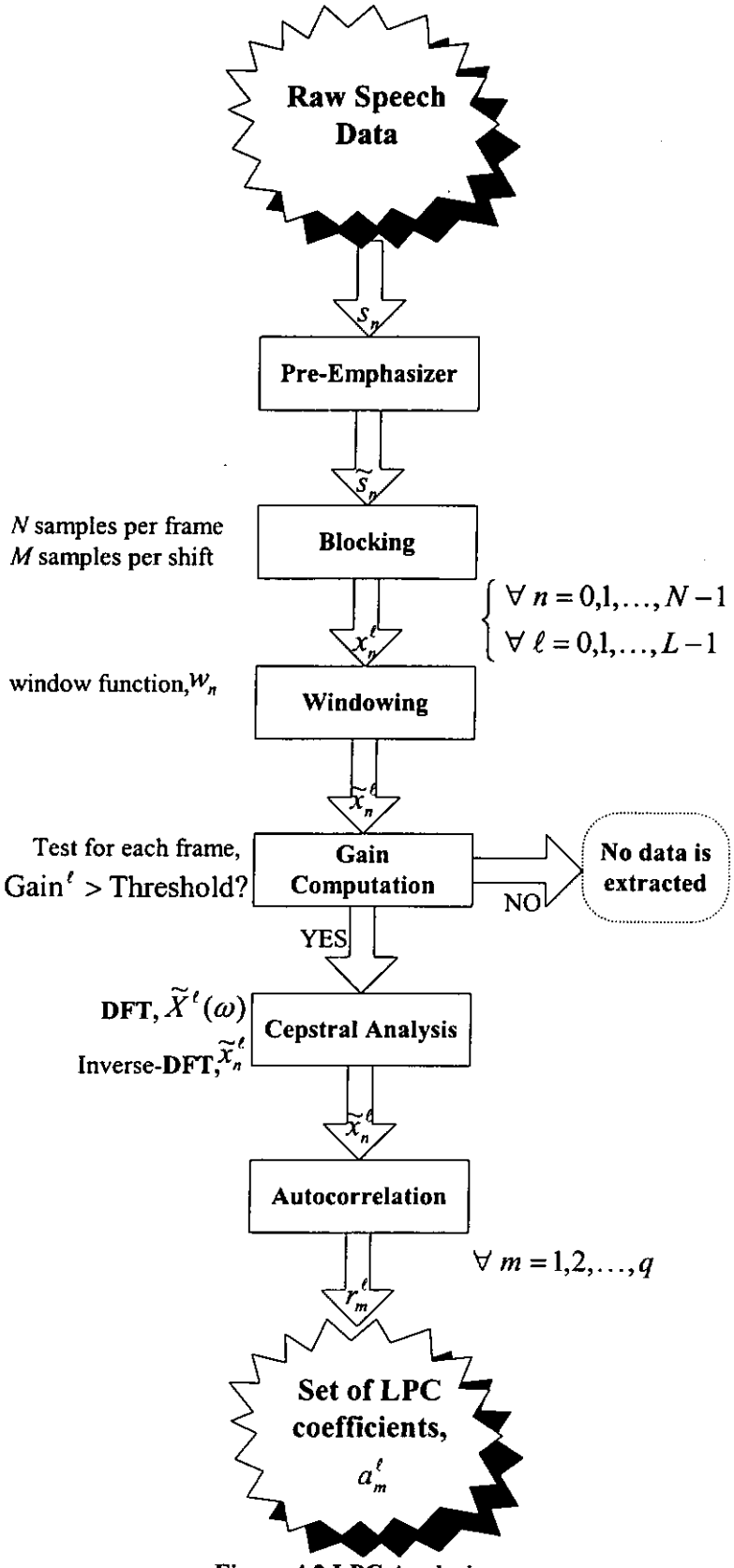


Figure 4.2 LPC Analysis

4.4 Features Extraction

The first step in any pattern-classification systems is to evaluate some representations of the input patterns, for instance in the case of speech some forms of power spectrum are often evaluated. In general, when features of speech are extracted within successive analysis windows of 20 milliseconds that often overlap by 10 milliseconds, the information in the speech signal can be condensed to a set of vectors. For the one-dimensional monaural speech (time series) classification, the input speech is transformed into a sequence of feature vectors that are sampled at a rate which is generally much lower than the original sequence.

In speech preprocessing modeling, it is desirable to find a good set of generalized features to reduce the dimensionality of the orginal speech sequence. In this chapter, some existing **LPC**-based parametric representations are introduced for our research study, which include *Weighted Cepstral Coefficients* (**WCEP**), *Mel-frequency Cepstral Coefficients* (**MFCC**), and *Relative Spectral Coefficients* (**RASTA-PLP**).

4.4.1 Weighted Cepstral Coefficients

The **LPC**-based cepstral coefficients set, c_m, contains the coefficients of the Fourier Transform representation of the log magnitude spectrum, which can be derived from the **LPC** coefficients set. This is a more robust and reliable feature set than the basic **LPC** coefficients with reflection parameters [34]. Based on the pre-defined **LPC** coefficients set in Equation 4.11, c_m. can be presented as

$$c_0 = \ln (\text{Gain})^2$$

$$c_{m^*} = \begin{cases} a_{m^*} + \sum_{k=1}^{m^*-1} \left(\frac{k}{m^*} \right) c_k \cdot a_{m^*-k} & \text{if } 1 \le m^* \le q \\ \sum_{k=1}^{m^*-1} \left(\frac{k}{m^*} \right) c_k \cdot a_{m^*-k} & \text{if } m^* > \text{q} \end{cases} \qquad (4.12)$$

where $(\text{Gain})^2$ denotes the square of gain computation in the LPC analysis (refer to Equation 4.4). Q is set to be the order of the cepstral coefficients, with the condition $Q \approx \frac{3}{2} q$.

In order to minimize the sensibility of the low-order cepstral coefficients to overall spectral slope and the sensibility of the high-order cepstral coefficients to noise, a set of appropriate Weighted Cepstral Coefficients will be calculated by introducing a Tapered Window function.

$$\hat{c}_{m^*} = \hat{w}_{m^*} \cdot c_{m^*} \qquad 1 \le m^* \le Q \qquad (4.13)$$

where

$$\hat{w}_{m^*} = \left[1 + \frac{Q}{2} \sin \left(\frac{\pi m^*}{Q} \right) \right] \qquad 1 \le m^* \le Q$$

4.4.2 Mel-frequency Cepstral Coefficients

The Mel-scale, the result of a numeric approximation developed to describe psychoacoustical experiments and presents a great number of approximations, was first suggested by Douglas O'Shaughnessy [43]. The psychoacoustic perceptual scale, "Mel-scale", is commonly used in the analysis of speech signals for speech recognition to establish a nonlinear spectral characterization for raw input of speech [7]. The relation between the linear scale and the nonlinear Mel-scale is given by

$$Mel = 2595 \log_{10}\left[\left(\frac{Freq}{700}\right)+1\right] \tag{4.14}$$

where *Freq* is the speech frequency in Hz and *Mel* is the Mel-scale value. Equation 4.14 shows that the Mel-scale is nearly linear for frequencies close and up to 1000 Hz and assumes a logarithmic behavior for frequencies above 1000 Hz. For frequencies below 1000 Hz, the value of Mel-scale is higher than the value in Hz. For frequency above 1000 Hz, the value of Mel-scale is lower than the value in Hz. The **LPC**-based **MFCC** can be obtained by subsituting the Mel-scale in Equation 4.14 into the cepstral analysis process (refer to subsection 4.3.4) of the **LPC** Analysis. Later on, a set of the coefficients is generated after the **LPC** to **CEP** conversion.

4.4.3 Relative Spectral Coefficients

The development of the Relative Spectral Coefficients (**RASTA-PLP**) [14] makes the conventional Perceptual Linear Predictive (**PLP**) technique (one of short-term spectrum techniques) more robust to linear spectral distortions. Less sensitivity to the slowly

changing or steady-state factors in speech is achieved by replacing a conventional critical-band short-term spectrum with a spectral estimate, in which each frequency channel is band-pass filtered by a filter with a sharp spectral zero at the zero frequency.

The key idea of **RASTA-PLP** modeling [15] is to suppress constant factors in each spectral component of the short-term auditory-like spectrum prior to the estimation of the all-pole model. Steps for each pre-defined speech frame include:

- Compute the *Power Spectrum* from the original windowed speech input.
- Compute the *Critical-band Power (Auditory) Spectrum* and take its logarithm.
- Transform *Spectral Amplitude* through a compressing static nonlinear transformation.
- Filter the time trajectory of each *Transformed Spectral Component*.
- Transform the *Filtered Speech Representation* through expanding static nonlinear transformation.
- In conventional **PLP**, add the *Equal Loudness Curve* and multiply by 0.33 to simulate the *Power Law* of hearing.
- Take the inverse logarithm (exponential function) of this *Relative Log Spectrum*, yielding a *Relative Auditory Spectrum*.
- Compute an *All-Pole Model* of the resulting spectrum, following the conventional **PLP** technique.

From the above steps, the low cut-off frequency of the filter determines the fastest spectral change of the log spectrum which is ignored in the output, while the high cut-off frequency determines the fastest spectral change which is preserved. The high-pass

portion of the equivalent band-pass filter is expected to alleviate the effect of convolutional noise introduced in the channel. The low-pass filtering helps to smooth some of the fast frame-to-frame spectral changes present in the short-term spectral estimate due to analysis artifacts.

4.5 Conclusion

In general, the **LPC** analysis is good for formant estimation. The all-pole-prediction filter models the vocal tract and the angular position of the poles of the filter gives the formant frequencies. However, the use of this analysis has a very serious drawback, which is the good prediction filter order is dependent on the formants of the speech to be estimated. The analysis is unable to accurately track the formant trajectories with a unique order of prediction filter.

The Mel-scale results from acoustic perception experiments and establishes a nonlinear spectral characterization for the speech signal. The **LPC**-based **MFCC** modeling explores the concept of "centralization" of the frequency spectrum. Low frequencies are mapped to higher frequencies and high frequencies are mapped to lower frequencies. This modeling is capable of extracting the formants with a fixed order prediction filter [7], and therefore, the use of **LPC**-based **MFCC** assures a significant improvement in recogniton phase within speech.

