TOWARD SPEAKER INDEPENDENT ISOLATED WORD RECOGNITION FOR LARGE LEXICONS: A TWO-CHANNEL, TWO-PASS APPROACH

By

JERRY N. LARAR

A DISSERTATION PRESENTED TO THE GRADUATE SCHOOL
OF THE UNIVERSITY OF FLORIDA
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

UNIVERSITY OF FLORIDA

1985

To Lilliana

ACKNOWLEDGEMENTS

The completion of this research has confirmed that the gratitude traditionally extended on this page is indeed sincere. First and foremost, thanks are due to the author's advisor and committee chairman, Dr. D.G. Childers, for his invaluable assistance throughout this research. The guidance, encouragement, and financial support provided are most appreciated. The author also thanks Drs. G.P. Moore, J.R. Smith, L.W. Couch, and H.B. Rothman for their time and interest in serving on the supervisory committee.

An invaluable part of the learning experience has been the insightful discussions with committee members and with Drs. B. Yegnanarayana and D. Hicks. Equally stimulating were the many extensive discussions with friends and former colleagues, Drs. J. Naik, A. Krishnamurthy, and T. Achariyapaopan, as well as with Y. Alsaka and other present colleagues at the Mind-Machine Interaction Research Center.

The author wishes to express appreciation to his family, to Ms. L. Capitanio, and to her sister Anna, for their love, constant encouragement, and help in many ways. He also thanks Ms. D. Hagin for being a cooperative speaker and a fine typist.

TABLE OF CONTENTS

Abstract of Dissertation Presented to the Graduate School
of the University of Florida in Partial Fulfillment of the
Requirements for the Degree of Doctor of Philosophy

TOWARD SPEAKER INDEPENDENT ISOLATED WORD RECOGNITION
FOR LARGE LEXICONS: A TWO-CHANNEL, TWO-PASS APPROACH

By

Jerry N. Larar

May, 1985

Chairman: D.G. Childers
Major Department: Electrical Engineering

A phonetic feature based strategy is an alternative to the classical pattern matching approach to isolated word recognition. High performance recognition can be achieved using detailed phonetic knowledge. For large vocabulary recognition, partial phonetic information can be used to reduce the number of likely match candidates before a detailed phonetic analysis is performed. Phonetic labeling, however, is nontrivial for a multispeaker application.

This study is an example of an intermediate step in the transition to a solely feature based system. The emphasis is on lexical access via broad acoustic-phonetic feature representation, but pattern matching is still employed in the final decision process. A manageable subset of the lexicon is retrieved in the first stage of recognition using a speaker independent word representation indicative of manner of

articulation, stress, and fricative location. The second stage utilizes dynamic time warping (DTW) techniques for detailed pattern matching.

The synchronized electroglottographic (EGG) signal is used as a second channel of data to ensure reliable first pass utterance representation. The glottal sensing characteristics of the EGG aid endpoint detection, voiced-unvoiced-mixed-silent classifications, and pitch tracking. Simple algorithms for these functions are described.

Lexical entries with the same initial representation are grouped together to form equivalence classes. The within class ordering considers the relative frequency of occurrence of the words. Further search space reduction is shown to be possible using only the most likely candidates. The pattern matching stage can use class specific matching techniques since the word representation is known from the first pass.

The recognition techniques are evaluated using eight speakers and a difficult 100 word vocabulary. Results show that performance can be made to depend solely on the detailed matching stage at the cost of an increased number of DTW comparisons. This number is still significantly less than the size of the vocabulary. Alternatively, very few candidates can be matched with a predicted increase in lexical access error. Class specific local weighting techniques improve performance for the alpha subset of the vocabulary.

CHAPTER 1
INTRODUCTION

The basic human ability to disseminate ideas and knowledge through verbal communication has greatly facilitated the advancement of our society. In the last two decades, the proliferation of computing machinery has necessitated extension of the communication link to our electronic contemporaries. The realization of this need for a suitable man-machine interface has motivated a great deal of research in automatic speech recognition.

Almost a decade after Pierce's [1] caustic assessment of advances in and the motivation for research in speech recognition, Neuberg [2] noted that practical speech recognition still had not made any great strides forward. The last few years have seen many successful limited recognition systems demonstrated, and some marketed [3,4]. The general consensus continues to be, however, that the greatest future contributions will be in the form of solid advances in one direction rather than more complicated, comprehensive speech recognition systems with marginal improvements in performance [5].

The various systems in existence today reflect efforts that have been directed toward a wide range of sub-tasks related to the ultimate goal of unrestricted continuous speech input to machines. Whether the task is phoneme, word, connected word, or continuous speech recognition, researchers must overcome the fundamental problem of speaker variability. Even for the single speaker case, it is highly unlikely

that two repetitions of the same utterance will have identical temporal and spectral characteristics. Thus, the task of recognition is essentially that of determining the information carrying features common to all utterances of the same message and using those features to classify utterances according to their intended meanings.

The purpose of this chapter is to provide an introduction to the various techniques employed in speech recognition. After a brief overview of common methodologies, the issues addressed in this research will be presented. The chapter concludes with a summary of the organization of the remaining text.

1.1 General Speech Recognition

The typical approach has been to treat speech recognition as a classical problem in pattern recognition. This involves comparing the parameter or feature representation of the incoming utterance with the previously stored prototype reference patterns of each of the words in the vocabulary. Only limited success in direct extrapolation of this pattern recognition paradigm to the case of continuous speech has been realized. The problem is that the intelligence conveyed by speech is encoded locally into spectral features and globally into structural features [6]. A more general model of the speech communication process and hence better system performance is achieved when greater consideration is given to the structural and linguistic aspects of speech. This strategy was utilized in the speech understanding systems spawned by the Advanced Research Projects Agency's (ARPA) speech understanding project [7].

Further work is still needed toward more appropriate acoustic analysis and feature extraction for pattern matching, and on the augmentation of pattern recognition principles with formal language theory and semantic analyses. The scope of discussion can be narrowed by focusing on aspects of the former. Specifically, the concerns of this study are limited to the task of recognizing words spoken in isolation.

1.2 Isolated Word Recognition (IWR)

The multitude of techniques for isolated word recognition usually can be described in terms of two basic approaches. The unknown utterance can be treated either as an isolated acoustic pattern which is compared with stored reference templates, or features may be extracted and used to identify the unknown word.

Generalizing even further, we can fit most techniques into the canonic pattern recognition model shown in Figure 1.1. This model is invariant to different vocabularies, users, feature sets, and distance measures. The basic structure is also easy to implement and works well in practice. Although a number of realizations exist and some modifications have been made to the various stages, recognition schemes will incorporate this model into some aspect of the procedure used if not relied upon entirely. For instance, the distinction made above between feature and template matching is important yet misleading. The essential discriminatory technique used in both approaches follows the model of Figure 1.1. The major difference is in the type of features measured. Template matching connotes a form of spectral time sequence matching. Feature based systems rely on further reduction of the data to descriptors such as formant frequencies, spectral energy ratios, etc.

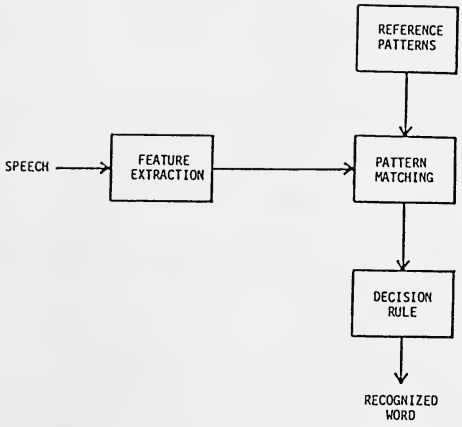


Figure 1.1 Pattern recognition model for IWR [6].

The following discussion of isolated word recognition systems proceeds in the context of Figure 1.1. Techniques to implement the functions of the different steps in the model are given first for speaker dependent systems and then for speaker independent systems. Afterwards, newer variations of the basic approach to the IWR problem are considered.

1.2.1 Speaker Dependent IWR

For the purpose of data reduction, feature measurement is employed to transform the incoming digitized speech signal into a smaller number of parameters that accurately represent the prominent characteristics of the original waveform. Many different feature sets have been used in IWR systems. Features range from easily obtainable signal energy and zero crossing rates to the more complex filter bank and linear predictive coding (LPC) spectral representations. The choice of a feature set is dependent upon constraints such as vocabulary, computational complexity, and storage requirements.

The most popular approach to IWR involves direct matching of input and reference spectra. The feature sets are then usually obtained from either a filter bank or LPC model. The early vowel recognition work of Peterson and Barney [8] used formant information from spectrograms as the spectral features for classification purposes. Modern word recognition systems now derive this information from on line filter bank or LPC analyses depending on the level of performance attainable for the given task [9,10].

In a filter bank analysis model such as that depicted in Figure 1.2, the speech signal is passed through a bank of Q bandpass filters

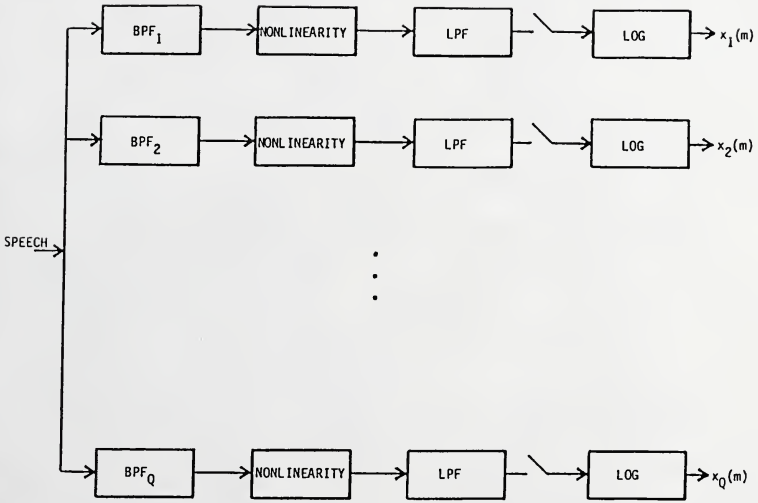


Figure 1.2 Block diagram of filter bank system [9].

that span the frequency range of interest. The number of filters used, Q, varies from about 5 to 40. Dautrich, et al. [9] found that the best IWR performance was obtained with 15 uniformly spaced filters or 13 nonuniformly spaced filters positioned along a critical band scale.

Typically, the output of each bandpass filter is rectified, lowpass filtered, and resampled at a lower rate for efficient storage before the output is logarithmically compressed. This yields estimates of the speech energy in the passbands of the Q filters. Collectively, the parallel outputs $x_1(m)$, $x_2(m)$, ..., $x_Q(m)$ comprise a Q^{th} order feature vector X(m). The variation of X with m defines a pattern, P, given by

$$P = \{X(1), X(2), \ldots, X(M)\} \tag{1.1}$$

where

$$X(1) = \{x_1(1), x_2(1), \ldots, x_Q(1)\} \tag{1.2}$$

and similarly for all X(m) such that $1 \leq m \leq M$. The spectral description of Eq. (1.1) is the pattern used by the recognition system.

The signal processing in Figure 1.2 need not be done digitally. In fact, inexpensive analog filter bank implementations are commonly used in commercial devices [4]. Variations in digital schemes also exist. An example is Klatt's FFT implementation of the filter bank [11].

The other frequently used spectral representation is the LPC based feature set originally proposed for use in IWR by Itakura [12]. A block diagram of a system for extracting LPC features is shown in Figure 1.3. The speech is first preemphasized and then blocked into N sample frames spaced M samples apart. The separation, M, may be chosen to be less than N so that the overlap provides smoothing between adjacent feature vectors. Then, a window is applied to each frame to taper the data before an autocorrelation analysis is performed.

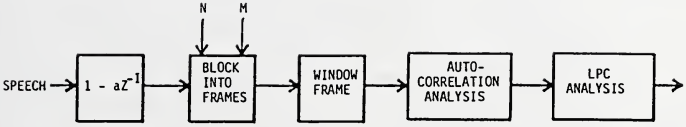


Figure 1.3 LPC analysis system [6].

Once the autocorrelation coefficients, R(m), m = 0, 1, ..., p, are computed, the set

$$X(j) = \{R_j(0), R_j(1), \ldots, R_j(p)\} \tag{1.3}$$

can be used as the feature vector. This is often just an intermediate step preceding a least squares solution of a set of linear simultaneous equations to obtain the LP coefficients. In the frequency domain, this is equivalent to an all-pole fit to the spectrum of the present frame of data. The resulting model is of the form

$$A_j(z) = \frac{G}{\sum_{m=0}^{p} a_m^j z^{-m}} \tag{1.4}$$

where $a_0^j = 1$, G is a gain factor, and the set of coefficients

$$X(j) = \{a_0^j, a_1^j, \ldots, a_p^j\} \tag{1.5}$$

define the all pole model and can also be used as the features for frame j. A short review of the basic concepts of linear prediction is provided in Appendix A.

At this point, the two most common procedures for creating spectral feature sets have been presented. Any deviation from these methods typically occurs in the last step of actual feature vector formation. For example, cepstral coefficients may be computed from the filter bank outputs and then stored [13]. Proportional normalization of the spectral energy is used in [14] to compensate for inter-utterance signal differences. Both normalization and spectral channel smoothing are employed in [15]. The vector of LPC coefficients has been augmented with zero crossing and energy measurements as well as with stationarity information [16,17].

Now that a pattern of feature vectors has been created, the next task is to identify the utterance, or to label the segment if a phonemic string approach is used. This entails determining similarity between test and reference utterances. The accuracy of the actual recognition system is determined by what takes place in this matching stage where it is decided which reference pattern most closely resembles the unknown test utterance. Assessment of pattern similarity involves not only distance computation, but also some form of time alignment since speaking rates are never constant. For word pattern matching, these two functions are often performed simultaneously using a dynamic time warping (DTW) algorithm [12,18-23]. Phonemic labelling and string matching use techniques ranging from dynamic programming [24] to the phonemic subspace decomposition and feature distance string comparison methods of [25].

The function of time alignment between a test pattern T(n) and a reference pattern R(m) is depicted in Figure 1.4. Most template based IWR systems use dynamic programming to find the nonlinear alignment function w(n) which maps R onto the corresponding parts of T. The mapping w minimizes some measure of distance between the two patterns. The proper alignment is found by solving the optimization problem

$$D(T,R) = \min_{w(n)} \left[\sum_{n=1}^{N} d(T(n),R(w(n))) \right] \tag{1.6}$$

where $d(T(n),R(w(n)))$ is commonly the Itakura distance [12] between frame n of the test pattern and frame w(n) of the reference pattern. The Itakura measure is described in Appendix A and Chapter 6 gives more insight into dynamic time warping.

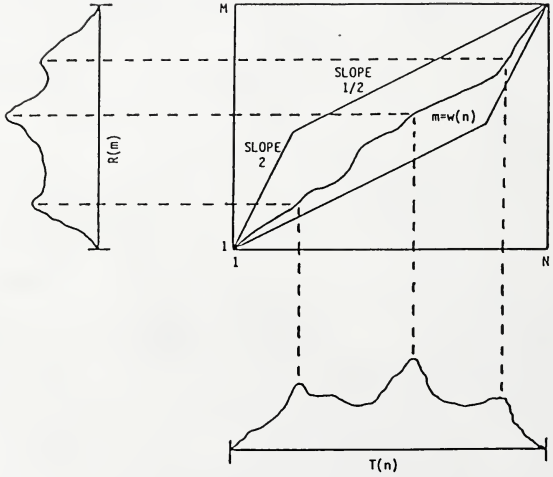


Figure 1.4 Nonlinear alignment of patterns [6].

The description just given of IWR systems based on the model in Figure 1.1 implies speaker dependence if the reference patterns are created by a single speaker. Intraspeaker variability in factors such as amplitude, timing, articulation, etc. requires that each user trains the system by repeating every word in the vocabulary one or more times during a training phase. This adapts the recognizer to the speaker and may yield better than 98% accuracy for the given application [6,9,10, 12-16,26]. When the user's voice changes or when the size or confusability of the vocabulary increases, the performance of these systems degrades substantially. Also, for applications such as remote data entry and retrieval, order entry, switchboard functions, etc., it is not at all practical to retrain the system for each new speaker, especially when the vocabulary size is large.

1.2.2 Speaker Independent IWR

One of the largest impediments to progress in the field of automatic speech recognition is the substantial variation in speech between individuals. Such differences have made the development of speaker independent recognition systems a formidable task. So great are the interspeaker differences due to vocal tract size and length, dialect, and the amount of coarticulation, that the majority of existing recognition systems employ some form of speaker adaptation in order to achieve reasonably high recognition accuracy [26].

Recognition is more difficult in speaker independent systems because spoken words from people for whom no previous voice samples have been stored must be recognized. A common line of thinking has been that if speaker specific word samples are not available to the recognition

system, then speech samples must be collected from hundreds of people before a new unknown person's speech can be accurately recognized. Very large data bases for an entire population of speakers must therefore be collected, processed, and clustered to obtain representative models of the different types of speech for each word in the lexicon [3].

This approach has been used in many successful speaker independent recognition systems based on the model in Figure 1.1 [27-30]. Isolated occurrences of feature sets of a word (from many different speakers) are converted to reference patterns for the recognizer during a training mode. One method downgrades spurious features by averaging together the templates for all occurrences of a given word to form a single reference template. More effective, though, is clustering conversion where P occurrences of each vocabulary word are grouped together to form Q clusters. For each of the Q clusters, an averaging technique is used to obtain a single reference template. This results in storage of multiple templates for each word to account for interspeaker variations and thus requires use of a K-nearest neighbor decision rule.

Other approaches usually concentrate on key feature representations for greater discriminability between words. The features selected must also have a speaker independent interpretation. This idea was used early on for digit recognition [31], recently in a syntactic pattern recognition scheme [32], and many times between these studies in conjunction with template clustering.

1.2.3 New Directions in IWR

Most of the recognition schemes mentioned thus far utilize some form of initial spectral analysis. Feature based systems follow this with additional data reduction stages prior to word identification. Final word matching may take place using a decision tree of statistical discriminators [33] or a maximum likelihood approach based on the frequency of occurrence of certain features [34]. A drawback of the statistical feature based approaches is the general lack of knowledge of which features to use. Much progress remains to be made in the understanding and modeling of the mechanism of human speech perception. The ability to perform fine phonetic distinctions [33], however, demonstrates feature based systems as a viable approach.

An alternative route to follow involves the direct use of the spectral representation of the utterance for recognition. The input signal is compared with previously stored representations of each lexical entry in order to identify the spoken word. Popular techniques for this approach were described in the last two sections. A number of successful IWR systems have been implemented using these techniques, yet several open issues still remain. Two major problems that persist are the reduction in accuracy for complex vocabularies and the high computational cost incurred by DTW matching, particularly for large lexicons.

For a vocabularly of similar sounding words, the pattern matching algorithm must focus on the regions that serve to maximize discriminability among the words. This has been effected using one [35] and two-pass techniques [28,36] with varying amounts of success. Computational requirements have been reduced within the DTW framework by

limiting the number of match candidates with some form of lexical pruning [36-38]. Another approach uses vector quantization [39] of the LPC parameters to reduce the DTW distance computations to simple table lookup operations [40]. The cost of DTW can be circumvented by implementing an IWR system without explicit time normalization. Examples of such systems are those using vector quantization as in [41,42], and those that combine vector quantization and hidden Markov modeling [29,43,44].

In assessing the state of the art in speaker dependent and speaker independent IWR, one may say that much progress has been made since highly accurate task specific systems are now commonplace. From a broader perspective, however, it is clear that much work still remains to be done toward truely robust, multitalker, multitask (large vocabulary) IWR. Exactly which is the best path to follow remains to be seen. Perhaps a combination of template matching and feature based schemes will prove to be optimal.

1.3 Research Issues

In general, existing speaker dependent and speaker independent IWR systems work quite well for small vocabularies of moderate complexity. Many practical applications, however, require fairly large lexicons as well as operation in multitalker environments. Speaker independent IWR for large lexicons remains an active area of research since only limited success towards this end has been attained. This lack of significant progress constitutes the underlying problem and impetus for this research.

The objective of this study is to explore techniques for reliable speaker independent IWR for the case of large lexicons. The approach taken requires two passes; the utterance is first associated with a reduced equivalence class of words and then detailed pattern matching is performed. Emphasis is placed on reliable lexical access. A two-channel approach is utilized to accomplish accurate and efficient classification of the spoken word. The glottal sensing characteristics of the electroglottograph (EGG) make its output attractive for use as the second channel of information. This work demonstrates the practicability of a two-pass, two-channel (speech and EGG) speaker independent IWR system. The chapters that follow describe the design, implementation, and testing of a modular, minicomputer-based laboratory realization.

1.4 Description of Chapters

An overview of the IWR system is given in Chapter 2. After presenting rationale for the two-pass approach, lexical access procedures are described. Discussion then turns to use of electroglottography within the adopted framework. Finally, the within class matching procedures are reviewed.

The techniques associated with reference data collection and preprocessing are discussed in Chapter 3. In Chapter 4, methodologies for endpoint detection, acoustic classification, pitch extraction, and spectral analysis are described. Chapter 5 presents the details of equivalence class formation and lexical access. Chapter 6 expounds on the detailed word matching procedures including time normalization, template formation, and the decision rule. The IWR system's performance

is evaluated in Chapter 7. Results are presented for both actual and predicted performance. Chapter 8 summarizes results and identifies areas where extensions in future endeavors may prove fruitful.

CHAPTER 2
APPROACH TO THE IWR PROBLEM

Speaker independent recognition of words from a large vocabulary is a task comprised of a conglomeration of separate problems. Aside from basic recognition, the issues of speaker independence and large vocabulary application warrant foremost consideration. All aspects are inter-related and necessarily contribute to the general approach.

The IWR methodologies proposed in this study include template matching as well as extraction of crude acoustic-phonetic features. Recognition is accomplished in two passes; first a set of words with the same feature representation is retrieved from the lexicon and then a DTW algorithm is used to match the spectral template of the test word to those for the reference words in the set. This chapter provides an introduction to the various components of the IWR system developed. Later chapters treat topics associated with the stages more thoroughly.

2.1 Two-Pass Recognition

For relatively small vocabularies (up to 50 words), it is not unreasonable to exhaustively search the entire set of reference templates for the best match to the test utterance. This standard pattern recognition approach has also been applied successfully to moderately large vocabularies (up to 150 words) as in [45]. When the vocabulary size is increased to several hundreds or thousands of words, a number of problems surface.

One major problem is the impracticality of retraining a large vocabulary system for each new user. To avoid this problem, recognition can be based on a fundamental subunit such as a phoneme or demisyllable [46]. The system training time would then be reduced to roughly that for a small vocabulary. Another solution is to avoid training by implementing a speaker independent system. The template formation procedure outlined in Section 1.2.2 will result in from 6 to 12 reference patterns for each word. Assuming adequate storage is available, the recognizer's response time is the limiting factor. The computational cost becomes prohibitively expensive because the unknown test utterance has to be compared with up to 12 reference templates for each word in the vocabulary. In addition to the critical requirement of response time, recognition error rates increase with each doubling of vocabulary size [47].

The difficulties above suggest a procedure in which only some subset of the entire vocabulary is considered. Clearly, there is no need to attempt a match with inappropriate candidates. Totally unambiguous words can be eliminated reliably without the substantial computation needed for precise acoustic matching. The basic idea is to achieve effective lexical pruning with a computationally inexpensive first pass. Once the search space has been reduced to only those words that are similar in some sense to the test utterance, the detailed matching techniques can be applied. This approach has been used in DTW systems for the alpha-digit task [36], connected word recognition [38], and for an office correspondence corpus [37]. A phonetic approach for large vocabulary recognition was used in [48]. Several preliminary studies have demonstrated the usefulness of acoustic-phonetic

categorization of the utterance to retrieve a small set of match candidates from a large vocabulary [49-51]. Results from these investigations were considered in formulating the procedures to be described next.

2.1.1 Lexical Access

The two pass approach provides an economical solution to the large vocabulary recognition problem. A vocabulary of moderate size (100 words) was used in this work for practical considerations, but the procedures are directly extensible to large vocabularies. The purpose of this section is to present the basics of a first pass word representation that ideally corresponds to only a small fraction of the words in the vocabulary. The representation must be independent of the vocabulary, independent of the speaker, and must be robustly obtained from the utterance.

Most of the techniques for search space reduction in [48-51] represent the word in ways that meet the criteria above. In [49] and [51], the broad phonetic categories of 1) vowel, 2) stop, 3) nasal, 4) liquid or glide, 5) strong fricative, and 6) weak fricative are used. Some of the distinctions required are by no means trivial (e.g. liquids/nasals) and are therefore prone to errors. A slightly better choice of categories given in [48] includes 1) unvoiced plosive, 2) voiced plosive, 3) unvoiced fricative, 4) voiced fricative, 5) vocalic nucleus, and 6) all other sounds. A fair amount of difficulty is expected in measuring voice onset time to make the voiced/unvoiced distinction for plosives. Also, voiced fricatives are usually hard to detect from the speech signal alone. Some of the suprasegmental cues

used in [50] are largely durational. A measure may then retain characteristics of the speaker and not always correspond to what was expected from the orthographic representation.

In this study, a broad acoustic-phonetic categorization of the test utterance is utilized for lexical access. The word is represented by a sequence of voiced, unvoiced, mixed (voiced and unvoiced), or silent segments. These classes approximate vowels, unvoiced fricatives or plosives, voiced fricatives or plosives, and the stops prior to a plosive release, respectively. Prosodic cues of stress and intonation are also extracted for possible use. All of the features used are combined to eliminate durational information. The only temporal structure remaining is in specification of the relative position of certain features (e.g. stress and unvoiced regions).

Since time is normalized out of the word representation, insertion-deletion problems are not handled during lexical access. Word equivalance classes are formed in a preliminary training mode and overlap is allowed to account for normal variability in pronunciation. The words of the lexicon are stored in a form keyed by their equivalence class representation. A single table lookup will therefore yield all of the entries with the same general acoustic features. An approach to efficient lexical access based on a linear code assumption was initially considered. Enough independent features were not available from the word to exploit the error correcting capabilities so the method was not used. A brief description of the approach is given in Appendix B.

Use of electroglottography. Increasing the number of classes used to describe an utterance will result in fewer words matching a particular representation. The tradeoff will be increased

classification error if the procedures are not suitably robust. The method of representation adopted for this work is limited to a small number of categories since only reliably obtained information is used. For optimal performance it is imperative that words are represented in a consistent manner. To assure the necessary level of performance, the electroglottographic signal is used as a second channel of information.

The electroglottograph is essentially an impedance measuring device used to monitor vocal fold activity. The output from the device is a direct indicator of glottal vibration and therefore can be used as a source of voicing information that is decoupled from the effects of the supraglottal system. This auxiliary signal introduces new levels of accuracy and simplicity to voicing decisions and pitch tracking.

