

# **EVALUATION OF GLOTTAL CHARACTERISTICS FOR SPEAKER IDENTIFICATION**

A thesis presented for the degree of  
Doctor of Philosophy  
in Electrical and Electronic Engineering  
at the University of Canterbury,  
Christchurch, New Zealand.

by  
Andrew George Elder  
B. E. (Hons 1)  
June 1992.



---

## ABSTRACT

Based on the assumption that the physical characteristics of people's vocal apparatus cause their voices to have distinctive characteristics, this thesis reports on investigations into the use of the long-term average glottal response for speaker identification. The long-term average glottal response is a new feature that is obtained by overlaying successive vocal tract responses within an utterance.

The way in which the long-term average glottal response varies with accent and gender is examined using a population of 352 American English speakers from eight different accent regions. Descriptors are defined that characterize the shape of the long-term average glottal response. Factor analysis of the descriptors of the long-term average glottal responses shows that the most important factor contains significant contributions from descriptors comprised of the coefficients of cubics fitted to the long-term average glottal response. Discriminant analysis demonstrates that the long-term average glottal response is potentially useful for classifying speakers according to their gender, but is not useful for distinguishing American accents.

The identification accuracy of the long-term average glottal response is compared with that obtained from vocal tract features. Identification experiments are performed using a speaker database containing utterances from twenty speakers of the digits zero to nine. Vocal tract features, which consist of cepstral coefficients, partial correlation coefficients and linear prediction coefficients, are shown to be more accurate than the long-term average glottal response. Despite analysis of the training data indicating that the long-term average glottal response was uncorrelated with the vocal tract features, various feature combinations gave insignificant improvements in identification accuracy.

The effect of noise and distortion on speaker identification is examined for each of the features. It is found that the identification performance of the long-term average glottal response is insensitive to noise compared with cepstral coefficients, partial correlation coefficients and the long-term average spectrum, but that it is highly sensitive to variations in the phase response of the speech transmission channel.

Before reporting on the identification experiments, the thesis introduces speech production, speech models and background to the various features used in the experiments. Investigations into the long-term average glottal response demonstrate that it approximates the glottal pulse convolved with the long-term average impulse response, and this relationship is verified using synthetic speech. Furthermore, the spectrum of the long-term average glottal response extracted from pre-emphasized speech is shown to be similar to the long-term average spectrum of pre-emphasized speech, but computationally much simpler.



---

## ACKNOWLEDGEMENTS

In the course of my research and thesis preparation I have been fortunate to receive assistance, advice, support and encouragement from many people.

I would particularly like to thank my supervisor, the late Professor Richard Bates, for his guidance and encouragement during the first stages of my research. His energy, enthusiasm and prolificacy were inspirational. I also thank Mr Bill Kennedy for his guidance and advice throughout my research and for the supervision that he provided since Professor Bates died. Dr Kathy Garden's insights were also much appreciated.

Many people have assisted in the massaging of this thesis into coherent English and I would like thank Mr Bill Kennedy, Professor Richard Bates, and Dr Kathy Garden for their comments. The proof-reading by friends and colleagues has also improved the readability of this document - thank you.

Throughout my research I enjoyed stimulating discussions with Speech Group members: William Thorpe, Tracy Clark, Catherine Watson, John Kirkland, Lim Ching Aun and Nigel Brieseman. Drs. Harsha Sirisena and Murray Smith gave valuable advice on statistical aspects of my work. Electrical Engineering Department staff members Mike Cusdin, Dave van Leeuwen and Mike Shurety have always willingly helped with any hardware or computing difficulties.

I acknowledge financial support from the Telecom Corporation of New Zealand Ltd.

Good friends, near and far away, flatmates (a large number over the years), fellow postgrads, particularly current members of R6 and past members of R4, have all contributed towards making my time as a postgrad enjoyable. I am especially grateful for the support offered me at times when my thesis seemed daunting. A healthy number of excursions to the hills have also helped maintain my (in)sanity.

Last, but not least, I would like to acknowledge the support of my family and I thank my parents for their love and understanding.



---

**CONTENTS**

|  |      |
|--|------|
| <b>PREFACE</b>                                     | xiii |
| <b>GLOSSARY OF ABBREVIATIONS</b>                   | xix  |
| <b>CHAPTER 1 SPEECH SOUNDS AND CHARACTERISTICS</b> | 1    |
| 1.1 Physiology of speech production                | 1    |
| 1.2 Types of speech sounds                         | 4    |
| 1.2.1 Phonemes                                     | 4    |
| 1.2.2 Voiced sounds                                | 4    |
| 1.2.3 Unvoiced sounds                              | 4    |
| 1.2.4 Aspirated sounds                             | 5    |
| 1.2.5 Stop consonants                              | 5    |
| 1.2.6 Nasals                                       | 6    |
| 1.3 Speech characteristics                         | 7    |
| 1.3.1 Loudness and intensity                       | 7    |
| 1.3.2 Pitch variations                             | 7    |
| 1.3.3 Frequency content of speech                  | 7    |
| 1.3.4 Emotional effects                            | 9    |
| 1.4 Voice quality                                  | 10   |
| <b>CHAPTER 2 SPEECH ANALYSIS AND MODELLING</b>     | 13   |
| 2.1 Signal analysis                                | 13   |
| 2.1.1 The Fourier transform                        | 13   |
| 2.1.1.1 Useful properties of the Fourier transform | 14   |
| 2.1.1.2 The discrete Fourier transform             | 15   |
| 2.1.2 The Z-Transform                              | 16   |
| 2.1.3 Sampling considerations                      | 16   |
| 2.2 The source filter model                        | 19   |
| 2.2.1 Source models                                | 19   |
| 2.2.2 Filter models                                | 21   |
| 2.3 Acoustic model of the vocal tract              | 23   |
| 2.3.1 Simplifying assumptions                      | 24   |
| 2.3.2 A single tube                                | 24   |
| 2.3.3 Lattice formulation                          | 24   |
| 2.3.4 Area function                                | 27   |
| 2.4 Prosodic characteristics                       | 28   |
| 2.4.1 Speech segmentation                          | 28   |
| 2.4.2 Intensity                                    | 29   |
| 2.4.3 Pitch estimation                             | 29   |
| 2.4.3.1 Time domain methods                        | 29   |
| 2.4.3.2 Frequency domain methods                   | 30   |