The **RASTA-PLP** modeling, which is based on the filtering of time trajectories of outputs from critical-band filters, has been introduced for estimating a robust time-

varying spectrum. It is said to show an order-of-magnitude improvement in error rate over some conventional spectral estimations, such as **LPC** analysis or the conventional **PLP** [15]. Firstly, the **RASTA-PLP** modeling can be used for the enhancement of noisy speech by applying an overlap-add analysis re-synthesis method to the cubic root of the power spectrum of noisy speech. Secondly, the modeling increases the dependence of the data on its previous context and works well in tasks with whole word models as presented in our **ASR** system in next chapter. Finally, the modeling in the logarithmic (cepstral) domain can handle both additive and convolutional noise reasonably well. This improvement is consistent across different speech databases and different recognition approaches, especially for large vocabulary continuous **ASR** system.

Chapter 5

Speech Recognition Mechanisms

5.1 Introduction

Speech recognition technology was launched nearly half a century ago and has been adopted in today's commercial business, industry, and customer areas in different applications and systems. The goal of the technology has always been to allow speakers to obtain information and perform transactions through machines simply by speaking naturally. Although it is not yet a reality for free-form conversation, significant progress has been made in many real-world **ASR** systems presently, such as Dragon NaturallySpeaking System [27], MIT SUMMIT System [55], OfficeTALK [21], SPHINX System [26], and IBM ViaVoice System [17].

Hong Kong is a multicultural society. Many residents will find it much more comfortable to use their mother tongues to communicate with others from the same country or to import any important information in their daily life. When considering the big differences of the characteristics and the formations of the three common spoken languages (*Cantonese, Mandarin, and English*) in Hong Kong, designing an integrated **ASR** system to work on is required. In this chapter, the modified class-dependent discretization

algorithm was introduced to classify our word-based speech continuous data set for model building and further computation. The three speech recognition mechanisms include our proposed *Improved Naïve Bayesian Classification* (**INBC**) [29], *Hidden Markov Modeling* (**HMM**) with *Viterbi* algorithm [43], and *Multi-Layers Backpropargation Modeling* (**ML-BkProp**) [3].

5.2 Class-Dependent Discretization

Most empirical learning systems are given a set of pre-classified instances. Each is described by a vector of attribute values and a mapping from the attribute values to classes. The attributes can be grouped into continuous attributes, whose values are numeric, and discrete attributes with unordered nominal values. The class-dependent discretization in this chapter refers to the discretization of all word-based speech of one of the three Hong Kong dialects' sets after feature extraction. Unlike most traditional discretization procedures, a class-dependent discretizer can automatically determine the most preferred number and width of intervals of continuous data, and significantly improve the classification performance of many existing learning algorithms [33].

Suppose we have a set of Z training cases and each of these cases has been pre-classified into one of p classes (**a word-based speech data**), where each class represents a label $C_k, k = 1,\ldots,p$. The whole set of data is described by n distinct attributes (**values of the coefficients set**), $A_1,\ldots,A_n$. For any attribute A_i, there is a domain of attribute values defined as, $V_{A_i,j}\{j = 1,\ldots,Z\}$ and these values can be either numeric, symbolic, or both.

5.2.1 The Discretization Criterion

Based on the concept of Class-Attribute dependence, the discretization criterion seeks to maximize the dependency relationship between the target class and a continuous-valued attribute and minimize the amount of information loss due to discretization [4].

Let a boundary set of an attribute be $B_i = \{e_0, e_1, \ldots, e_{L_i}\}$ with a set of ordered endpoints, where e_0 denotes the lower boundary value and e_{L_i} denotes the upper boundary value of the observed attribute, A_i. The ordered boundary points have $e_{\beta-1} < e_\beta$ with $\beta = 1,2,\ldots,L_i$, where L_i represents the total number of intervals of the continuous attribute values of the observed attribute, A_i. Q_i is a 2-dimensional Quanta Matrix of the observed attribute A_i as shown in Table 5.1. The matrix's element $q_{\alpha\beta}$ denotes the total number of the observed attribute values that fall within the partition $[e_{\beta-1}, e_\beta]$ and belong to class C_α from the set Z.

Class	Boundary $[e_0,e_1]$ $\cdots$ $[e_{\beta-1},e_\beta]$ $\cdots$ $[e_{L_i-1},e_{L_i}]$	$\sum$ Rows
C_1	q_{11} $\cdots$ $q_{1\beta}$ $\cdots$ q_{1L_i}	row$[q_1]$
$\vdots$	$\vdots$ $\vdots$ $\vdots$	$\vdots$
C_α	$q_{\alpha 1}$ $\cdots$ $q_{\alpha\beta}$ $\cdots$ $q_{\alpha L_i}$	row$[q_\alpha]$
$\vdots$	$\vdots$ $\vdots$ $\vdots$	$\vdots$
C_p	q_{p1} $\cdots$ $q_{p\beta}$ $\cdots$ q_{pL_i}	row$[q_p]$
$\sum$ Columns	col$[q_1]$ $\cdots$ col$[q_\beta]$ $\cdots$ col$[q_{L_i}]$	Z

Table 5.1 The 2-Dimensional Discretization Quanta Matrix

The theory of maximum mutual information discretization states that the absolute mutual information is greatest when the total number of intervals is the largest possible. To partition the original Z continuous-valued data of an attribute into L_i initial discrete intervals, the maximum allowable number of intervals would be considered as,

$$L_i \leq \frac{Z}{(\lambda \times p)} \tag{5.1}$$

where p is the total number of output classes and $\lambda = 3$ is a parameter with the suggested value for liberal estimation according to Wong and Chiu [53].

5.2.2 Boundary Improvement

The boundary improvement process attempts to adjust the initial boundary set between the lower boundary pair and the upper boundary pair starting from the first ordered interval $[e_0, e_1]$ and ending at the last interval $[e_{L_i-1}, e_{L_i}]$. Within the boundary set, each interval can be perturbed either boundary up or boundary down in order to maximize the value of the Class-Attribute Mutual Information (**CAMI**) [29,33].

Let's consider the quanta matrix as shown in Table 5.1, since the partitioned attribute is treated as an ordered discrete random variable, the estimated joint probability and marginal probabilities can be easily computed as follows,

$$\begin{aligned} p\left(C_k = C_\alpha, V_{A_i,j} \in [e_{\beta-1}, e_\beta]\right) &= \frac{q_{\alpha\beta}}{Z} = P_{\text{row}[\alpha],\text{col}[\beta]} \\ p(C_k = C_\alpha) &= \frac{\text{row}[q_\alpha]}{Z} = P_{\text{row}[\alpha]} \\ p\left(V_{A_i,j} \in [e_{\beta-1}, e_\beta]\right) &= \frac{\text{col}[q_\beta]}{Z} = P_{\text{col}[\beta]} \end{aligned} \tag{5.2}$$

The **CAMI** between the classes C_k and the observed attribute interval boundaries of A_i with its associated Quanta Matrix set Q_i is,

$$\text{CAMI}(C_{k\in[1,p]}) = \sum_{C_k} P_{\text{row}[\alpha],\text{col}[\beta]} \log\left(\frac{P_{\text{row}[\alpha],\text{col}[\beta]}}{P_{\text{row}[\alpha]} \times P_{\text{col}[\beta]}}\right) \geq 0 \qquad (5.3)$$

Both the boundary set and the relevant quanta matrix must be modified after each interval adjustment. In order to obtain an optimal estimation of global interdependence, the process is repeated until no improvement is found [29,33]. The pseudo-code of boundary improvement method is shown in Figure 5.1.

```
Boundary Improvement Algorithm
If the observed attribute values V_{A_i,j} are continuous Then
    Sort the values and set an Initial Boundary (using Eqn (5.1) for L_i setting)
    Set the Quanta Matrix Q_i as shown in Table 5.1
    Calculate CAMI(C_{k∈[1,p]}) from Eqn (5.3)
    For (boundary interval = 1 to L_i) do
    Begin
        Left Adjustment:
            Shift the interval one data point to left
            Update the Quanta Matrix Q_i* and calculate new CAMI*(C_{k∈[1,p]})
            If (CAMI*(C_{k∈[1,p]}) > CAMI(C_{k∈[1,p]})) Then
                Set Q_i = Q_i* and CAMI(C_{k∈[1,p]}) = CAMI*(C_{k∈[1,p]})
                Loop back to Left Adjustment

        Right Adjustment:
            Shift the interval one data point to right
            Update the Quanta Matrix Q_i* and calculate new CAMI*(C_{k∈[1,p]})
            If (CAMI*(C_{k∈[1,p]}) > CAMI(C_{k∈[1,p]})) Then
                Set Q_i = Q_i* and CAMI(C_{k∈[1,p]}) = CAMI*(C_{k∈[1,p]})
                Loop back to Right Adjustment
    End
```

Figure 5.1 Pseudo-code of Boundary Improvement Algorithm

5.2.3 Interval Reduction

After improvement of the boundary set is completed, elements of some adjacent intervals of the modified quanta matrix may have a very similar frequency distribution with respect to the observed class. Hence one can conclude that the frequency distribution among them is insignificantly interdependent and the two intervals should be merged [53].

The Class-Attribute Interdependence Redundancy (**CAIR**) is equivalent to **CAMI** (from Equation 5.3) divided by Joint Entropy (**JE**), that is **CAIR = CAMI/JE** and is bounded by zero,

$$\mathrm{JE}(C_{k\in[1,p]}) = -\sum_{C_k} P_{\mathrm{row}[\alpha],\mathrm{col}[\beta]} \log\left(P_{\mathrm{row}[\alpha],\mathrm{col}[\beta]} \right) \tag{5.4}$$

Without considering the whole boundary set of the attribute A_i, the partial **CAMI** and the partial **JE** only involve any two adjacent boundaries' pairs within the attribute, that is $[e_{\beta-1}, e_\beta]$ and $[e_\beta, e_{\beta+1}]$. Both of these partial values can be calculated from the frequency elements and subtotal training cases of any two neighboring boundaries in order to check the statistical significance of their frequency distribution. In order to achieve an optimal number and an appropriate width of intervals of the observed attributes, the interval reduction algorithm is applied repeatedly until all pairs of adjacent intervals within the boundary set pass the statistical interdependence test as below,

$$\text{Partial CAIR} \geq \frac{\chi^2_{(p-1)(L_i-1)}}{2m_1 \times \text{Partial JE}} \tag{5.5}$$

where $\chi^2_{(p-1)(L_i-1)}$ represents a Chi-square distribution with degrees of freedom $(p-1)\times(L_i-1)$; p denotes the total number of output classes and L_i denotes the total number of the current intervals of the observed attribute. The pseudo-code of interval reduction method is shown in Figure 5.2.

```
Interval Reduction Algorithm
For each pair of adjacent intervals (β, β+1) do
Begin
    Calculate Partial CAIR = Partial CAMI ÷ Partial JE based on Eqn (5.3) and Eqn (5.4)
    If the Statistical Interdependence Test from Eqn (5.5) fails Then
        Merge the two intervals β and β+1
        Update the new boundary set
End
```

Figure 5.2 Pseudo-code of Interval Reduction Algorithm

5.3 Improved Naïve Bayesian Classification (INBC)

Naïve Bayesian Classifier (**NBC**) or Naïve Bayes (**NB**) is a simplified form of Bayes' rule that assumes independence of the observations. Some research results [1,2,23] demonstrated that **NBC** has competitive performance in comparison with other learning algorithms if the normal distribution assumption holds.