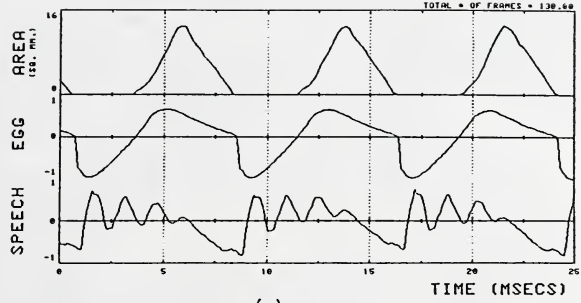
Electroglottography is one of many techniques for indirect observation of vocal fold activity [52]. Some other techniques include ultra-high-speed cinematography, photoglottography, x-rays, etc. [53]. Indirect techniques are required due to the relative inaccessibility of the larynx. The particular advantage of electroglottography is that it is an inexpensive, noninvasive procedure that can be used with almost all speakers for any phonation.

The EGG signal is obtained from a high frequency probe current passed through the neck at the level of the larynx. Plate electrodes are positioned on both sides of the thyroid cartilage so that the current passes from one electrode through the laryngeal structure and is picked up by the other electrode. The vibration of the vocal folds changes the impedance seen by the device. The sensing electrode therefore picks up an amplitude modulated radio frequency signal. A detector circuit yields the demodulated output which is believed to be

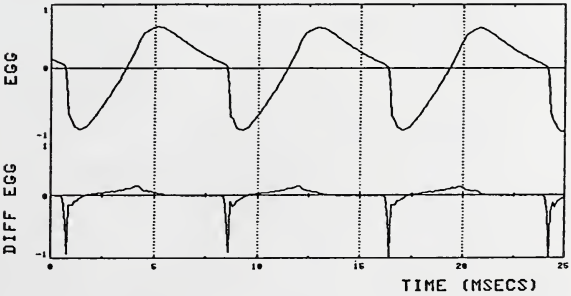
proportional to the amount of lateral contact area between the vocal folds. A comprehensive review of electroglottography is available in [54].

Application of the EGG to speech analysis, synthesis, recognition, and speaker identification is due to the interpretation of its output as a signal varying with vocal fold movement [54-61]. The present understanding of the EGG results from numerous investigations that have correlated the EGG with other synchronously obtained glottographic waveforms [59,62-66]. Some important properties of the EGG signal are that 1) the rising portion of the EGG corresponds to separation of the vocal folds and usually has a break in the slope, 2) the EGG maxima are during glottal open phase, 3) the moment of first glottal contact corresponds to the "knee" in the EGG where the greatest negative slope is found, 4) the EGG minima are during glottal closed phase, 5) the zero mean EGG signal is quite regular and has exactly two zero crossings per cycle; a positive zero crossing occurs approximately when the folds separate on opening, and the negative crossing is approximately at the time of first lateral contact on closing, and 6) the EGG exhibits no movement in the absence of voicing provided there are no irrelevant structural adjustments (and hence impedance variations).

An example of synchronized speech, EGG, and glottal area measured from an ultra-high speed film is given in Figure 2.1.a. The EGG and differentiated EGG are shown in part (b) to illustrate the useful peaks in the differentiated signal. Maxima occur at the instants of glottal opening (last separation) and the minima are at the instants of glottal closure (first contact). Consideration of EGG characteristics in relation to glottal vibratory motion observed from ultra-high-speed



(a)



(b)

Figure 2.1 Synchronized waveforms: (a) glottal area, speech, and EGG, (b) EGG and differentiated EGG.

films [65,66] has resulted in the model of the EGG waveform in Figure 2.2 [55].

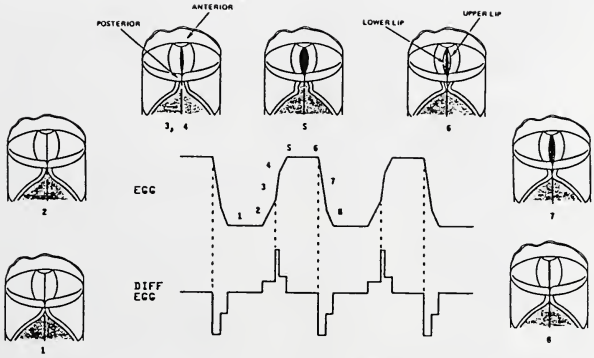
The fact that the EGG is a reliable source of glottal vibratory information makes it quite useful for the present study. The two channels (speech and EGG) of data facilitate the two-pass approach taken towards recognition. The pattern matching performed in the second pass will be discussed next.

2.1.2 Pattern Matching

The first pass by the recognizer used a broad acoustic-phonetic representation to obtain an equivalence class of words from an indexed set of references. The second stage of recognition discriminates among members of the class to find the reference word that most closely matches the test utterance. Since the search space has been reduced to a manageable size, the computationally burdensome template matching procedures can be used.

An autocorrelation LPC analysis is performed to determine the parameters of the model in Eq. (1.4). The feature vectors used to describe an utterance are then sets of coefficients as given in Eq. (1.5). Pattern similarity is measured using dynamic time warping with the Itakura distance [12]. The word spoken is recognized as the word that corresponds to the reference pattern with the smallest distance from the test pattern. If none of the references are suitably close, no word is matched and the test utterance must be repeated.

The reference patterns are time sequences of vectors of the form in Eq. (1.5) except they were not computed from a single analysis. To



1 VOCAL FOLDS MAXIMALLY CLOSED. COMPLETE CLOSURE MAY NOT BE OBTAINED. FLAT PORTION IDEALIZED.
2 FOLDS PARTING ALONG LOWER MARGIN (LIP) TOWARD UPPER MARGIN.
3 FOLDS OPENING ALONG UPPER MARGIN, USUALLY OPENING IS ALSO FROM POSTERIOR TO ANTERIOR.
4 FOLDS CONTINUING TO OPEN.
5 FOLDS APART, NO LATERAL CONTACT.
6 FOLDS STARTING TO CLOSE, LOWER MARGIN FIRST PROGRESSING ANTERIOR TO POSTERIOR, BUT STILL NO CONTACT.
7 FOLDS MAKING FIRST LATERAL CONTACT ALONG LOWER MARGIN AND AT ANTERIOR.
8 FOLDS COMPLETING CLOSURE WITH INCREASING LATERAL CONTACT.

Figure 2.2 Model of EGG waveform.

attain speaker independent recognition, feature vectors from many different realizations of the same word were combined to form the reference pattern. Both male and females speakers were used in the training phase.

Certain words in the lexicon are pronounced more consistently than others. Thus, for a given first pass representation, the probability of occurrence will differ for the words in an equivalence class. The members of the class are therefore ranked according to their a posteriori probabilities and matched in that order. A likelihood rejection threshold can also be set to further reduce the number of match possibilities. The system is updated (or learns) after a recognition run so that the equivalence classes are kept current with the new relative frequency information.

2.2 Overview of the System

A block diagram of the two-channel, two-pass IWR system outlined in this chapter is given in Figure 2.3. The blocks in the figure represent functions performed and not necessarily stages. That is, the functions depicted by two different blocks may actually be carried out simultaneously in one stage.

The system utilizes concepts from both template based and feature based procedures. Broad acoustic-phonetic information is combined with prosodic cues to identify an acoustically similar subset of words in the vocabulary. Spectral matching techniques are then used for the detailed pattern matching. Since the general characteristics of the words in the class are known, class specific matching can be performed. This entire

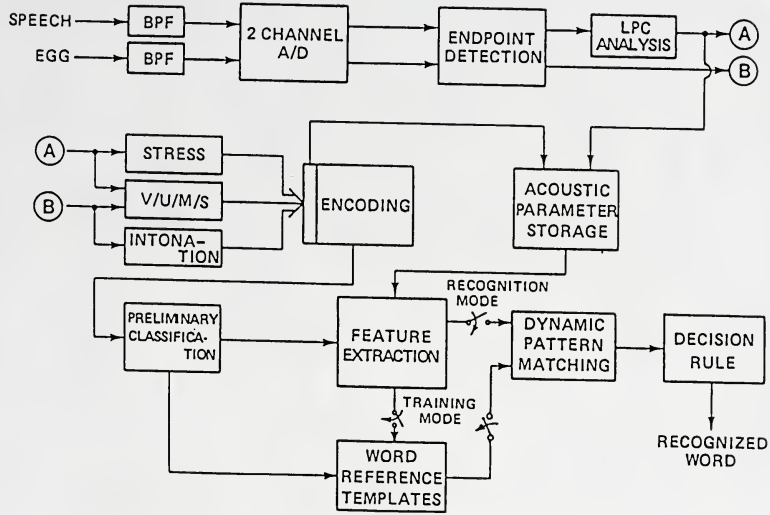


Figure 2.3 Block diagram of IWR system.

approach has been implemented in a modularized fashion so that stages can be readily modified in subsequent studies.

CHAPTER 3
DATA COLLECTION AND PREPROCESSING

The approach taken in this study towards large vocabulary IWR has been set forth in Chapter 2. Key aspects of the system were highlighted and the details were relegated to later chapters. This is the first of such chapters in which specifics of each stage are given.

In both the training and testing modes, the IWR system requires synchronously obtained speech and electroglottographic signals. The procedures for digitizing, aligning, and filtering of these signals are discussed in this chapter. A description of the vocabulary entries used is given first.

3.1 The 100 Word Lexicon

The procedure by which words are chosen for recognizer evaluation is often dictated by research objectives. If simply the best possible recognition results are desired, a method to maximize the dissimilarity between the words of the vocabulary should be used [67,68]. Selection flexibility can also be restricted by vocabulary size.

For this study, words were selected considering phonetic content, syllabic structure, and usefulness in performance assessment. Practical utility was also important, so the criteria above were satisfied with selections from a set of words deemed useful by the research sponsor. The one hundred words chosen are displayed in Table 3.1. The digits 0-9 and the letters of English alphabet comprise a highly complex subset

TABLE 3.1
THE 100 WORDS FOR IWR STUDY

ZERO	ONE	TWO	THREE
FOUR	FIVE	SIX	SEVEN
EIGHT	NINE	A	B
C	D	E	F
G	H	I	J
K	L	M	N
O	P	Q	R
S	T	U	V
W	X	Y	Z
ABORT	AFFIRMATIVE	ALPHA	ALTITUDE
AUTHORIZE	BRAVO	BREAK	BRIEFING
CHARLIE	CHECK	COMMUNICATE	COMPILE
COMPUTER	COORDINATES	DECOY	DEGREES
DELTA	DESTROY	ECHO	EMERGENCY
ENTER	EQUALS	ESTIMATE	EXECUTE
HOLD	HORIZON	HOSTILE	IFF
IN	LANDING	LIGHT	LOAD
LOW	MIKE	MILITARY	MINUS
MINUTES	MISSION	NEGATIVE	NORMAL
OFF	ON	OUT	PASS
PLUS	POSITION	PRINT	RECORD
REJECT	RUN	SCRAMBLE	SELECT
STANDBY	STATUS	STOP	TAKEOFF
TERMINATE	TEST	TIMES	TRANSMIT
URGENCY	VELOCITY	VERIFY	ZEBRA

recognized as one of the most difficult for IWR systems due to the acoustic similarities of the various words. An overall characteristic of the lexicon is that the syllable counts are roughly consistent with a 1250 word frequency weighted excerpt from a 20,000 word dictionary [49]. Monosyllabic words comprised 51% of the vocabulary. The 2, 3, and 4 syllable words accounted for the other 30%, 12%, and 7%, respectively.

3.2 Speech and EGG Digitization

All of the isolated words were produced with the speakers situated inside an Industrial Acoustics Company single wall sound room. The speech was obtained with an Electro Voice RE-10 dynamic cardioid microphone and the EGG signal was from a Fourcin device. Amplification of the speech was accomplished with a Realistic SA-500 amplifier and the EGG signal was boosted with a Sony STR-VX5 amplifier. Both signals were bandlimited with 6 pole, 5 kHz low pass filters. The two channels were alternately sampled at 20 kHz by a Data General Nova 4 A/D system with 12 bits of precision.

The utterances were directly digitized and simultaneously recorded on a Sony TC-FX66 cassette deck. An extender attached to the microphone kept the speaker a fixed distance away. With the microphone and EGG electrodes in place, the speaker would run the data collection program on a terminal inside the sound room. The program prompts the speaker by sounding a bell and presenting the word to be spoken in the center of the screen. Execution of the program suspends until the speaker initiates digitization (by depressing any key) since timing is critical. Immediately after digitization, another bell sounds to indicate termination of the sampling process. An option exists to

repeat the word if it is thought that part of the word may have been missed or spoken abnormally. Although a repeat provision exists for known errors, digitization problems may not become evident until the sampled data file is manually checked. At some later point in time, the recorded speech and EGG can be digitized from tape if the directly digitized data can not be used. The first choice of the directly digitized data avoids any distortion that may be introduced in the recording process [59,69].

In the single word testing mode, recordings were not made since abnormally spoken words were easily repeated. Data collection for automatic testing or reference development involved speaking the entire 100 word vocabulary. For the latter, a second repetition of the vocabulary was also collected. The speakers were instructed to repeat a word if they felt it was rushed, mispronounced, too low, etc. The entire 100 word vocabulary with words repeated as necessary took an average of 24 minutes to collect. The second time through took an average of 20 minutes; more relaxed and natural pronunciations were evident in the second session.

When directly digitizing, exactly 1.28 seconds of speech and EGG were automatically stored on disk. Thus, for each subject, 5.12 megabytes of data were collected per session. This made on-line verification of the data much too time consuming and the recordings provided the most effective means of ensuring a complete data set for each speaker. The bell sounding upon presentation of the word to be spoken was a convenient indicator of a word to follow on the recordings. This facilitated recollection of data from the tapes when determined to be necessary.

3.3 Synchronization of Data

Since the speech and EGG channels were alternately sampled, the resulting file of digitized data had the two signals interleaved. The trivial task of demultiplexing can be performed efficiently after careful consideration of memory limitations, and reassignment and I/O tradeoffs. Once the data were demultiplexed, the speech and EGG were time aligned to account for the acoustic propagation delay from the larynx to the microphone.

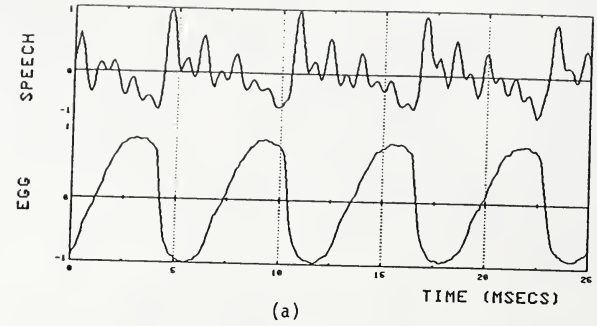
The microphone was kept a fixed 7.62 centimeters (3 inches) away from the speakers' lips to reduce breath noises and to simplify the alignment process. Synchronization of the waveforms had to account for the distance from the vocal folds to the microphone. To do so, an average vocal tract length of 17 cm was assumed. The number of samples to discard from the beginning of the speech record was then

$$\#\ \text{samples} = \text{Int}\ [(24.62/34442)10000 + .5] \quad . \tag{3.1}$$

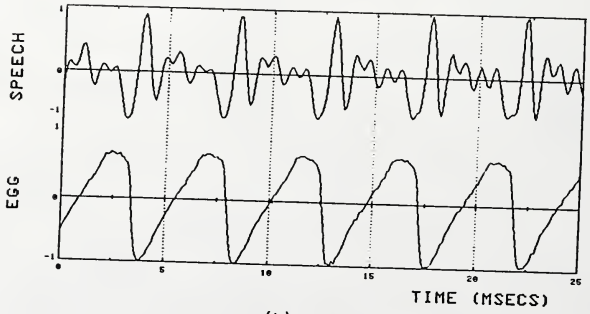
The matter of varying vocal tract lengths among males and females was largely resolved with the 17 cm compromise. Equation (3.1) shows that a 7 sample correction is actually appropriate for tract lengths from 15 to 18 cm long. Examination of the data also supported use of this figure for adult speakers. Examples of aligned speech and EGG signals for a male and female speaker are shown in Figure 3.1.

3.4 Bandpass Filtering

The analog anti-aliasing filters unfortunately had passbands down to 0 Hz. This presented a problem because of audio amplifier d.c.



(a)



(b)

Figure 3.1 Aligned speech and EGG signals for (a) male and (b) female speakers.

offset, 60 Hz contamination, and low frequency trends in the EGG signal. Some high frequency noise also persisted in the EGG signal. These artifacts had to be removed to avoid any loss of accuracy in the acoustic classification stage.

A linear phase, 257 tap, FIR filter was used for both the speech and EGG. The 80 Hz - 4800 Hz bandpass filter was designed using the windowed Fourier series method. The magnitude response characteristics of the filter are shown in Figure 3.2.

Direct implementation of the filtering operation involves convolving the 257 point impulse response of the filter with the 12,800 samples of speech or EGG. A more efficient approach to filtering computes the IDFT of the product of speech and filter DFT's. The overlap and add method [70] is one such approach and was utilized for filtering of the long data records.

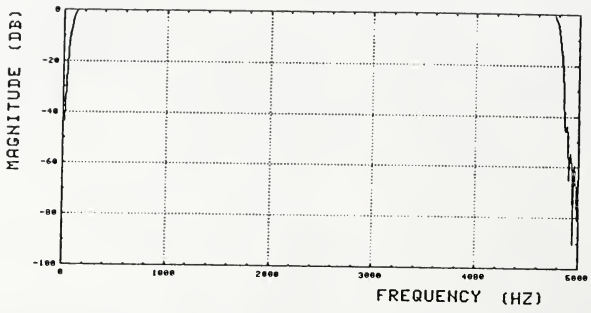


Figure 3.2 Magnitude frequency response of linear phase filter.

CHAPTER 4
ACOUSTIC SEGMENTATION AND ANALYSIS

With the speech and EGG sampled at 10 kHz each, a sizeable amount of data represents every utterance. Fortunately, the fact that the speech signal is highly redundant allows data rate reduction without significant loss of information. The assumption of local stationarity permits a short time spectral characterization useful in the final matching stage.

Initial lexical access requires an even further reduced word representation. This has been achieved with a suitable combination of segmental and suprasegmental features as will be described in the next chapter. The missing intermediate steps of broad phonetic classification and prosodic analysis are discussed in this chapter.

Subsequent processing relies on the word representations extracted from the two channels of data. This chapter describes the data reduction techniques employed. Algorithms are discussed and results are given for procedures amenable to separate testing.

4.1 Endpoint Detection

Associated with each digitized word are 1.28 second records of speech and EGG. Necessary computations can be significantly reduced if the word boundaries are located and the extraneous data discarded. Finding the speech signal imbedded within the background noise (or

silence) must be done carefully to avoid large errors. Incorrect endpoint detection adversely affects the performance of the recognizer.

Since the isolated words were spoken in a sound room, the endpoint detection process is fairly simple. The lack of clicks, pops, and other transients that may occur in telephone transmission systems makes an algorithm such as [71] unnecessary. A fast, efficient algorithm based on energy and zero crossing rates is well suited for this application [72].

Considering only two simply measured parameters, energy and zero crossing rate, the endpoint detection algorithm of [72] effectively estimates the word boundary locations. The algorithm was designed to isolate enough of the word for use in some recognition scheme and not necessarily the entire word. Using this same design objective, accuracy and reliability have been improved by incorporating the EGG signal into the algorithm with only minor modifications.

The glottal sensing characteristics of the EGG were exploited to yield improvements when a word starts or ends with a low energy voiced segment. This also includes the tapering seen at voicing onset and offset. An example of a situation where the EGG would improve results is shown in Figure 4.1. Since the algorithm uses energy measurements for initial endpoint determination, the EGG gives better results due to the AGC circuitry. That is, the glottal vibration associated with a low energy speech signal (voiced) produces an EGG signal of considerably greater energy than that in an unvoiced region. The corresonding low energy speech signal may be mistaken for background noise. The EGG signal is therefore used to locate voicing onset and offset via simple energy measurements. Zero crossing measurements from the speech signal are used to identify initial and trailing unvoiced regions.

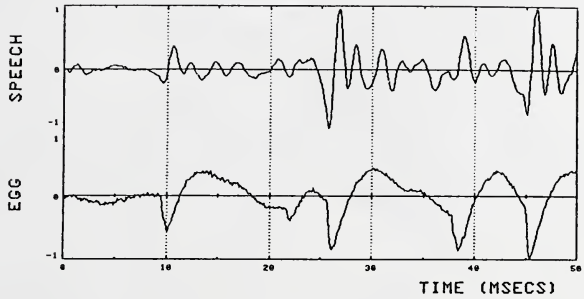


Figure 4.1 Speech and EGG for low energy voiced segment.

An improved algorithm. The modified endpoint detection algorithm gives a first estimate of the word boundaries based on EGG energy levels. Low EGG energy means that the amount of vocal fold contact area is somewhat constant and does not imply that the segment is silent. The possibility remains that a constriction in the vocal tract resulted in unvoiced speech. Thus, speech zero crossing rates are considered in a second step to update the endpoint estimates when preceding or trailing fricatives exist. The flow diagram of the algorithm given in Figure 4.2 depicts the basic functions.

Though the implementation is quite similar to, and the objective is the same, the new procedure differs from that of [72] in an important way. This newer version gives greater attention to detection of initial or final fricatives. Accurate location of voicing onset and offset using the EGG defines more precisely the regions where slightly increased zero crossing rates would be expected. Variations in these regions are then appropriately interpreted. Figure 4.3 illustrates the difference in location of the energy based estimates using the speech (S_1) and EGG (E_1) signals. Also shown are the updated endpoints (S_2) found using the speech zero crossing rates.

To obtain the necessary energy and zero crossing information, the speech and EGG signals are first segmented into 128 10 ms frames. A short frame length was chosen to allow for short duration transients in the classification stage that will be described in Section 4.2. Nonoverlapping frames are used because of dimensional considerations. The energy value computed is actually the short time average magnitude of the measurement interval. For frame n,

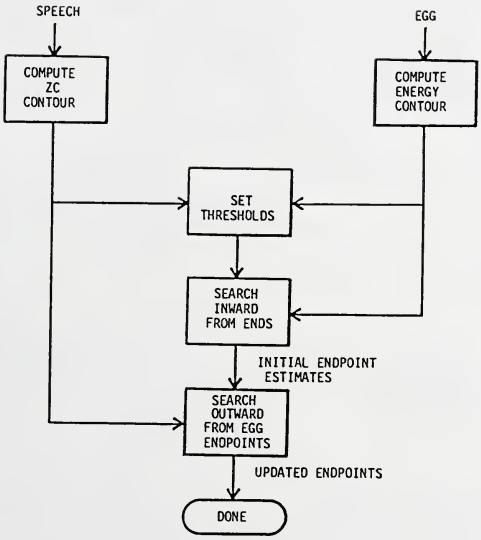


Figure 4.2 Flow diagram for endpoint detection algorithm [72].

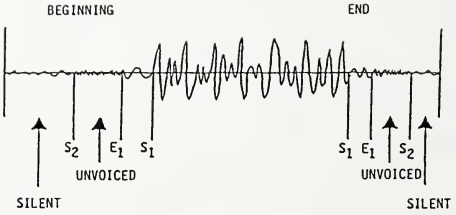


Figure 4.3 Example of speech (S1,S2) and EGG (E1) based endpoints.

$$ENG(n) = \sum_{i=1}^{100} |x((n-1)100 + i)| \quad , \tag{4.1}$$

where x(i) represents samples of the speech or EGG signal. The zero crossing rate, ZCR(n), is defined to be the number of positive zero level crossings in the 100 sample frame. The energy and zero crossing contours are smoothed slightly with a zero phase filter. The three point filter with coefficients {.19,.62,.19} preserves most of the variations in the original contour.

Finding endpoints requires decisions based on relative comparisons of the energy and zero crossing measurements. The comparisons are made with respect to thresholds derived from silent region information. It is assumed that 10 frames of silence exist at either the beginning or end of the data record. For both regions, then, the mean and standard deviation of the energy ($\overline{ENG}$ and SIGENG), and zero crossing ($\overline{ZCR}$ and SIGZCR) contours, are computed. Statistics for the first interval are used unless any of the following conditions hold:

Speech

a) $\overline{ENG2} < (.95)\overline{ENG1}$ and $\overline{ZCR2} < (.95)\overline{ZCR1}$ (4.2.a)

b) $\overline{ENG2} < (1.1)\overline{ENG1}$ and $\overline{ZCR2} < (.90)\overline{ZCR1}$ (4.2.b)

c) $\overline{ENG2} < (.90)\overline{ENG1}$ and $\overline{ZCR2} < (1.1)\overline{ZCR1}$ (4.2.c)

d) $\overline{ENG2} < (1.1)\overline{ENG1}$ and SIGZCR2 < (.90)SIGZCR1 (4.2.d)

EGG

e) $\overline{ENG2} < (.95)\overline{ENG1}$. (4.2.e)

If any part of Eq. (4.2) is satisfied, the interval at the end of the record is used. The conditions above assure that the interval with the

smaller amount of speech-like activity is used to represent the background silence. In all but a few instances, the first interval was used due to the delay between the start of digitization and the time the word was spoken.

Once the silence statistics are known, several speech zero crossing thresholds and EGG energy thresholds can be computed. The first zero crossing threshold is found as

$$ZCT1 = \begin{cases} \max\{Z1,\ 13.8\} & ;\ ZCRMAX > 15.0 \\ \max\{Z1,\ ZCRMAX\text{-}1,\ 13.0\} & ;\ ZCRMAX < 15.0 \end{cases} \qquad (4.3.a)$$

where

$$Z1 = \max\{Z2,\ Z3\} \qquad (4.3.b)$$

$$Z2 = \min\{25.0,\ \overline{ZCR} + (2.10)SIGZCR\} \qquad (4.3.c)$$

$$Z3 = .27(ZCRMAX - \overline{ZCR}) + \overline{ZCR}\ . \qquad (4.3.d)$$

The other threshold is at the upper end of the amplitude distribution,

$$ZCT2 = \overline{ZCR} + (3.0)SIGZCR\ . \qquad (4.4)$$

Since the zero crossing rate is a rough indicator of spectral content for speech signals, the thresholds of Eqs. (4.3) and (4.4) set frequency cutoffs below which fricatives will not be detected. Equation (4.3.a) shows that if the highest frequency content of the word (ZCRMAX) is below 1500 Hz, the low end cutoff will be at 1300 Hz. The threshold, ZCT1, will assume a value depending on the peak zero crossing rate over all frames and the silent region statistics.