|  |           |
|--|-----------|
| 2.4.3.3 Hybrid frequency domain and time domain methods            | 31        |
| 2.4.4 Voiced/unvoiced decision methods                             | 32        |
| 2.5 Linear predictive coding of speech                             | 33        |
| 2.5.1 An historical perspective                                    | 34        |
| 2.5.2 Lip and glottal effects                                      | 35        |
| 2.5.3 Linear prediction  | 36        |
| 2.5.3.1 Solution of LPC equations                                  | 37        |
| 2.5.4 Lattice filtering  | 38        |
| 2.6 Spectral estimation  | 40        |
| 2.6.1 Fast Fourier Transform                                       | 40        |
| 2.6.2 Linear predictive coding                                     | 41        |
| 2.6.2.1 Formants   | 42        |
| 2.6.3 Cepstral analysis  | 44        |
| 2.7 Vector quantization  | 45        |
| 2.7.1 Introduction   | 46        |
| 2.7.1.1 The information source                                     | 46        |
| 2.7.1.2 Preliminary notation                                       | 47        |
| 2.7.1.3 A quantizer for speech transmission                        | 48        |
| 2.7.2 Advantages of vector quantization                            | 49        |
| 2.7.2.1 Vector properties  | 49        |
| 2.7.2.2 Theoretical performance                                    | 51        |
| 2.7.3 Distortion measures  | 52        |
| 2.7.4 Quantizer design   | 55        |
| 2.7.4.1 Notation   | 55        |
| 2.7.4.2 Centroid calculation                                       | 56        |
| 2.7.4.3 Iterative codebook design                                  | 58        |
| 2.7.4.4 Aspects of codebook storage and codebook searching         | 61        |
| 2.8 Shift-and-add  | 63        |
| 2.8.1 Historical background  | 64        |
| 2.8.2 SAA processing of speech                                     | 64        |
| 2.8.3 Computation details of the SAA algorithm                     | 65        |
| 2.8.4 The effect of pre-emphasis on the LTAGR                      | 68        |
| 2.8.5 Incorporation of the voicing decision into SAA               | 70        |
| 2.8.6 Relationship to the source filter model                      | 71        |
| 2.9 Measures of speech noise                                       | 72        |
| 2.9.1 Signal-to-noise ratio  | 72        |
| 2.9.2 Speech correlated noise                                      | 74        |
| 2.10 Summary   | 75        |
| <b>CHAPTER 3 SPEAKER RECOGNITION FUNDAMENTALS</b>                  | <b>77</b> |
| 3.1 Introduction   | 77        |
| 3.1.1 Terminology  | 77        |
| 3.1.2 Fingerprints vs. voiceprints                                 | 79        |
| 3.1.3 Applications   | 82        |
| 3.2 Factors that affect recognition performance by human listeners | 84        |
| 3.2.1 Familiarity  | 84        |
| 3.2.2 Duration   | 85        |
| 3.2.3 Pitch  | 85        |
| 3.2.4 Perceptual factors   | 86        |
| 3.3 Classes of techniques useful for speaker recognition           | 87        |
| 3.3.1 Dynamic techniques   | 87        |

|         |  |     |
|---------|--|-----|
| 3.3.1.1 | Hidden Markov Models                                       | 88  |
| 3.3.1.2 | Dynamic time warping                                       | 90  |
| 3.3.2   | Statistical techniques                                     | 91  |
| 3.3.3   | Vector quantization techniques                             | 92  |
| 3.4     | Statistical methods for assessing features                 | 95  |
| 3.4.1   | F-ratio  | 96  |
| 3.4.2   | Discriminant analysis                                      | 97  |
| 3.4.2.1 | Stepwise inclusion of features                             | 99  |
| 3.4.3   | Factor analysis  | 99  |
| 3.4.4   | Factor rotation  | 100 |
| 3.5     | Comparative performance of specific features               | 101 |
| 3.5.1   | Statistical vs dynamic techniques                          | 101 |
| 3.5.2   | Transitional vs instantaneous features                     | 102 |
| 3.5.3   | Vocal tract features                                       | 102 |
| 3.5.4   | Glottal flow   | 103 |
| 3.5.5   | The effect of removing glottal characteristics             | 103 |
| 3.5.6   | The effectiveness of different phonemes                    | 104 |
| 3.5.7   | Feature spacing and dimension                              | 104 |
| 3.6     | Effect of voice distortion on speaker recognition accuracy | 105 |
| 3.6.1   | Mimicry  | 105 |
| 3.6.2   | Disguise   | 105 |
| 3.6.3   | Noise and transmission distortion                          | 106 |
| 3.6.4   | Voice variation with time                                  | 107 |
| 3.6.5   | Health   | 108 |
| 3.7     | Real-time speaker verification                             | 108 |
| 3.8     | Summary  | 110 |

## CHAPTER 4 FEATURES USED IN IDENTIFICATION EXPERIMENTS

|         |   |     |
|---------|---|-----|
| 4.1     | The speech database   | 111 |
| 4.1.1   | Observations on the speech database   | 113 |
| 4.2     | Vocal tract features  | 113 |
| 4.2.1   | Computation   | 113 |
| 4.2.2   | Vector quantization of vocal tract features   | 115 |
| 4.2.2.1 | Distortion and centroid computation   | 115 |
| 4.2.2.2 | Centroid splitting  | 116 |
| 4.2.2.3 | Verification of the VQ training algorithm   | 117 |
| 4.2.2.4 | Examples of applying the VQ training algorithm to speech                              | 118 |
| 4.3     | The long-term average glottal response  | 119 |
| 4.3.1   | Real-time shift-and-add   | 120 |
| 4.3.1.1 | Hardware  | 120 |
| 4.3.1.2 | Real-time shift-and-add implementations   | 121 |
| 4.3.1.3 | Comments  | 123 |
| 4.3.2   | Descriptors of the long-term average glottal response                                 | 123 |
| 4.3.3   | Variation of the long-term average glottal response with changes in accent and gender | 127 |
| 4.3.3.1 | Factor analysis   | 128 |
| 4.3.3.2 | Discriminant analysis   | 131 |
| 4.4     | Long-term average spectrum  | 136 |
| 4.4.1   | Introduction  | 136 |

|   |   |            |
|---|---|------------|
| 4.4.2   | Comparison of various LTAS calculation methods  | 136        |
| 4.4.2.1   | LTAS methods  | 136        |
| 4.4.2.2   | Effects of removing unvoiced speech and silent periods from the speech                      | 137        |
| 4.4.2.3   | Pitch synchronous spectral estimation   | 138        |
| 4.4.2.4   | The effect of pre-emphasis  | 140        |
| 4.4.3   | Other applications of the LTAS  | 140        |
| 4.5   | The spectrum of the long-term average glottal response                                      | 141        |
| 4.5.1   | Calculation of the spectrum of the long-term average glottal response                       | 141        |
| 4.5.1.1   | Comparison of the spectrum of the LTAGR and the LTAS  | 142        |
| 4.5.1.2   | Computational requirements  | 144        |
| 4.6   | Summary   | 146        |
| <b>CHAPTER 5 SPEAKER IDENTIFICATION EXPERIMENTS</b> |   | <b>149</b> |
| 5.1   | Terminology   | 149        |
| 5.2   | Training of speaker templates   | 149        |
| 5.2.1   | Normalization of the LTAGR and LTAS-VE  | 150        |
| 5.2.2   | Analysis of speaker templates   | 150        |
| 5.2.3   | Correlation between intraspeaker and interspeaker distances for entire speech               | 154        |
| 5.3   | Statistical significance of identification results  | 154        |
| 5.3.1   | A distribution for modelling the identification error rate                                  | 154        |
| 5.3.2   | Binomial confidence limits  | 156        |
| 5.3.3   | Comparisons between two systems   | 158        |
| 5.4   | Evaluation of different speaker identification systems                                      | 160        |
| 5.4.1   | Identification using the entire utterance   | 160        |
| 5.4.2   | Comparison of identification using voiced and entire utterances                             | 163        |
| 5.4.2.1   | Performance of the LTAGR and LTAS-VE  | 165        |
| 5.4.2.2   | Various LTAGR based methods   | 167        |
| 5.4.3   | The effect of varying pre-emphasis on vocal tract features                                  | 169        |
| 5.4.4   | Combining features  | 169        |
| 5.4.4.1   | Method 1 (distance normalization, all features)   | 170        |
| 5.4.4.2   | Method 2 (distance normalization, CEP and LTAGR)  | 171        |
| 5.4.4.3   | Method 3 (feature weighting with intraspeaker distance)                                     | 171        |
| 5.4.4.4   | Method 4 (modified feature weighting)   | 171        |
| 5.4.4.5   | Method 5 (presort using LTAGR)  | 171        |
| 5.4.4.6   | Discussion of feature combination results   | 173        |
| 5.5   | Factors that reduce the accuracy of speaker identification                                  | 173        |
| 5.5.1   | The effect of noise   | 174        |
| 5.5.1.1   | Gaussian noise  | 174        |
| 5.5.1.2   | Speech correlated noise   | 174        |
| 5.5.2   | The effect of non-ideal frequency response  | 176        |
| 5.5.2.1   | The telephone channel   | 176        |
| 5.5.2.2   | Magnitude   | 177        |
| 5.5.2.3   | Phase   | 180        |
| 5.5.3   | Summary of the effects of noise and frequency response distortion on speaker identification | 182        |
| 5.6   | Computational requirements  | 184        |
| 5.6.1   | Feature calculation   | 184        |