5.3.1 Formulation of NB

Refer to the general setting in Section 5.2, let A_i be a set of attributes and each of these attributes have a certain number of possible values $V_{A_i,j}$ with $\{i=1,\ldots,n; j=1,\ldots,m\}$. In general, *n* is the number of the sets of attributes and *m* is the total number of instances in the set, which contains both the training cases (m_1) and test cases (m_2). C_k is a set of target classes with $\{k=1,\ldots,p\}$.

The **NB** formula is used to estimate the posterior probability that an instance belongs to a particular class given the observed attribute values for the instance. The class with the highest estimated posterior probability [2] is finally selected. Assume $V_{\{A_i\}} = V_{A_i,j'}$ with $\{i = 1,\ldots,n; j' = 1,\ldots,m_1\}$ are values in attributes set, A_i, the probability of **NB** is formulated as

$$\begin{aligned} P(C_k \mid V_{\{A_i\}}) &= P(C_k \mid V_{\{A_1\}} V_{\{A_2\}} V_{\{A_3\}} \cdots V_{\{A_n\}}) \\ &= P(C_k)\frac{P(C_k \mid V_{\{A_1\}})}{P(C_k)}\frac{P(C_k \mid V_{\{A_1\}} V_{\{A_2\}})}{P(C_k \mid V_{\{A_1\}})}\frac{P(C_k \mid V_{\{A_1\}} V_{\{A_2\}} V_{\{A_3\}})}{P(C_k \mid V_{\{A_1\}} V_{\{A_2\}})}\cdots \\ &= P(C_k)\frac{P(C_k \mid V_{\{A_1\}})}{P(C_k)}\frac{P(C_k \mid V_{\{A_2\}})}{P(C_k)}\frac{P(C_k \mid V_{\{A_3\}})}{P(C_k)}\cdots \\ &= P(C_k)\prod_{i=1,j'=1}^{n,m_1}\frac{P(C_k \mid V_{A_i,j'})}{P(C_k)} \end{aligned} \tag{5.6}$$

Equation 5.6 shows that the posterior probability can be computed by multiplying the prior probability of C_k by a set of factors $P(C_k \mid V_{A_i,j'})/P(C_k)$ with $\{i = 1,\ldots,n; j' = 1,\ldots,m_1\}$, only if the assumption of independence of the observations holds.

5.3.2 Estimation of NB

In the defined training set, let $N(V_{A_i,j})$ be the count of the same value $V_{A_i,j}$ of attribute A_i being observed, and similarly, let $N(C_k V_{A_i,j})$ be the count of value $V_{A_i,j}$ of attribute A_i being observed with respect to class C_k.

Equation 5.6 can be estimated by applying the Relative Frequency (**RF**) approximation,

$$P(C_k \mid V_{A_i,j}) = \frac{N(C_k V_{A_i,j})}{N(V_{A_i,j})} \tag{5.7}$$

However such an approximation seems not to be reliable when $N(V_{A_i,j})$ tends to zero. Thus, the Initial Probability Density (**IPD**) approximation was introduced to solve this problem. It involved two parameters a and b for estimating conditional probabilities [2], such as

$$P(C_k \mid V_{A_i,j}) = \frac{N(C_k V_{A_i,j}) + a}{N(V_{A_i,j}) + \xi} \tag{5.8}$$

where $\xi (= a + b)$ denotes the amount of noise in the domain and it can be used to estimate the two parameters, $a = P(C_k) \times \xi$ and $b = \xi - a$.

5.3.2 INBC Algorithm

From Trivedi [52], we note that if the size of learning field (sample) is increased within the same data set (population), a better classification rate can be achieved under the definitions of the consistency and unbiasedness of an estimate. Our proposed **INBC** algorithm [28,29,31] simply uses the past tested data to update the learning field for producing the optimal performance. For instance, we have a fundamental training data set (the first phase of data), which contains all word-based speech data of one of the three common Hong Kong dialects' sets. We then divide the second phase of data (some previous tested speech data) into a fixed number of sets (StepSize) to enhance the original speech data set gradually. The classification rate of the third phase of data (current speech

data for recognition) can be computed with the new and updated training model. The pseudo-code of our proposed **INBC** algorithm is shown in Figure 5.3.

The Improved Naive Bayes Classification Algorithm

Calculate $N(C_k V_{A_i,j'})$, $N(V_{A_i,j'})$ and $P(C_k)$ from the first phase of data
Prompt for the *StepSize*, a fixed number of data records, for evaluating the performance
Initialize *start-pos*=0; *end-pos*=0; *index*=1
Do
  Set *start-pos* = *end-pos* + 1
  Set *end-pos* = *index* × *StepSize*
  Prompt for the ***NB*** Approx. (***RF*** from Eqn (5.7) or ***IPD*** from Eqn (5.8))
  For each $k = 1$ to p **do**
  Begin

$$\text{Calculate } P(C_k \mid V_{A_i,j''}) = P(C_k)\prod_{i=1}^{n}\prod_{j''=\text{start-pos}}^{\text{end-pos}}\frac{NB \text{ Approx.}}{P(C_k)}$$

    Find the highest conditional probability of $A_{i\in[1,n]}$
  End
  Update $N(C_k V_{A_i,j'})$, $N(V_{A_i,j'})$ and $P(C_k)$ with the second phase of data
  Set *index* = *index* + 1
While *end-pos* < m_2
For the rest of each *start-pos* ≤ j'' ≤ m_2

$$\text{Calculate } P(C_k \mid V_{A_i,j''}) = P(C_k)\prod_{i=1}^{n}\prod_{j''=\text{start-pos}}^{m_2}\frac{NB \text{ Approx.}}{P(C_k)} \text{ for all } k$$

  Find the highest conditional probability of $A_{i\in[1,n]}$
Evaluate the classification rate of all k and the overall performance

Figure 5.3 Pseudo-code of Improve Naive Bayes Classification Algorithm

5.4 Hidden Markov Modeling (HMM)

In general, **HMM** are popular and dominant models used in speech recognition, they can be used for a study of observed symbols arranged in a discrete-time series with the observable state sequence [37]. In a **HMM**, the output for each state corresponds to an output probability distribution instead of a deterministic event. Such output probabilities impose a veil (not observable) between the state sequence and the observer of the time

sequence. Thus, signal modeling based on **HMM** can be considered as a technique that extends conventional stationary spectral analysis principles to the analysis of time-varying signals [45]. By using **HMM**, the speech signal variability in parameter space and time can be modeled. The **HMM** learning procedure is achieved by presenting speech data to HMM and automatically improving the models by data. In fact, the more data that is presented to the model, the higher the recognition accuracy achieved will be.

5.4.1 Definition of HMM

HMM can be represented by using the compact notation $\lambda = (A, B, \pi)$. Specification of an **HMM** involves the choice of the number of states, N, the number of discrete symbols, L, and specification of the three probability densities with matrix form A, B, and π. Transitions must start from one of initial states in a set, S_I, and end at one of final states in a set, S_F. Total numbers of both initial states and final states are denoted by N_I and N_F, respectively.

T = length of the observation sequence, $O_1, O_2, \ldots, O_T$
N = number of states in the model
L = number of observation symbols
S = {s} be a set of states, where state i at time t is denoted by $s_t = i$
$v = \{v_1, v_2, \ldots, v_L\}$ be a discrete set of possible symbol observations
$A = \{a_{ij} \mid a_{ij} = P(s_{t+1} = j \mid s_t = i)\}$ be the state transition probability distribution, where a_{ij} denotes the transition probability from state i to state j
$B = \{b_j(O_t) \mid b_j(O_t) = P(O_t \mid s_t = j)\}$, for each state, there is a corresponding output probability, and all of these output probabilities represent random variables or stochastic processes to be modeled. In the discrete HMM, it refers to the probability of generating some discrete symbols v_k in state j, which can be denoted simply by $b_j(k)$
$\pi = \{\pi_i \mid \pi_i = P(s_1 = i)\}$ be the initial state distribution

Figure 5.4 The Model Notation for an HMM

The following are three fundamental questions concerned with **HMM**, it is shown that those problems are closely related under the same probabilistic framework.

- Problem on Evaluation

  Given a sequence of observation vectors $O = O_1, O_2, \ldots, O_T$ and a **HMM** $\lambda = (A, B, \pi)$, how to choose the model with best matches the observations for the purpose of recognition (how to compute $P(O, S \mid \lambda)$)?

- Problem on Decoding

  Given a sequence of observation vectors $O = O_1, O_2, \ldots, O_T$ and a **HMM** $\lambda = (A, B, \pi)$, what is the most likely state sequence $\boldsymbol{S} = s_1, s_2, \ldots, s_T$ according to some optimality criterion?

- Problem on Optimization

  Given a sequence of observation vectors $O = O_1, O_2, \ldots, O_T$ and a **HMM** $\lambda = (A, B, \pi)$, how to adjust and optimize the model parameters to maximize $P(O, S \mid \lambda)$?

5.4.2 Viterbi Algorithm

HMM works by comparing the probability for the observed speech to the probabilities for other speech in the predefined dictionary by use of the Viterbi Algorithm. The algorithm is used to find the single best sequence, where s_t is in the best path with highest probability for $P(O, S \mid \lambda)$ maximizing. It is very similar to the Dynamic Time Warping (**DTW**) algorithm [44], where the transition information is neglected and speech data such as those **LPC**-based coefficients (as shown in Chapter 4) are usually stored without further parameterization.

The Viterbi algorithm can also be used in score evaluation since it finds the maximum of $P(O,S \mid \lambda)$ over all of S. The algorithm can operate in the logarithm domain, using only additions for the sake of efficiency, to find the single best state sequence, s_t. Computation steps of the Viterbi algorithm is shown in Figure 5.5.