The lower EGG energy threshold is

$$ENT1 = \min\{E1, E2\} \tag{4.5.a}$$

where

$$E1 = .10(ENGMAX - \overline{ENG}) + \overline{ENG} \tag{4.5.b}$$

$$E2 = (4.0)\overline{ENG} \tag{4.5.c}$$

and the upper energy threshold is

$$ENT2 = (5.0)(\min\{E2, E3\}) \tag{4.6.a}$$

where

$$E3 = .03(ENGMAX - \overline{ENG}) + \overline{ENG} \quad . \tag{4.6.b}$$

The thresholds of Eqs. (4.5) and (4.6) are either a percentage of the adjusted peak energy or a multiple of the average silence energy level.

Accurate estimates of voicing onset and offset are found from the EGG energy contour by searching inward from the first and last frames. If the energy exceeds ENT1 and successive frames rise above ENT2 before falling below the lower threshold, then the first frame exceeding ENT1 is preliminarily labeled as an endpoint. This is how the endpoints E_1 in Figure 4.3 are found.

The first set of endpoints is updated by searching outward from their present position in the speech zero crossing contour. Twenty-five frames are examined at each end and a record is kept of the number of times ZCTĪ is exceeded. If ZCT1 is exceeded in three or more frames, the farthest point in time away from the initial endpoint estimate is used as the update. When the lower threshold is exceeded only one or

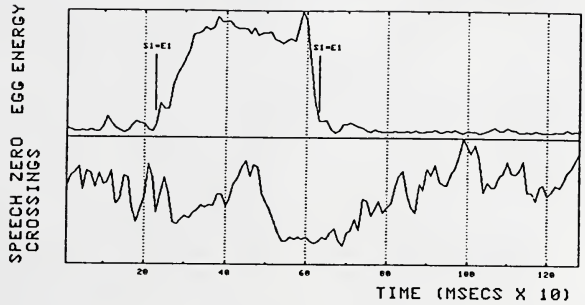
two times, the endpoint is updated if the zero crossing rate was greater than min{17.5, ZCT2}. This improves performance for weak and/or short duration leading or trailing fricatives.

Some examples of automatically detected endpoints are shown in Figure 4.4. The initial estimates are labeled E_1 and the updated, or final, endpoints are at S_1. For the all voiced word in part (a), $\{S_1\} = \{E_1\}$ as expected. Part (b) illustrates zero crossing extensions.

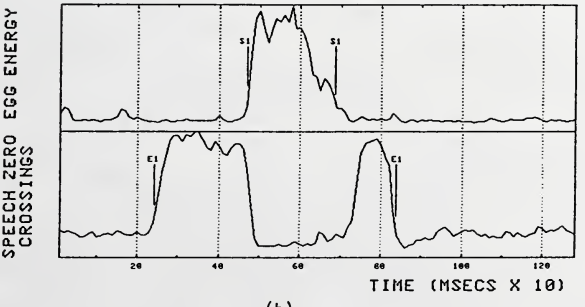
4.2 V/U/M/S Classification

Having located the boundaries of the utterance surrounded by silence, the next objective is to classify all frames between the endpoints as either voiced (V), unvoiced (U), mixed (M), or silent (S). This four way classification is an extension of the basic V/U/S problem in speech analysis-synthesis schemes. In the commonly used source-filter model of speech production, the excitation is assumed to be either a quasiperiodic sequence for voiced sounds, or a random, noise-like source for unvoiced sounds. The classification requirements then reduce to just a V/U decision. The U/S distinction is taken care of during synthesis since the gain factor for a silent region is much smaller than that of an unvoiced region.

Even for the V/U distinction, standard speech based schemes have problems because of the large frame size used, low levels of voicing, or the strength of the first formant. Classification difficulty increases when the U/S decision is also needed. Additional measured features (energy, zero crossing rate, etc.) have been used in a statistical approach for high quality speech with fairly reliable results [73].



(a)



(b)

Figure 4.4 Automatically detected endpoints for the words (a) "one" and (b) "six."

Other parametric techniques have also been shown to perform satisfactorily [74]. Consideration of telephone quality speech has resulted in use of an optimized set of parameters [75], and an approach utilizing all of the information in the speech signal [76].

The four way, V/U/M/S, decision is usually not attempted due to the difficulty in making the mixed designation from the speech signal alone. Sophisticated pattern recognition techniques used for V/U/M classification achieved 95% accuracy overall, but under 83% of the mixed frames were assigned correctly [77]. Interest in making a mixed classification continues because of applications in speech synthesis and recognition. Higher quality, natural sounding synthetic speech requires use of a mixed excitation for voiced fricatives [60]. The procedures for lexical access outlined in Chapter 2 use the mixed classification as an additional discriminator to distinguish between word classes.

A requirement of the features used for initial word classification was that robust extraction from the speech signal must be possible. Use of the electroglottograph as an independent measure of vocal fold activity makes V/U/M/S classification no more difficult than V/U/S classification. Section 4.1 showed that the V/US decision is performed reliably when the EGG is used. The U/S decision, however, is not aided by the EGG since both categories correspond to the absence of glottal vibration.

For just a V/US decision, thresholding of the EGG signal suffices since the EGG is ideally zero during nonvoiced regions and periodic and nonzero during voiced regions. The U/S and V/M decisions require a somewhat greater degree of sophistication. Both decisions have the common task of detecting the presence of aperiodic components in

speech. As was the case for endpoint detection, this is not very difficult since data collection took place in a reasonably high signal-to-noise ratio environment. An accurate, hybrid speech-EGG algorithm for V/U/M/S classification was implemented and will be described in Section 4.2.2. Procedures investigated for improving the reliability of the U/S decision are presented next.

4.2.1 U/S Considerations

Several easily computed features of the EGG and speech signals can be used to perform accurate V/U/M/S classification of the utterance. The few remaining errors that occur are usually due to U/S confusions. In an attempt to achieve nearly perfect V/U/M/S classification, auxiliary spectral based procedures were considered.

4.2.1.1 Two pole analysis

The first technique uses a second order LPC analysis to represent the gross characteristics of the short-time spectrum [78]. For a two pole analysis, one complex conjugate pair or two real poles will be located in the region of greatest spectral energy concentration. The normalized error from the analysis increases with the spread of spectral energy. A two pole analysis was used along with other measurements to make a six way classification in [31]. With regard to U/S discrimination, the idea was to obtain an independent indicator to favor a decision for one of the classes.

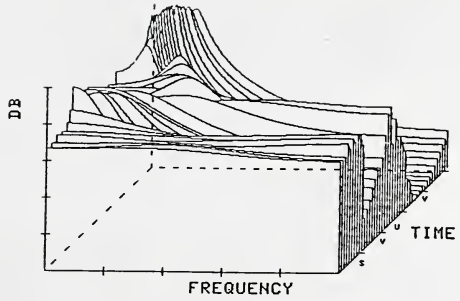
Experimentation showed that for many unvoiced segments the pole location was not too meaningful since the bandwidths were very large. The silent regions tended to have high pole frequencies and large bandwidths. A significant amount of unvoiced frames had similar

characteristics. The normalized error was comparable in both regions. Figure 4.5 shows spectra for voiced, unvoiced, and silent regions from the word "echo." Part (a) gives a 3-D time sequence of the spectra starting just before voice onset. Typical voiced and unvoiced spectra are shown in part (b) along with an unvoiced frame from just after the plosive release. Results such as those in Figure 4.5, and the fact that this type of information was not definitive when the zero crossing based decisions were in error, made this approach unattractive for the stated purpose.

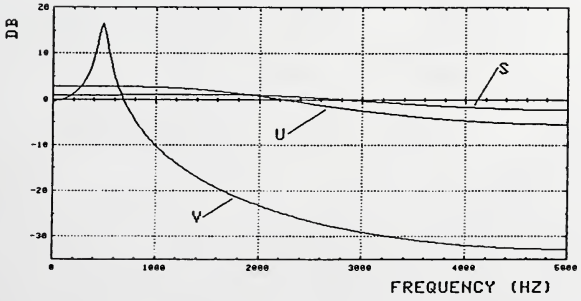
4.2.1.2 Spectral classification procedure

A problem with the use of the second order model above was that the unvoiced and silent frames had globally similar spectra. Increasing the order of the analysis provides more spectral detail and perhaps better class discrimination. The procedure suggested in [76] uses an averaged spectral representation for voiced, unvoiced, and silent classes and an LPC distance to measure similarity between the test and reference frames. An energy distance is nonlinearly combined with the LPC distance to make the final class decision.

The V/U/S classifier was implemented as described in [76], except a 12 pole autocorrelation analysis was used and the frame size was increased to 200 samples. Since the objective was to aid the U/S decision, the algorithm was trained with special emphasis given to the hard to detect weak fricatives for the unvoiced class. This helped increase the dissimilarity between the silent and unvoiced spectra. The average spectra computed for the three classes are shown in Figure 4.6(a). An example of V/U/S classification results for the word "two" is given in part (b). This plot was typical in that the algorithm

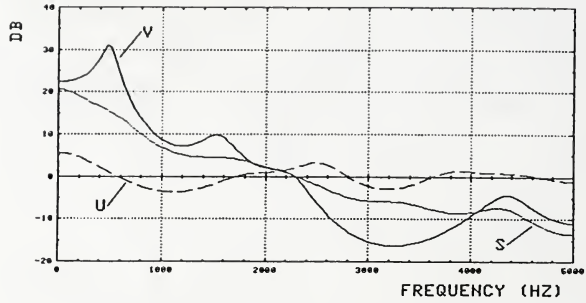


(a)

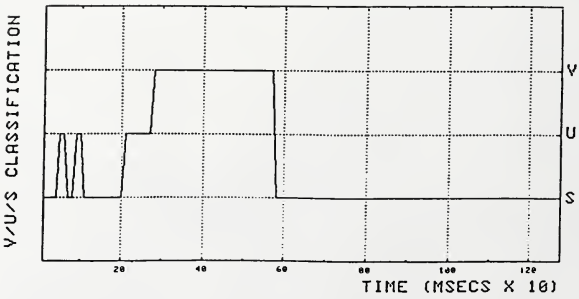


(b)

Figure 4.5 LPC spectra for voiced (V), unvoiced (U), and silent (S) regions in the word "echo."



(a)



(b)

Figure 4.6 (a) Averaged LPC spectra and (b) classification results for the word "two."

generally performed well, but most of the errors were again due to U/S confusions. Thus, the increased order spectral representation did not exhibit the potential to improve the U/S performance of the algorithm described in the next section either.

4.2.2 Speech-EGG Based Algorithm

An algorithm using both the speech and EGG signals for voiced, unvoiced, mixed, and silent classifications is presented in this section. In addition to simplicity, another major feature of the method is accuracy. The EGG is used to advantage as a glottal sensor in the same way as for endpoint detection. In fact, the two algorithms were combined and the same thresholds used.

The endpoint detection algorithm produced two sets of endpoints; the first set from the EGG energy contour gave the bounds on the voiced region(s), and the updated set from the speech zero crossing contour indicated the presence of leading or trailing fricatives. Without any further consideration, the regions outside of the outer pair of endpoints are designated as silent frames. The parts of the word between the two sets of endpoints can next be labeled as unvoiced frames. If the two sets of endpoints are the same at the beginning or end, the word will have no leading or trailing fricative, respectively. At either end of the word, a silent region between fricative frames is considered as part of a single unvoiced segment. For example, the beginning of the word stop will have only unvoiced frames preceding the voiced region. The initial assignments are depicted by the unvoiced and silent regions in Figure 4.3.

The V/U/M/S algorithm uses the speech zero crossing contour, the EGG energy contour, and the speech energy contour, SENG(n), which was not used by the endpoint detection procedure. The thresholds used for classification include ZCT1 and ENT1 from endpoint detection and new speech energy thresholds defined as

$$ENT3 = \min\{E4, E5\} \tag{4.7.a}$$

where

$$E4 = .03(SENGMAX - \overline{SENG}) + \overline{SENG} \tag{4.7.b}$$

$$E5 = (4.0)\overline{SENG} \tag{4.7.c}$$

and

$$ENT4 = \overline{SENG} + (3.0)SSIGENG \quad . \tag{4.8}$$

The threshold ENT3 is either 3% of the peak speech energy adjusted for the background noise level or it is set to four times the average silent region energy. If the speech energy is assumed to have an approximately Gaussian distribution, silence energy measurements would not exceed the threshold ENT4. For any distribution, roughly 90% of the measurements will be less than this threshold.

The V/U/M/S classification procedure is illustrated in Figure 4.7. After the outer silent and unvoiced frames are assigned, all of the frames between the inner set of endpoints are classified. A frame is designated as silent if both the speech and EGG have low energy and the speech has a low zero crossing rate. During silence, the vocal folds do not exhibit any regular vibration; so the EGG energy is not expected to deviate much from the mean computed from the 10 frame silent

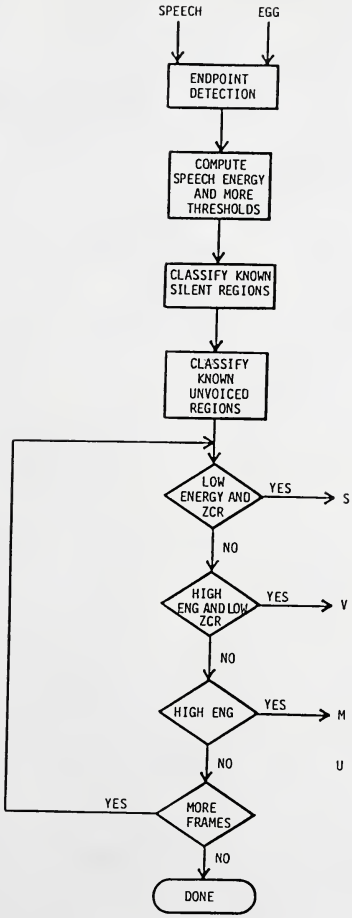


Figure 4.7 Flow diagram for V/U/M/S classifications.

interval. However, if the speech energy is between the upper and lower thresholds and was low in the preceding frame, the present frame is classified as silent provided that the EGG energy and speech zero crossing rate remained low. Thus, a frame n is silent if

$$SENG(n) < ENT3 \text{ and } ENG(n) < ENT1 \text{ and } ZCR(n) < ZCT1 \quad , \qquad (4.9.a)$$

or

$$SENG(n-1) < ENT3 \text{ and } SENG(n) < ENT4 \text{ and } ENG(n) < ENT1 \text{ and } ZCR(n) < ZCT1 \quad . \qquad (4.9.b)$$

If not classified as silent, a frame is then checked to see if the EGG energy exceeds the lower threshold while the speech zero crossing rate remains low. The frame is classified as voiced if

$$ENG(n) > ENT1 \text{ and } ZCR(n) < ZCT1 \quad . \qquad (4.10)$$

At this point, a frame can have high EGG energy only if it also had a high speech zero crossing rate. This means that the vocal tract was excited not only by vocal fold vibrations but also by turbulence from a constriction at some point along the tract. The frame would then be classified as mixed if

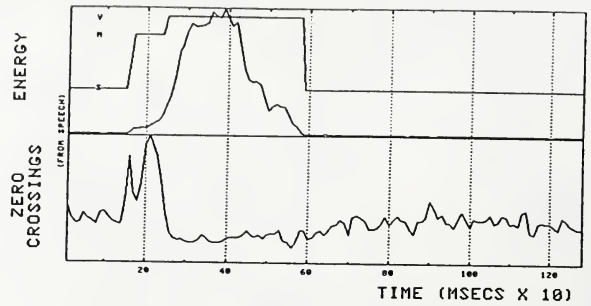
$$ENG(n) > ENT1 \quad . \qquad (4.11)$$

After attempting classification as either silent, voiced, or mixed, a frame that does not meet any of the criteria above is said to be unvoiced.

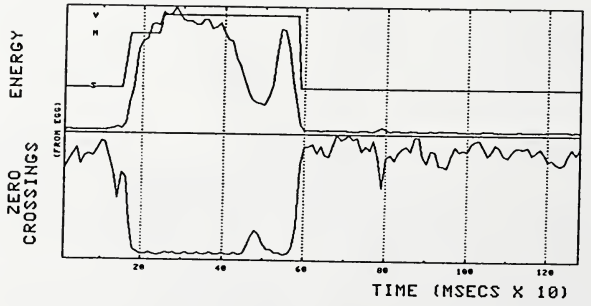
Use of the algorithm as described so far indicated that when a classification error occurred, it was typically an isolated event. That is, single frames classified different from adjacent frames were by far the most common type of error. The situation was rectified by use of a smoothing filter. The filtering operation was implemented based on a set of correction rules. Knowledge obtained from observation of the different types of single frame errors was used to derive the various rules. A frame classified different from both of its neighbors was changed depending on the original assignments of the three successive frames. Single voiced frames were changed to mixed, unvoiced, or silent in this order if either neighbor was classified as such. Similarly, unvoiced frames were converted to M, S, or V, mixed frames to U, V, or S, and silent frames to U, M, or V. Only the V/U/M/S results between the inner set of endpoints were smoothed to allow single unvoiced frames at the beginning and end of a word.

Classification results. Examples of V/U/M/S classification results are shown in Figures 4.8 and 4.9. In both figures, part (a) shows the results with speech derived contours and part (b) shows the same results with EGG derived contours. The accuracy of the V/U/M/S algorithm was judged with respect to a manually determined reference obtained from a visual inspection of the original speech and EGG waveforms, the corresponding energy contours, and of the respective zero crossing contours. The algorithm's decision was correct if it agreed with the manual classification.

All frames of the words thirteen, seven, zero, ten, five, and twelve were classified manually and compared with the automated classifications. Results from these comparisons are presented in Table 4.1.

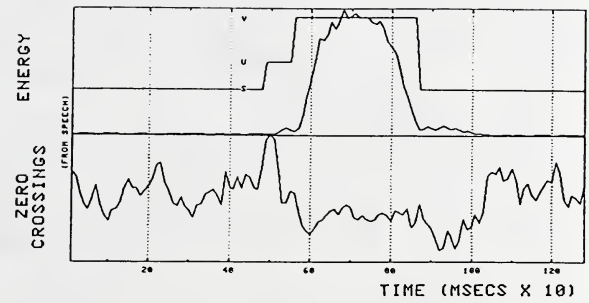


(a)

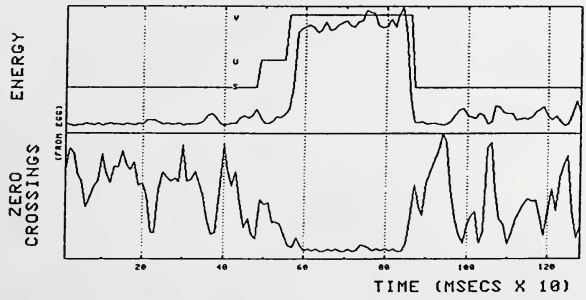


(b)

Figure 4.8 V/U/M/S classification results for the word "zero."



(a)



(b)

Figure 4.9 V/U/M/S classification results for the word "four."

TABLE 4.1
COMPARISON OF MANUAL AND
AUTOMATIC V/U/M/S CLASSIFICATION

WORD	SS	SU	SM	SV	US	UU	UM	UV	MS	MU	MM	MV	VS	VU	VM	VV
THIRTEEN	83				1	3										41
SEVEN	72	8				14	1						1			32
ZERO	85					1				7	1					34
TEN	86	5				3										34
FIVE	83	8				4										33
TWELVE	72	10				2										40
TOTALS	481	31			1	27	1			11	1		1			214
%	93.9	6.1			3.4	93.1	3.45			91.7	8.3		.5	3		99.5

OVERALL 733 correct, 35 errors
% 95.45 4.55

An SS entry denotes a manually determined silent frame classified as silent by the algorithm, SU is a silent frame classified as unvoiced, etc. For the set of words used to test the algorithm, less than 5% error resulted overall. This figure, however, is biased towards performance for the larger classes. The large number of SU errors were found mostly at the beginning or end of the word where silent frames between frication and voicing onset or offset were deliberately classified as unvoiced. Refiguring overall performance without these errors gives a 99.45% correct classification rate.

4.3 Fundamental Frequency Estimation

Interest in monitoring the suprasegmental cue of pitch translates into the task of tracking its primary acoustic correlate, the fundamental frequency of voiced speech. This is not easily accomplished and has therefore resulted in a myriad of schemes. The various approaches have been categorized as either time domain or short term analysis techniques and are surveyed in [79]. An evaluation of seven different pitch detectors is given in [80]. Some difficulties associated with the methods tested include the lack of perfect glottal periodicity, source-tract interaction, and talker dependence.

Pitch tracking is greatly simplified when the EGG signal is considered. In Section 2.1.1, the ideal EGG signal was described as periodic with exactly two zero level crossings per cycle. The periodicity of the EGG signal is precisely that of vocal fold vibration (and thus vocal tract excitation). A cycle by cycle measure of pitch is directly available as the elapsed time between two successive invariant features of the EGG waveform. Pitch tracking with the aid of the EGG

signal virtually eliminates all of the previously encountered problems with the time domain methods.

Several new EGG based pitch tracking schemes have recently been reported in [59] and [60]. These methods utilize the robustly detected prominent features of the EGG signal. Aside from the details of implementation, the basic differences between these new algorithms are whether the pitch is computed per frame or for each period, and which EGG feature is used. Usually, the pitch value is found from the EGG zero crossings or the distance between minima in the differentiated EGG. A new algorithm that uses both features to compute an average pitch per frame is described next.

EGG based procedure. The algorithm described here determines the average pitch frequency of a frame known to be mixed or voiced by processing the EGG data. Since the pitch frequency is found for each frame and not each pitch period, data outside the frame of interest are required to allow tracking for a reasonable frequency range. Required inputs are therefore 3 frames of EGG data centered about the frame of interest and the size of the frame used. With a sampling rate of 10kHz and a 100 point frame size, the lowest detectable pitch frequency is about 67 Hz. The lower bound is given by

$$FO_{min} = [(.5)(3.0)(\text{frame size})(T_s)]^{-1} \tag{4.12}$$

where T_s is the sampling period. The highest detectable pitch frequency is approximately 667 Hz, and is set by

$$FO_{max} = [(15.0)T_s]^{-1} \quad . \tag{4.13}$$

The bounds given in Eqs. (4.12) and (4.13) were imposed largely by heuristics and were found to be adequate for all subjects.

The pitch detection algorithm returns two values for F0; one based on zero crossing measurements and one based on the minima of the differentiated EGG signal. A simple yet effective way of monitoring the fundamental frequency of a voiced speech segment is to count zero crossings of the EGG. The EGG is already zero mean due to the preprocessing described in Chapter 3 and the signal is quite regular in nature. Both of these factors facilitate the detection of zero level transitions. By differentiating the EGG signal, sharp peaks are produced at the break in the EGG closing phase. The greatest negative slope occurs at this point in each cycle, thus yielding another reliable F0 indicator.

Employing the two pitch related parameters mentioned above, the tracking algorithm proceeds as shown in Figure 4.10. The reasoning behind using both the EGG zero crossings and the minima of the differentiated EGG is as follows. The positive zero level crossing corresponds only approximately to the separation of the folds during the opening phase of vibration. The exact point of the zero crossing depends on such things as the time constant of the electronics, any local mean removal, and the distortion characteristics of the tape recorder used. The differentiated EGG minima, on the other hand, almost always correspond to a particular physical event. The slope of the EGG has its greatest negative value at the point of first contact when the vocal folds are closing. This event is relatively unaffected by the type of problems the zero crossing locations are susceptable to.

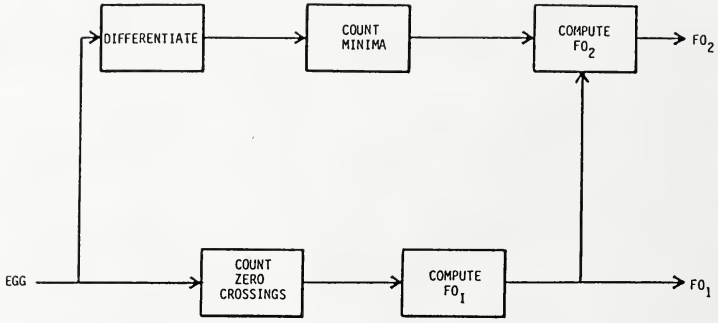


Figure 4.10 Flow diagram for pitch estimation using the EGG.

The blocks of Figure 4.10 depict the four basic components of the pitch detection algorithm; zero crossing counting, EGG differentiation, minima counting, and pitch computation. By accomodating a collection of special cases, the original procedures for an ideal EGG signal evolved into a fairly general algorithm applicable to any input.

Locating EGG zero crossings was only slightly hampered by the noise on the EGG signal and by the few incorrectly classified nonvoiced regions. A zero crossing, ZC, is said to have occurred at sample n when the conditions below are satisfied:

$$ZC(i) = n \quad , \quad i=1,\ldots,NPER+1$$

if and only if

$$\text{a)} \quad EGG(n) \geq 0 \text{ and } EGG(n-1) < 0; \tag{4.14.a}$$

$$\text{b)} \quad \max\{EGG(n-2), EGG(n-3), EGG(n-4)\} < 0; \tag{4.14.b}$$

$$\text{c)} \quad \sum_{j=n+1}^{n+6} (\text{sign}\{EGG(j)\}*1+1)/2 > 4; \text{ and} \tag{4.14.c}$$

$$\text{d)} \quad ZC(i) - ZC(i-1) > 15 \quad . \tag{4.14.d}$$

Before other factors are considered, condition (4.14.a) must hold. Once it is known that adjacent points represent a positive zero level transition, a check is made to see if the three points preceding the non-negative value are less than zero. This condition, (4.14.b), and (4.14.c) which requires at least four of the six samples following the transition to be positive, allow for some noise in the EGG in the

transition region. Then, lastly, it is required that at least 15 samples separate successive zero crossings, thus allowing a maximum pitch frequency as given in Eq. (4.13), namely 667 Hz.

Differentiation of the EGG signal would normally be accomplished using the first back difference approximation for the continuous operator s. This would give

$$DEGG(n) = EGG(n) - EGG(n-1) \quad . \tag{4.15}$$

Use of the relation in (4.15) did not prove very effective when there was noise on the EGG and when the negative peaks were not very prominent. A smoothing differencer of the form

$$DEGG(n) = EGG(n+1) + EGG(n) - EGG(n-1) - EGG(n-2) \tag{4.16}$$

or,

$$H(z) = \frac{DEGG(Z)}{EGG(Z)}$$

$$= Z + 1 - Z^{-1} - Z^{-2} \tag{4.17}$$

was considerably more effective and thus adopted for use.

Before finding the minima of the differentiated EGG signal, DEGG, a first estimate for the pitch frequency, is computed from the EGG zero crossings. If the total number of pitch periods, NPER, is zero, the pitch frequency is set to zero. Otherwise, the following quantities are computed:

a) starting period,

$$SP = \text{Int}[\frac{NPER}{3}] + 1 \quad ; \tag{4.18.a}$$

b) number of periods of interest,

$$NPI = (NPER - 2)\ \text{Int}[\frac{NPER}{3}] \quad ; \tag{4.18.b}$$

c) average number of points per period,

$$ANP = \frac{1}{NPI} \sum_{i=SP}^{SP+NPI-1} [ZC(i+1) - ZC(i)] \quad . \tag{4.18.c}$$

Finally, the possibly averaged pitch frequency for the frame of interest is

$$FO_1 = 10000/ANP \tag{4.19}$$

assuming a sampling frequency of 10KHz.