|   |   |            |
|---|---|------------|
| 5.6.2   | Distance measures   | 185        |
| 5.7   | Summary   | 186        |
| <b>CHAPTER 6 CONCLUSIONS AND SUGGESTIONS FOR FURTHER RESEARCH</b> |   | <b>187</b> |
| 6.1   | Conclusions   | 187        |
| 6.1.1   | The long-term average glottal response  | 187        |
| 6.1.1.1   | Analysis of the long-term average glottal responses belonging to a large database | 188        |
| 6.1.2   | Speaker identification performance  | 188        |
| 6.1.2.1   | Analysis of templates formed from the training data                               | 189        |
| 6.1.2.2   | Evaluation of different speaker identification features                           | 189        |
| 6.1.2.3   | Sensitivity of features to noise and distortion                                   | 190        |
| 6.2   | Suggestions for further research  | 191        |
| 6.2.1   | The long-term average glottal response  | 191        |
| 6.2.2   | Speaker identification  | 192        |
| 6.2.2.1   | Methods of improving the accuracy of results                                      | 192        |
| 6.2.2.2   | Weighting individual samples in the LTAGR   | 192        |
| 6.2.2.3   | Further evaluation of the LTAGR for presorting the speaker population             | 193        |
| 6.2.2.4   | A method of assessing identification accuracy                                     | 193        |
| 6.2.2.5   | An alternative VQ structure   | 194        |
| 6.2.2.6   | Future trends   | 196        |
| <b>REFERENCES</b>   |   | <b>197</b> |

CONTENTS  
INTRODUCTION  
REFERENCES  
ACKNOWLEDGEMENTS  
NOTES

---

## PREFACE

The research reported in this thesis stems from the application of techniques developed in one research discipline to another discipline. In particular, an astronomical image processing technique is applied to speech to extract a long-term average glottal response as a new feature for speaker identification. The technique was suggested by my supervisor, the late Professor R.H.T. Bates, who was actively involved in the area of general inverse problems (especially computed tomography and ultrasonic imaging), various biomedical applications, radio antenna engineering and speech processing. It was from amongst this broad range of research areas that my research project arose.

When I began my postgraduate studies there were three significant areas of research being pursued by Professor Bate's Speech Group that had a direct bearing on my research activity. These fell broadly under the headings of speech synthesis, speech therapy and, as it was then called, extraction of the 'glottal pulse'. I will discuss each of these briefly in the following paragraphs.

In the area of speech synthesis I investigated different methods of producing good quality speech modelled by linear predictive coefficients(LPC) and examined the effect of varying parameters such as the number of poles, the glottal excitation, the window, and so on. All of the work was performed using a signal processing program, called SGPRC, which was written by Nigel Brieseman, a postgraduate student at the time. I wrote SGPRC routines to produce synthetic speech, but apart from producing interesting effects by adjusting the pitch of the synthetic speech and making nasal sounding speech by significantly reducing the number of LPC coefficients, no particularly interesting research directions emerged. Although this research area was not pursued any further, it gave me insight into the effect of varying the aforementioned parameters on synthetic speech. This experience was useful at a later date when I performed LPC analysis of speech for speaker recognition.

The second area that I was involved in was an ongoing speech therapy project which aimed to assist the therapist and client by providing feedback, in real-time, of certain vocal parameters. The real-time feedback was provided by digital signal processing (DSP) hardware that was designed by a Masters student, Stephen Turner. The DSP hardware, and appropriate software, performed speech analysis to obtain the pitch, intensity and LPC coefficients and the graphics capability of a host PC was used to plot the results. A colleague, Tracy Clark, and I both worked on the speech therapy project and between us specified and wrote several therapy modules. It soon became apparent that in order to further development of the speech therapy aid more DSP boards were required. I modified Turner's original design to make it more manufacturable and had several produced by Departmental technicians. Although the investigations into speech therapy are now being pursued by Catherine Watson and I am no longer directly involved, the speech therapy DSP hardware was useful for the preliminary experimental investigations reported in the next paragraph.

Professor Bates' long standing interest in astronomical signal processing led to the development of several useful image processing techniques. One of these techniques, called shift-and-add (SAA), was modified and applied to speech by Nigel Brieseman to

perform ‘glottal pulse’ (here called the long-term average glottal response, or LTAGR) recovery with the aim of iteratively improving the estimation of the linear prediction model of speech. In order to quickly evaluate the variation between long-term average glottal responses (between people and within a single person) the SAA algorithm was implemented in real-time using our DSP hardware (see §4.3.1). The real-time SAA algorithm (SAA1 in §4.3.1) worked well and at the suggestion of Dr. Richard Fright, then a postdoctoral fellow, I added the ability to store the ‘glottal pulse’ so that comparisons could be made between different long-term average glottal responses. Informal experimentation revealed that people tended to have different long-term average glottal responses and this motivated a thorough investigation into the efficacy of the long-term average glottal response for speaker identification. The following paragraphs outline a series of investigations that firstly examine the effects of speaker’s gender and accent on the long-term average glottal response and, secondly, compare the long-term average glottal response against other ‘vocal tract’ features for performing speaker identification.

I examined the sensitivity of the LTAGR to speaker’s gender and accent (§4.3.3.1) using a database of 352 American English speakers that were preclassified into 8 different accent regions (§4.3.3). A total of 21 descriptors were defined to represent the information recorded in the LTAGR (§4.3.2). The correlations between these 21 descriptors were examined by extracting orthogonal factors from the descriptors using factor analysis (§3.4.3,§4.3.3.1). This revealed that the largest factor, which accounted for 67.7% of the variance of the descriptors, was strongly correlated with descriptors that represented the overall shape of the LTAGR. The next two factors, which accounted for 11.5% and 4% of the total variance, were most highly correlated with the number of peaks in the LTAGR. It was apparent from these results that the overall shape of the LTAGR is important and there is no particular descriptor that can be identified as containing most of the information in the LTAGR.

The usefulness of the LTAGR for identifying the gender and accent of speakers was investigated by employing discriminant analysis and then classifying speakers according to their gender and accent (§4.3.3.2). The database of 352 American speakers gave an accent classification error rate of 63.1% and a gender classification error rate of 4.3%. This implies that the LTAGR is potentially useful for determining the gender of a speaker.