Step 1: Initialization
For all states i,

$$x_1(i) = \pi_i b_i(O_i)$$

$$y_1(i) = 0$$

Step 2: Recursion
From time t = 2 to T for all states j

$$x_t(i) = \max_i [x_{t-1}(i)a_{ij}]b_j(O_t)$$

$$y_t(i) = \arg\max_i [x_{t-1}(i)a_{ij}]$$

Step 3: Termination
The optimized results are

$$P^* = \max_s [x_T(S)]$$

$$s_T^* = \arg\max_s [x_T(S)]$$

Step 4: Path (state sequence) backtracking
From time T-1 to 1,

$$s_t^* = y_{t+1}(s_{t+1}^*)$$

Figure 5.5 Viterbi Algorithm

5.5 Multi-Layers Backpropagation Modeling

Fundamentally, Neural Networks are multi-layer networks of nodes (perceptrons), these nodes are interconnected by links and each link has an associated weight [10]. The three kinds of nodes on the corresponding layers are

- **Input Nodes**, for introducing information from the environment to the network model
- **Output Nodes**, for showing the final outcomes

- **Hidden Nodes**, for storing all "hidden" information from the environment.

The Backpropagation model [3] is one of the general-purpose multi-layer neural networks, which has been used for many current speech recognition problems. The model is applied to feed-forward networks and uses the Delta rule, which starts with the calculated difference between the actual outputs and the desired outputs. Using this error, connection weights are increased in proportion to the error times with a scaling factor for global accuracy. Now training inputs are applied to the input layer of the network, and desired outputs are compared at the output layer. During the learning process, a forward sweep is made through the network, and the output of each element is computed layer by layer. The difference between the output of the final layer and the desired output is back-propagated to the previous layers, usually modified by the derivative of the transfer function, and the connection weights are normally adjusted using the Delta Rule [46]. Thus, the process proceeds for the previous layers until the input layer is reached.

5.5.1 Notation of Multi-Layers Backpropagation (ML-BkProp)

ML-BkProp is simply a gradient descent method to minimize the total squared error of the output computed by the network model with multiple hidden-to-output layers. Our proposed **ML-BkProp** network is extended to two-hidden-layered network model with adjustable weights connecting the hidden nodes and the output. Such a network involves steps to flow the speech information up from the input nodes (total number of speech observations) through successive layers of hidden nodes (a set of features), and to create the final response at the layer of the output nodes (the overall performance of the speech data). Figure 5.6 shows the notation of our purposed **ML-BkProp** network.

x = input training vector, $\{x_1,\ldots,x_i,\ldots,x_n\}$

t = output target vector, $\{t_1,\ldots,t_k,\ldots,t_m\}$

θ_k = portion of error correction weight adjustment for $w_{k'k}$ that is due to an error at output unit Y_k and such information is propagated back to the hidden units (in second layer)

$\theta_{k'}$ = portion of error correction weight adjustment for $w_{jk'}$ that is due to an error at second layer hidden unit $Z_{k'}$ and such information is propagated back to the hidden units (in first layer)

θ_j = portion of error correction weight adjustment for v_{ij} that is due to an error at first layer hidden unit Z_j and such information is propagated back to the input units of the net

α = learning rate

v_{0j} = bias on the first layer hidden unit j

$w_{0k'}$ = bias on the second layer hidden unit k'

w_{0k} = bias on output unit k

X_i = input unit i

Z_j = hidden unit j

$Z_{k'}$ = hidden unit k'

Y_k = output unit k

Figure 5.6 Notation for Multi-Layers Backpropagation Model

5.6 Performance Evaluation

Other than speech segmentation, both preprocessing and recognition phases are also playing important roles in **ASR** systems. In order to investigate and compare the performance of our integrated **ASR** system for the two phases, different types of parametric representations and recognition models from both Chapter 4 and Chapter 5 are being considered here. The experiments are based on the same experimental environment as described in Section 3.4. From the segmentation results in Chapter 3, our proposed **LFS** method can successfully extract 427 Cantonese digits, 433 Mandarin digits, and 400 English digits out from the male-speaker set, and 422, 417, and 428 of Cantonese, Mandarin and English digits were extracted out respectively from the female-speaker set.

The Zero Crossing method can extract 390, 400, and 378 digits of the three dialects from the male-speaker set, whereas 390, 385, and 419 related digits were extracted out from the female-speaker set.

For the system training and testing purposes, 3 sets of the six codebook files were constructed for each dialect and for each gender after applying **LFS** method for repeated-words segmentation. Each set of the six codebook files was obtained from one of the three parametric representations (the order of each type of coefficients is set as 12) as described in Chapter 4. Each codebook file contains 300 training voice vectors and 100 testing voice for all digits between 0 to 9 of a particular gender and dialect. Each voice vector has 36 data values (12 are *Mean*, 12 are *Standard Deviation*, and 12 are *Standard Error*) and 1 label to describe a digit. In our experiments, our proposed **INBC** will enhance the training model up to 350 training voice vectors only. Sets of our training **HMM** models are word-based models for each digit of a particular gender and dialect. Each of such **HMM** word model contains 8 states and 32 data values per each state, with a left-to-right topology. Our training **ML-BkProp** model for each dialect of a particular gender with 36-12-12-10 neurons' structure for the input layer, the first hidden layer, the second hidden layer, and the output layer respectively. The performance of our integrated **ASR** system for the three dialects is shown in both Table 5.2 and Table 5.3.

WCEP (in %)	*Cantonese*			*Mandarin*			*English*		
	M	**F**	**All**	**M**	**F**	**All**	**M**	**F**	**All**
INBC	74.1	73.6	73.85	71.4	68.7	70.05	66	72	69
HMM	72.4	71.9	72.15	76.6	73.7	75.15	76.9	83.7	80.4
ML-BkProp	74.4	73.8	74.1	74.3	71.5	72.9	75.5	82.2	78.85
Overall	73.6	73.1	73.4	74.1	71.3	72.7	72.8	79.3	76.1
MFCC (in %)	*Cantonese*			*Mandarin*			*English*		
	M	**F**	**All**	**M**	**F**	**All**	**M**	**F**	**All**
INBC	76	75.4	75.7	74.2	71.4	72.8	66.7	72.7	69.7
HMM	75.4	74.8	75.1	77.3	74.4	75.85	79	86.1	82.55
ML-BkProp	74.1	73.5	73.8	76.3	73.4	74.85	78.1	85.1	81.6
Overall	75.2	74.6	74.9	75.9	73.1	74.5	74.6	81.3	78
RASTA-PLP (in %)	*Cantonese*			*Mandarin*			*English*		
	M	**F**	**All**	**M**	**F**	**All**	**M**	**F**	**All**
INBC	71.4	70.8	71.1	70.2	67.5	68.85	64	69.7	66.85
HMM	70.8	70.3	70.55	73.9	71.1	72.5	75	81.7	78.35
ML-BkProp	70.3	69.8	70.05	73.1	70.4	71.75	74.2	80.0	77.5
Overall	70.8	70.3	70.6	72.4	69.7	71	71.1	77.1	74.2

Table 5.2 Overall Performance for Speech Preprocessing

Table 5.2 shows the overall performance of the **ASR** system with different speech preprocessing methodologies for the three languages falling into the 70% to 80% range. The quality of both Cantonese and Mandarin speech, taken from the male speakers, is slightly better than for the female speakers. There is only a difference of about 3% in the accuracy rates for the two sets of speakers. However, a difference of more than 6% in the accuracy rates of English speech for the two genders is found.

From Table 5.2, the best performance for recognizing Cantonese can be achieved by applying **MFCC** into **INBC**, whereas the best performance for recognizing Mandarin and English can be achieved by applying **MFCC** into **HMM** with *Viterbi* algorithm. Such results reveal that the well known **HMM** may not be suitable for recognizing all the three sets of languages. It is also indicated that an integrated **ASR** system (the composition of different algorithms from segmentation, preprocessing, and recognition phases) is needed for different kinds of spoken-languages recognition. Our proposed **INBC** is shown to be good at Cantonese and Mandarin recognition rather than English by applying different parametric representations. Less than 70% accuracy of the overall performance of **INBC** for English recognition becomes the worse case in our **ASR** system.

INBC (in %)	*Cantonese*			*Mandarin*			*English*		
	M	**F**	**All**	**M**	**F**	**All**	**M**	**F**	**All**
WCEP	74.1	73.6	73.85	71.4	68.7	70.05	66	72	69
MFCC	76	75.4	75.7	74.2	71.4	72.8	66.7	72.7	69.7
RASTA-PLP	71.4	70.8	71.1	70.2	67.5	68.85	64	69.7	66.85
Overall	73.8	73.3	73.6	71.9	69.2	70.6	65.6	71.5	68.5
HMM (in %)	*Cantonese*			*Mandarin*			*English*		
	M	**F**	**All**	**M**	**F**	**All**	**M**	**F**	**All**
WCEP	72.4	71.9	72.15	76.6	73.7	75.15	76.9	83.7	80.4
MFCC	75.4	74.8	75.1	77.3	74.4	75.85	79	86.1	82.55
RASTA-PLP	70.8	70.3	70.55	73.9	71.1	72.5	75	81.7	78.35
Overall	72.9	72.3	72.6	75.9	73.1	74.5	77	83.8	80.4
ML-BkProp (in %)	*Cantonese*			*Mandarin*			*English*		
	M	**F**	**All**	**M**	**F**	**All**	**M**	**F**	**All**
WCEP	74.4	73.8	74.1	74.3	71.5	72.9	75.5	82.2	78.85
MFCC	74.1	73.5	73.8	76.3	73.4	74.85	78.1	85.1	81.6
RASTA-PLP	70.3	69.8	70.05	73.1	70.4	71.75	74.2	80	77.5
Overall	72.9	72.4	72.7	74.6	71.8	73.2	75.9	82.4	79.3

Table 5.3 Overall Performance for Speech Recognition

5.7 Conclusion

The task of an **ASR** system is to take the acoustic waveform (as input) produced by the speaker and to produce a sequence of linguistic words corresponding to the input utterance by machine. However, many potential problems exist in the system, such as the quality of voice, speaking speed and loudness, accents, speaking styles and languages used of individual speakers. A successful **ASR** system must take into account all of these problems.

In this chapter, we proposed the modified class-dependent discretization algorithm for the three speech recognition mechanisms, **INBC**, **HMM** and **ML-BkProp**. Unlike most traditional discretization procedures, the class-dependent discretizer can automatically determine the most preferred number and width of intervals of continuous data, and significantly improve the classification performance of many existing learning algorithms.