As mentioned earlier, the pitch frequency is computed for each frame which may or may not contain one pitch period as defined by the algorithm. For this reason, zero crossings and DEGG minima are located within three frames of data centered about the frame of interest. Equations (4.18.a)-(4.18.c) show that the algorithm tries to compute pitch from the inner pitch periods surrounding the frame of interest. Equations (4.18.a) and (4.18.b) establish which periods will be used in the computation and indicate that for anything other than exactly one or three pitch periods in the three frames, the resulting pitch frequency will be an averaged value.

Now, using the mean, $\overline{DEGG}$, and the smallest value of the differentiated EGG signal, $DEGG_{min}$, a threshold, T, is set as

$$T = (.20)(DEGG_{min} - \overline{DEGG}) + \overline{DEGG} \ . \tag{4.20}$$

This threshold is at a value that is one fifth of the way between the mean and the smallest value in the three frames of differentiated EGG data. Another threshold,

$$MP = \max \{15, ANP/5\} \ , \tag{4.21}$$

is the maximum number of points allowed for the search window in which a local minima is to be found. The quantity ANP/5 is 40% of half the number of points in the average pitch period as determined from the EGG data for the same frame in equation (4.18.c). This number was determined based on observations of the maximum rates at which the pitch period changes from cycle to cycle. The smallest search window would be for MP equal to 15. This corresponds to the maximum detectable frequency of 667Hz.

A search window starts at an n for which DEGG(n) is less than T and continues on for MP samples. Counts KW and KT are kept of the number of points in the present window, and of the number of points below the threshold T within that window, respectively. A sample is said to be a local minima when the following conditions hold:

$$MINLOC(i) = n \quad , \quad i=1,\ldots,NPER+1$$

if and only if

a) $DEGG(n) < T$ (4.22.a)

b) $DEGG(n) = \min\{DEGG(n)\}$, n within the window (4.22.b)

c) $KW < MP$ (4.22.c)

d) $KT > 3$ (4.22.d)

e) $$\left[\frac{\sum_{j=n+4}^{J+n+4} (\text{sign}\{DEGG(j)\}*1+1)/2}{J+1} \right] > .65 \qquad (4.22.e)$$

where $J = \max\left\{\frac{ANP}{8}, 4\right\}$.

The conditions of (4.22) say that in order for a sample to be designated a local minima it must be less than the threshold T and smaller than any other point in the present window. Also, there must be at least 3 points in the window that fall below the threshold. The last requirement is that two thirds of some number of points following the minima are positive. The number of positive values required ranges from about one eighth to one fourth of the pitch period and is based on the number of points in an average period as determined from the EGG zero crossings. This criterion stems from the observation that the slope of the EGG always becomes positive a short time after the break in the closing phase.

A step necessary to assure that the minima found above are free of window size errors entails checking the spacing of the designated samples. When a minima occurs at a window boundary, it is likely that the surrounding points will have the smallest values in the adjacent

frames. This problem is overcome by finding one minima from each close pair. If

$$\text{MINLOC}(i+1) - \text{MINLOC}(i) < \text{MP} \quad , \tag{4.23}$$

then one minima is found as

$$\min\{\text{DEGG}(n)\}, \text{ for } \text{MINLOC}(i) - \text{MP}/2 \leq n \leq \text{MINLOC}(i+1) + \text{MP}/2 \tag{4.24}$$

and then MINLOC(i) = n. All other minima previously found are shifted over one location and all adjacent pairs are checked again. When this procedure is finished, NPER is updated and Eqs. (4.18) and (4.19) are used to compute the second estimate of the pitch frequency, FO_2, using MINLOC instead of ZC.

The algorithm yields FO_1, based on the EGG zero crossing rate, and FO_2, based on the number of minima in the differentiated EGG signal. The value used as the pitch frequency for the given frame is determined by the calling routine. Since FO_2 is thought to be the most accurate of the two values computed, an attempt is made to favor FO_2 for most situations. The final assignment is made according to

$$\text{PITCHFREQ}(i) = \begin{cases} FO_1; \text{ if } FO_2 = 0, \text{ or} \\ \quad \text{if } \dfrac{|FO_2 - FO_1|}{\min\{FO_1, FO_2\}} > .13 \\ \quad \text{and } |FO_1 - \text{PITCHFREQ}(i-1)| \\ \quad\quad < |FO_2 - \text{PITCHFREQ}(i-1)| \\ FO_2; \text{ otherwise} \quad . \end{cases} \tag{4.25}$$

This says that the only cases where FO_1 is used are where FO_2 is 0 Hz, or when FO_1 and FO_2 differ by more than 13% of the smaller value and FO_1 is closer to the value taken as the pitch frequency for the previous frame.

Examples of the performance of the EGG based pitch tracker are given in Figure 4.11. Pitch contours are shown for the words "one," "seven," and "emergency." These results were verified on a frame by frame basis so are known to be quite accurate. To gauge the improvement in accuracy over speech based techniques, results were compared with pitch contours from algorithms similar to those of [81] and [82]. The example shown in Figure 4.12.a compares the EGG method with results from a speech autocorrelation procedure. In Figure 4.12.b, the comparison is with a contour derived from the LPC error signal.

The plots in Figure 4.12 illustrate improved results for the word "thirteen." In many all voiced words, the results from the three methods were nearly identical except for discrepencies in location of voicing onset and offset with the speech based methods. Unvoiced regions often introduced classification errors which resulted in yet more problems for the speech based methods. The EGG based algorithm consistently performed best across all speakers and words.

4.4 LPC Analysis

The spectral representation used for detailed word matching is the LPC feature set corresponding to the model of Eq. (1.4). The way the parameters of the model are obtained is discussed briefly in Appendix A. The autocorrelation method was used for stability and computational considerations. Analysis conditions represented by the various blocks

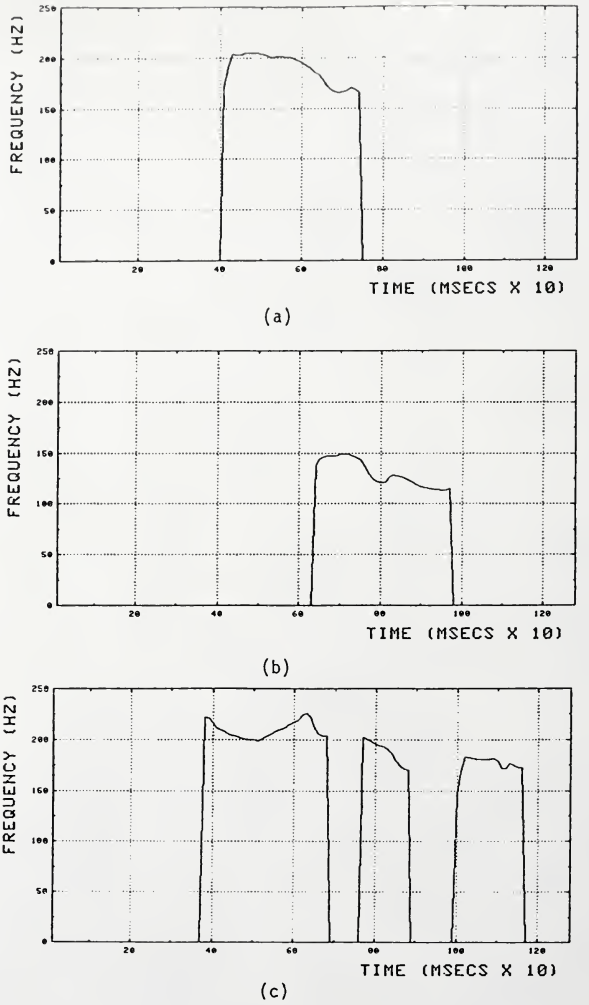
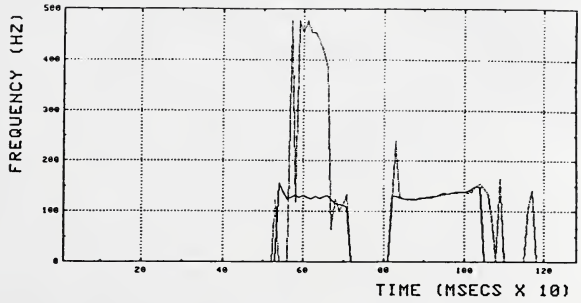
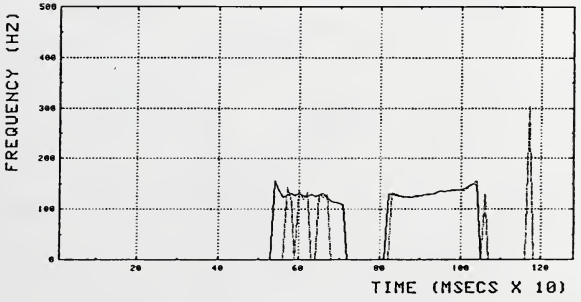


Figure 4.11 Pitch tracking results for the words (a) "one," (b) "seven," and (c) "emergency."



(a)



(b)

Figure 4.12 Comparison of an EGG derived pitch contour (solid line) with results from (a) a speech autocorrelation procedure and (b) an LPC residual based technique.

of Figure 1.3 were chosen based on the results of several studies [83-85].

The effect of analysis order on recognition performance was found to be quite pronounced in [85]. This result stems from the fact that two poles should be allotted for each resonance in the system to be modelled. For the 4.8 kHz bandwidth used in this study, four to six formants were expected. Thus, a 14 pole analysis was used to also allow for source contributions.

Analysis frames were N samples long and spaced M samples apart. The size of the analysis frame is less important than the shift rate for recognition accuracy [85]. Halving the number of computations by doubling the shift rate increased the error by 4%. A rate of 67 frames per second was desirable for the 6.67 kHz sampling frequency used in that study. For the 10 kHz sampling frequency used here, a shift rate of 100 frames per second (M = 100 samples) was chosen. The frame size, N, was set to 200 samples and each frame was centered about the 100 sample frame employed in the preceding sections.

Each frame was preemphasized and Hanning windowed prior to analysis. The first order digital system,

$$H(z) = 1 - az^{-1} \quad , \tag{4.26}$$

was used for preemphasis with a = .95. For a model order of 14, preemphasis does not have much affect on the average recognition error rate. It is important, however, for numerical stability in the LPC analysis [86].

The theoretical stability guaranteed by the autocorrelation method is contingent upon adequate computational precision. For this reason, the LPC coefficients were computed using double precision arithmetic. An analysis was performed on frames corresponding to the isolated word. Frames were analyzed sequentially from two frames preceding on through the second frame following the word boundaries found by the endpoint detector. The two frame safety margin at each end provides additional flexibility with regard to endpoint constraints for pattern matching. This will be discussed further in Chapter 6. Rarely in these or other silent regions of the word, would computational difficulties arise because of an underdetermined set of normal equations. The problem was usually avoided due to amplification of background noise. Any time it did occur, or if the frame was unstable for some other reason, the next stable frame would be repeated an appropriate number of times.

4.5 Discussion

The EGG zero crossing contours in Figures 4.8 and 4.9 clearly convey information regarding voiced regions. In a voiced frame, the EGG zero crossing rate is much lower than when voicing is not present. The nonvoiced segments have a high zero crossing rate due to the internal electronics of the EGG. Since the same voicing information is available in the EGG energy contour, but in a more usable form, the zero crossing contour is computed and displayed for manual verification and comparison purposes only. Complete sets of energy, zero crossing, V/U/M/S, and pitch contours are shown for three words in Figures 4.13-4.15.

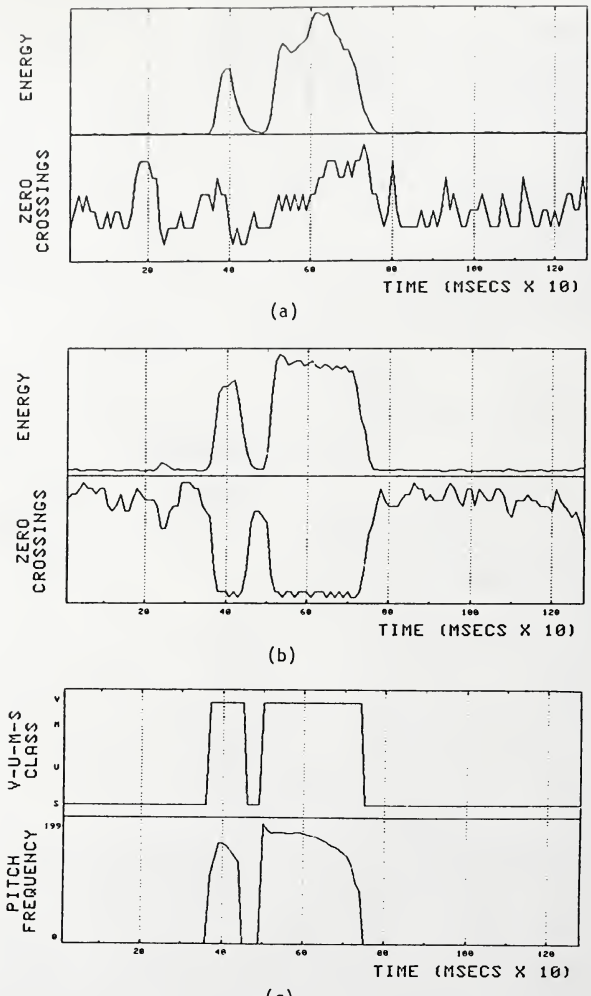


Figure 4.13 Energy and zero crossing contours from (a) speech and (b) EGG for the word "abort"; part (c) shows the corresponding V/U/M/S and pitch contours.

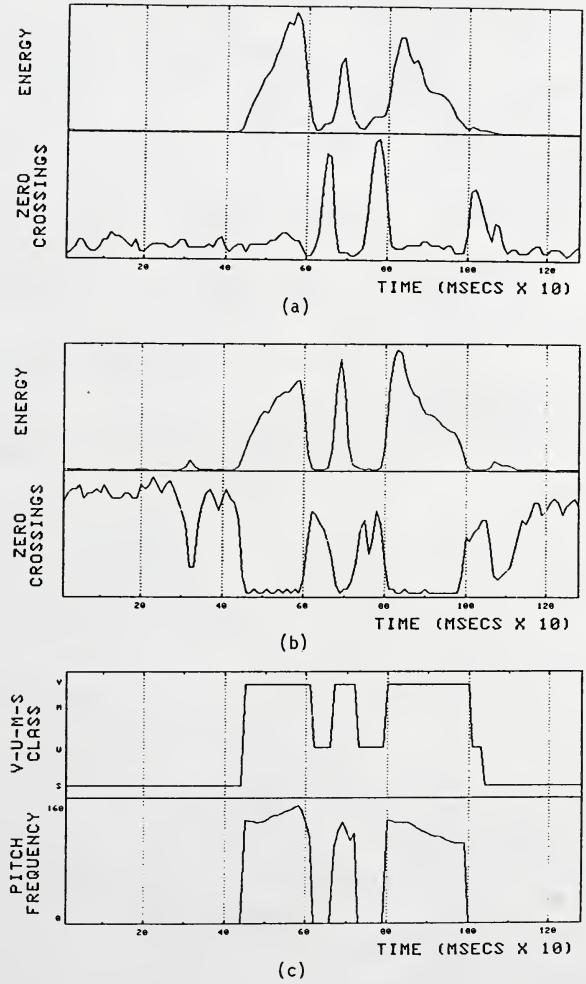
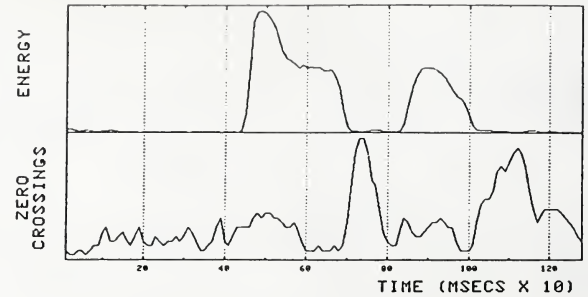
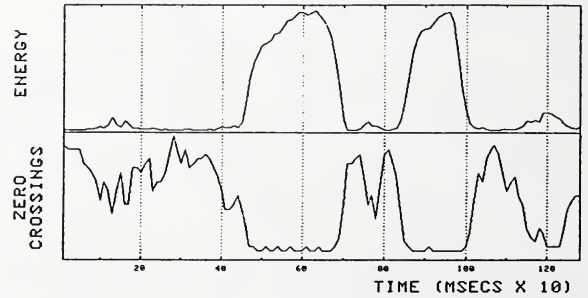


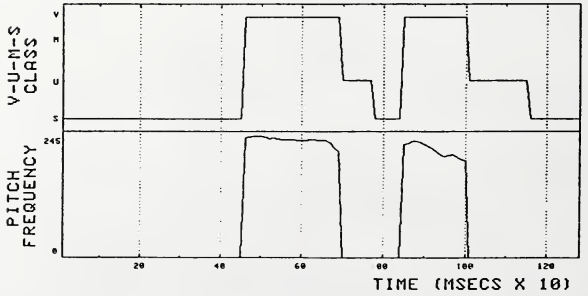
Figure 4.14 Energy and zero crossing contours from (a) speech and (b) EGG for the word "altitude"; part (c) shows the corresponding V/U/M/S and pitch contours.



(a)



(b)



(c)

Figure 4.15 Energy and zero crossing contours from (a) speech and (b) EGG for the word "takeoff"; part (c) shows the corresponding V/U/M/S and pitch contours.

Many algorithms combine the V/US classification and pitch estimation procedures. A logical reason for this is that they both may utilize the same indicator. For example, a peak in the autocorrelation of the LPC residual represents voicing (periodicity) and its position specifies the pitch period. Similar reasoning may suggest use of the easily computed pitch contour as a voicing indicator. A problem with this approach is the variability of the EGG zero crossing rate. Thresholds can not specify a reasonable range of pitch values and simultaneously give reliable voicing information. On the other hand, the EGG was shown to yield accurate pitch estimates when the frame was known to contain voicing.

In summary, the techniques used to extract all the information needed for subsequent word recognition procedures have been presented in this chapter. Once the word is found in the digitized data set, all frames are classified as being either voiced, unvoiced, mixed, or silent. An estimate of the fundamental frequency of the speech signal is computed for each voiced (or mixed) frame. Finally, the parameters of a 14 pole LPC model are derived sequentially for the entire word. The next chapter describes how the broad phonetic classification and prosodic cues are used to access the lexicon.

CHAPTER 5
EQUIVALENCE CLASS FORMATION AND LEXICAL ACCESS

The two-pass approach requires that a coarse stage of recognition takes place prior to the finer matching necessary for exact word identification. Information derived from the speech and EGG signals to be used in both recognition stages is available after application of the techniques of the last chapter. The subject of this chapter is the way by which the V/U/M/S classifications, pitch, and energy measurements are used to associate similar utterances for the first stage of recognition. Topics discussed include the word representation used, construction of equivalence classes, and methods for accessing the various classes of words.

5.1 Word Representation

Once again, the purpose of the first stage of recognition is to eliminate the unlikely words from the pool of match candidates. This must be done in a highly reliable manner to avoid irrecoverable errors due to early elimination of the correct word. At the same time, the procedure should be computationally inexpensive and should provide substantial lexical pruning. Studies of potential vocabulary filters have shown that appropriate word descriptors can indeed be chosen to meet the objectives above [49,50]. These studies, however, assumed that the desired features can be robustly obtained from the speech signal. Subsequent work reporting preliminary results found that the actual

implementation must also consider analysis errors and normal variability in pronunciation [48,51,87]. As pointed out in Chapter 2, some of the representations used are particularly prone to error or some amount of speaker dependence.

The word descriptors used in the present investigation are derived from the simple energy and zero crossing parameterizations of the input waveforms described in the last chapter. This reduced set of data is used for a very broad acoustic-phonetic segmentation of the utterance. No attempt is made to perform a detailed phonetic transcription of the word. The V/U/M/S representation approximates the phonetic categories of [48,49,51] without the final relabeling. Also, acoustic correlates of stress and intonation are available to augment the segmental representation.

5.1.1 V/U/M/S String

The output of the V/U/M/S classifier is a string of length 128 representing the characteristics of each 10 ms frame with one of four symbols. A possibility is

SS...SSMMMVV...VVSSUUVV...VVSS...SS (5.1)

which includes unnecessary surrounding silent classifications. To describe only within word characteristics, the string can be reduced to

#M#V#S#U#V (5.2)

or just

MVSUV . (5.3)

The string in Eq. (5.2) represents acoustically consistent segments of duration specified by the preceding number (#). Since duration is generally rather variable, the further reduced form in Eq. (5.3) would be a more reliable (repeatable) representation.

Another consideration in how the string should be reduced is the way in which it will be utilized. The V/U/M/S representation is obtained from a test utterance and then compared with all of those for the reference words. Some form of string matching is therefore required to find which word (class) the input is closest to. This task is facilitated with a time normalized sequence where the information is position dependent. Thus, the representation of Eq. (5.3) was encoded to

VUMS (5.4)

where V now denotes the number of voiced regions in the entire word, U denotes the number of unvoiced regions, etc. Notice that only the number of regions need be specified since the order of presentation is fixed.

5.1.2 Fricative Location

The encoded form of Eq. (5.4) is reduced in discriminating power relative to the original representation since the natural sequencing is also normalized out. For instance, the sequence UVUV becomes 2200 which is the same as the encoded representation of VUVU. An effective way of compensating for the lost symbol order was found in a three way specification of fricative (and stop) positioning. Much information is regained since, by definition, the voiced regions must be distinct.

The mixed, unvoiced, and silent regions, and combinations thereof, were assigned as either word-initial, word-medial, or word-final. To be in an initial or final position, the corresponding end frame must be classified as either U or M. A medial designation was given to the occurrence of any fricated sequence of frames that was bounded on both sides by voiced regions. Adjacent U, M, and/or S frames were considered as part of the same cluster.

Fricative positional information was added to the representation of Eq. (5.4) by including another category. A word is now described by

$$\text{VUMSF} \tag{5.5}$$

where F indicates the relative position of the fricative(s). The value of F is conveniently found to be the decimal equivalent of the binary number 100 for word-initial fricatives, 010 for word-medial fricatives, and 001 for word-final fricatives. For multi-fricative words, the appropriate combination of numbers is summed. The example sequences UVUV and VUVU used earlier become 22006 and 22003, respectively.

5.1.3 Prosodic Cues

Apart from some recent studies [50,87], the use of prosodics in IWR has been conspicuously absent. Suprasegmental information such as stress patterns, intonation, and timing structures are useful tools with which lexical pruning can be aided. The present system considers stress and intonation primarily for location of the emphasized portion of the word. Durational information is retained but not used at this time.

Stressed regions tend to have a high and rising fundamental frequency, high energy level, and long duration. Figure 5.1 shows an example where the F0 contour is rising (and high) and the speech energy is also peaked. Combination of the various stress cues results in more accurate automatic stress location [88]. There are some problems, however, in using the F0 contour. To start with, the F0 fluctuation can be due to the particular vowel. High vowels have higher F0 than low vowels. Rising F0 may then signal the presence of a high vowel and not represent a stressed region. Other problems include the rising F0 after a voiced obstruent, and the falling F0 after an unvoiced consonant that is followed by a stressed region [89].

The main problem in using the pitch contour for stress location encountered in this study was apparently the level of accuracy. At voicing onset and offset a large amount of variability in F0 was typically found. The worst case was when vocal fry was present. The increase in speech signal amplitude found between primary excitations corresponded to an increase in lateral vocal fold contact area as indicated by the synchronized EGG signal. This situation is illustrated in Figure 5.2.a. Since the pitch tracking algorithm follows the rate of excitation by the glottal source (vocal fold closure), the result is an increase in estimated pitch in the regions of vocal fry. An example of increased F0 due to vocal fry at voice onset and offset is shown in Figure 5.2.b. After preliminary use of the rising F0 cue as a stress indicator, it was obvious that a significant increase in algorithm complexity was necessary in order to attain reliable results. An alternative approach was to modify the pitch tracker so that only general trends in F0 are obtained [88]. Instead of doing either, the

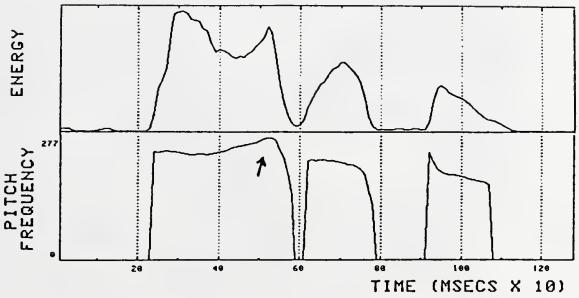
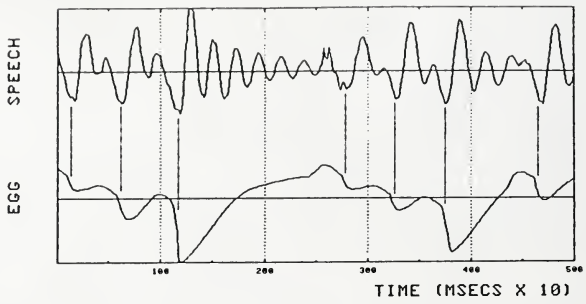
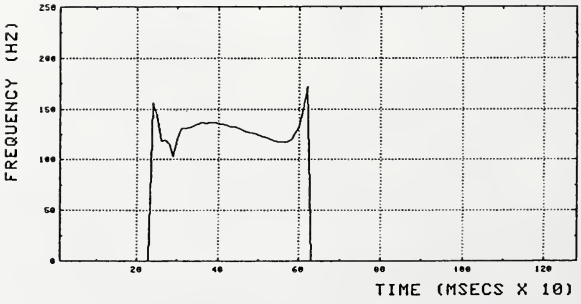


Figure 5.1 Example of high speech energy and rising F0 in the word "affirmative."



(a)



(b)

Figure 5.2 Example of vocal fry for the word "one" in (a) the speech and EGG waveforms and its effect on (b) the pitch contour.

algorithm remained unaltered and was used for observation of the finer details in the pitch contour. For stress detection, consistency increased when the speech energy contour was the sole cue.

The acoustic segmentation employed does not perform syllabification per se. A word like "normal" has exactly one voiced region whereas "reject" has two. Only in the latter case do the voiced regions correspond to the vocalic nuclei of the syllables. Also, in some monosyllabic words, the speech energy contour shows phonemic dependence. An example is the word "one" shown in Figure 5.3. The energy is seen to decrease in the nasalized region.