The suitability of the LTAGR as a feature for speaker identification was evaluated by comparing it against other features that are commonly used for speaker identification (§5.4). The features selected here were linear prediction coefficients, cepstral coefficients, partial correlation coefficients and the long-term average spectrum. The linear prediction coefficients, cepstral coefficients and the partial correlation coefficients were selected because they have been shown by other researchers to be useful for performing speaker identification (§3.5). The justification for selecting the long-term average spectrum as a feature was that, like the long-term average glottal response, it records the long-time average characteristics of a person’s voice (§4.4).

Since the long-term average spectrum and the LTAGR might reasonably be expected to measure similar effects(§4.4.1), an investigation into the similarities and differences between these two features was undertaken (§4.5.1.1). It was found that the spectrum of the LTAGR matches the LTAS quite closely under certain conditions (§4.5.1.1). Furthermore, the LTAGR is shown to have a 5-10 times computational advantage over the LTAS (§4.5.1.2).

Repeatable speaker identification experiments using recorded (digitized) speech were designed to further investigate the usefulness of the long-term average glottal response, and real-time processing became less critical. I collected and digitized a

database of utterances from 20 speakers to use for speaker identification experiments (see §4.1). The experiments were performed using SGPRC and several extensions to SGPRC that I programmed in the course of my research.

Initially, characteristic descriptors of the LTAGR were used to perform speaker identification. However after much investigation I concluded that descriptors offered no advantage over the entire LTAGR waveform (§5.4.2.2). I also implemented one of the 'standard' speaker identification schemes that represent each speaker by a quantized (VQ) codebook containing 'characteristic' vectors (§2.7.4.3, §4.2.2). This method appealed to me because the matching of features against a person's codebook does not require any time alignment information, and accurate identification results were reported in the literature (§3.3.3). I used vector quantization to construct speaker codebooks for several different types of features. The identification error rate using the LTAGR was higher than that obtained using standard features such as partial correlation coefficients and cepstral coefficients (see §5.4).

The degree of independence between the long-term average glottal response and vocal tract features was evaluated by examining the correlation between the long-term average glottal response and the vocal tract features (§5.2.2). Since the vocal tract filter features appeared to be independent of the LTAGR, I went on to investigate possibilities for combining these features to improve identification accuracy (§5.4.4). A small reduction in the identification error rate was observed for speaker templates containing 8 or 16 'characteristic' vectors, but this was not consistent across all template sizes (§5.4.4.6).

I also examined the effects of various types of distortion and noise on the speaker identification accuracy of LTAGR and linear prediction based features (§5.5). Identification experiments showed that the LTAGR is significantly better at performing speaker identification on noisy speech than any of the other features (§5.5.1). However, the LTAGR is particularly sensitive to variations in the phase response of the channel over which the speech is recorded (§5.5.2.3). The other features, which do not model the phase, were not so affected by the phase response.

As research into the accuracy of the above features progressed I discovered that there were certain fluctuations in identification results that were difficult to account for. I therefore examined suitable methods for determining the confidence intervals for experimental results. This led to the use of statistically based methods for assessing whether results from two identification experiments were significantly different (§5.3).

The following six paragraphs outline the structure of this thesis. The first 3 chapters contain principally background on the production and characteristics of human speech, speech processing techniques and speaker recognition, while Chapters 4, 5 and 6 report on the original contributions of my work.

Speaker recognition is based on the properties of speech and these properties depend upon the speech production mechanism. Chapter 1 is an introduction to the physiological processes that produce human speech. Terminology for describing speech is introduced and the characteristics of different classes of sounds are discussed.

In order to perform speaker recognition, it is necessary to extract parameters from the speech that are useful for characterizing the identity of the speaker. Chapter 2 pulls together into a single concise treatment the wide range of signal processing techniques used in this thesis to parameterize speech. Standard LPC analysis techniques are introduced, and the relationship between the vocal tract model and partial correlation coefficients is discussed. Methods of spectral estimation are described since both cepstral and linear prediction coefficients model the spectrum of the speech signal. Vector quantization is introduced since it is used to form speaker templates (or codebooks) from partial correlation, cepstral and linear prediction coefficients. The last

section in Chapter 2 introduces shift-and-add and describes an original contribution I made concerning the way in which the voiced/unvoiced decision is implemented. The relationship between the LTAGR and the glottal excitation and vocal tract filters of synthetic speech is examined.

Chapter 3 contains a wide ranging review of techniques that are useful for speaker recognition. The performance of human speaker recognition is discussed, along with factors that affect that performance. Speaker recognition methodologies are divided into those that use dynamic features, those that use statistical features and those that use vector quantized features. The performance of several different systems described in the literature is critically assessed, followed by a brief discussion of practical considerations that are important when performing speaker recognition.

Chapter 4 is the first chapter which contains major contributions of my work and is mainly concerned with defining the various features that I use for characterizing voices. The 20 speaker speech database that I collected and digitized for use in speaker identification experiments is described. The chapter reports in detail the testing of my implementation of the vector quantization training algorithm first reported by Linde, Buzo and Gray. Characteristic descriptors that I abstract to describe the LTAGR are defined and investigations I performed to determine the sensitivity of these descriptors to the speakers' accent and gender are presented. Since both the long-term average spectrum and the LTAGR are long-term average features the similarities and differences are discussed in some detail in this chapter. Several different methods of calculating the long-term average spectrum are evaluated and the spectral information recorded in the long-term average spectrum is compared with that recorded in the LTAGR.

Chapter 5 is concerned primarily with reporting results from various speaker identification experiments. The accuracy of speaker templates for various features is examined along with the effect of unvoiced speech and silence on the templates. Statistical tests are described for determining whether recognition results between identification experiments are significantly different. The recognition performance of cepstral coefficients, partial correlation coefficients, the LTAGR and long-term average spectrum are examined individually and in certain combinations. The effect of noise and distortion on the identification accuracy of different features is also explored. The computational overheads when using the different features described in this thesis is also discussed.

Chapter 6 contains conclusions and suggestions for future research.

Publications and presentations prepared during the course of my Ph.D. research are listed below.

ELDER, A.G., BATES, R.H.T., BRIESEMAN, N.P., CLARK, T.M., FRIGHT, W.R., GARDEN, K.L., KENNEDY, W.K., SQUIRES, P.L., TURNER, S.G. and THORPE, C.W. (1987), 'Real-time speech therapy aid', In *Proc. NELCON, (New Zealand National Electronics Conference)*, Auckland, 1-3 September, pp. 115-118.

BATES, R.H.T., BRIESEMAN, N.P., CLARK, T.M., ELDER, A.G., FRIGHT, W.R., GARDEN, K.L., KENNEDY, W.K., SQUIRES, P.L., THORPE, C.W., TURNER, S.G. and JELINEK, H.J. (1987), 'Interactive speech-defect diagnostic/therapeutic /prosthetic aid', In LETELLIER, J.P. (Ed.), *Real Time Signal Processing X*, Proceedings of SPIE - The International Society for Optical Engineering, 20-21 August, pp. 131-139.

WATSON, C.I., CLARK, T.M., ELDER, A.G. and THORPE, C.W. (1988), 'Multifarious real-time speech processing applications', In *Proc. NELCON, (New Zealand National Electronics Conference)*, Christchurch, September, pp. 65-70.