INBC [32] is one kind of statistical model that uses the past tested data to update the current learning field to produce the optimal performance. **HMM** [25] is a flexible general method for modeling many uncertainty factors in speech recognition associated with space and time series. **ML-BkProp** network [3] is one of the general-purpose multi-layer neural networks, it is simply a gradient descent method to minimize the total squared error of the output computed by the network model with the two extended hidden-to-output layers. Although it is shown that **HMM** has presented speech recognition with a solid theoretical basis and has resulted in significant advances in the overall performance, this mechanism may not be suitable for recognizing all three sets of

languages in this study. From the results summary tables in last section, it is shown that the best performance for recognizing Cantonese can be achieved by applying **MFCC** into **INBC**, whereas the best performance for recognizing Mandarin and English can be achieved by applying **MFCC** into **HMM** with *Viterbi* algorithm. Thus, we may conclude that an integrated **ASR** system (the composition of different algorithms from segmentation, preprocessing, and recognition phases) is needed for different kinds of spoken-languages recognition.

Chapter 6

A Zoological Fortune Telling System

6.1 Interactive Voice Response (IVR)

In the past, the human-machine interaction has been largely dependent on keyboard strokes or other mechanical means rather than speech. Nowadays, such spoken language understanding applications are available on automatic dictation (especially for Chinese and Japanese), database query, command and control, and computer-assisted instruction [9]. We may also use a computer telephony system regularly in our real life without realizing this technology, it allows us to interact with a computer through the telephone system.

In general, **IVR** system can provide smart support for business and organizations [5] of all sizes in the areas of

- Touch-tone Banking Service
- Hotel or Airline Reservation Systems
- Voice Recognition Systems for Directory Assistance
- Technical Support Questions or other Common Queries
- Information Hotlines

- Real Estate Systems for meeting the Buyer's Criteria
- Automatic Customer Service Systems for checking Customers Order Status

Most of today's businesses meet their strategic and operational goals through innovative electronic communications solutions. In order to strengthen their competitive edge, some of them are looking to cut costs with streamlined business processes, and the others to increase sales and enhance customer service. Some real applications [8] with **IVR** systems are listed below:

1. An **IVR** system was implemented within the *American Industrial Hygiene Association* that allows user to enter specimen analysis results by telephone. Such information are then logged into a databank and scored.

2. The *American Association of Airport Executives* uses an **IVR** system to provide consulting services.

3. The *World Bank Corporation* applies **IVR** system to handle the heavy call volume with a vast number of queries every day from academic and research communities, governments, international organizations and the business community.

4. The steel organization, *U.S. Steel*, offers an **IVR** fax-on-demand system to search for the requested material safety data sheets to send to customers, distributors, medical personnel, some emergency planning commissions, fire departments, and other parties.

5. The *Internal Revenue Service* of the U.S. government offers a cost-effective fax-on-demand solution for tax form requests to handle more than 45000 cases per day.

6.2 Zoological Fortune Telling Application

A zoological fortune telling system [20] is presented to implement our integrated **ASR** system for the three studied dialects (Cantonese, Mandarin, and English). It is believed that the development of an integrated **ASR** system can be applied for a Voice Response System, which can improve traditional human-computer interactions to support multimodal. In our **ASR** fortune telling application, the three essential phases of speech recognition are applied here, which include segmentation modeling (**Zero Crossing** and **LFS**), preprocessing modeling (**WCEP**, **MFCC**, and **RASTA-PLP**) and recognition mechanisms (**INBC**, **HMM**, and **ML-BkProp**).

According to the fortune telling application we studied here, each individual's birth date is associated with one of the twelve animals representing the person's personality as shown in Figure 6.1.

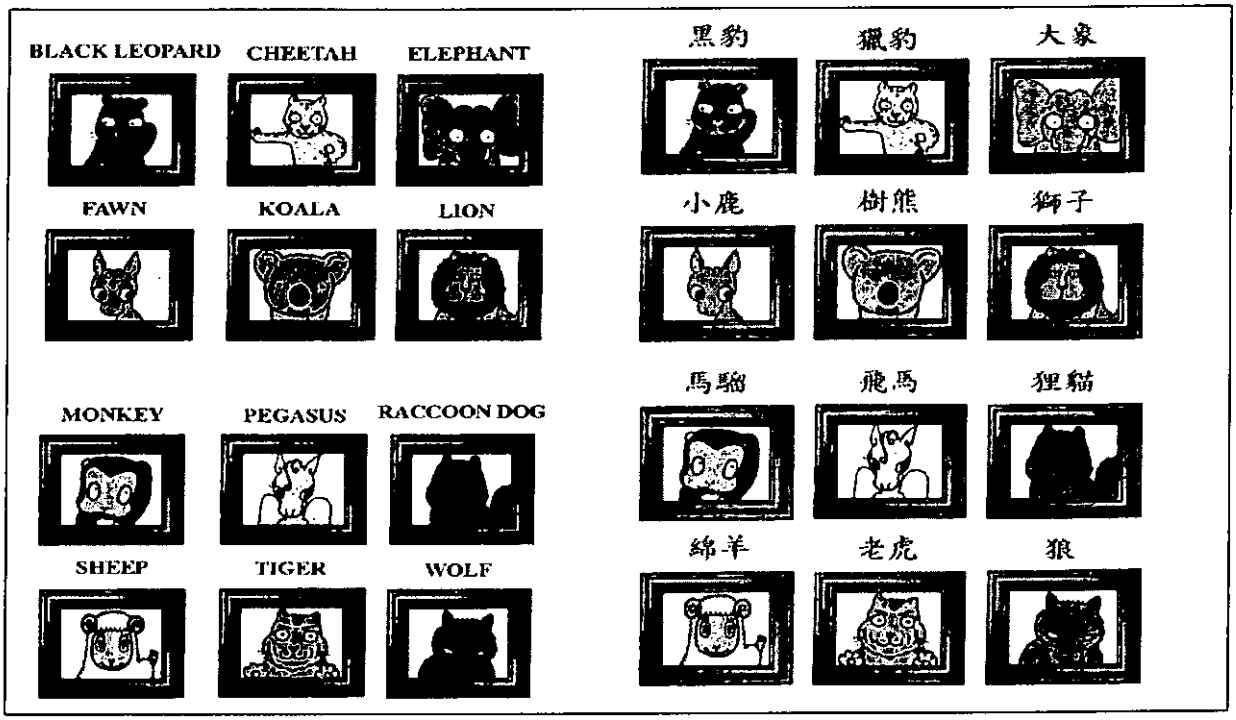


Figure 6.1 The Twelve Representative Animals for English (left) and for Chinese (right)

The basic calculation procedure of the zoological fortune telling system is

Step 1: Find the "number" which crosses between the individual's birth year and birth month as shown in Table 6.1.

	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
1945	6	37	5	36	6	37	7	38	9	39	10	40
1946	11	42	10	41	11	42	12	43	14	44	15	45
1947	16	47	15	46	16	47	17	48	19	49	20	50
1948	21	52	21	52	22	53	23	54	25	55	26	56
1949	27	58	26	57	27	58	28	59	30	0	31	7
1950	32	3	31	2	32	3	33	4	35	5	36	6
1951	37	8	36	7	37	8	38	9	40	10	41	11
1952	42	13	42	13	43	14	44	15	46	16	47	17
1953	48	19	47	18	48	19	49	20	51	21	52	22
1954	53	24	52	23	53	24	54	25	56	26	57	27
1955	58	29	57	28	58	29	59	30	1	31	2	32
1956	3	34	3	34	4	35	5	36	7	37	8	38
1957	9	40	8	39	9	40	10	41	12	42	13	43
1958	14	45	13	44	14	45	15	46	17	47	18	48
1959	19	50	18	49	19	50	20	51	22	52	23	53
1960	24	55	24	55	25	56	26	57	28	58	29	59
1961	30	1	29	0	30	1	31	2	33	3	34	4
1962	35	6	34	5	35	6	36	7	38	8	39	9
1963	40	11	39	10	40	11	41	12	43	13	44	14
1964	45	16	45	16	46	17	47	18	49	19	50	20
1965	51	22	50	21	51	22	52	23	54	24	55	25
1966	56	27	55	26	56	27	57	28	59	29	0	30
1967	1	32	0	31	1	32	2	33	4	34	5	35
1968	6	37	6	37	7	38	8	39	10	40	11	41
1969	12	43	11	42	12	43	13	44	15	45	16	46
1970	17	48	16	47	17	48	18	49	20	50	21	51
1971	22	53	21	52	22	53	23	54	25	55	26	56
1972	27	58	27	58	28	59	29	0	31	1	32	2
1973	33	4	32	3	33	4	34	5	36	6	37	7
1974	38	9	37	8	38	9	39	10	41	11	42	12
1975	43	14	42	13	43	14	44	15	46	16	47	17
1976	48	19	48	19	49	20	50	21	52	22	53	23
1977	54	25	53	24	54	25	55	26	57	27	58	28
1978	59	30	58	29	59	30	0	31	2	32	3	33
1979	4	35	3	34	4	35	5	36	7	37	8	38
1980	9	40	9	40	10	41	11	42	13	43	14	44
1981	15	46	14	45	15	46	16	47	18	48	19	49
1982	20	51	19	50	20	51	32	52	23	53	24	54
1983	25	56	24	55	25	56	26	57	28	58	29	59
1984	30	1	30	1	31	2	32	3	34	4	35	5
1985	36	7	35	6	36	7	37	8	39	9	40	10

Table 6.1 A Reference Table for People who were born between 1945 to 1985

Step 2: Add the value of the day of birth on the "number", and subtract 60 from the final result when it is over 60.

Step 3: Check the representative animal from the 12 possibilities in Table 6.2. Different numbers represent different personalities even for people who belong to the same animal class.

Black Leopard	5, 44, 50, 53, 56, 59
Cheetah	1, 7, 42, 48
Elephant	12, 18, 31, 37
Fawn	11, 17, 32, 38
Koala	4, 10, 16, 33, 39, 45
Lion	51, 52, 57, 58
Monkey	3, 9, 15, 34, 40, 46
Pegasus	21, 22, 27, 28
Raccoon Dog	2, 8, 41, 47
Sheep	14, 20, 23, 26, 29, 35
Tiger	6, 43, 49, 54, 55, 60
Wolf	13, 19, 24, 25, 30, 36

Table 6.2 Twelve Animals and their Representative Numbers

The zoological fortune telling system supports any one of the three dialects input (Cantonese, Mandarin and English). At the front page of the application as shown in Figure 6.2, the three buttons represent the three databases we can link with.