The considerations above led to the practical definition of a stressed region as that part of a word where voicing (energy) was relatively strong. Words with a single voiced region were characterized as having emphasis on the first half, second half, or a substantial distribution across both halves of the voiced region. To make the distinction, the speech energy, SENG(n), was summed over the first half of the voiced region to obtain SE1, and the second half to get SE2. If $SE2/SE1 \leq .65$, then the greater energy was located in the first half. Similarly, if $SE1/SE2 \leq .65$, then the second half was determined to be the strongest. If neither condition held, the energy was assumed to be somewhat centralized in the voiced segment. The conservative threshold of .65 was chosen to reduce the variability of the stress assignment in repetitions of the same word.

In words with multiple voiced regions, the one with the greatest energy was selected as stressed. Since the voiced regions vary in duration, the summed energy was divided by the number of frames in the given region. The maximum of the averaged energy values corresponds to

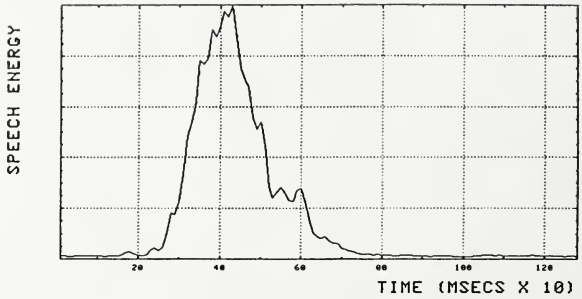


Figure 5.3 Speech energy contour for the word "one."

a stressed voiced segment if it is at least 20% greater than the average energy of any other voiced region. Otherwise, no specific stress location is assumed.

To include the stress position in the word representation, one more category has to be added to Eq. (5.5). Using

VUMSFs, (5.6)

the s denotes the stressed part of a single voiced region in the same way the fricative locations were specified. The decimal value of s will be 4, 1, or 2 for a stressed first half, second half, or mid region, respectively. When V is greater than one (multiple voiced regions), s is simply the number of the voiced region found to be stressed. If no single region was found to have greater emphasis than the others, s was set to zero.

5.2 The Equivalence Class

Stress and fricative positional information has been combined with the V/U/M/S classification to form the codeword in Eq. (5.6). The encoded representation is generally associated with a small group of words from the lexicon that has the same acoustic-phonetic characteristics. All words that are equivalent, to the level of precision afforded by the descriptors used, are designated as members of the same equivalence class. The codeword derived from a test utterance can then be used to retrieve the class of words that are similar to the word spoken.

The codewords are formed from descriptors obtained from the data using robust and computationally efficient algorithms. Adequate lexical

pruning power has been inferred from preliminary studies using related vocabulary filters. A manual encoding and classification of all words in the vocabulary used in this work also demonstrated the potential of the VUMSFs representation. The results of this classification are summarized in Table 5.1. An encouraging observation from the results was that the average class size was only 2.56 words. The maximum class size was 10 and some 19 words were uniquely identified in classes of size 1. The average class size is computed irrespective of the likelihood of the various class sizes to occur. A more appropriate statistic is the expected class size as used in [50],

$$ECS = \sum_i CS(i)p(CS(i)) \quad , \tag{5.7}$$

where CS(i) is the class size and p(CS(i)) is the probability of getting a class of that size. Using Eq. (5.7), the preliminary results from manual classification of the 100 word vocabulary yield an expected class size of 4.78.

Actual pronunciations of the same vocabulary entry can show a significant amount of variability, particularly for words with unvoiced regions. The word "five" is typical of a case where the leading fricative may range in intensity from strong to essentially nonexistant and still be perceived by the human listener to be the same word. The machine listener must also accomodate such variability. With the fixed length word representation of Eq. (5.6), a coding approach such as that in Appendix B would handle the insertion-deletion problem by decoding (correcting) to the closest word class. Since enough robustly detected independent features are not available, the approach taken here is to enumerate all likely variations for all words in the vocabulary.

TABLE 5.1
EQUIVALENCE CLASSES FOUND FROM MANUAL ASSIGNMENTS

Class Size	Number of Classes
1	19
2	8
3	4
4	3
5	0
6	0
7	2
8	1
9	1
10	1

Class formation. Allowing for alternate word pronunciations will make the experimental results differ from the manual classifications in two ways; the number of classes, and the expected class size will both increase since words will likely belong to more than one class. An objective in class formation was to always have the spoken word as a member of the class corresponding to the codeword representation. This way, the recognizer would never start the second pass of recognition with a zero probability of correct match.

A collection of vocabulary words and expected variations can be generated from an orthographic representation and a set of rules. This system, however, utilizes a spectral representation in the second pass so a training session is still necessary. Indeed, the spectral templates can also be generated from a smaller set of basic components, but a set of rules is again needed. Data collection is unavoidable in either approach since the rules should be derived from actual observations. Once adequate rules are developed, less storage would be needed and new words could be added to the vocabulary without additional training.

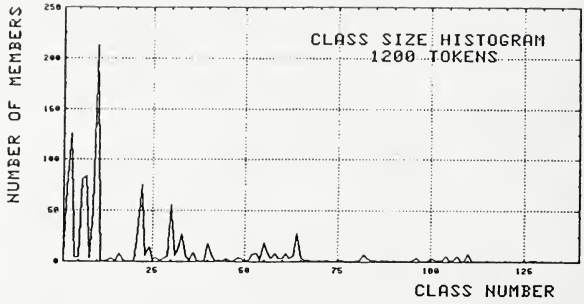
To achieve a degree of speaker independence, multi-speaker data were collected in a training phase following the procedures outlined in Chapter 3. Seven cooperative speakers were initially used to obtain different realizations of the 100 vocabulary words. Each speaker went through the entire vocabulary twice. Thus, a total of 1400 words was available for equivalence class formation. The seven speakers included 3 adult males, 3 adult females, and 1 child. Unfortunately, the child's data contained an excess of breath noises and could not be used. The remaining 12 realizations of each word were prepared for subsequent processing.

Equivalence classes were formed after analysis and encoding of the 1200 useable words. All words with the same codeword representation were assigned to the same equivalence class. Instead of the 39 classes found by manual classification, the number of classes found for the 100 words from the 12 speakers was 140. Classification results are displayed in the histograms of Figure 5.4. Part (a) shows the distribution of pronunciations (or tokens) versus the class number, and part (b) indicates how many of the tokens for each class in (a) are actually different words. The classes are ordered numerically from 100001 to 431233 and the empty ones were skipped.

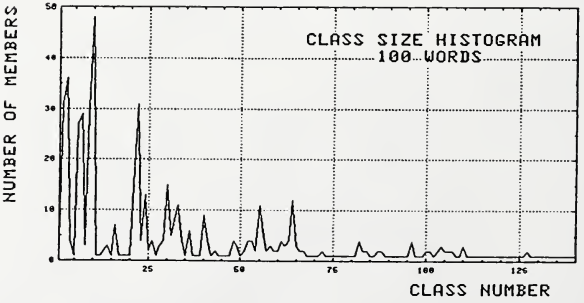
The results in Figure 5.4 show that the average class size was not far from what was expected. For the 140 classes, the average size was 3.94 words. Clearly from part (b), though, the class sizes are quite disparate. The likelihood of getting one of a few large classes is seen to be quite high in part (a). This fact is reflected in the expected class size which is 24.07. The considerable class overlap was caused mainly by weak fricatives. Class sizes ranged from 1 to 48 where the V, VU, UV, and UVU classes were the largest.

5.3 Database Organization

The first pass analysis by the recognizer yields a VUMSFs representation of the test utterance. This codeword is exactly the label of the equivalence class that is expected to contain the word spoken. A logical structure for the lexicon is therefore an inverted form keyed by the various codewords. For search considerations, the classes are stored in increasing numerical order. The plots of Figure 5.4 suggest that an intraclass ordering may also prove beneficial. Note



(a)



(b)

Figure 5.4 Equivalence class histograms for (a) the different tokens and (b) the different words.

that the transition from part (a) to part (b) involves more than uniform scaling. This is due to the pronunciation variability of the different words.

Intraclass ranking. Grouping of the vocabulary words by their equivalence class label is the mechanism by which unlikely words are excluded from considera-tion. After finding the codeword from the test utterance, the group of words with the same label are accessed directly. This pattern matching function can also be formulated in a statistical framework [90].

The codeword representations, are defined to be observations of a random vector $\underline{c}$. The pattern $\underline{c}$ is described by a multivariate probability density function, $p(\underline{c}|\omega_i)$, where $\underline{c}$ is known to belong to pattern class (word) ω_i, i=2, ..., N. For the vocabulary used, N is equal to 100. The problem of deciding to which class $\underline{c}$ belongs is solved by choosing that class for which $p(\omega_i|\underline{c})$ is maximum. The a posteriori probability, $p(\omega_i|\underline{c})$, is the probability of $\underline{c}$ belonging to class ω_i. The classifier that maximizes this a posteriori probability is called the Bayes' classifier.

More correctly, Bayes' classifier minimizes the total expected loss incurred by assigning $\underline{c}$ to class ω_i. A loss equal to L_{ij} occurs when $\underline{c}$ is assigned to ω_j when it actually came from ω_i. Assuming that all errors are equally costly, the loss function

$$L_{ij} = 1 - \delta_{ij} \tag{5.8}$$

can be used to assign no loss to a correct decision and a normalized loss of unity to any error. The conditional average loss is given by

$$l_j(\underline{c}) = \sum_{i=1}^{N} L_{ij} p(\omega_i|\underline{c}) \quad . \tag{5.9}$$

Assigning the pattern to the class with the smallest conditional loss, $l_j(\underline{c})$, j=1, ..., N, will minimize the total expected loss with respect to all decisions. Thus, $\underline{c}$ is assigned to class ω_i if

$$l_i(\underline{c}) < l_j(\underline{c}), \; j=1, \ldots, N, \; j \neq i \quad . \tag{5.10}$$

The decision rule of Eq. (5.10) can be rewritten using Bayes' formula

$$p(\omega_i|\underline{c}) = \frac{p(\underline{c}|\omega_i)p(\omega_i)}{p(\underline{c})} \quad , \tag{5.11}$$

where

$$p(\underline{c}) = \sum_{i=1}^{M} p(\underline{c}|\omega_i)p(\omega_i) \tag{5.12}$$

is the unconditional probability function of $\underline{c}$ and $p(\omega_i)$ are the a priori probabilities of the N classes. Substituting Eqs. (5.8) and (5.11) in Eq. (5.9) and neglecting the common $1/p(\underline{c})$ factor for all l_j's gives

$$\begin{aligned} l_j(\underline{c}) &= \sum_{i=1}^{M} p(\omega_i|\underline{c}) - p(\omega_j|\underline{c}) \\ &= p(\underline{c}) - p(\underline{c}|\omega_j)p(\omega_j) \quad . \end{aligned} \tag{5.13}$$

The decision rule of Eq. (5.10) then chooses ω_i if

$$p(\underline{c}) - p(\underline{c}|\omega_i) < p(\underline{c}) - p(\underline{c}|\omega_j)p(\omega_j)$$

or

$$p(\underline{c}|\omega_i)p(\omega_i) > p(\underline{c}|\omega_j)p(\omega_j),\ j=1,\ \ldots,\ N,\ j\neq i \quad . \tag{5.14}$$

An alternate form of Eq. (5.14),

$$p(\omega_i|\underline{c})p(\underline{c}) > p(\omega_j|\underline{c})p(\underline{c}),\ j=1,\ \ldots,\ N,\ j\neq i \quad , \tag{5.15}$$

is obtained by substituting in Eq. (5.11). Notice that Eq. (5.14) can be reduced to

$$p(\underline{c}|\omega_i) > p(\underline{c}|\omega_j),\ j=1,\ \ldots,\ N,\ j\neq i \tag{5.16}$$

since all classes are assumed to be equally likely (i.e. $p(\omega_i) = p(\omega_j)$, $j=1,\ \ldots,\ N$). Also, Eq. (5.15) can be expressed as

$$p(\omega_i|\underline{c}) > p(\omega_j|\underline{c}),\ j=1,\ \ldots,\ N,\ j\neq i \tag{5.17}$$

since $p(\underline{c})$ is independent of j.

The decision rules of Eqs. (5.16) and (5.17) provide two equivalent approaches to the classification problem. Both assume that the conditional densities are completely known. Choice of one over the other depends on the use of $p(\underline{c}|\omega_i)$ or $p(\omega_i|\underline{c})$. This work assumes that $p(\underline{c}|\omega_i)$ can be estimated from the training samples.

In the vocabulary (reference) development stage, a codeword is computed for each of the 12 tokens corresponding to a realization of each of the 100 different words. Based on the relative frequency of occurrence of the codewords for each word, the class conditional densities $p(\underline{c}|\omega_i)$ can be computed for i=1, ..., 100. Therefore, to find the pattern class to which the test codeword corresponds, i is found such that $p(\underline{c}|\omega_i)$ is maximum for $1 \leq i \leq 100$. This gives the most likely word for the particular observation of the random vector $\underline{c}$.

Not only should $p(\underline{c}|\omega_i)$ be maximized over all words ω_i, but the words should also be ordered by decreasing probability. Normalization by $p(\omega_i)/p(\underline{c})$ after reordering will give a display of $p(\omega_i|\underline{c})$. For every observation of $\underline{c}$, the words ω_i having $p(\omega_i|\underline{c})\neq 0$ are precisely the members of the equivalence classes as defined in Section 5.2.1. The difference now is that a within class ranking of the words from most to least likely exists.

Vocabulary entries indexed by a common equivalence class label are stored in order of decreasing likelihood. Along with each word, the a posteriori probability is also stored. A shortcoming of the probability density estimation procedure is that the number of training samples used may not be large enough. For this reason, all probabilities are updated after each recognition run as new relative frequency information becomes available. Equivalence classes may be reordered as probabilities change and new classes are inserted into the database when reasonable unknown codewords are encountered. Continuous updating of the reference data serves to stablize the word ordering in each class. After eight updates, the number of classes increased to 172, the average class size was 4.10, and the expected class size was 25.82.

5.4 Lexical Access

The codeword obtained from the test utterance is the label of the equivalence class to be searched in the second recognition stage. Since the lexicon is stored in an indexed sequential form, the words of each class are easily accessed after a binary search of the known codewords. The 1200 tokens initially used to form the reference database provide a good representation of the numerous classes possible. Because this set was by no means all inclusive, the updating procedures allow the addition of alternate word pronunciations. If a codeword is new not only to one word but to all words, no equivalence class will be found upon lexical access. Only after the attempted match can the database be updated to include the new class.

To avoid certain termination after an unsuccessful access of the lexicon, limited codeword correction procedures were implemented. A three stage hierarchy of rules changed the test codeword to the first corrected version matched. Stress location correction was attempted in the first pass since this type of error is least significant. Next, single U, M, or S errors in the codeword were considered. The last pass looked for a match when single U, M, or S errors and one or more F errors were allowed. If still no match was found after limited corrections, the recognition process would end and the lexicon would be updated. Manual intervention in the updating process was retained to prevent reference modification with codewords from mispronunciations or with those derived from data sets containing other artifacts.

CHAPTER 6
DETAILED PATTERN MATCHING

In the second stage of recognition, detailed matching techniques are utilized to determine which reference word is most similar to the test utterance. From the first recognition pass, an ordered list of likely candidates is available. The number of words on the list is some fraction of the vocabulary so that computation is not wasted on inappropriate candidates. This chapter is concerned with the pattern matching procedures of the second stage of recognition. Topics to be considered include nonlinear time alignment of the utterances, feature vector formation, the decision rule for matching, and some local weighting procedures.

6.1 DTW Alignment

Multiple productions of the same word never have identical temporal characteristics. Therefore, when two utterances are to be compared, some form of time alignment or normalization is in order. A linear transformation of the time scale of either the test or reference utterance is one way to solve the alignment problem. A drawback of this approach is that the actual time scale differences usually are due to nonuniform variations of specific portions of the word. Simply stretching or compressing a word to match endpoints does not account for the nonlinear intraword variations. Optimal alignment of the

corresponding portions of the test and reference utterances will substantially reduce recognition errors [10].

This study employed the most common approach to nonlinear time alignment which uses a technique known as dynamic time warping [18,23]. The function of mapping parts of a reference pattern onto a test pattern by way of a nonlinear warping path found from a DTW procedure was illustrated in Figure 1.4. The time scale of a pattern representing the reference utterance,

$$R = \{R(1),R(2),\ldots,R(M)\} \quad , \tag{6.1}$$

is warped so that specific events line up with the same events in the pattern for the test utterance,

$$T = \{T(1),T(2),\ldots,T(N)\} \quad . \tag{6.2}$$

The correct alignment is given by the warping function,

$$m = w(n) \quad , \tag{6.3}$$

that is found by solving Eq. (1.6),

$$D(T,R) = \min_{w(n)} \left[\sum_{n=1}^{N} d(T(n),R(w(n))) \right] \quad . \tag{6.4}$$

The local distance $d(T(n),R(w(n)))$ used between test frame n and reference frame $m = w(n)$ is the log likelihood measure proposed by Itakura [12],

$$d(T(n),R(W(n))) = \log \left[\frac{\underline{a}_R R \underline{a}_R'}{\underline{a}_T R \underline{a}_T}\right] \quad , \tag{6.5}$$

where R is the correlation matrix of the speech segment in the test frame. The vectors $\underline{a}_T$ and $\underline{a}_R$ contain the LPC coefficients for the test and reference frames, respectively.

Solution of Eq. (6.4) is equivalent to finding the best path through the M by N grid in Figure 1.4. This is accomplished using a simple recursion and the local and global path contraints of [12]. The standard approach defines the accumulated distance function $D_A(n,m)$ as the total distance along the optimal alignment path from the starting point in the grid to position (n,m). The accumulated distance is found using the recursion

$$D_A(n,m) = d(T(n),R(m)) + \min_q [D_A(n-1,m-q)] \tag{6.6}$$

where the allowable values of q are determined by the local continuity constraints

$$q = \begin{cases} 0,1,2 & \text{if } w(n-1) \neq w(n-2) \\ 1,2 & \text{if } w(n-1) \neq w(n-2) \ . \end{cases} \tag{6.7}$$

The limitation on q guarantees that the alignment path does not stay flat for two consecutive frames. Boundary conditions, or endpoint constraints, specified as

$$\begin{aligned} w(1) &= 1 \\ w(N) &= M \end{aligned} \tag{6.8}$$

assume that the test and reference utterance endpoints have been accurately located. The constraints of Eqs. (6.7) and (6.8) restrict the range of grid points through which the alignment path can traverse. Thus, global path constraints are also implied and are depicted in Figure 1.4 as the search region bounded by the parallelogram with lines of slope 2 and 1/2.

The final value of the recursion in Eq. (6.6) is

$$D_F = D_A(N,M) \tag{6.9}$$

which is found after sequential iterations from n = 1 to n = N. The quantity D_F is the distance between the test and reference patterns after optimal alignment. The actual warping path w(n) is found by backtracking from the endpoint (N,M) using the framewise transitions in time reversed order.

The work of [91] showed that different ratios of reference to test pattern length can shrink the search region parallelogram. Excessive shrinking negates the advantages of DTW alignment. To overcome this problem, both patterns can be normalized to a standard length prior to DTW alignment [91]. The linear warping maximizes the region of possible paths giving the DTW algorithm a better chance of attaining proper alignment.

The normalize-and-warp procedure is used in the present work with linear normalization of the test and reference patterns to a predetermined length greater than or equal to the longest possible pattern. Since the time axes are never contracted, short duration events will not disappear from the patterns. The warping path for linear time scale expansion is given by

$$m = w(n) = [(n-1)\ \frac{(NP-1)}{125} + 1] \tag{6.10}$$

where NP is the length of the pattern to be warped, and the final length after warping is 126.

The endpoint constraints in Eq. (6.8) are appropriate for entirely voiced words since the endpoint detection algorithm accurately finds voicing onset and offset with the aid of the EGG. The constraints are also adequate for most words with initial and trailing fricatives since the regions can be scaled by the DTW algorithm. When a weak fricative region is missed at one end of the word, large local distances will result for a few frames. To allow a small amount of flexibility in endpoint location, each linearly warped test and reference pattern is augmented with the two frames preceding and following the endpoints found in the original pattern. All patterns are now 130 frames in length.

Relaxing the endpoint constraints of Eq. (6.8) will yield the desired flexibility in the end regions [92]. The new endpoint conditions are

$$\begin{aligned} 1 \leq w(1) &\leq 1 + \delta \\ M - \delta \leq w(N) &\leq M \end{aligned} \tag{6.11}$$

where the offset parameter δ is set to 2. Since either the test or reference pattern may be missing an end fricative, the continuity constraints are also relaxed at the ends. Two horizontal transitions are allowed for $n \leq 3$ and $n \geq 128$. The endpoint considerations are illustrated by the parallogram search region in Figure 6.1.

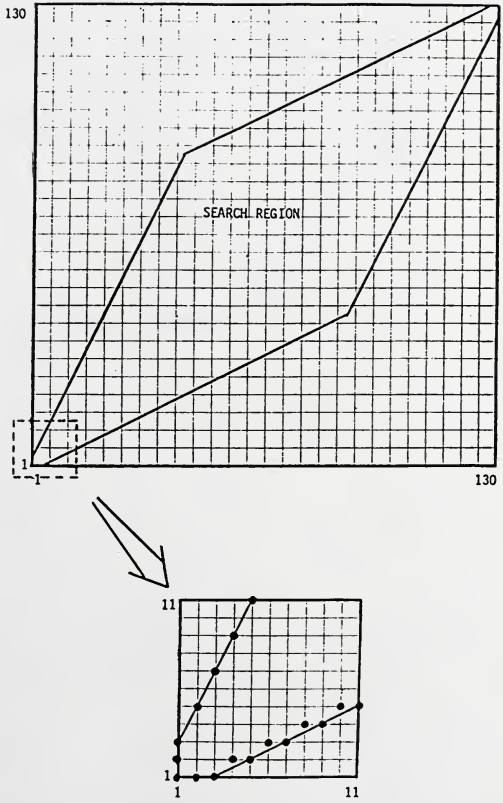


Figure 6.1 Allowable search region (parallelogram) for DTW matching.

6.2 Template Formation

Procedures for simultaneous distance computation and time alignment were given in the preceding section. The patterns of Eqs. (6.1) and (6.2) are derived from an LPC analysis performed every 10 ms. Thus, a time sequence of LPC feature vectors represents each utterance in the word matching stage. The exact form of the feature vector depends on whether it is for a test or reference pattern.

The local distance of Eq. (6.5) can be expressed in the computationally more attractive form [12],

$$d(T(n),R(m)) = \log \left[\sum_{k=0}^{p} \frac{r(k)}{E} r_a(k) \right]$$

$$= \log [T(n)R'(m)] \qquad (6.12)$$

where T(n) and R(m) are feature vectors (see Appendix A) for frames n amd m of the test and reference patterns, respectively. From Eq. (6.12),

$$T(n) = \frac{1}{E} (r(0),r(1),\ldots,r(p)) \qquad (6.13)$$

and

$$R(m) = (r_a(0),r_a(1),\ldots,r_a(p)) \qquad (6.14)$$

are the forms of the p+1 component vectors representing each frame of the two patterns.

6.2.1 Test Pattern

An unknown test utterance is analyzed sequentially using the procedures specified in Section 4.4. The feature vectors for the test pattern consist of the p + 1 speech autocorrelation coefficients divided by the total squared error from the LPC analysis. The NP frames, or vectors, in the initial pattern are linearly warped to the predetermined length by repeating frames as necessary according to Eq. (6.10). The final 130 frame test pattern is stored until needed for the dot product distance computation in Eq. (6.12).

6.2.2 Reference Pattern

In order to establish the identity of a spoken word, some basis for comparison is needed. The stored reference patterns obtained in a training mode serve this purpose. Since the recognizer must be aware of various normal pronunciations, each word is usually spoken several times. A reference template can then be created for each token [12], or a robust training procedure can be used to form a single template [93]. For speaker independent operation, the recognizer must also account for interspeaker variations. An effective way of doing this is by using clustering techniques on a large number of repetitions of each word collected from many speakers [27,94]. This will yield a reference template for each of the different clusters found for every word. A disadvantage of this approach is the amount of storage required for large vocabulary applications. Also, for recognition, an additional DTW will be needed for each alternative representation stored.

Another possibility for creating speaker independent templates is to average all replications of each word to give a single reference

pattern [95]. This approach produces generally reliable references and, more importantly for this study, keeps the number of comparisons and amount of storage required to a minimum. A disadvantage is that highly dissimilar tokens may average to a reference that does not represent the word well. If the memory and computational requirements can be tolerated, the clustering approach would give better results [27]. Unfortunately, the tradeoff is not favorable for large vocabulary recognition.

The averaging procedures used to create reference templates for the present system employ a multilevel combination approach. The feature vectors combined were obtained from an LPC analysis of the same utterances used to develop the equivalence class database. The set of words included two tokens of every lexical entry collected from each of 6 native American speakers. The 3 males and 3 females had no speaking characteristics from any regional dialect. Preliminary reference templates were stored as vectors of the normalized autocorrelation coefficients $r(k)/r(0), k=0,1,\ldots,p$. All 12 templates for each word were linearly warped to the standard length of 130 frames. Their original lengths were noted so that an average duration could be found for each word.

The templates of a given word were combined using the three stage scheme illustrated in Figure 6.2. At level A, the six male and six female templates were combined in pairs corresponding to the duplication of words by each speaker. For each pair, a template was nonlinearly warped onto the one that was originally closer to the average word duration. After warping, backtracking was used to obtain the warping path, $w(n)$, so that the corresponding frames of the two templates could be averaged. Two templates, R_i and R_j, were combined such that

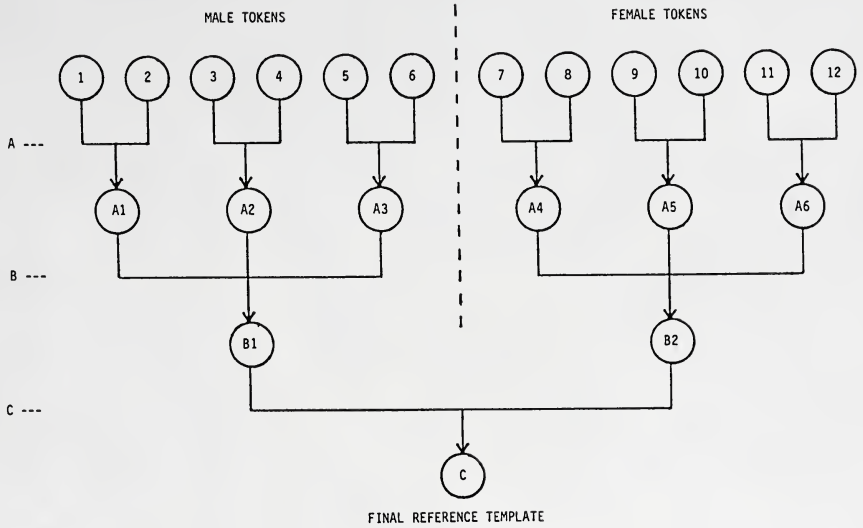


Figure 6.2 Token combination order for reference template formation.

$$(R_i + R_j)/2 = (R_i(n) + R_j(w(n)))/2,\ n = 1,2,\ldots,130 \qquad (6.15)$$

where for some frame n, the 1^{th} component of the vector is $(\hat{r}_i(1) + \hat{r}_j(1))/2$ and the $\hat{r}(1)$ are the normalized autocorrelation coefficients.