MILLER, A.J., KENNEDY, W.K., ELDER, A.G., THORPE, C.W. (1989), 'Speech Trans-

- mission over Packet Networks', In *Proc. NELCON, (New Zealand National Electronics Conference)*, pp. 64-68.
- LIM, C.A., ELDER, A.G., CLARK, T.M. and BATES, R.H.T. (1990), 'Software implementation of Hidden Markov model for recognition of isolated digits uttered by New Zealand speaker', In *Proc. NELCON, (New Zealand National Electronics Conference)*, pp. 287-294.
- ELDER, A.G., KENNEDY, W.K., BATES, R.H.T (1991), 'The effect of voice/unvoiced decisions on the accuracy of speaker identification', In *Proc. NELCON, (New Zealand National Electronics Conference)*, pp. 93-98.
- ELDER, A.G., KENNEDY, W.K., BATES, R.H.T (19XX), 'A comparison of glottal and vocal tract characteristics for speaker identification', *Computer Speech and Language*, Submitted for publication.



---

## GLOSSARY OF ABBREVIATIONS

Selected abbreviations that appear in this thesis are:

|               |   |
|---------------|---|
| <b>CEP</b>    | Cepstral coefficients                                   |
| <b>EER</b>    | Equal-error ratio                                       |
| <b>HMM</b>    | Hidden Markov model                                     |
| <b>IEEE</b>   | Institute of electrical and electronic engineers        |
| <b>LBG</b>    | Linde, Buzo and Gray                                    |
| <b>LPC</b>    | Linear predictive coding                                |
| <b>LTAGR</b>  | Long-term average glottal response                      |
| <b>LTAS</b>   | Long-term average spectrum                              |
| <b>LTAS-E</b> | Long-term average spectrum of pre-emphasized speech     |
| <b>LTAS-P</b> | Long-term average spectrum computed pitch synchronously |
| <b>LTAS-V</b> | Long-term average spectrum of voiced speech             |
| <b>MNRU</b>   | Modulated noise reference unit                          |
| <b>MSE</b>    | Mean-square error                                       |
| <b>PARCOR</b> | Partial correlation coefficients                        |
| <b>PCA</b>    | Principal component analysis                            |
| <b>PIN</b>    | Personal identity number                                |
| <b>Q(dB)</b>  | signal-to-speech-correlated noise ratio                 |
| <b>SAA</b>    | Shift-And-Add   |
| <b>SGPRC</b>  | Sigproc - a signal processing package                   |
| <b>SNR</b>    | Signal-to-noise ratio                                   |
| <b>VQ</b>     | Vector quantization                                     |
| <b>VUV1</b>   | Voiced/unvoiced decision method 1                       |
| <b>VUV2</b>   | Voiced/unvoiced decision method 2                       |



## CHAPTER 1

---

### SPEECH SOUNDS AND CHARACTERISTICS

Human beings have a unique articulatory mechanism that allows them to produce a wide range of sounds. Over many years combinations of these sounds have had particular meanings associated with them and have evolved into what we now call languages. The act of articulating a series of sounds according to the conventions of a language is called *speaking* and the emitted sound, perceived by the ears is called *speech*. The convenience with which speech can be used to convey thoughts, concepts, feelings and ideas is the motivation for research into automatic systems that recognize speech. The remainder of this chapter explains how human speech is produced and describes the way the character of the speech varies as different sounds are uttered.

Section 1.1 introduces the physiology of speech production and provides background for the discussion of the range of sounds in the English language in §1.2. Characteristics of human speech are discussed in §1.3. Section 1.4 defines terminology for describing the quality of a person's voice.

#### 1.1 PHYSIOLOGY OF SPEECH PRODUCTION

Described crudely, speech is the result of air molecules colliding with each other to force an acoustic wave to propagate through the atmosphere. Speech is manifested physically as the variations of pressure associated with the aforesaid acoustic wave. A person generates such a pressure waveform by causing the lungs to force air through the glottis and on out through the vocal tract. The term *speech signal* is used from now on to denote the content of recordings made by apparatus capable of sensing the pressure waveform.

The automatic operation of our lungs during speech production comprises a complicated chain of events (Lieberman and Blumstein, 1988). Firstly, the intercostal muscles, which are interwoven with the ribs, work to expand the volume around the lungs. The lungs expand elastically to fill the increased volume and, as a result, air is taken into the lungs. A simplistic model of this expansion, consisting of balloons (lungs) within pistons (ribcage) is depicted in Fig. 1.1. Speech production occurs by air being moved out through articulators. The force required to do this is mostly provided by the lungs, since elastic lung expansion causes energy to be stored when the lungs are inflated.

After leaving the lungs, air passes through the larynx (Fig. 1.2), which converts the steady flow of exhaled air into a series of 'puffs' or 'pulses' that form the *excitation* for voiced speech sounds. Inside the larynx are two fleshy cords that stretch across it. These are called the *vocal cords* or *vocal folds*. Cartilages that support the vocal cords allow them to be held open, or closed, under variable tension. The slit between the vocal cords is termed the *glottis*. When the tensed vocal cords are held together across the larynx the glottis is closed. The lungs, attempting to exhale air, produce a pressure beneath the glottis. This subglottal pressure increases until the vocal cords are forced apart and a rush of air occurs through the glottis. The subsequent suction, together with the tension on the vocal cords, causes the glottis to close (it is worth

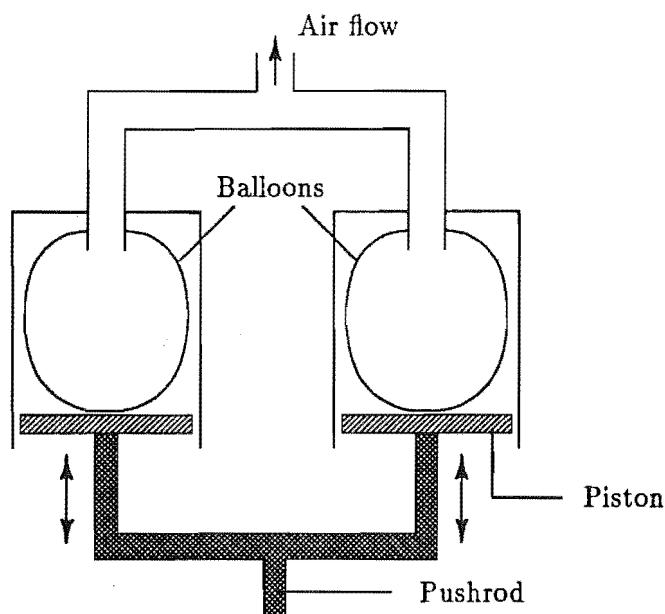
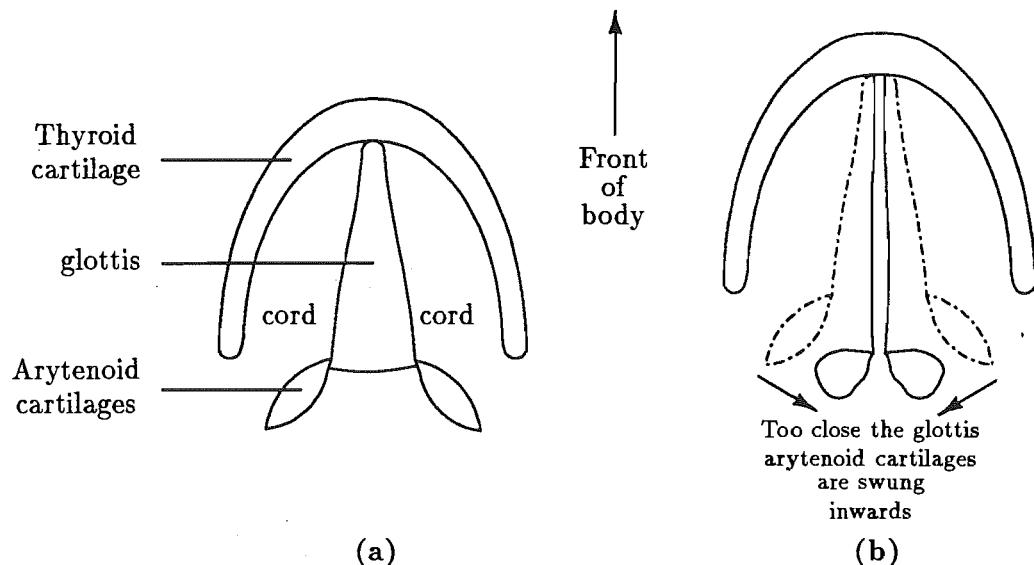