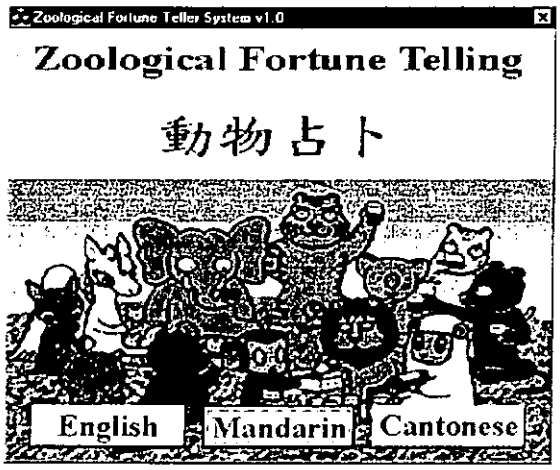


Figure 6.2 Front Page of the Application

A brief introduction to the system is displayed after we choose our favorite database to work on. In Figure 6.3, the two independent English and Chinese screen versions are shown separately.

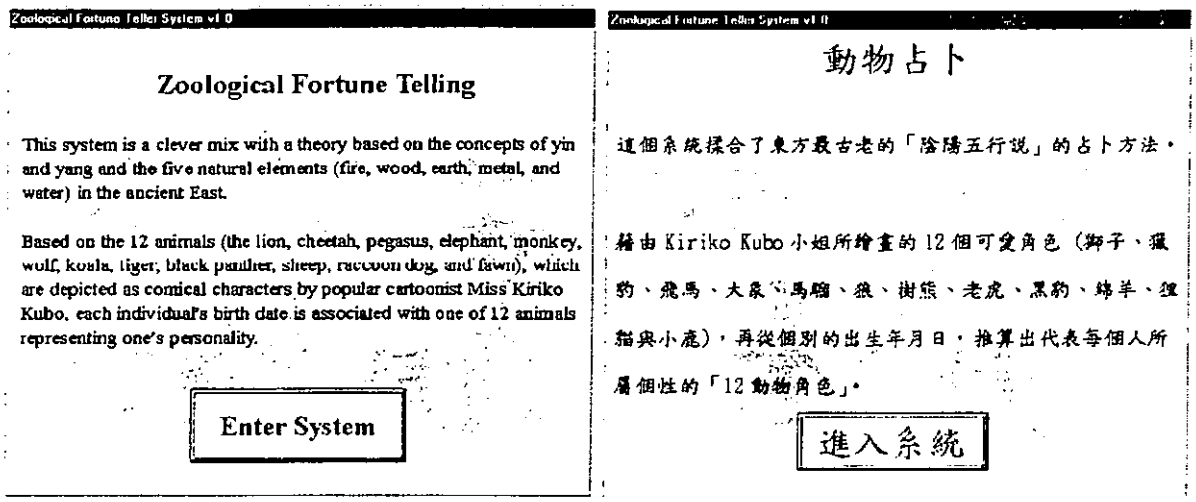


Figure 6.3 Introduction Page of the Application for English (left) and for Chinese (right)

The main page of the English application is shown in Figure 6.4. It contains a menu bar for setting our preferable background, some text boxes for inputting either "in speech form" or "in words form" and some buttons for performing some appropriate actions. By default, the system with **Zero Crossing**, **LPC-CEP**, and **INBC** models is set for evaluation.

Actions upon choosing items from the File menu (New, Analysis, Exit) are the same as pressing the right hand side buttons (Start/Reset, Analysis, Exit) of the page. The "**Start/Reset**" button clears all inputs in the page and starts with our default setting (the first item of each menu will be chosen). The "**Analysis**" button analyzes the whole set of our inputted information in the page and prepares to display the result page of the application. The "**Exit**" button exits from the Zoological Fortune Telling application. The main page of the application for Chinese is shown in Figure 6.5.

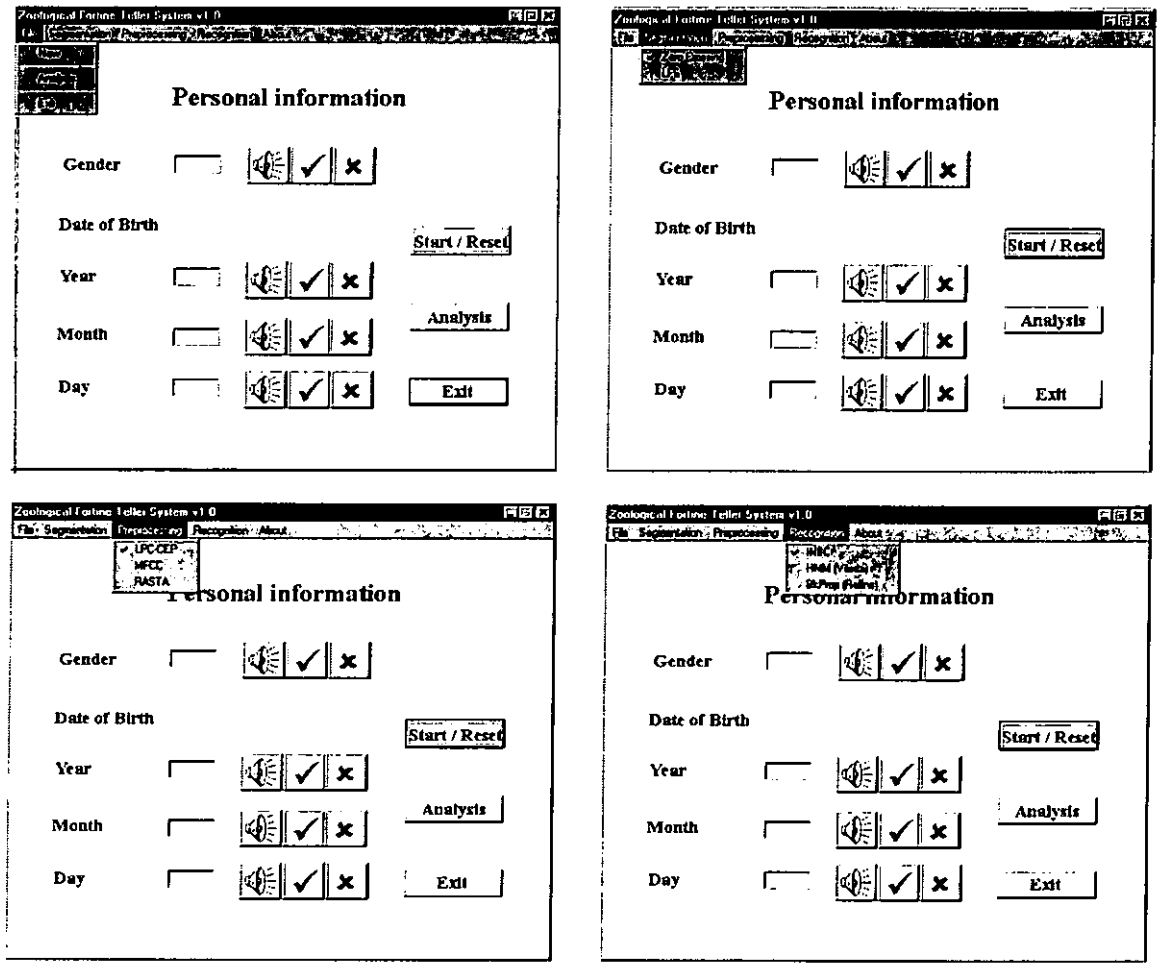


Figure 6.4 Main Page of the Application for English

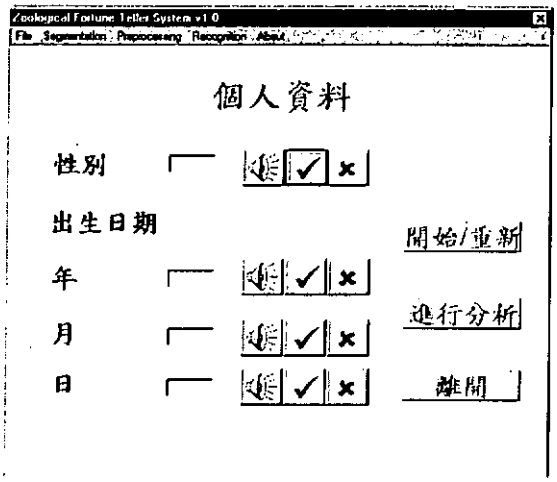


Figure 6.5 Main Page of the Application for Chinese

In both Figure 6.4 and Figure 6.5, there is a set of three buttons next to each information box of the main page. The "loud-speaker" button allows the user input "in speech form". The "tick" button confirms the current "the words format" input. The "cross" button erases the current "the words format" input.

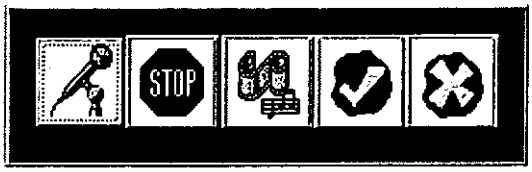


Figure 6.6 A Dialog Box for Speech Recording

When pressing the "loud-speaker" button, a dialog box for speech recognition as shown in Figure 6.6, is displayed. The first button starts the speech recording. The second button stops the speech recording. The third button plays the recorded voice. The fourth button starts to recognize the inputted speech. The fifth button cancels this voice input.

In order to give the whole view of the application, we input the author's birthday (1971-10-24) into main pages the application as shown in Figure 6.7.

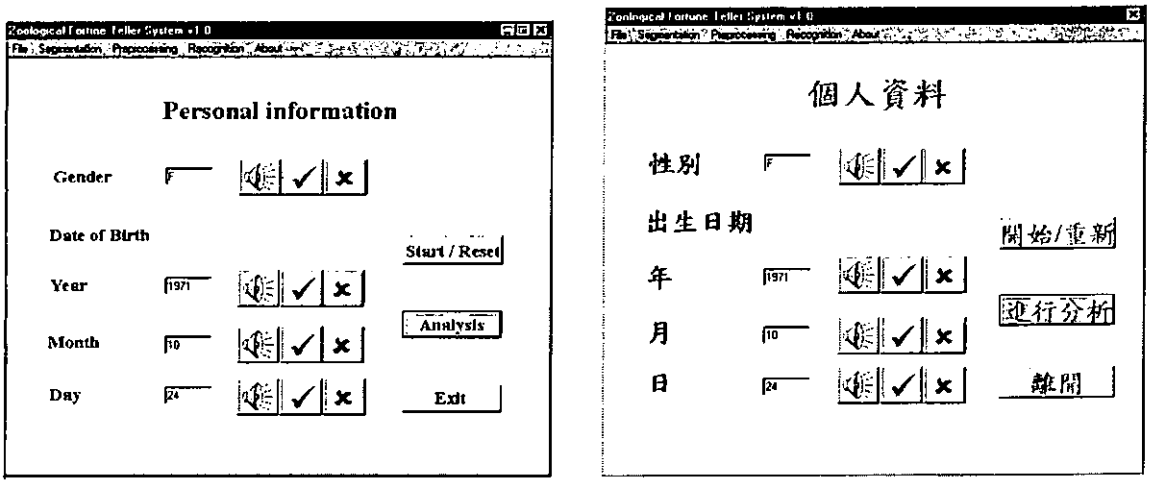


Figure 6.7 Main Page (with inputs) of the Application for English (left) and for Chinese (right)

After the analysis process, the final result will be spoken by the system while the related result page is displayed. As shown in Figure 6.8, some particular information, such as "General Character", "Love Attitude", and "Personal Analysis" for the user will also be provided.