The next level of combination, B, averaged patterns in two three element groups. Two of the templates in each group were warped onto the one that was nearest average in duration and then all three were combined. In the last stage, the single male and female tokens remaining were combined in the same manner as the others. The reference pattern that resulted allowed almost equal contribution from all tokens. Slight bias towards the templates closest to the average length was intended with the warping direction used.

After combining all of the tokens into one pattern, one more step was necessary to put R in the form required for Eq. (6.12). Each frame in the pattern was converted to LPC coefficients and then the autocorrelation of the LPC's, $r_a(k)$, was computed. The final reference template consisted of the vectors R(m), m = 1,2,...,130, given in Eq. (6.14) where

$$r_a(k) = \begin{cases} \sum_{i=0}^{p} a_i^2 & ;\ k = 0 \\ 2\sum_{i=0}^{p-k} a_i a_{i+k} & ;\ 1 \le k \le p\ . \end{cases} \qquad (6.16)$$

The form of the feature vectors for test and reference patterns given in Eqs. (6.13) and (6.14) is for computational economy during word recognition. The DTW alignment required for reference pattern formation

also uses the distance as expressed in Eq. (6.12); except, while in the training mode, the input patterns expected are vectors of normalized autocorrelation coefficients and additional transformations are needed before the distance is computed.

6.3 The Decision Rule

Each reference word in the equivalence class selected by the first stage of the recognizer is warped onto the test utterance using the DTW procedures of the previous sections. The objective of the two-pass approach is to significantly reduce the number of DTW comparisons required. Chapter 5 showed that the average number of words in a class is less than 5% of the vocabulary. Thus, if the classes were equally likely, a significant savings would be consistently realized. When the likelihood of the various class sizes was considered, the expected class size was found to be roughly 25% of the vocabulary. This was because a lot of the words had alternate representations appearing in a few of the same classes. The word entries in a large class (or any class) occurring infrequently need not always be considered as match candidates in the second state of recognition.

Determination of which reference pattern matches the unknown input and when to stop considering other references is made by the decision rule. The basic rule for matching says to choose the nearest neighbor (i.e., closest reference pattern). All words in an equivalence can be considered and an expected reduction in DTW computations of 75% will result. Even greater savings are possible when the least likely words in the class are not considered. The words of an equivalence class are matched in the order presented, from most to least likely. To reduce

recognition time, the stipulation can be made that only class members with some minimum likelihood will be matched. Thus, the likelihood rejection threshold, T_R, can be defined as the minimum acceptable likelihood ratio of any member of the class to the most likely member (the first entry). By performing DTW matching for only those class members such that

$$\frac{p(\omega_i|\underline{c})}{p(\omega_1|\underline{c})} > T_R \quad , \quad 1 < i < M \quad , \tag{6.17}$$

unlikely references will not be considered and recognizer response time will be improved.

Once all match candidates have been considered, the test utterance is recognized as the word corresponding to the reference pattern that resulted in the smallest accumulated distance from the unknown input. If no reference patterns are suitably close, no match is found and recognition terminates with a repeat utterance prompt. Acceptable closeness is determined to be the maximum of the 11 intertoken distances computed during reference template formation plus an experimentally determined constant. This allows for some variation beyond the worst case encountered. When no references are close enough to the test and T_R is chosen to be nonzero, additional references are considered until one matches. This may (but rarely does) continue until all words in the class are examined anyway. If several reference patterns match the test equally well, all are output as possibilities in order of closeness.

6.4 Class Specific Matching

Some limiting factors affecting the accuracy of the recognition system are the manner by which the reference templates were created and the word matching procedure. Reference template averaging is known to be less effective than a clustering conversion procedure used in conjunction with the K-nearest neighbor decision rule [6]. However, as discussed in earlier sections, recognizer response time and storage considerations make 6 to 12 templates per word undesirable for large vocabulary applications. Potential for improvement exists with the two pass approach since general word characteristics are known prior to DTW matching in the second stage. In this section, initial efforts directed toward improving the recognizer's performance for the classes associated with the complex alpha task (e.g., A, B, C, ..., Z) will be described.

The poor performance of most DTW based recognizers on a vocabulary consisting of the letters of the alphabet often stems from the method used to assess pattern similarity. The standard dynamic time warping algorithm is inherently limited by one of its normally useful features. Discriminating among an arbitrary group of dissimilar words is effectively accomplished by matching the reference that is minimally different in a global sense. For the letters of the alphabet, this strategy is not as powerful due to the acoustic similarity of the members of the vocabulary. All of the words are globally similar to some degree and the local differences between speech patterns are weighted equally in the total distance computation. Thus, word differences of minor significance are given a disproportionate contribution to the DTW distance when the size of the dissimilar region is small.

The problem described above is somewhat alleviated by the recognition scheme employed in this research. The {A,J,K}, {B,C,E} types of confusions are reduced by separation of the members with and without fricative components. The stress location helps to further distinguish members of the resultant classes. When a low likelihood rejection threshold is used, class overlap increases and some of the confusions reappear. To improve recognition, the DTW matching procedure needs to be modifiifed in such a way to focus on the parts of the words that contain the differences necessary to discriminate among members of the class. Approaches to this are varied and include compression of stationary regions by trace segmentation prior to DTW as in [96]. Regional emphasis has also been achieved using a network-type word structure [35], partial word matching [36], and locally weighted distances [28] to focus recognition. The procedure used here is essentially an extension of [36] and a simplification of [28].

The techniques of [28] require computation of M(M-1) weighting fuctions, W(m), to emphasize the dissimilar portions of the M words in the class. The weighting functions were derived from the distances between all words in the classes found using a clustering procedure on templates from the alpha-digit task. Use of the weighting functions improved recognition by 3.5 to 6.6 percent. In [36], words from the alpha-digit task were first classified based on their consonant-vowel composition. Then, within class matching was accomplished by warping of only the consonant portion and transition region and discarding the steady state vowel. Recognition performance improved by over 3.5 percent when using this approach on a subset of words containing the vowel /i/, but decreased 1.5 percent when applied to the consonant-vowel subdictionaries.

The approach taken in this study to improve the recognizer's performance on the alpha task also considers the characteristics of the words in the class to be matched. The objective is to find a weighting function, W(n), that will emphasize the local distances in an appropriate manner. The minimuim overall distance of Eq. (6.4) can be modified as

$$D(T, R) = \sum_{n=1}^{N} W(n)d(T(n), R(w(n))) \qquad (6.18)$$

to focus on selected portions of the word. Exactly which region to emphasize is dependent on the word class to which the test utterance belongs. For the alpha task, all words will necessarily belong to an equivalence class with exactly one voiced region. The class may also have an initial or trailing fricative (unvoiced or mixed) region. Whatever the case, all words in a class will have the same acoustic components. Some word pairs will have identical consonants and others vowels. The strategy employed here makes the assumption that transitional regions convey proportionally more discriminative information than do the stationary regions. This has been substantiated with respect to unvoiced-voiced boundaries by studies of formant transitions and gross spectral shape at consonantal release [97,98]. Characteristics of onset spectra have been established that provide cues to the place of articulation [97]. Eleven of the words in the alpha task have an initial fricative and four have a trailing fricative. In the absense of fricative regions, the unstressed parts of the word are considered to be the most likely regions where some degree of nonstationarity may exist.

Though some parts of the utterance may be considered more important than others, all regions retain a nonzero contribution to the overall distance of Eq. (6.18). Thus, one requirement of the weighting function, W(n), is that $W(n) \neq 0$ for $n = 1, 2, \ldots, N$. Values of W(n) are determined solely from characteristics of the test utterance. This gives one weighting function for all words in a class. It is for this reason that all parts of the word must have some influence on the computed distance. Three basic weighting functions were used for all single voiced region classes falling under all voiced, initial fricative, or final fricative categorization. Specifics on the three types of weighting used follow.

6.4.1 All Voiced Case

In the alpha task, the all voiced words are members of the subset {A,E,I,L,M,N,O,R,U,W,Y}. These words either have different vowels or are maximally different in the low energy (unstressed) region. The one exception to this is W. Valid members of this all voiced class are given above but recall that all words from the entire lexicon with the same codeword representation will actually be in the equivalence class. In view of these considerations, the weighting function was chosen so that the "easier to recognize" stressed region is deemphasized. This forces a better match in the possibly nonstationary regions.

The weighting function for this class is given as

$$W(n) = \begin{cases} L/K & ; \quad 1 \leq n < c_i-(NW-1)/2 \\ L/K - \dfrac{1}{K\sigma(2\pi)^{1/2}} e^{-(n-c_i)^2/2\sigma^2} & ; \quad c_i-(NW-1)/2 \leq n \leq c_i+(NW-1)/2 \\ L/K & ; \quad c_i+(NW-1)/2 < n \leq 130 \end{cases} \tag{6.19}$$

where NW is the number of frames deemphasized, c_i is the center of the deemphasized region for the i = 1,2, or 3 different stress possibilities, σ is the broadness parameter of the Gaussian weighted stressed region, and K is the normalization factor. Values for the different parameters were determined based on preliminary experimentation using a number of reasonable combinations. The values chosen are given in Table 6.1.

The stressed region was deemphasized 13.5 percent by finding L such that

$$(130-NW)L = 3\left[(NW)L - \sum_n \frac{1}{\sigma(2\pi)^{1/2}} e^{-(n-c_i)^2/2\sigma^2}\right]$$

or

$$L = \frac{3}{(4(NW)-130)\sigma(2\pi)^{1/2}} \sum_n e^{-(n-c_i)^2/2\sigma^2} . \tag{6.20}$$

Normalization requires $\sum_{n=1}^{130} W(n) = 130$, so then

TABLE 6.1
WEIGHTING FUNCTION PARAMETERS FOR THE ALL VOICED CASE

Parameter	Value
NW	50
σ	10
c_1	40.5
c_2	65.5
c_3	90.5

$$\frac{1}{K}\left[(130-NW)L + \left((NW)L - \frac{1}{\sigma(2\pi)^{1/2}} \sum_{n} e^{-(n-c_i)^2/2\sigma^2}\right)\right] = 130$$

$$K = \frac{130-NW}{65(2NW-65)\sigma(2\pi)^{1/2}} \sum_{n} e^{-(n-c_i)^2/2\sigma^2} \quad . \tag{6.21}$$

The summations in Eqs. (6.20) and (6.21) are over the NW samples $c_i - (NW-1) \leq n \leq c_i + (NW-1)/2$. Examples of this weighting function are shown in Figure 6.3.

6.4.2 Initial Fricative Case

The subset of words from the alpha task that fall in this category is {B,C,D,G,J,K,P,Q,Z}. As discussed earlier, the unvoiced-voiced transitions are the regions chosen for emphasis. Considering this, the following weighting function was used

$$W(n) = \begin{cases} L/K & ; \quad 1 \leq n \leq a \\ L/K + \frac{2}{Kd}(n-a)e^{-(n-a)^2/d} & ; \quad a < n \leq a + NW \\ L/K & ; \quad a + NW < n \leq 130 \end{cases} \tag{6.22}$$

where NW is now the number of frames emphasized, a is the frame just before the emphasized region, d is the broadness parameter for the Rayleigh weighting (the maximum of W(n) is at $a + (d/2)^{1/2}$), and K is again the normalization factor. Frame a was located prior to voice

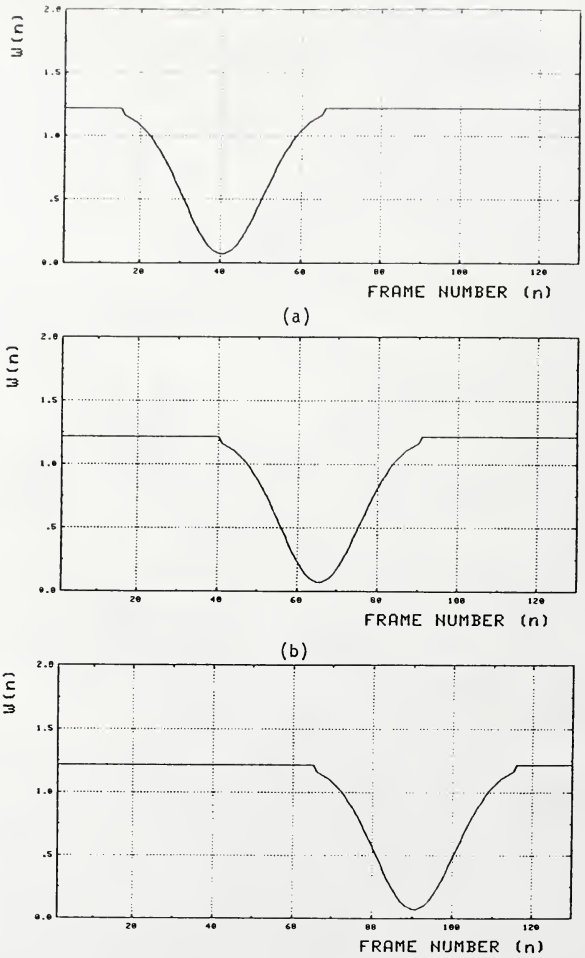


Figure 6.3 Examples of weighting functions for all voiced classes with (a) initial, (b) medial, and (c) final stressed regions.

onset and depended on NW. The parameter values in Table 6.2 specified that the emphasized region include 12 frames of the fricative prior to voice onset and the 17 frames of initial voicing that follow. The Rayleigh function was used since it most heavily weights the frames just around the U-V transition and decreases weighting of the voiced region as the distance from the transition increase. It also conveniently limits emphasis of the preceding fricative frames. The contribution of the emphasized region was increased by 10 percent, thus resulting in

$$L = \frac{2}{25(130-3NW)} \sum_{n} (n-a)e^{-(n-a)^2/d} \tag{6.23}$$

and

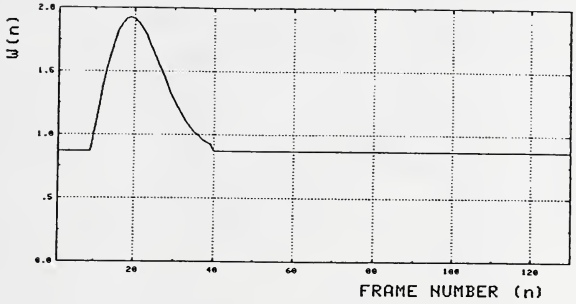
$$K = \frac{3(130-NW)}{130(130-3NW)25} \sum_{n} (n-a)e^{-(n-a)^2/d} \quad . \tag{6.24}$$

6.4.3 Final Fricative Case

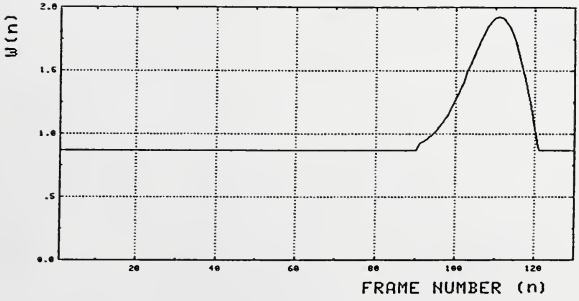
For the words, {F,H,S,X}, with final fricatives, the voiced-to-unvoiced transition was emphasized. The Rayleigh weighting described above was used in a time reversed manner with frame a taken as the 13th frame after voice offset. All of the other parameter values were unchanged. Examples of the weighting functons used for both fricative cases are provided in Figure 6.4.

TABLE 6.2
WEIGHTING FUNCTION PARAMETERS FOR FRICATIVE CLASSES

Parameter	Value
NW	30
d	50
a	onset - 13 (initial fricative)
	offset + 13 (final fricative)



(a)



(b)

Figure 6.4 Typical weighting functions for (a) initial and (b) final fricatives.

CHAPTER 7
EVALUATION OF THE IWR SYSTEM

The various components of the IWR system diagrammed in Figure 2.3 have been described individually in the previous chapters. Preliminary results were presented and procedures for improved recognizer performance were suggested. Yet to be given, however, is a discussion of the system in a more global sense. Overall performance in terms of recognition results is perhaps of major concern. Also of interest are operational characteristics and specifics of implementation. This chapter describes aspects of system evaluation after the processing structure (or layout) is considered.

7.1 System Implementation

The functions depicted in Figure 2.3 were incorporated into a number of FORTRAN 5 main program modules as depicted in the flow diagram of Figure 7.1 Programs were written for a Data General Corporation NOVA 4 minicomputer used in conjunction with an Okidata 169 megabyte fixed disk drive and a DGC 20 megabyte removable disk (10 Mb each) drive. To overcome core limitations, swapping and chaining of routines was necessary. Word recognition proceeded in a fully automatic manner since the appropriate routines were invoked under program control.

Two basic modes of operation existed and are denoted by the R and T levels in Figure 7.1. The modules in the top part of the figure were

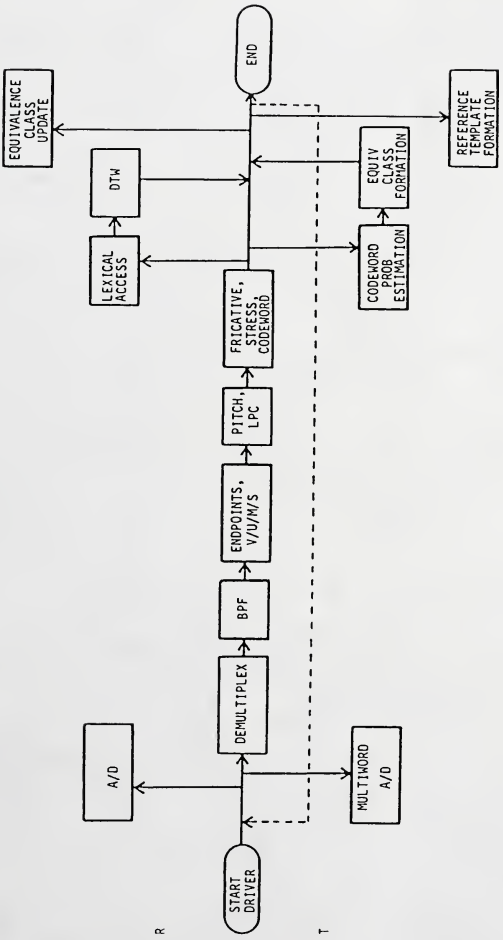


Figure 7.1 Modularized implementation of the IWR system.

run during recognition, and the ones on the botton were used during training. Several modules were common to both modes.

7.1.1 Training Mode

The mode in which the IWR system creates references for all of the words in the vocabulary is known as training. For the fixed vocabulary size this was a one time, off-line process. Training for this system involved the collection and preprocessing of the 100 words spoken twice by each of 6 adult speakers. The speech and EGG data were then used for the V/U/M/S, pitch, and LPC analyses. Codewords were created for each of the 12 renditions of every vocabulary word and the probabilities $p(\underline{c}|\omega_i)$ were also calculated. The codewords and associated probabilities were then used to form the reference database needed for lexical access. Spectral reference templates for the second stage of recognition were created in a separate procedure from the available LPC representations of words.

Since 200 utterances from each of the 14 original speakers resulted in 71.68 megabytes of speech and EGG data, multiword processing was a necessity. Hence, the feedback loop in Figure 7.1 allowed repetition of any or all of the processing. A single run through the 100 word vocabulary took under 25 minutes to collect and about 5 hours to demultiplex and filter. The endpoint detection, V/U/M/S classification, pitch estimation, LPC analysis, and codeword formation, on the other hand, took only 86 minutes for 100 words. Once all 1200 words were processed, the spectral templates took 50 hours of computation, whereas the equivalence classes took a matter of minutes to create.

7.1.2 Recognition (Testing) Mode

For actual word recognition, the system is used in the testing mode. No additional training is required prior to use by a new speaker. The equivalence class and template data were obtained a priori in a speaker independent manner during the training mode. In standard operation, a speaker would produce an utterance in the sound room, and after the sequence of required processing, the recognizer displays the matched word. The intermediate steps of demultiplexing on through codeword formation are the same as for the training mode. Lexical access is just a matter of retrieving the group of words stored for the particular test codeword. The delay encountered before a word is recognized grows linearly with the number of words in the equivalence class to be matched. This is due to the 12 seconds required for each DTW comparison.

Operating in a single word recognition mode is appropriate for demonstration purposes. However, for system evaluation, it is not at all practical to separately speak then recognize all 100 words. Testing is automated by collecting words as for vocabularly development and then sequentially matching each of the stored words without operator intervention. After a full recognition run, the equivalence class information can (optionally) be updated with the stored codewords whether or not the correct word was matched.

7.2 System Evaluation

The bottom line in the evaluation of any speech recognizer is the performance attainable. Several ways of assessing performance of the procedures implemented will be discussed in this section. The overall

performance of the IWR system will be considered after lexical access procedures are evaluated.

7.2.1 Lexical Access

The success of the lexical access procedure is predicated upon accurate and reliable acoustic classification of an utterance. Chapter 4 showed that the EGG is useful for reliable voicing discrimination and thus aids all but the U/S decisions. The data collection environment made U/S discrimination no real problem. Often, though, it is difficult to consistently identify the presence of unvoiced segments of different pronunciations of the same word. This does not necessarily imply that the acoustic classifier failed to perform adequately. Rather, the dramatic intra/inter-speaker variations were to blame. A manual assessment of the classifier's performance indicated a high level of accuracy was possible over all V, U, M and S frames considered by the operator.

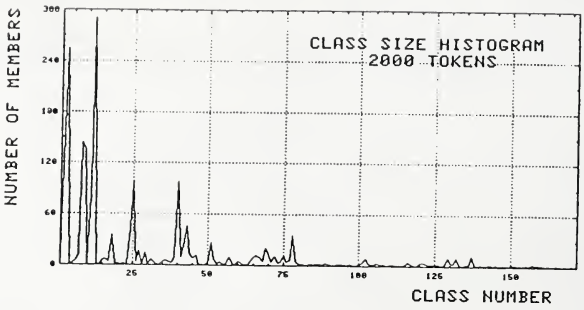
Since the reference codewords are obtained in the same manner as the test codewords, absolute accuracy of the broad acoustic-phonetic representation is not quite as important as consistency. Accuracy is important to the extent that a particular discriminating feature should not be missed. The basic components of the word must be consistently identified in order to achieve the lexical pruning predicted from a manual classification of the words in the vocabulary.

To account for normal variability in pronunciation, this system employs equivalence class updating procedures in addition to the 12 tokens intitially used for each word. Clearly from Figure 5.4, the desire to include all reasonable variations of vocabulary entries leads to significant class overlap. After eight updates, the class size

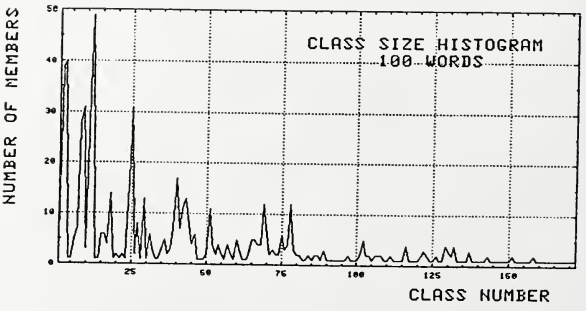
histograms using 2000 tokens are as shown in Figure 7.2. The number of different classes has increased to 172, the average class size to 4.10, and the expected class size to 25.82. The effect of updating is to stablize the probabilities $p(\underline{c}|\omega_i)$. In doing so, all reasonable codeword representations become known for a particular word. This implies that the error in lexical access can be made negligibly small with a sufficient number of "learning" runs. The recognizer's performance will then depend solely on the DTW matching results.

An error is made in lexical access when the class of words returned does not contain the test utterance. The approach in this study has been to avoid this not by using broad word representations, but by accounting for word variability. The end effect is the same in that the expected class size is increased. Alternate word pronunciations, however, need not be considered as match candidates when they occur infrequently. The likelihood rejection threshold proposed in the last chapter serves to decrease the size of the equivalence class by limiting the number of likely candidates.

When only the most likely entries in an equivalence class are used as possible matches, some amount of error will be incurred. The cost/benefit tradeoff can be evaluated in terms of the probability of lexical access error and the expected class size, respectively. An estimate of the error is found empirically by counting the number of errors made for a given threshold using the known equivalence classes. This is essentially a resubstitution estimate [99] since the same data are used to estimate the conditional densities and then to evaluate the error. The estimate becomes more reliable as the number of equivalence class updates increases.



(a)



(b)

Figure 7.2 Equivalence class histograms after eight updates for distinct (a) tokens and (b) words.

The expected class size of Eq. (5.7) can be rewritten as

$$ECS = \sum_i cs(i)p(cs(i))$$

$$= \sum_j M_j p(\underline{c}_j) \qquad (7.1)$$

where M_j is the size of the class corresponding to codeword j and $p(\underline{c}_j)$ is the probability of getting class j. The probability of error is

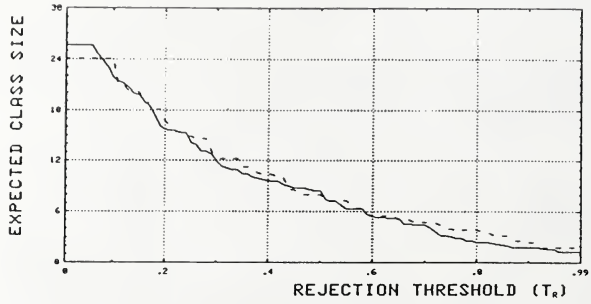
$$PE = \sum_j p(E|\underline{c}_j)p(\underline{c}_j)$$

$$= \sum_j (1 - \sum_i p(\omega_i|\underline{c}_j))p(\underline{c}_j)$$

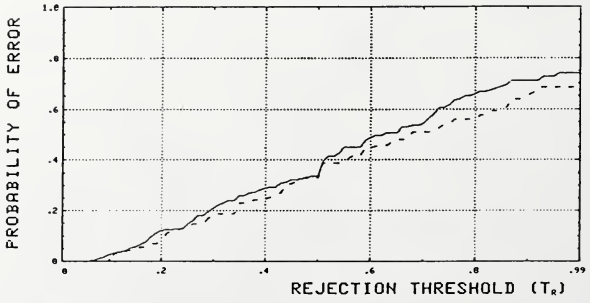
$$= 1 - \sum_j p(\underline{c}_j) \sum_i p(\omega_i|\underline{c}_j) \qquad (7.2)$$

where the index j ranges over all classes and the index i ranges over the members of class j that satisfy Eq. (6.17). The expected class size and the probability of error given in Eqs. (7.1) and (7.2) are both dependent on the number of words in each class. This number, M_j, is directly a function of the likelihood rejection threshold, T_R. Evaluating the equations above for $0 \leq T_R \leq 1 - \Delta T_R$ yields the expected lexical access performance for the entire range of threshold values.