Figure 1.1. A model of the lung mechanism for speech production.

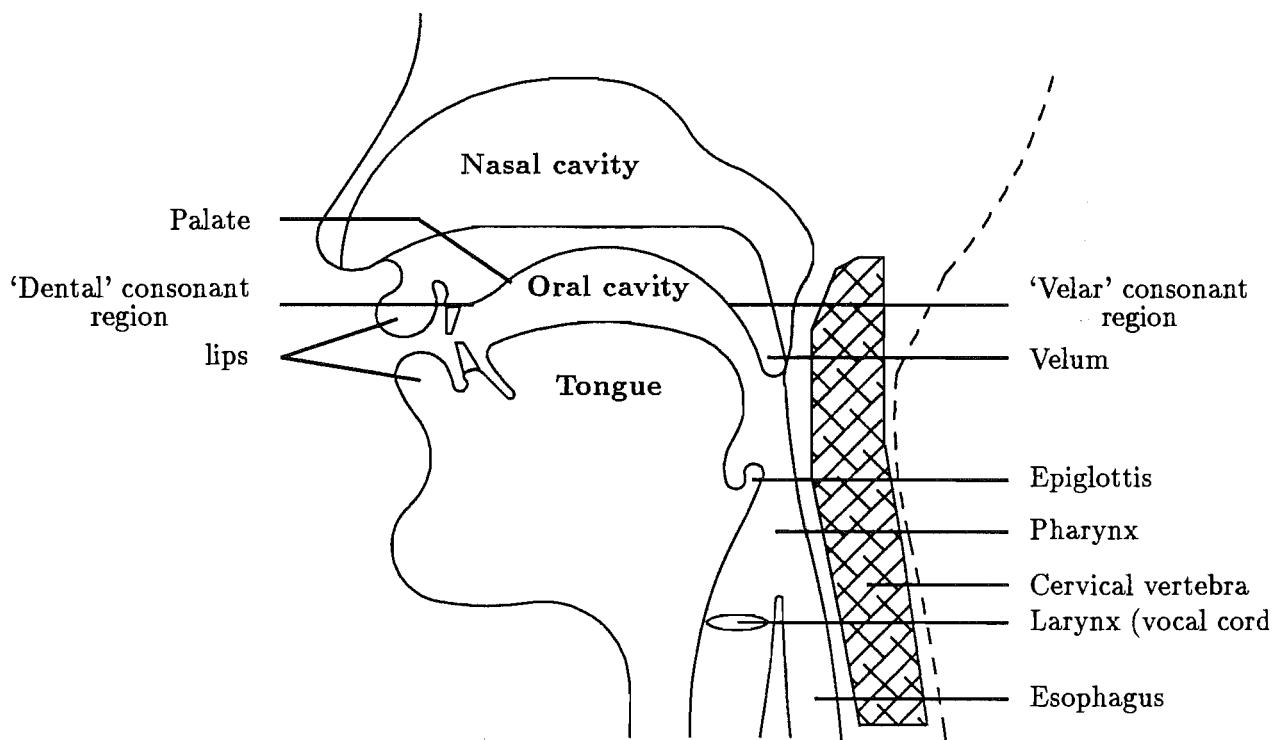
noting that the Bernoulli relation for fluid flow explains the reduction of pressure at the glottis assisting it to close (Ward Smith, 1980)). The result is that a single ‘pulse’ of air, known as the *glottal pulse*, passes through the glottis. The periodic repetition of the glottal pulse gives the *pitch* of a voice, but more importantly provides excitation for the upper vocal tract.

The oral cavity and the nasal cavities above the larynx are called the supralaryngeal vocal tract or, more commonly, the *vocal tract*. Articulators in the vocal cavity control the cross-sectional area of the vocal tract, thus modifying the frequencies that are present in the glottal pulse as it travels towards the lips. As early as 1779, Kratzenstein (1782) used a set of tubes to filter the output of a vibrating reed, thereby demonstrating the different vocal tract configurations for five Russian vowels. A few years later, in 1791, von Kempelen demonstrated a speaking machine of a more sophisticated nature (Flanagan, 1972). His machine utilized a soft leather resonator to simulate the vocal tract. The shape of the leather resonator was modified by hand to produce different sounds. The use of a tube of varying diameter to make up the model vocal tract and the concept of a vocal tract filter is part of the *source filter model*, which is explained in detail in §2.3. The vocal tract has other speech functions in addition to filtering the glottal pulse. For certain sounds the air-flow from the lungs is completely stopped, while for others the articulators within the vocal tract generate the excitation.

Linguists describe speech sounds in terms of the combinations of articulators utilized to produce them. The first of these categories of sound, *bilabial*, refers to sounds that are made with both lips while sounds formed by the lips and teeth are described as *labiodental* (Skinner and Shelton, 1978, p67). *Dental* sounds are produced by the tongue and the teeth. Because the tongue can touch either the tip of the teeth, or the base of the teeth (near the gum), dental refers to the former and the latter is termed *alveolar*. Just behind the teeth is the hard palate, and sounds made by the tongue and hard-palate are called *palatal*. Towards the throat from the *hard palate* is the *soft palate*, or *velum* (Hardcastle, 1976, p120). *Velar* utterances (sometimes called *guttural*) are



**Figure 1.2.** A depiction of the glottis as viewed from above: (a) normal breathing, (b) complete closure of the glottis for speech (based on Lieberman and Blumstein (1988, p99)).



**Figure 1.3.** A cross-section of the human head (based on Lieberman and Blumstein (1988) and O'Connor (1973) ).

|       |                      | Tongue position    |        |                    |
|-------|----------------------|--------------------|--------|--------------------|
|       |                      | Front              | Middle | Back               |
| Close | /i/<br>pe <u>a</u> t |                    |        | /u/<br>boot        |
|       | /ɪ/<br>pi <u>t</u>   |                    |        | /ʊ/<br>pu <u>t</u> |
|       | /e/<br>pe <u>t</u>   | /ɜ/                |        |                    |
|       | /æ/<br>ca <u>t</u>   | /ʌ/<br>bu <u>t</u> |        | /ɔ/<br>po <u>t</u> |
|       |                      |                    |        |                    |
|       | /ɑ/<br>pa <u>r</u> t |                    |        | /ə/<br>po <u>t</u> |

Table 1.1. Vowel phonemes in New Zealand English (after Maclagan (1982)).

produced by the back of the tongue and the soft palate forming a constriction.