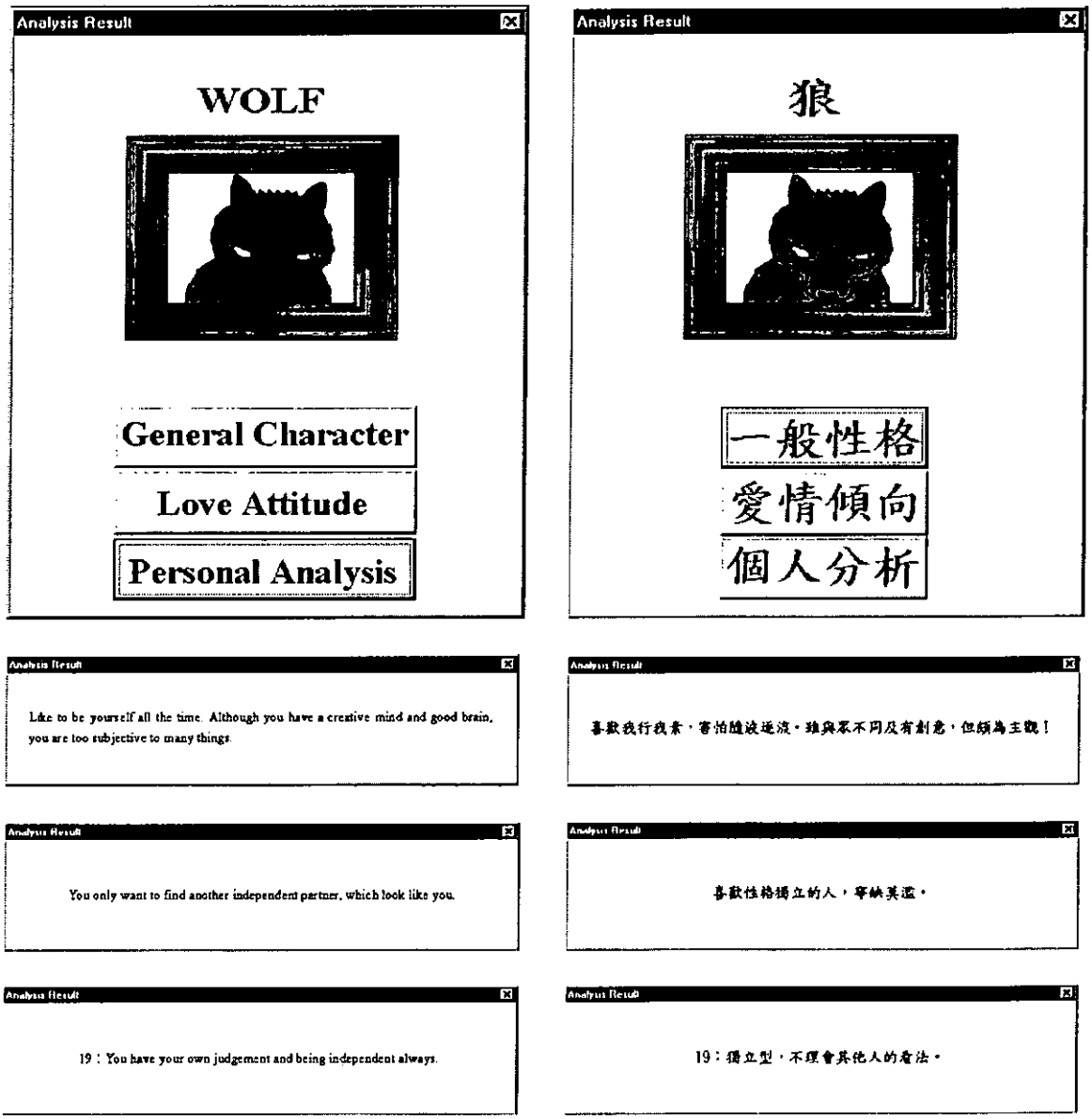


Figure 6.8 Result Page of the Application for English (left) and for Chinese (right)

In Figure 6.9, the dialog box is used to display the information about Zoological Fortune Telling Application.

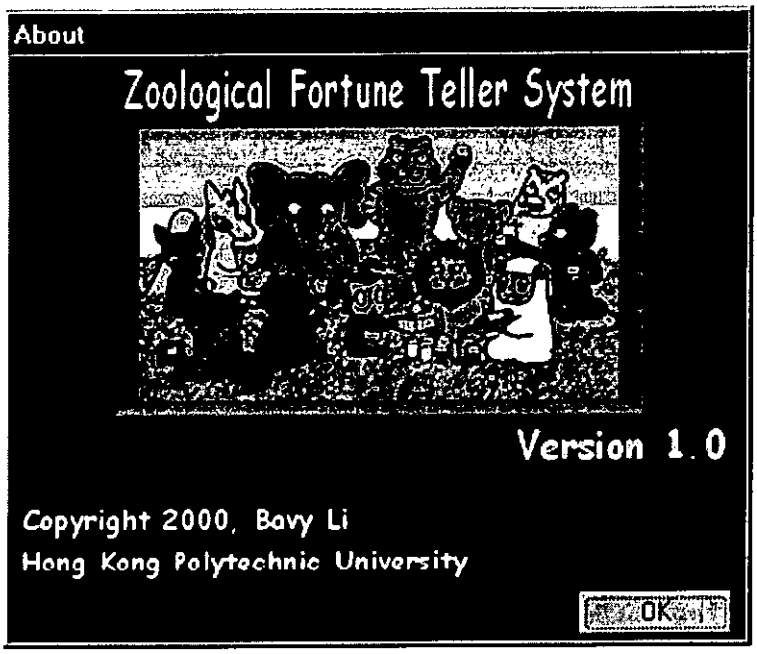


Figure 6.9 About Box of the Application

Chapter 7

Conclusion and Potential for Extension

7.1 Conclusion

In this thesis, we have presented an integrated **ASR** system for the three common dialects (*Cantonese*, *Mandarin*, and *English*) in Hong Kong for speech segmentation, preprocessing and recognition. In order to illustrate our **ASR** system, which can recognize the three mentioned dialects and apply for a Voice Response System, an **ASR** zoological fortune telling application has been constructed as shown in Chapter 6. In the following, we give a conclusion of this study.

Chapter 1: We have presented a brief introduction on **ASR** technology and briefly looked at some factors affecting its feasibility and performance. We have also discussed the motivation, the objectives and the suggested solution of this thesis.

Chapter 2: At the beginning, we introduced the three common dialects (Cantonese, Mandarin and English) in Hong Kong and their characteristics. We also gave a brief

description of some current approaches of **ASR** system including *Acoustic Phonetic*, *Pattern Recognition*, and *Artificial Intelligence*.

Chapter 3: We gave a description of the traditional *Zero Crossing* and our proposed *Linguistically Free Segmentation* (**LFS**) modeling and compared their performance. The **LFS** modeling consists of *Convex Hull*, *Spectral Variation Function*, *Normal Decomposition*, and *Level Building Dynamic Programming based* algorithm for determining both the optimal number and the optimal boundaries of segmented speech. At the end of the chapter, our proposed **LFS** was shown to be much more stable than the traditional word-spotting algorithm by considering their standard deviation.

Chapter 4: In this chapter, we presented a description of speech preprocessing modeling with *Linear Predictive Coding* (**LPC**) analysis. Some existing **LPC**-based parametric representations were introduced and discussed, such as *Weighted Cepstral Coefficients*, *Mel-frequency Cepstral Coefficients*, and *Relative Spectral Coefficients*. In fact, speech-preprocessing technology is the only way to produce feature vectors (coefficients) on each speech segments for further computation and analysis without loss of information.

Chapter 5: We have introduced the modified class-dependent discretization algorithm for model building and the three speech recognition mechanisms include our proposed *Improved Naïve Bayesian Classification* (**INBC**), *Hidden Markov Modeling* (**HMM**) with *Viterbi* algorithm, and *Multi-Layers Backpropargation Modeling* (**ML-BkProp**). Through our experiments, the best performance for recognizing Cantonese can be achieved by applying **MFCC** into **INBC**, whereas the best performance for recognizing Mandarin and English can be achieved by applying **MFCC** into **HMM** with *Viterbi*

algorithm. It reveals that no single combination of algorithms from segmentation, preprocessing to recognition with the best performance on recognizing all different spoken-languages.

Chapter 6: We have briefly described current Interactive Voice Response technique and introduced an **ASR** Zoological Fortune Telling application to implement our integrated **ASR** model for the three studied dialects.

7.2 Limitation of the Research

This research has a number of limitations, which include:

1. As mentioned from the thesis title, it aims for constructing an **ASR** and Voice Response system. However, this thesis put many efforts on designing an integrated **ASR** model rather than applying Voice Response technique in our speech system. It is believed that the development of an integrated **ASR** system can be applied for a Voice Response system and there is no single combination of algorithms from segmentation, preprocessing to recognition that can help different spoken-languages with the best performance. Therefore, the part of applying a Voice Response system will be set as the future enhancement of our integrated **ASR** system and will not be emphasized in this thesis.

2. In this thesis, the term "multi-lingual" means that our integrated **ASR** system can support the three common dialects (Cantonese, Mandarin, and English) in Hong Kong independently.

3. Due to the difficulties on collecting raw speech data for the purpose of training and testing our ASR system, only small vocabulary set of the three languages was collected from 5 males and 5 females. Therefore, only two sets of five repetitions of digits (from digit 0 to digit 9) of the three dialects were recorded and used in our experiments. It is believed that the size of such digits-based speech data is enough for building sets of word-based models for further models training and testing. Any complicated words of the three languages will not be considered in this thesis.

7.3 Future Research

Our integrated **ASR** system can be further enhanced with various improvements as described in the following:

We may consider enhancing our system to support some continuous related-closed spoken dialects, such as Hakka, Tiew Chow, Shanghaiese, Fukien and Hainanese in Mainland China. On the other hand, we might want our system to support human-machine interaction in a natural way (users require no special training) and in a natural conversation (different dialects can be mixed into one sentence).

Since more and more spoken language understanding applications are available in the market nowadays, our system is expected to be able to support different kinds of users by handling information-intensive problems, such as information enquiry, travel planning, stock trading, and office management.