For $\Delta T_R = .01$, the relationships between the expected class size, probability of error, and rejection threshold are illustrated in parts (a) and (b) of Figure 7.3 The dashed line represents performance estimated from the original 1200 tokens and the solid line is after 8



(a)



(b)

Figure 7.3 Expected (a) class size and (b) PE as a function of T_R.

updates (2000 tokens). Notice that the expected class size approaches 1 but does not quite make it at T_R = .99. This is due to equally likely leading entries in some of the classes. Also, the PE is far from 1 at T_R = .99 for the reason above combined with the fact that many classes have only a single entry. The plots also show that the expected class sizes are generally smaller for T_R > .05 after updating than they were prior to doing so. The PE, however, was found to increase for the different values of T_R after updating. The explanation for this is that updating reinforces some preliminary structure in the distributions and also adds more spread with newly encountered pronunciations. Thus, many unlikely variations may increase the probability of error while their individual probabilities are reduced. For the same T_R, fewer words will below to a class since $p(\omega_i|\underline{c})/p(\omega_1|\underline{c})$ tends to become smaller.

With T_R = 0, all known pronunciations are considered as valid and the recognizer will go into the detailed matching stage with a nonzero probability of finding the correct word. A significant amount of variability can still be tolerated when T_R is a small number since it is the cutoff for a likelihood ratio. Examining candidates that may be only 30% as likely as the leading entry reduces the expected class size by half and increases the probability of error to .20. Much of the error, however, occurs in the larger classes since the broadest (or most basic) representations are the ones excluded first. So, the pruning capability of the first pass word representation can be greatly enhanced by requiring only slightly more controlled pronuniciations.

7.2.2 Pattern Matching

The codeword representation of the various acoustic components in the utterance is not necessarily unique. In fact, the word-to-codeword transformation is expected to be a many-to-one mapping. After eight updates, 48% of the equivalence classes have only one member. Another 18% of the classes have just two members. However, when the likelihoods of the classes are considered, the expected size of the equivalence class dramatically increases. The job of the second stage of recognition is to perform the detailed pattern matching necessary to distinguish between equivalence class members. Matching of the correct reference to the test pattern is performed using the DTW techniques described in the previous chapter. The results of the procedures utilized are discussed in this section for different subsets of the vocabulary.

Given an adequate number of preliminary recognition sessions, probability density estimates improve and insignificant erorr is made on lexical access. Due to the difficulty (storage, processing, etc.) associated with complete 100 word runs, evaluation was limited to eight such sessions. The average number of first pass errors reduced by 50% over the last four speakers, to about twelve. Since there is no chance for correct recognition if the test utterance does not belong to the equivalence class, the second stage's performance was gauged after updating of the various classes. Performance of the recongizer on the entire 100 word vocabulary given in Table 3.1 will be considered first.

7.2.2.1 Overall results

Recognition results were obtained for 4 male and 4 female speakers in a multiword test mode. Two of the speakers were from the original set of 6 used for vocabulary development (training) and the other 6 were totally unknown to the system. New data were collected from the male and female used for training so none of the test tokens were used for template formation. In the automatic test mode, it was not possible to respeak an utterance if no references were sufficiently close to it. Therefore, the decision rule simply ordered the list of references by closeness to the test pattern and then the position of the correct template was stored. This approach effectively forced a decision for every utterance.

The results from the 100 word recognition experiments are summarized in Table 7.1. The first two entries are for the male and female speakers that were also used for training. Following these are results for the three new males and then the three new females. No significant difference in performance exists for the male and female speakers. Better overall results were obtained for the first two speakers thus indicating residual speaker dependent characteristics. Observation of the DTW distances showed that the correct candidate was frequently within a small range of the top candidate. In the single word mode, all close candidates would be output. The automatic test mode just ranks all references numerically. The average error rates are seen to be about 34% for the top match and 24% for the two best candidates. The recognition scores in Table 7.1 are appropos for the single template speaker independent pattern matching technique used. The particularly complex alpha-digit task included in the 100 word

TABLE 7.1
RECOGNITION ACCURACY FOR THE 100 WORD VOCABULARY

	Accuracy (%)	
Speaker	Top Candidate	Two Top Candidates
RH	73.91	81.52
LC	75.27	83.87
JA	58.62	72.41
JL	63.29	72.15
LS	71.26	81.61
AC	64.29	71.43
JC	65.31	73.47
DH	60.22	73.12
Average	66.48	76.30

vocabulary also serves to degrade recognition performance. Results for when this difficult subset is excluded will be considered next.

7.2.2.2 64 word subset

When the digits zero-nine, and the letters of the alphabet are excluded from the test set, recognition scores noticably improve. Table 7.2 shows that the average error rates decreased to 26% and 17% for the closest and two closest reference templates, respectively. Even greater improvements are expected if the alpha-digit subset is also excluded from the reference set. The results presented are for a test set with fewer confusable entries, however, all words are still matched with the original equivalence class members derived from the 100 word vocabulary. Confusable monosyllabic entries (e.g., on-in-N, load-low-O, etc.) were found to be the primary sources of error. The 64 word subset is 28% monosyllabic whereas 51% of the entries in the original set have one syllable. The poorest results were obtained when this subset was excluded from the original test set. Recognition results for the complex alpha-digit task were also obtained using the references derived from the entire 100 word vocabulary. This represents increased difficulty over the standard task since many other confusable entries are added to the various classes.

7.2.2.3 Alpha-digit task

The useful, but difficult to recognize alpha-digit subset was included in the vocabulary for evaluation purposes. Though other confusable entries appear in the equivalence classes, rough performance comparisons are possible with test results for this 36 word task. Results for the letters of the alphabet are given in Table 7.3 and the

TABLE 7.2
RECOGNITION ACCURACY FOR THE 64 WORD SUBSET

Speaker	Accuracy (%)	
	Top Candidate	Two Top Candidates
RH	85.71	92.86
LC	82.81	90.63
JA	63.16	78.95
JL	68.08	74.47
LS	81.13	86.79
AC	70.91	76.36
JC	72.58	82.26
DH	66.67	77.19
Average	74.06	82.93

TABLE 7.3
RECOGNITION ACCURACY FOR THE LETTERS OF THE ALPHABET

Speaker	Accuracy (%)	
	Top Candidate	Top Two Candidates
RH	50.00	53.85
LC	54.16	58.33
JA	42.86	52.38
JL	31.82	50.00
LS	44.00	64.00
AC	40.91	54.55
JC	46.15	50.00
DH	42.31	65.38
Average	44.27	56.25

digit results are shown in Table 7.4. Performance for the letters and digits is assessed separately to illustrate the great complexity of the alpha subset. The combined alpha-digit task gave an average error rate of 47%, which is comparable to that found in [27] when each word was represented by one template.

In Section 6.4, modifications to the standard DTW approach were suggested that would focus attention on certain parts of the word patterns for the alpha task. The specifications given minimally emphasize or deemphasize selected regions to improve performance for known classes of words. The local distance weighting procedure starts with the understanding that the DTW algorithm will warp the appropriate reference regions onto the test utterance. Failure to do so may result in errors that appear to indicate that the wrong portion of the test utterance was emphasized.

A summary of recognition results using class specific weighting for the alpha task is presented in Table 7.5. The table shows that selected emphasis increased the error rate only slightly for one of the subjects. Data were available for 6 of the 8 subjects used in the other tests. The results in the table consider the two top templates for a match. When recognition was changed by weighting, results were improved by an average of 4%. The improvements in recognition were important but also of interest was the fact that an average of 20% of the words exhibited a relative change in score when the word was not recognized. Of the words that changed, almost equal numbers moved in either direction. The scores that improved, however, moved up an average of twice as far as those that moved down when performance declined.

TABLE 7.4
RECOGNITION ACCURACY FOR THE DIGITS

Speaker	Accuracy (%)	
	Top Candidate	Top Two Candidates
RH	90.00	90.00
LC	80.00	100.00
JA	77.78	77.78
JL	62.50	87.50
LS	89.89	88.89
AC	85.71	85.71
JC	70.00	80.00
DH	70.00	70.00
Average	77.94	83.82
Combined Alpha-Digit Average	53.08	63.95

TABLE 7.5
RECOGNITION ACCURACY FOR LETTERS OF THE ALPHABET

	Accuracy (%)	
Speaker	Class Weighting	T_R=.30
RH	61.54	57.69
JA	52.38	52.38
LS	60.00	60.00
AC	54.55	54.55
JC	57.69	65.38
DH	69.23	73.08
Average	59.59	60.96

Another consideration for the second stage of recognition is the effect of a nonzero likelihood rejection threshold, T_R, on overall matching performance. Section 7.2.1 showed that response time is expected to improve since class sizes will be reduced. Also, the probability of lexical access error increases and thereby decreases the probability of a correct match after the second pass. The results for the locally weighted alpha task with T_R = .30 are given in the right hand column of Table 7.5. An error was made when the correct word was excluded from the class or when the wrong pattern was matched. One observation from this test was that the correct pattern was excluded from consideration due to T_R 17% of the time yet the error rate increased only slightly for one speaker. This suggests that the unlikely pronunciations for a word (as indicated by the codeword) also have correspondingly harder to match spectral representations. Another result to support this is that performance actually improved for two speakers. By eliminating some of the spectral overlap due to word variability, the intraclass pattern matching confusions were also reduced.

CHAPTER 8
CONCLUDING REMARKS

Several aspects of this study are reviewed in this chapter. After some discussion of recognition concepts, modifications are proposed for future investigations. Lastly, the general results of this work are summarized.

8.1 Discussion

A particularly difficult vocabulary was used to evaluate the isolated word recognition system. If the confusability was reduced by replacing the alpha-digit subset with multisyllabic words, recognition results would undoubtably be better. The results presented are thus fairly conservative performance indicators.

Alternate approaches may yield better results for all or part of the 100 word vocabulary used here, but performance may not be vocabulary independent. Recall that the intended application of the techniques presented in this work is for large vocabulary, speaker independent tasks. A system performing well for one small set of words may be useless for large vocabularies if the techniques employed are not directly extensible. The present system was tested on a 100 word vocabulary due to resource limitations; however, the recognition concepts apply to larger lexicons.

The primary aspect of large vocabulary IWR addressed in this work was the reduction of system response time. Closely related to this

problem were the issues of storage limitations and overall accuracy. Within the template matching framework, system response time was shortened as the number of reference patterns examined was reduced. The approach taken achieved this in two ways. First, the number of reference patterns was kept to a minimum by creating a single averaged template from many different tokens of the same word. Therefore, only one reference template was examined for each match candidate. This sped up recognition and reduced the reference storage requirements. The number of references patterns was also reduced by allowing only some subset of the lexicon to be searched for a match. As long as the correct word was in the subset, no accuracy was lost. When the correct word was not among those considered, recognition of the utterance spoken was not possible.

This study demonstrated that in addition to robust detection of the first pass features for lexical access, normal variations in pronunciation must be accounted for. This was necessary in order to avoid irrecoverable errors from the first stage. The cost of no lexical access error was a reduction in pruning power of the first pass representation.

Accuracy was somewhat comprimised by the use of a single reference template. Computation and storage, however, were the foremost concerns. The recognition strategy was to search slightly larger equivalence classes so that negligible probabiltiy of lexical access error existed. To do so, few templates per reference could be examined. Only one reference was used to allow for large classes. Using knowledge of the general characteristics of the word class, second stage matching was to be modified to provide greater discriminability

between the members. This was implemented for the word classes corresponding to the alpha task. Results showed that the class specific weighting could improve performance for globally similar words. Increasing the likelihood rejection threshold produced more homogeneous equivalence classes.

8.2 Research Extensions

The recognizer heretofore described is an experimental system designed to study effective lexical access procedures for large lexicons. For a very large number of words (20,000), the proportion of monosyllabic entries drops as the number of multisyllabic words increases. This would serve to balance the distribution of words across equivalence classes and keep the expected class size relatively small. The preliminary work in [49] showed that effective pruning of a 20,000 word lexicon can be achieved using a similar broad characterization of the utterance. The present study showed that results from word transcriptions are highly optomistic unless pronunciation variability is considered. Therefore, future work directed towards increasing the dimension of the first pass representation may be necessary if the classes grow too large. Additional robustly detected features with adequate discriminating power would keep the search space managable.

Keeping the number of references examined to a minimum is important but a storage problem may exist due to just the large number of lexical entries. Using a lower order LPC analysis and a shorter standard utterance length are ways to greatly reduce the memory requirements. The recommended approach, however, is to also vector quantize the test and reference templates. A reference pattern could then be stored as a

sequence of indices, and distances can be precomputed. Since pattern comparisons would be faster and memory requirements reduced, multiple reference templates can be stored to improve the detailed matching results.

Another important continuation of this work is to study class specific matching tecniques. The information known about the test utterance from the first recognition pass need not be discarded after lexical access. Local weighting techniques can be used to improve DTW matching or class dependent features can be extracted to move away from the template based procedures.

8.3 Summary

This study considered problems associated with large vocabulary IWR within the context of a template matching scheme. A two-channel, two-pass approach was used in which a broad initial classification of the utterance preceded detailed pattern matching. The electroglottographic signal facilitated endpoint detection and acoustic classification in the first pass where a robustly obtained word representation was used to access the lexicon. Knowledge of the word class and lexical structure aided the DTW comparisons in the second stage.

The design, implementation, and testing of the modular isolated word recognizer resulted in the presentation of new procedures for 1) two channel endpoint location, V/U/M/S classification, and pitch tracking, 2) a robust segmental and suprasegmental word representation, 3) lexical structure and organization, 4) match limiting after lexical access, and for 5) class specific local weighting. The two-channel, two-pass scheme was shown to yield a previously reported level of

recognition accuracy for a subset of the vocabulary. The major feature, however, was seen to be the substantial lexical pruning possible for the varying amounts of lexical access error specified by a likelihood rejection threshold.

APPENDIX A
LPC ANALYSIS AND DISTANCE MEASURE

Many speech recognition systems characterize utterances based on information derived from a linear predictive coding (LPC) analysis of the acoustic signal. The popularity of this approach warrants the dedication of at least a few pages of an appendix to show exactly where the feature vectors of LPC coefficients come from, and how pattern similarity is determined. This will be the extent of the review. Comprehensive treatments of these topics can be found in [12,78,86], and [100].

A.1 Linear Prediction

A simplified model of speech production is shown in Figure A.1. The excitation u(n) represents either a quasiperiodic train of impulses or random noise when the speech output s(n) is voiced or unvoiced, respectively. For both types of excitation, the choice

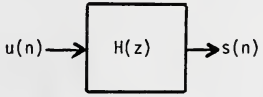


Figure A.1 Discrete speech production model.

of parameters for the filter H(z) will determine the characteristics of the sound generated. It is desirable, then, to develop a parametric model for the behavior of s(n) for applications such as prediction or data compression.

Thus, let s(n) be the output of a system with unknown input u(n) such that

$$s(n) = - \sum_{k=1}^{p} a_k s(n-k) + Gu(n) \tag{A.1}$$

where a_k, $1 \leq k \leq p$, and G are the system parameters. From Eq. (A.1), the output, s(n), is seen to be predictable from a linear combination of P past outputs and the present input u(n).

The transfer function of the system of Figure A.1 is obtained by Z transforming and rearranging Eq. (A.1) to get

$$H(z) = \frac{G}{1 + \sum_{k=1}^{p} a_k z^{-k}} \quad . \tag{A.2}$$

The all pole model of Eq. (A.2) has been by far the most commonly used representation of the system in Figure A.1. Given samples of the acoustic signal, the problem is to determine the model parameters in some manner. The coefficients a_k will be found using the least squares approach in [100] for a deterministic signal.

Assuming that the input u(n) is unknown, the signal s(n) is predicted only from p past samples

$$\hat{s}(n) = - \sum_{k=1}^{p} a_k s(n-k) \tag{A.3}$$

and the prediction error, or residual, is

$$e(n) = s(n) - \hat{s}(n)$$

$$= s(n) + \sum_{k=1}^{p} a_k s(n-k) \quad . \tag{A.4}$$

In the method of least squares, the parameters a(k) are found by minimizing the total squared error with respect to each of the parameters. The total squared error is given by

$$E = \sum_{n} e^2(n)$$

$$= \sum_{n} (s(n) + \sum_{k=1}^{p} a_k s(n-k))^2 \quad . \tag{A.5}$$

The range of summation in Eq. (A.5) will be specified later. First, E is minimized by setting

$$\frac{\partial E}{\partial a_i} = 0, \quad 1 \leq i \leq p \quad . \tag{A.6}$$

From Eqs. (A.5) and (A.6),

$$\frac{\partial E}{\partial a_i} = 0 = \frac{\partial}{\partial a_i} [\sum_{n} (s(n) + \sum_{k=1}^{p} a_k s(n-k))^2]$$

where, for $1 \leq i \leq p$,

$$0 = 2 \sum_{n} s(n)s(n-i) + \sum_{n} [\sum_{k=1}^{p} a_k s(n-k)s(n-i) + \sum_{k=1}^{p} a_k s(n-k)s(n-i)]$$

or,

$$\sum_{k=1}^{p} a_k \sum_n s(n-k)s(n-i) = -\sum_n s(n)s(n-i), \quad 1 \leq i \leq p \quad . \tag{A.7}$$

Equations (A.7) are often referred to as the normal equations. This set of p equations in p unknowns can be solved for the predictor coefficients, a_k, that minimize E in Eq. (A.5).

The minimum total squared error, E_{min}, is found expanding Eq. (A.5),

$$E = \sum_n s^2(n) + 2 \sum_{k=1}^{p} a_k \sum_n s(n)s(n-k) + \sum_{j=1}^{p} a_j \sum_{k=1}^{p} a_k \sum_n s(n-k)s(n-j)$$

and then substituting in Eq. (A.7),

$$E_{min} = \sum_n s^2(n) + 2 \sum_{k=1}^{p} a_k \sum_n s(n)s(n-k) - \sum_{j=1}^{p} a_j \sum_n s(n)s(n-j)$$

$$= \sum_n s^2(n) + \sum_{k=1}^{p} a_k \sum_n s(n)s(n-k) \tag{A.8}$$

Two distinct methods for estimation of the a_k's result from different ways in which the range of summation over n in Eqs. (A.5), (A.7), and (A.8) is specified.

In the autocorrelation method, the error in Eq. (A.5) is minimized over all samples (i.e., $-\infty < n < \infty$), and Eqs. (A.7) and (A.8) reduce to

$$\sum_{k=1}^{p} a_k R(i-k) = - R(i), \quad 1 \leq i \leq p \tag{A.9}$$

and

$$E_{min} = R(0) + \sum_{k=1}^{p} a_k R(k) \quad , \tag{A.10}$$

respectively, where

$$R(i) = \sum_{n=-\infty}^{\infty} s(n)s(n+i) \tag{A.11}$$

is the even autocorrelation function of s(n). Assuming that the signal s(n) is nonzero only over a finite interval, say $0 \leq n \leq N-1$, a short-time analysis technique results. The autocorrelation function would then be given by

$$R(i) = \sum_{n=0}^{N-1-i} s(n)s(n+i) \quad . \tag{A.12}$$

The autocorrelation matrix formed by the R(i-k)'s is symmetric and Toeplitz. This special form allows solution of the autocorrelation normal equations (A.9) using a computationally efficient procedure such as Durbin's algorithm.

The second method, known as the covariance method, assumes that the error E in (A.5) is minimized over some finite interval $0 \leq n \leq N-1$. Equations (A.7) and (A.8) then reduce to

$$\sum_{k=1}^{p} a_k \phi(k,i) = - \phi(k,i), \quad 1 \leq i \leq p \tag{A.13}$$

and

$$E_{min} = \phi(0,0) + \sum_{k=1}^{p} a_k \phi(0,k) \quad , \tag{A.14}$$

respectively, where

$$\phi(i,k) = \sum_{n=0}^{N-1} s(n-i)s(n-k) \qquad (A.15)$$

is the even covariance function of a zero mean signal. The $\phi(i,k)$'s form is a symmetric non-Toeplitz matrix. The normal equations of (A.13) can be solved using a method such as Cholesky decomposition. In contrast to the autocorrelation method the covariance method makes no assumptions about the signal, s(n), other than that of all needed samples are known. As the range of values for n is made infinitely large, the covariance and autocorrelation methods become equivalent.

The autocorrelation method is used more frequently in recognition systems for several reasons. One reason is that the poles of H(z) will lie within the unit circle in the Z plane. This method therefore guarantees stability of H(z) if sufficient computational accuracy is used. Another advantage is that efficient algorithms exist for the solution of the normal equations in (A.7). The autocorrelation method also has a convenient interpretation in the spectral domain.

A.2 LPC Distance Measure

To match feature vectors of LPC coefficients, many IWR systems use a distance based on the one proposed by Itakura [12]. The development of the measure follows a maximum liklihood approach using the multidimensional Gaussian probability distribution governing the estimates, $\underline{\hat{a}} = (1, \hat{a}_1, \hat{a}_2, \ldots, \hat{a}_p)$, of the true coefficients, $\underline{a} = (1, a_1, a_2, \ldots, a_p)$.

The distance measure is obtained from the logarithm of the distribution,

$$P(\hat{\underline{a}}/\underline{a}) = \frac{1}{(2\pi)^{P/2} (\det S)^{1/2}} e^{-1/2(\hat{\underline{a}}-\underline{a})S^{-1}(\hat{\underline{a}}-\underline{a})'} \tag{A.16}$$

where S is the covariance matrix

$$S = \frac{R-1}{N} \hat{\underline{a}} R \hat{\underline{a}}' \quad , \tag{A.17}$$

and R is the correlation matrix of the speech segment with components

$$r(k) = \sum_{n=1}^{N-k} s(n)s(n+k) \quad . \tag{A.18}$$

The resulting metric,

$$d(\hat{\underline{a}},\underline{a}) = (\hat{\underline{a}}-\hat{\underline{a}}) \left[N \frac{R}{\hat{\underline{a}} R \hat{\underline{a}}'}\right] (\hat{\underline{a}}-\underline{a})' \quad , \tag{A.19}$$

decreases as the probability of getting $\hat{\underline{a}}$ when $\underline{a}$ are the true coefficients increases.

The log likelihood distance measure,

$$d(\hat{\underline{a}},\underline{a}) = \log \left[\frac{\underline{a} R \underline{a}'}{\hat{\underline{a}} R \hat{\underline{a}}'}\right] \quad , \tag{A.20}$$

was proposed by Itakura for computational reasons. Since $r(k) = r(-k)$, the numerator of Eq. (A.20) can be evaluated as

$$\underline{a} R \underline{a}' = \sum_{i=0}^{p} \sum_{j=0}^{p} a_i r(k-j) a_j \quad . \tag{A.21}$$

After a simple change in variable and suitable definition of a and r, Eq. (A.21) can be rewritten as

$$\underline{a}\, R\, \underline{a}' = \sum_{k=-p}^{p} r(k) \sum_{i=0}^{p} a_i a_{i+k}$$

$$= r(0) \sum_{i=0}^{p} a_i^2 + 2 \sum_{k=1}^{p} r(k) \sum_{i=0}^{p} a_i a_{i+k}$$

$$= \sum_{k=0}^{p} r(k) r_a(k) \quad , \tag{A.22}$$

where

$$r_a(k) = \begin{cases} \sum_{i=0}^{p} a_i^2 & ; \quad k = 0 \\ 2 \sum_{i=0}^{p-k} a_i a_{i+k} & ; \quad 1 \leq k \leq p \end{cases} \tag{A.23}$$

is the autocorrelation of the true LPC coefficients.

The quantity $\hat{\underline{a}}\, R\, \hat{\underline{a}}'$ in the denominator of Eq. (A.20) is just the total squared prediction error associated with the estimates $\hat{a}_k$. The measure of Eq. (A.20) can therefore be computed as

$$d(\hat{\underline{a}}, \underline{a}) = \log \left[\sum_{k=0}^{p} \frac{r(k)}{E} r_a(k) \right] \quad , \tag{A.24}$$

which requires (p+1) multiplications and additions, and one log when r(k)/E and $r_a(k)$ are the test and reference features, respectively [6].

APPENDIX B
APPLICATION OF ERROR CORRECTING CODES TO IWR

Some preliminary ideas regarding the use of error correcting codes in IWR are presented in this appendix. The intended application was for efficient lexical access once the first pass features have been extracted. Constraints imposed on the codewords, however, made the number of detected features inadequate for any advantage to be realized with this approach. By modifying the feature extraction and word representation procedures, the coding interpretation could conceivably prove useful in an IWR scheme.

B.1 Coding Theory Approach

Starting with the assumption that the same acoustic-phonetic word representation of Chapter 5 is used, the problem is to match the incoming test codeword with the stored patterns to determine which word the input is closest to. The first step is to transform the VUMSFs representation to a binary number. One way is to represent each of the six categories with its binary equivalent. A similar 18 bit description can be obtained for all of the reference words stored. The binary representation can be considered as a codeword and known methods of decoding can be used. Error correction capability is dependent upon the minimum distance of the resulting code.

For a linear block code, the parity check matrix H is needed in order to decode received vectors. This can be found from the generator matrix G, of the code if it is in a special form. That form is

$$G = [I_k | -A'] \tag{B.1}$$

where k is the dimension of the code. All that is known is the set of codewords and some basic knowledge. Namely, the set of all possible codewords is the space spanned by the rows of the generator matrix, G. The rows of G are, therefore, a set of basis vectors and the message bits are the basis coefficients for the codeword. Since the basis vectors of a linear space are not unique, many G matrices can give the same set of codewords. This says that any maximal set of linearly independent codewords taken from a given code can be used as the rows of a generator matrix for that code. Further, row operations on G do not alter the set of codewords generated. They just relabel the codewords since the message bits multiplying the generator matrix take on all possible combinations of values. Gauss-Jordan reduction can thus be performed on any generator matrix to put it into row echelon form.

Stacking the known codewords, $\underline{x}_i$, i=1, 2, ..., 100 as

$$G^* = \begin{bmatrix} \underline{x}_1 \\ \underline{x}_2 \\ \vdots \\ \underline{x}_{100} \end{bmatrix} \tag{B.2}$$

gives a starting matrix which can be reduced to a form like

$$G^* = \left[\frac{I_k | A'}{0}\right] = \left[\frac{G}{0}\right] . \tag{B.3}$$

The reduced matrix in Eq. (B.3) indicates two things. First, k is the number of linearly independent rows in G. This is the dimension of the code so there can be a maximum of 2^k codewords generated by this G. Next, the parity check matrix is found as

$$H = [A|I_{n-k}] \quad . \tag{B.4}$$

This is what is needed for decoding purposes.