Note that the combination of the *velum* and the *pharynx* (a muscular tube that extends from the esophagus to the base of the skull - see Fig. 1.3) is often called the *velopharyngeal system*. The word system is used because the velum and pharynx work in a coordinated fashion to produce nasal sounds.

## 1.2 TYPES OF SPEECH SOUNDS

This section discusses classification of sounds. In order to uniquely identify an individual sound, the phonetic alphabet is introduced. An individual sound typically belongs to more than one of the classes described below. For example, /k/ is unvoiced and also a stop consonant.

### 1.2.1 Phonemes

The word phoneme comes from the Greek words *phoneo*, to speak, and *phonema*, which means sound or speech. Each individual phoneme, or sound, has a phonemic symbol associated with it. The international phonetic alphabet is the preferred symbol system used throughout this thesis. Tables 1.1 and 1.2 define the sound that a particular symbol corresponds to. Phonetic symbols are bracketed by the '/' character to distinguish them from text (e.g. /i/ in sit).

### 1.2.2 Voiced sounds

By definition, *voiced sounds* occur when the vocal cords open and close regularly. The vowels, and certain consonants (for example /r/), fall into this category.

Vowels, which constitute one particular class of voiced sounds, are characterized by the vocal tract being relatively open, with only a small amount of coupling into the nasal cavity occurring. As a result of this, vowels contain more speech energy than unvoiced sounds.

### 1.2.3 Unvoiced sounds

*Unvoiced sounds*, in contrast to voiced sounds, rely on air turbulence to provide excitation. The turbulence can be generated at various locations within the vocal tract, the

| Manner<br>of<br>articulation | V/<br>UV | Place of articulation (obstruction) |                    |                    |                    |                     |                    |                   |
|------------------------------|----------|-------------------------------------|--------------------|--------------------|--------------------|---------------------|--------------------|-------------------|
|                              |          | Bilabial                            | Labio-<br>dental   | Dental             | Alveolar           | Palatal             | Velar              | Glottal           |
| stops                        | V        | /b/<br><u>bib</u>                   |                    |                    | /d/<br><u>did</u>  |                     | /g/<br><u>gig</u>  |                   |
|                              | UV       | /p/<br><u>pet</u>                   |                    |                    | /t/<br><u>tot</u>  |                     | /k/<br><u>key</u>  |                   |
| fricative                    | V        |                                     | /v/<br><u>very</u> | /ð/<br><u>then</u> | /z/<br><u>zoo</u>  | /ʒ/<br><u>azure</u> |                    |                   |
|                              | UV       |                                     | /f/<br><u>fat</u>  | /θ/<br><u>thin</u> | /s/<br><u>sat</u>  | /ʃ/<br><u>sham</u>  |                    | /h/<br><u>had</u> |
| affricative                  | V        |                                     |                    |                    |                    | /dʒ/<br><u>june</u> |                    |                   |
|                              | UV       |                                     |                    |                    |                    | /tʃ/<br><u>chin</u> |                    |                   |
| nasal                        | V        | /m/<br><u>map</u>                   |                    |                    | /n/<br><u>nip</u>  |                     | /ŋ/<br><u>sing</u> |                   |
| liquid                       | V        |                                     |                    |                    | /l/<br><u>full</u> | /r/<br><u>run</u>   |                    |                   |
| glide                        | V        | /w/<br><u>why</u>                   |                    |                    |                    | /j/<br><u>yet</u>   | /w/<br><u>why</u>  |                   |

Table 1.2. Consonant phonemes (from Edwards and Shriberg (1983)). The V/UV label identifies whether sounds are voiced or unvoiced.

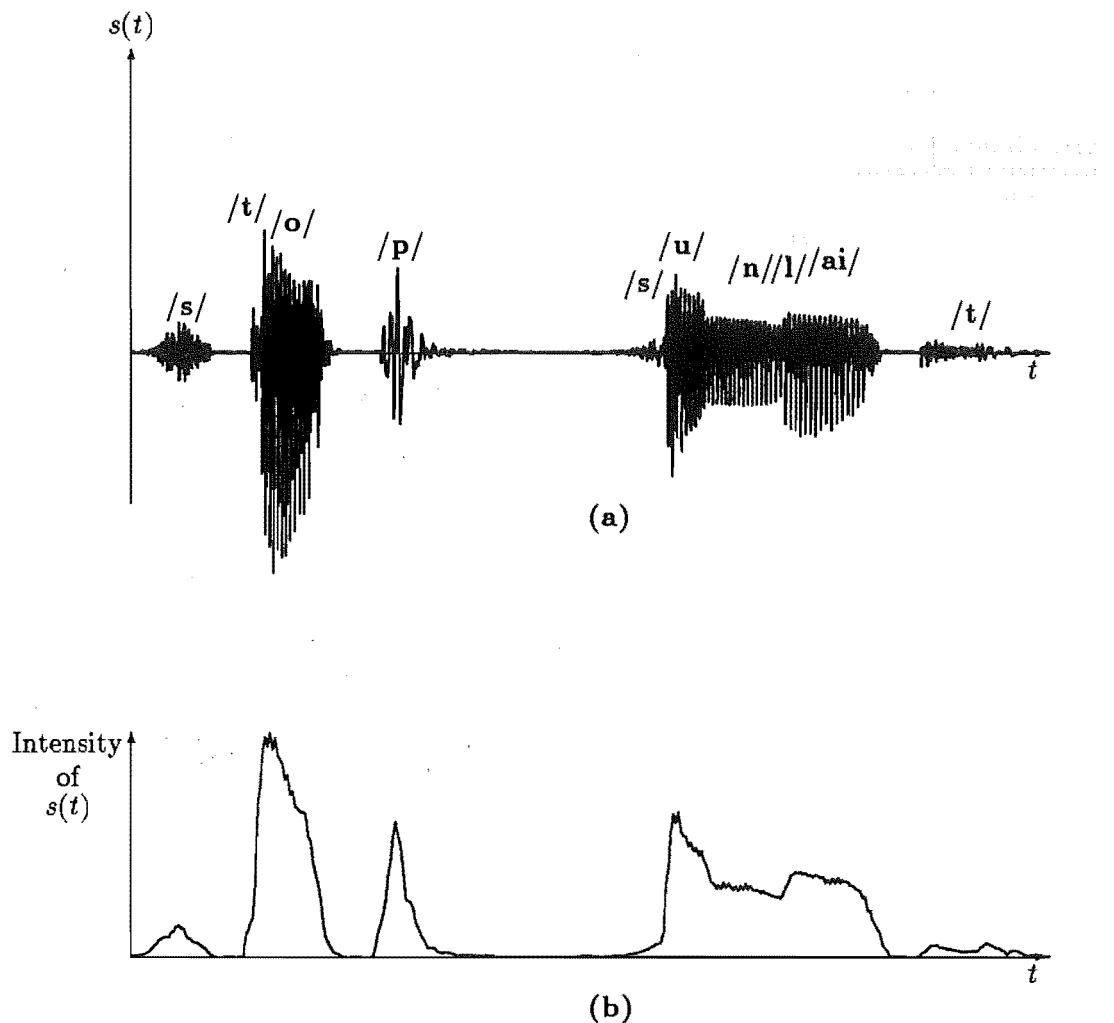
actual location depending upon the sound being formed. For example, the /s/ sound is due to turbulence being generated between the tongue and the teeth, while the /p/ sound relies on the explosive outrush of air from the lips. Fig. 1.4 illustrates the /s/ and /p/ sounds. Note that the /s/ sound is of low energy and has the appearance of noise. The /z/ sound is an example of a sound that has mixed voiced and unvoiced excitation, with the former predominating. The completely unvoiced version of /z/, is /s/. Similar pairs of voiced/unvoiced sounds are common among English consonants.