Finally, we may consider upgrading our systems to support multimodal. Response to a speech input by our system can also be in the forms of text, graphical and pictorial output rather than speech. Such systems can improve traditional human-computer interactions by permitting users to speak and write in their own native languages, to move or gesture while speaking, to view a synthesized human face (with lip movements and emotional expressions) on monitor while listening and/or to retrieve or manipulate different forms (speech, text, graphics, or set of actions) of output.

Bibliography

1. W. Buntine, "*Learning Classification Rules using Bayes*", in Proc. of the 6th International Workshop on Machine Learning, pp. 94-96, 1989.
2. Cestnik, "*Estimating Probabilities: A crucial task in machine learning*", in Proc. of the European Conference on Artificial Intelligence, pp. 147-149, 1990.
3. Y. Chauvin and D.E. Rumelhart, **Backpropagation: Theory, Architectures, and Applications**, New Jersey: Lawrence Erlbaum Associates, Inc., Chapter 1, 1995.
4. J.Y. Ching, A.K.C. Wong and K.C.C. Chan, "*Class-Dependent Discretization for Inductive Learning from Continuous and Mixed-Mode Data*", IEEE Trans on Pattern Analysis and Machine Intelligence, vol. 17, no. 6, pp. 1-11, 1995.
5. Christell & Associates, Interactive Voice Response Computer Telephone Integration, <http://www.christell.com/>, 2000.
6. J.W. Cooley and J.W. Tukey, "*An Algorithm for the Machine Computation of Complex Fourier Series*", Math. Computing, vol. 19, pp. 297-301, 1965.
7. A.M. De-Lima-Araujo and F. Violaro, "*Formant Frequency Estimation Using a Mel Scale LPC Algorithm*", in Proc. of ITS '98, IEEE Int'l., Vol. 1, pp. 207-212, 1998.
8. Dialogic Communications, Interactive Voice Response, <http://www.dccusa.com/ivr/>, 2001.
9. ElectricRates, Interactive Voice Response System, <http://www.electricrates.com/>, 1996.
10. L. Fausett, **Fundamentals of Neural Networks - Architectures, Algorithms, and Applications**, Prentice Hall, 1994.
11. K. Fukunaga, **Introduction to Statistical Pattern Recognition**, San Diego, Academic Press Inc., 2nd Ed., Chapter 11, 1990.
12. J.R. Glass and V.W. Zue, "*Multi-level Acoustic Segmentation of Continuous Speech*", in Proc. ICASSP, pp. 429-432, 1988.
13. Ben Gold and Nelson Morgan, **Speech and Audio Signal Processing – Processing and Perception of Speech and Music**, John Wiley & Sons, Inc., USA, 1999.
14. H. Hermansky, N. Morgan, A. Bayya, and P. Kohn, "*RASTA-PLP Speech Analysis Technique*", in Proc. IEEE Int'l Conf. Acoustic, Speech and Signal Processing (San Francisco), pp. I-121-I-124, 1992.

15. H. Hermansky and N. Morgan, "*RASTA Processing of Speech*", IEEE Trans. Speech Audio Process, vol. 2, pp. 578-589, 1994.

16. J.K.T. Huang and T.D. Huang, **An Introduction to Chinese, Japanese and Korean Computing**, World Scientific, 1989.

17. IBM Voice Systems, ViaVoice System, <http://www-4.ibm.com/software/speech/>, 2001.

18. J.C. Junqua and J.P. Haton, **Robustness in Automatic Speech Recognition: Fundamentals and Applications**, Kluwer Academic Publisher, 1996.

19. J. Karat, J. Lai, C. Danis and C. Wolf, "*Speech User Interface Evolution*", IBM T.J. Watson Research Center, 1999.

20. Kubo Kiriko and Nora, **Zoological Fortune Telling**, Shogakukan Inc., Japan, 1999.

21. Kolvox Communications Inc., OfficeTALK, <http://www.kolvox.com/>, 2001.

22. T.P. Krauss, L. Shure, and J.N. Little, **Signal Processing Toolbox User's Guide**, The Maths Works Inc., 3rd reprints, February, 1995.

23. P. Langley, W. Iba, K. Thompson, "*An Analysis of Bayesian Classifiers*", in Proc. of the 11th National Conference on Artificial Intelligence, pp. 223-228, 1992.

24. C.H. Lee and L.R. Rabiner, "*A Frame Synchronous Network Search Algorithm for Connected Word Recognition*", IEEE Trans. ASSP, November, vol. 37, no. 11, pp. 1649-1658, 1989.

25. K.F. Lee, "*Automatic Speech Recognition – The Development of the Sphinx System*", Kluwer, Norwell, Mass., 1989.

26. K.F. Lee, H.W. Hon and D.R. Reddy, "*An Overview of the SPHINX Speech Recognition System*", IEEE Trans ASSP, vol. 38, pp. 600-610, 1990.

27. Lernout and Hauspie Company, Dragon NaturallySpeaking System, Version 5.0, <http://www.dragonsys.com/>, 2000.

28. B. Li and J. Liu, "*An Empirical Study of Speech Systems for Dialect Recognition*", 1998 Postgraduate Research Day, ACM Hong Kong Chapter, October 1998.

29. B. Li and J. Liu and H.H. Dai, "*Forecasting from Low Quality Data with Applications in Weather Forecasting*", Informatica, An Int'l Journal of Computing and Informatics, Vol. 22, No. 3, pp. 351-358, December 1998.

30. B. Li and J. Liu, "*A Comparative Study of Speech Segmentation and Feature Extraction on the Recognition of Different Dialects*", In Proceedings of IEEE International Conference on Systems, Man, and Cybernetics (SMC'99), Tokyo, Japan, pp. 538-542, October 1999.

31. B. Li and J. Liu, "*A Comparative Study of Speech Segmentation and Preprocessing for Automatic Multi-lingual Recognition*", 1999 Postgraduate Research Day, ACM Hong Kong Chapter, October 1999.

32. B. Li, J. Liu and J. You, "*An Empirical Study of Linguistic-Free and Linguistic-Constrained Segmentation Methods for Multi-lingual Speech Recognition*", 2000

International Workshop on Multimedia Data Storage, Retrieval, Integration and Applications, Hong Kong, January 2000.

33. J.N.K. Liu, B.N.L. Li and T.S. Dillon, "An improved Naive Bayesian Classifier Technique coupled with a novel input solution method", IEEE Trans. on Systems, Man, and Cybernetics, Part C, Vol. 31, No. 2, May 2001.
34. J. Makhoul, "*Linear Prediction: A Tutorial Review*", IEEE Proceedings, vol. 63, pp. 561-580, 1975.
35. P. Mermelstein, "*Automatic Segmentation of Speech into Syllabic Units*", Journal of Acoustical Society of America, October, vol. 58, no. 5, pp. 880-883, 1975.
36. Ministry of Education, **Order of Ju Yin Characters**, Executive Order Number 75, The Ministry of Education, November 23, 1918.
37. N. Morgan and H. Bourlard, "*Continuous Speech Recognition: An Introduction to the Hybrid HMM / Connectionist Approach*", Signal Process. Mag. vol. 12, pp. 25-42, 1995.
38. H. Murveit, M. Cohen, P. Price, G. Baldwin, M. Weintraub and J. Bernstein, "*SRI's DECIPHER System*", in Proc. Speech Natural Lang. Workshop, Philadelphia, pp. 238-242, 1989.
39. C.S. Myers, L.R. Rabiner and L. Rosenberg, "*Performance Tradeoffs in Dynamic Time Warping Algorithms for Isolated Word Recognition*", IEEE Trans. ASSP, vol. 28, pp. 623-635, 1980.
40. C.S. Myers and L.R. Rabiner, "*A Level Building Dynamic Time Warping Algorithm for Connected Word Recognition*", IEEE Trans. ASSP, April, vol. 29, pp. 284-297, 1981.
41. S.J. Netwon, **Voice in Office System**, NCC Publication, 1985.
42. Newell, J. Barnett, J. Forgie, C. Green, D. Klatt, J.C.R. Licklinder, J. Munson, R. Reddy and W. Woods, "*Speech Understanding Systems: Final Report of a Study Group*", North- Holland / American Elsevier, Amsterdam, Netherlands, 1973.
43. J.W. Picone, "*Signal Modeling Techniques in Speech Recognition*", in Proc. IEEE, Vol. 81, No. 9, pp. 284-297, April 1981.
44. L.R. Rabiner and B.H. Juang, **Fundamentals of Speech Recognition**, Prentice Hall, Englewood Cliffs, New Jersey, 1993.
45. L.R. Rabiner and B.H. Juang, "*An Introduction to Hidden Markov Models*", IEEE ASSP Magazine, pp. 4-16, Jan 1986.
46. T. Robinson, M. Hochberg and S. Renals, "*The Use of Recurrent Neural Networks in Continuous Speech Recognition*", in C.H. Lee, F.K. Soong and K.K. Paliwal, eds., Automatic Speech and Speaker Recognition, Kluwer, Boston, Mass., 1996.
47. R.D. Rodman, **Computer Speech Technology**, Artech House, Boston, London, 1999.
48. R.W. Schafer and L.R. Rabiner, "*Design of Digital Filter Banks for Speech Analysis*", Bell Syst. Tech. Journal, December, vol. 50, no. 10, pp. 3097-3115, 1971.

49. T. Svendsen and F. Soong, "*On the Automatic Segmentation of Speech Signals*", in Proc. ICASSP, pp. 3.4.1-3.4.4, 1987.

50. The Linguasphere Observatory, "*What Languages are Most Spoken in the World?*" <http://www.linguasphere.org/language.html>, 2000.

51. The Speech Processing Technical Committee, "*The Past, Present, and Future of Speech Processing*", IEEE Signal Processing Magazine, May, pp. 24-48, 1998.

52. K.S. Trivedi, **Probability and Statistics with Reliability, Queuing, and Computer Science Application**, Prentice-Hall, Inc., Englewood Cliffs, N.J., pp. 469-471, 1982.

53. A.K.C. Wong and D.K.Y. Chiu, "*Synthesizing Statistical Knowledge from Incomplete Mixed-Mode Data*", IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 9, No. 6, pp. 796-805, 1987.

54. S.L. Wong, **A Chinese Syllabary Pronounced According to The Dialect of Canton**, Chung Wah Book Shop, Revised Edition, 1993.

55. V. Zue, J. Glass, M. Phillips and S. Seneff, "*The MIT SUMMIT Speech Recognition System: A Progress Report*", in Proc. Speech Natural Lang. Workshop, Philadelphia, pp. 179-189, 1989.