Before discussing decoding, several points should be made. First, the generator matrix can always be reduced to the echelon form since Rank G $\leq$ n, where n is the length of the codewords. The minimum distance, d, of the code can be found from G* since

$$d = \min_{\underline{x} \in C, \underline{x} \neq 0} \text{wt } (\underline{x}) \tag{B.5}$$

and G* is a display of all existing codewords. The minimum distance of the code is just the minimum weight of any nonzero codeword. So, knowing both n and d before G* is reduced, k can be found since n-k $\geq$ d-1 for a [n,k,d] code [101]. This will indicate whether k is large enough to uniquely generate all of the words in the vocabulary. If not, this procedure can still be used to group the words into a number of equivalence classes.

For decoding, the defining equations of the code (the parity check equations) are utilized. These equations say that the parity checks are chosen so that the codewords satisfy $H\underline{x}' = 0$. Thus, the code vectors are in the null space of the parity check matrix. If a received vector $\underline{y}$ is multiplied by H, the result is $\underline{0}$ only if $\underline{y}$ is error free. So if

the channel causes an arbitrary sequence of errors $\underline{e} = (e_1, e_2, \ldots, e_n)$, $\underline{y} = \underline{x}+\underline{e}$ and the syndrome is

$$\underline{s} = H\underline{y}' = H(\underline{x}'+\underline{e}') = H\underline{x}'+H\underline{e}' = H\underline{e}' \quad . \tag{B.6}$$

In the present case, the possible channel errors are incorrect V/U/M/S classifications or incorrect fricative or stress locations. If the syndrome is zero, no errors were made and the received vector is the codeword. If the syndrome is not zero, it is equal to the sum of the columns of the parity check matrix where errors occurred.

With $\underline{y} = \underline{x}+\underline{e}$, the error is $\underline{e} = \underline{y}+\underline{x}$ and there will be 2^k possible decodings of $\underline{y}$. Maximum likelihood or minimum distance decoding says to choose the codeword corresonding to the smallest weight vector among the set $\{\underline{x}_m + \underline{y}\}$. The procedure is

a. Store the minimum weight vector $\underline{e}(s)$ satisfying $\underline{s} = H\underline{e}'$ for each of the 2^{n-k} possible syndromes in a table of 2^{n-k} n-bit entries.

b. Compute the syndrome $\underline{s}$ for the received vector $\underline{y}$.

c. Do a table lookup in step a to obtain $\underline{e} = \underline{e}(s)$ from $\underline{s}$.

d. Recover the most likely codeword as $\underline{x} = \underline{y}+\underline{e}$.

The maximum likelihood decoder will decode correctly if $wt(\underline{e}) < [\frac{d-1}{2}]$. This says that the code will correct [(d-1)/2] errors if d is the minimum distance. The decoding procedure grows in complexity as 2^{n-k} since the n dimensional error vectors must be stored.

The known codewords $\underline{x}_i$ used in Eq. (B.2) were obtained a priori in a training session. They are assumed to be from an expurgated linear code since for dimension k, 2^k vocabulary entries should exist. The

formulation above considers any received (test) codeword as having an error component if it has a nonzero syndrome. If more than $[\frac{d-1}{2}]$ errors occurred, the decoding procedure may mistakenly identify $\underline{y}$ as some other allowable codeword. This is where the difficulty arises with the number of features extracted for lexical access.

The minimum distance of the code must be large enough to allow correction of several errors in order to benefit from this approach. From Eq. (B.6), the obvious solution is to increase the weight of all codewords. What is not obvious is exactly how this should be done. A possibility is to systematically repeat portions of the codewords. This will increase d but the errors will show the same systematic increase so nothing will be gained. Any type of transformation of the features already extracted will have a similar result. The only appropriate way to increase d appears to be by lengthening the codeword with new information independent of components of the existing representation.

If an adequate number of features can be obtained reliably, the coding approach can be used in a single pass sytem for efficient word recognition. After feature extraction, recognition would consist of only a matrix multiplication, a table lookup, and then addition of two vectors. Computation can be (and has been) reduced by implementing all of the matrix operations in a way that considers each bit of an integer word as an element of a binary matrix.

REFERENCES

[1] J.R. Pierce, "Whither speech recognition?" (Letter), J. Acoust. Soc. Amer., vol. 46, no. 4, p. 1049, October 1969.

[2] E.P. Neuberg, "Needs vs. competence in speech recognition," Trends in Speech Recognition, W. Lea, Ed., Englewood Cliffs, N.J., Prentice-Hall, pp. 19-23, 1980.

[3] J.M. Baker, "How to achieve recognition: A tutorial/status report on automatic speech recognition," Speech Technology, vol. 1, no. 1, pp. 30-43, Fall 1981.

[4] T. Edman, "Speech recognition," Scientific Honeyweller, vol. 16, no. 3, pp. 44-55, 1982.

[5] W. Lea, Ed., Trends in Speech Recognition, Englewood Cliffs, N.J., Prentice-Hall, 1980.

[6] L.R. Rabiner and S.E. Levinson, "Isolated and connected word recognition - theory and selected applications," IEEE Trans. Comm., vol. COM-29, no. 5, pp. 621-659, May 1981.

[7] D.H. Klatt, "Review of the ARPA speech understanding project," J. Acoust. Soc. Amer., vol. 62, pp. 1345-1366, December 1977.

[8] G.E. Peterson and H.L. Barney, "Control methods used in a study of vowels," J. Acoust. Soc. Amer., vol. 24, pp. 175-184, March 1952.

[9] B.A. Dautrich, L.R. Rabiner, and T.B. Martin, "On the use of filter bank features for isolated word recognition," Proc. IEEE Conf. Acoust., Speech, Signal Processing, Boston, Mass., pp. 1061-1064, April 1982.

[10] G.M. White and R.B. Neely, "Speech recognition experiments with linear prediction, bandpass filtering, and dynamic programming," IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP-24, pp. 183-188, April 1976.

[11] D.H. Klatt, "A digital filter bank for spectral matching," Proc. IEEE Conf. Acoust., Speech, Signal Processing, Philadelphia, Pa., pp. 537-540, April 1976.

[12] F. Itakura, "Minimum prediction residual principle applied to speech recognition," IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP-23, pp. 67-72, February 1975.

[13] K. Elenius and M. Blomberg, "Effects of emphasizing transitional or stationary parts of the speech signal in a discrete utterance recognition," Proc. IEEE Conf. Acoust., Speech, Signal Processing, Paris, France, pp. 535-537, May 1982.

[14] S.K. Das, "Some experiments in discrete utterance recognition," IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP-30, no. 5, pp. 766-770, October 1982.

[15] M.H. Kuhn and H.H. Tomaschewski, "Improvements in isolated word recognition," IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP-31, no. 1, pp. 157-167, February 1983.

[16] J.M. Tribolet, L.R. Rabiner, and J.G. Wilpon, "An improved model for isolated word recognition," Bell Syst. Tech. J., vol. 61, no. 9, pp. 2289-2312, November 1982.

[17] M.K. Brown and L.R. Rabiner, "On the use of energy in LPC-based recognition of isolated words," Bell Syst. Tech. J., vol. 61, no. 10, pp. 2971-2987, December 1982.

[18] H. Sakoe and C. Chiba, "Dynamic programming algorithm optimization for spoken word recognition, "IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP-26, pp. 43-49, February 1978.

[19] J.L. Gauvain, J. Mariani, and J.S. Lienard, "On the use of time compression for word-based recognition," Proc. IEEE Conf. Acoust., Speech, Signal Processing, Boston, Mass., pp. 1029-1032, April 1983.

[20] C.C. Tappert and S.K. Das, "Memory and time improvements in a dynamic programming algorithm for matching speech patterns," IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP-26, no. 6, pp. 583-586, December 1978.

[21] G.M. White, "Dynamic programming, the Viterbi algorithm, and low cost speech recognition," Proc. IEEE Conf. Acoust., Speech, Signal Processing, Tulsa, Ok., pp. 413-417, April 1978.

[22] M.K. Brown and L.R. Rabiner, "An adaptive, ordered, graph search technique for dynamic time warping for isolated word recognition," IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP-30, no,. 4, pp. 535-544, August 1982.

[23] M. Kuhn, H. Ney, and H. Tomaschewski, "Fast nonlinear time alignment for isolated word recognition," Proc. IEEE Conf. Acoust., Speech, Signal Processing, Atlanta, Ga., pp. 736-740, March 1981.

[24] M. Wagner, "Automatic labelling of continuous speech with a given phonetic transcription using dynamic programming algorithms," Proc. IEEE Conf. Acoust., Speech, Signal Processing, Atlanta, Ga., pp. 1156-1159, March 1981.

[25] M. Jalanko, "Studies of learning projective methods in automatic speech recognition," Ph.D. dissertation, Dept. of Technical Physics, Helsinki Univ. of Technology, Otaniemi, Finland, 1980.

[26] T.B. Martin and J.R. Welch, "Practical speech recognizers and some performance effectiveness parameters," Trends in Speech Recognition, W. Lea, Ed., Englewood Cliffs, N.J., Prentice-Hall, pp. 24-38, 1980.

[27] L.R. Rabiner and J.G. Wilpon, "Considerations in applying clustering techniques to speaker-independent word recognition," J. Acoust. Soc. Amer., vol. 66, no. 3, pp. 663-673, September 1979.

[28] L.R. Rabiner and J.G. Wilpon, "A two-pass pattern-recognition approach to isolated word recognition," Bell Syst. Tech. J., vol. 60, no. 5, pp. 739-766, May-June 1981.

[29] L.R. Rabiner, S.E. Levinson, and M.M. Sondhi, "On the application of vector quantization and hidden Markov models to speaker-independent, isolated word recognition," Bell Syst. Tech. J., vol. 62, no. 4, pp. 1075-1105, April 1983.

[30] A.E. Rosenberg, L.R. Rabiner, and J.G. Wilpon, "Recognition of spoken, spelled names for directory assistance using speaker-independent templates," Bell Syst. Tech. J., vol. 59, no. 4, pp. 571-592, April 1980.

[31] M.R. Sambur and L.R. Rabiner, "A speaker-independent digit-recognition system," Bell Syst. Tech. J., vol. 54, no. 1, pp. 81-102, January 1975.

[32] E.C. Bronson, "Syntactic pattern recognition of discrete utterances," Proc. IEEE Acoust., Speech, Signal Processing, Boston, Mass., pp. 719-722, April 1983.

[33] R.A. Cole, R.M. Stern, M.S. Phillips, S.M. Brill, A.P. Pilant, and P. Specker, "Feature-based speaker-independent recognition of isolated English letters," Proc. IEEE Conf. Acoust., Speech, Signal Processing, Boston, Mass, pp. 731-733, April 1983.

[34] R. Bakis and N.R. Dixon, "Toward speaker-independent recognition-by-synthesis," Proc. IEEE Conf. Acoust., Speech, Signal Processing, Paris, France, pp. 566-569, May 1982.

[35] R.K. Moore, M.J. Russel, and M.J. Tomlinson, "The discriminative network: A mechanism for focusing recognition in whole-word pattern matching," Proc. IEEE Conf. Acoust., Speech, Signal Processing, Boston, Mass., pp. 1041-1044, April 1983.

[36] L.F. Lamel and V.W. Zue, "Performance improvement in a dynamic-programming-based isolated word recognition system for the alpha-digit task," Proc. IEEE Conf. Acoust., Speech, Signal Processing, Paris, France, pp. 558-561, May 1982.

[37] T. Kaneko and N.R. Dixon, "A hierarchical decision approach to large vocabulary discrete utterance recognition," IEEE Trans. Acoust., Speech, Signal Processing, vol. 31, no. 5, pp. 1061-1066, October 1983.

[38] C.S. Myers and L.R. Rabiner, "An automated directory listing retrieval system based on recognition of connected letter strings," J. Acoust. Soc. Amer. vol. 71, no. 3, pp. 716-727, March 1982.

[39] A. Buzo, A.J. Gray, R.M. Gray, and J.D. Markel, "Speech coding based upon vector quantization," IEEE Trans. Acoust., Speech, Signal Processing, vol. 28, no. 5, pp. 562-574, October 1980.

[40] L.R. Rabiner, K.C. Pan, and F.K. Soong, "On the performance of isolated word speech recognizers using vector quantization and temporal energy contours," Bell Syst. Tech. J., vol. 63, no. 7, pp. 1245-1261, September 1984.

[41] J.E. Shore and D.K. Burton, "Discrete utterance speech recognition without time alignment," IEEE Trans. Info. Theory, vol. 29, no. 4, pp. 473-491, July 1983.

[42] D.K. Burton, J.E. Shore, and J.T. Buck, "Isolated-word speech recognition using multi-section VQ code books, Naval Res. Lab. report 5367, July 1984.

[43] S.E. Levinson, L.R. Rabiner, and M.M. Sondhi, "Speaker independent isolated digit recognition using hidden Markov models," Proc. IEEE Conf. Acoust., Speech, Signal Processing, Boston, Mass., pp. 1049-1052, April 1983.

[44] L.R. Rabiner, S.E. Levinson, M.M. Sondhi, "On the use of hidden Markov models for speaker independent recognition of isolated words from a medium size vocabulary," Bell Syst. Tech. J., vol. 63, no. 4, pp. 627-643, April 1984.

[45] J.G. Wilpon, L.R. Rabiner, and A. Bergh, "Speaker-independent isolated word recognition using a 129-word airline vocabulary," J. Acoust. Soc. Amer., vol. 72, no. 2, pp. 390-396, August 1982.

[46] A.E. Rosenberg, L.R. Rabiner, J.G. Wilpon, and D. Kahn, "Demisyllable based isolated word recognition systems," IEEE Trans. Acoust., Speech, Signal Processing, vol. 31, no. 3, pp. 713-726, June 1983.

[47] L.R. Rabiner, A.E. Rosenberg, J.G. Wilpon, and W.J. Keilin, "Isolated word recognition for large vocabularies," Bell Sys. Tech. J., vol. 61, no. 10, pp. 2989-3005, December 1982.

[48] J.F. Mari and J.P. Haton, "Some experiments in automatic recognition of a thousand word vocabulary," Proc. IEEE Conf. Acoust., Speech, Signal Processing, San Diego, Ca., pp. 46.6.1-26.6.4, April 1984.

[49] D.W. Shipman and V.W. Zue, "Properties of large lexicons: Implications for advanced word recognition systems," Proc. IEEE Conf. Acoust., Speech, Signal Processing, Paris, France, pp. 546-549, May 1982.

[50] A. Waibel, "Towards very large vocabulary word recognition," Speech Project, Computer Sci. Dept., CMU, Pittsburgh, Pa., November 1982.

[51] D.P. Huttenlocher and V.W. Zue, "A model of lexical access from partial phonetic information," Proc. IEEE Conf. Acoust., Speech, Signal Processing, San Diego, Ca., pp. 26.4.1-26.4.4, April 1984.

[52] A.J. Fourcin and E. Abberton, "First application of a new laryngograph," Med. Biol. Illus., vol. 21, p. 172, 1971.

[53] D.G. Childers, "Laryngeal pathology detection," Crit. Rev. Bioeng., vol. 2, no. 4, p. 375, 1977.

[54] D.G. Childers and A.K. Krishnamurthy, "A critical review of electroglottography," CRC Crit. Rev. Bioeng., vol. 12, no. 2, pp. 131-161, 1985.

[55] D.G. Childers and J.N. Larar, "Electroglottography for laryngeal function assessment and other applications," IEEE Trans. Bioeng., vol. 31, no. 12, pp. 807-816, December 1984.

[56] J.J. Yea, A.K. Krishnamurthy, J.M. Naik, G.P. Moore, and D.G. Childers, "Glottal sensing for speech analysis and synthesis," Proc. Int. Conf. Acoust., Speech, Signal Processing, Boston, Mass., pp. 1332-1335, 1983.

[57] J.N. Larar, Y.A. Alsaka, and D.G. Childers," Variability in closed phase analysis of speech," Proc. IEEE Conf. Acoust., Speech, Signal Processing, Tampa, Fla., pp. 1089-1092, March 1985.

[58] A.K. Krishnamurthy, "Two channel (speech and EGG) analysis for formant tracking and glottal inverse filtering," Proc. IEEE Conf. Acoust., Speech, Signal Processing, San Diego, Ca., pp. 36.6.1-36.6.4, March 1984.

[59] J.M. Naik, "Synthesis and evaluation of natural sounding speech using the linear predictive analysis-synthesis scheme," Ph.D. Dissertation, Univ. of Florida, 1984.

[60] A.K. Krishnamurthy, "Study of vocal fold vibration and the glottal sound source using synchronized speech, electroglottography, and ultra-high speed laryngeal films," Ph.D. Dissertation, Univ. of Florida, 1983.

[61] E. Abberton, "Listener identification of speakers from larynx frequency," Proc. 8th Int. Congr. Acoust., London, Chapman & Hall, p. 273, 1974.

[62] T. Baer, I. Titze, and H. Yoshioka, "Multiple simultaneous measures of vocal fold activity," Vocal fold physiology: contemporary research and clinical issues, D.M. Bless and J.H. Abbs, Eds., San Diego, College Hill Press, p. 229, 1983.

[63] L. Boves and B. Cranen, "Evaluation of glottal inverse filtering by means of physiological registrations," Proc. IEEE Conf. Acoust., Speech, Signal Processing, Paris, France, p. 1988, May 1982.

[64] M. Fog-Pedersen, "Electroglottography compared with synchronized stroboscopy in normal persons," Folia Phoniatr., vol. 29, p. 191, 1977.

[65] D.G. Childers, G.P. Moore, J.M. Naik, J.N. Larar, and A.K. Krishnamurthy, "Assessment of laryngeal function by simultaneous, synchronized measurement of speech, electroglottography, and ultra-high speed film," Transcripts of the Eleventh Symposium Care of the Professional Voice, The Julliard School, New York; II, Medical/Surgical Sessions: Papers, L. Van Lawrence, Ed., The Voice Foundation, p. 234, June 1982.

[66] D.G. Childers, J.M. Naik, J.N. Larar, A.K. Krishnamurthy, and G.P. Moore, "Electroglottography, speech, and ultra-high speed cinematography," presented at the Int. Conf. Physiol. Biophys. Voice, University of Iowa, Iowa City, May 4-7, 1983.

[67] Y. Chen, "Vocabulary selection for high performance speech recognition," Proc. IEEE Conf. Acoust., Speech, Signal Processing, Boston, Mass., pp. 757-760, April, 1983.

[68] Y. Lee, H.F. Silverman, and N.R. Dixon, "Preliminary results for an operational definition and methodology for predicting large vocabulary DUR confusability from phonetic transcriptions," Proc. IEEE Conf. Acoust., Speech, Signal Processing, San Diego, Ca., pp. 26.2.1-26.2.4, April 1984.

[69] M.G. Berouti, D.G. Childers, and A. Paige, "A correction of tape recorder distortion," Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing, pp. 397-400, 1977.

[70] A. Oppenheim and R. Shaffer, Digital Signal Processing, Englewood Cliffs, N.J., Prentice-Hall, 1975.

[71] L.F. Lamel and V.W. Zue, "Performance improvement in a dynamic-programming-based isolated word recognition system for the alpha-digit task," Proc. IEEE Conf. Acoust., Speech, Signal Processing, Paris, France, pp. 558-561, May 1982.

[72] L.R. Rabiner and M.R. Sambur, "An algorithm for determining the endpoints of isolated utterances," Bell Syst. Tech. J., vol. 54, no. 2, pp. 297-315, February 1975.

[73] B.S. Atal and L.R. Rabiner, "A pattern recognition approach to voiced-unvoiced-silence classification with applications to speech recognition," IEEE Trans. Acoust., Speech, Signal Processing, vol. 24, no. 3, pp. 201-212, 1976.

[74] Daaboul and J.P. Adoul, "Parametric segmentation of speech into voiced-unvoiced-silence intervals," Proc. IEEE Conf. Acoust., Speech, Signal Processing, Hartford, Conn., pp. 327-331, May 1977.

[75] L.R. Rabiner, C.E. Schmidt, and B.S. Atal, "Evaluation of a statistical approach to voiced-unvoiced-silence analysis for telephone quality speech," Bell Syst. Tech. J., vol. 56, no. 3, pp. 455-482, March 1977.

[76] L.R. Rabiner and M. Sambur, "Voiced-unvoiced-silence detection using the Itakura distance measure," Proc. IEEE Conf. Acoust., Speech, Signal Processing, Hartford, Conn., pp. 323-326, May 1977.

[77] L.J. Siegel and A.C. Bessey, "Voiced/unvoiced/mixed excitation classification of speech," IEEE Trans. Acoust., Speech, Signal Processing, vol. 30, no. 3, pp. 451-460, June 1982.

[78] L. Rabiner and R. Shaffer, Digital Processing of Speech Signals, Englewood Cliffs, N.J., Prentice-Hall, 1978.

[79] W.J. Hess, "Algorithms and devices for pitch determination of speech signals," Phonitica, pp. 219-240, 1982.

[80] L.R. Rabiner, M.J. Cheng, A.E. Rosenberg, and C.A. McGonegal, "A comparative performance study of several pitch detection algorithms," IEEE Trans. Acoust., Speech, Signal Processing, vol. 24, no. 5, pp. 399-418, 1976.

[81] M.M. Sondhi, "New methods of pitch extraction," IEEE Trans. Audio Electroacoust., vol. 16, pp. 262-266, June 1968.

[82] J.D. Markel, "The SIFT algorithm for fundamental frequency estimation," IEEE Trans. Audio Electroacoust., vol. 20, pp. 367-377, December 1972.

[83] S. Chandra and W. Lin, "Experimental comparison between stationary and nonstationary formulations of linear prediction applied to voiced speech analysis," IEEE Trans. Acoust., Speech, Signal Processing, vol. 22, no. 6, pp. 403-415, December 1974.

[84] L.R. Rabiner, B.S. Atal, and M. Sambur, "LPC prediction error - analysis of its variation with the position of the analysis frame," IEEE Trans. Acoust., Speech, Signal Processing, vol. 25, no. 5, pp. 434-442, October 1977.

[85] L.R. Rabiner, J.G. Wilpon, and J.G. Ackenhusen, "On the effects of varying analysis parameters on an LPC-based isolated word recognizer," Bell Syst. Tech. J., vol. 60, no. 6, pp. 893-911, July-August 1981.

[86] J. Markel and A. Gray, Linear Prediction of Speech, New York, N.Y., Springer Verlag, 1976.

[87] A. Waibel, "Suprasegmentals in very large vocabulary isolated word recognition," Proc. IEEE Conf. Acoust., Speech, Signal Processing, San Diego, Ca., pp. 26.3.1-26.3.4, April 1984.

[88] W.A. Lea, "Prosodic aids to speech recognition," Trends in Speech Recognition, W.A. Lea, Ed., Englewood Cliffs, N.J., Prentice-Hall, pp. 166-205, 1980.

[89] W.A. Lea, "Prosodic correlates of linguistic structure," Topics in Speech Science, D.J. Broad, Ed., Santa Barbara, Ca., Speech Comm. Res. Lab., p. 394, 1977.

[90] J.T. Tou and R.C. Gonzalez, Pattern Recognition Principles, Reading, Mass., Addison-Wesley Pub. Co., 1974.

[91] C.S. Myers, L.R. Rabiner, and A.E. Rosenberg, "Performance tradeoffs in dynamic time warping algorithms for isolated word recognition," IEEE Trans. Acoust., Speech, Signal Procesing, vol. ASSP-28, pp. 622-635, December 1980.

[92] L.R. Rabiner, A.E. Rosenberg, and S.E. Levinson, "Considerations in dynamic time warping for discrete word recognition," IEEE Trans. Acoust., Speech, Signal Processing, vol. 26, pp. 575-582, December 1978.

[93] L.R. Rabiner and J.G. Wilpon, "A simplified, robust training procedure for speaker trained, isolated word recognition systems," J. Acoust. Soc. Amer., vol. 68, no. 5, pp. 1271-1276, November 1980.

[94] S.E. Levinson, L.R. Rabiner, A.E. Rosenberg, and J.G. Wilpon, "Interactive clustering techniques for selecting speaker-independent reference templates for isolated word recognition," IEEE Trans. Acoust., Speech, Signal Processing, vol. 27, pp. 134-141, 1979.

[95] M.R. Sambur and L.R. Rabiner, "A statistical decision approach to the recognition of connected digits," IEEE Trans. Acoust., Speech, Signal Processing, vol. 24, pp. 550-558, 1976.

[96] M. Kuhn, H. Ney, and H. Tomaschewski, "Fast nonlinear time alignment for isolated word recognition," Proc. IEEE Conf. Acoust., Speech, Signal Processing, Atlanta, Ga., pp. 736-740, March 1981.

[97] K.N. Stevens and S.E. Blumstein, "Invariant cues for place of articulation in stop consonants," J. Acoust. Soc. Amer., vol. 64, no. 5, pp. 1358-1368, November 1978.

[98] D. Kewley-Port, "Time-varying features as correlates of place of articulation in stop consonants," J. Acoust. Soc. Amer., vol. 73, no. 1, pp. 322-335, January 1983.

[99] P.A. Lachenbruch and R.M. Mickey, "Estimation of error rates in discriminant analysis," Technometrics, vol. 10, pp. 1-11, 1968.

[100] J. Makhoul, "Linear prediction: A tutorial review," Proc. IEEE, vol. 63, pp. 561-580, April 1975.

[101] F.J. MacWilliams and N.J. Sloane, The Theory of Error Correcting Codes, New York, North Holland, 1977.

BIOGRAPHICAL SKETCH

Jerry N. Larar was born in Washington, D.C., on February 3, 1960. He received the Bachelor of Science degree in electrical engineering from Florida Atlantic University, Boca Raton, in March 1980.

Since 1980, he has been with the Mind-Machine Interaction Research Center at the University of Florida, Gainesville. In December, 1981, he received the Master of Engineering degree in electrical engineering. After completing the requirements for the Ph.D. degree in electrical engineering, the author will join the technical staff of AT&T Bell Laboratories, in Murray Hill, N.J.

Mr. Larar is a member of Phi Kappa Phi, Tau Beta Phi, Eta Kappa Nu, and is a student member of the I.E.E.E.

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Donald G. Childers

Donald G. Childers, Chairman
Professor of Electrical Engineering

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Leon W. Couch

Leon W. Couch, II
Professor of Electrical Engineering

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

G. Paul Moore

G. Paul Moore
Distinguished Service Professor
Emeritus of Speech

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Howard B. Rothman

Howard B. Rothman
Associate Professor of Speech

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Jack R. Smith

Jack R. Smith
Professor of Electrical Engineering

This dissertation was submitted to the Graduate Faculty of the College of Engineering and to the Graduate School, and was accepted as partial fulfillment of the requirements for the degree of Doctor of Philosophy.

May, 1985

Hubert A. Bewe

Dean, College of Engineering

Dean for Graduate Studies and Research