#### 1.2.4 Aspirated sounds

Not all sounds fit neatly into the voiced/unvoiced classification. *Aspirated sounds* occur when the vocal cords are stretched across the larynx, but not tightly enough to close the glottis. Turbulence is produced as air is forced through the glottal constriction. An example of an aspirated sound is /h/. Aspirated sounds are easily confused with breathing noises because they have similar amplitude. This can cause difficulties when attempting to process speech automatically by computer.

#### 1.2.5 Stop consonants

When the vocal tract is completely closed for an instant, the sound produced as the closed vocal tract is abruptly opened, is called a *stop consonant*. The vocal tract closure



**Figure 1.4.** Waveforms of the words “stop sunlight,”: (a) pressure waveform, (b) intensity (see §1.3.1).

results in air pressure building up behind the blockage (closed vocal cords cause the same effect). The instant the blockage is removed, a rush of turbulent air escapes from the lips. Because of this sudden outrush of air, many texts refer to stop consonants as plosives (Catford, 1977, p73). The speech pressure waveform for a stop consonant has a very large pressure change when the vocal tract blockage is removed (see the /t/ and /p/ sounds in Fig. 1.4).

### 1.2.6 Nasals

A phoneme which sounds as though it comes from a speaker’s nose (Laver, 1980, p77) is called a *nasal sound*. Simplistically, nasal and non-nasal sounds can be thought of as being formed with an open and closed velum respectively. To examine the velopharyngeal setting for nasals, Lubker and Moll (1965) measured the nasal and oral air flows by fitting a special mask to a male speaker’s face. The mask directed air flow from the mouth and nose to separate flow measuring devices. They conclude that nasal air flow depends not only upon velopharyngeal opening but also upon the oral constriction. For example, they find that an adult male saying the word ‘nip’ has oral and nasal flows of

0 and 12 litres per minute for the /n/ sound. Oral and nasal flows of 16 and 4 litres per minute respectively are measured for /i/ sound, which indicates that the velum is not completely closed for non-nasal sounds. These flow rates are only approximate, but they highlight the fact that significant nasal air flow occurs for certain oral sounds. It is therefore incorrect to say that nasal sounds occur when there is nasal air flow. The main characteristic of a nasal sound is that the nasal air flow is significantly greater than the oral air flow.

### 1.3 SPEECH CHARACTERISTICS

Temporal variations of speech characteristics are the concern of this section. In particular, the variation of speech characteristics from sound to sound within an utterance is illustrated.

#### 1.3.1 Loudness and intensity

The term *loudness* is a perceptual measure of variations in amplitude of the speech pressure waveform. If we say that a particular noise sounds louder than another, we are forming an opinion from the perceived air pressure variations detected by our ears (Lieberman and Blumstein, 1988, p29). The acoustic measure of loudness is *intensity*, which is related to the energy present in the speech waveform. The intensity of a particular utterance is determined by the pressure of air coming from the lungs.

Variations of intensity occur within words, depending on the type of sound being produced (Cruttenden, 1986, p3). For example, vowel sounds are uttered with the mouth relatively open and have more energy than fricatives, which are produced when the vocal tract is constricted. Syllables consist of single uninterrupted sounds that contain one or more vowels. Syllable boundaries within words occur at consonants and are therefore usually identifiable by their low intensities (see Fig. 1.4).

#### 1.3.2 Pitch variations

The frequency of the glottal pulses, on which the perceived pitch (§1.1) depends, varies as we speak. Utterances of the average adult male and female lie within pitch ranges of 80-100Hz and 160-200Hz respectively (Fry, 1979, p68). Deliberate variations in pitch are used to add extra information, or meaning, to the words being uttered. In Chinese, word meanings are often dramatically dependent upon the modulation of pitch as they are spoken (Cruttenden, 1986, p8)(Lehiste, 1970, p92). Languages that use pitch in this way are called tone languages. However, in English, pitch variations primarily distinguish questions (rising pitch) from statements (falling pitch). For example, pitch variation is the distinguishing feature between the utterances '*You're going.*' and '*You're going?*'.

Apart from voluntary alterations of pitch, there are certain sounds such as /i/ and /u/ that have a higher pitch than other vowels because of the connections between muscles in the tongue and the larynx (Lehiste, 1970, p69). As we speak there are also patterns of pitch, rises and falls called *intonation* (Cruttenden, 1986, p9), that recur consistently.

#### 1.3.3 Frequency content of speech

As we make different sounds, the speech pressure waveform changes. This is the same as saying that the frequency content (*spectrum*) of the speech changes. The change occurs because certain frequencies are attenuated more than others as the glottal pulse travels

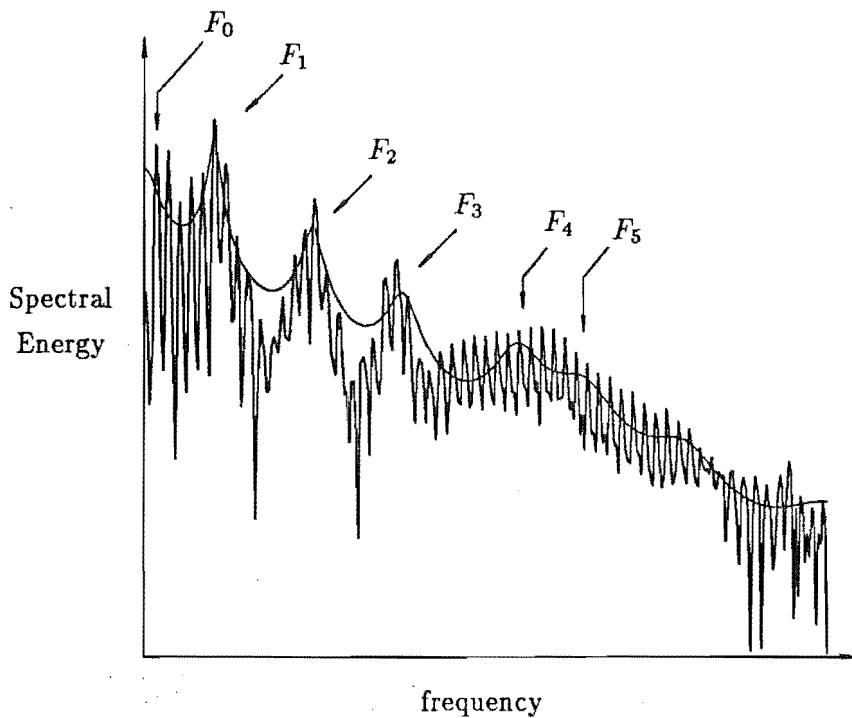


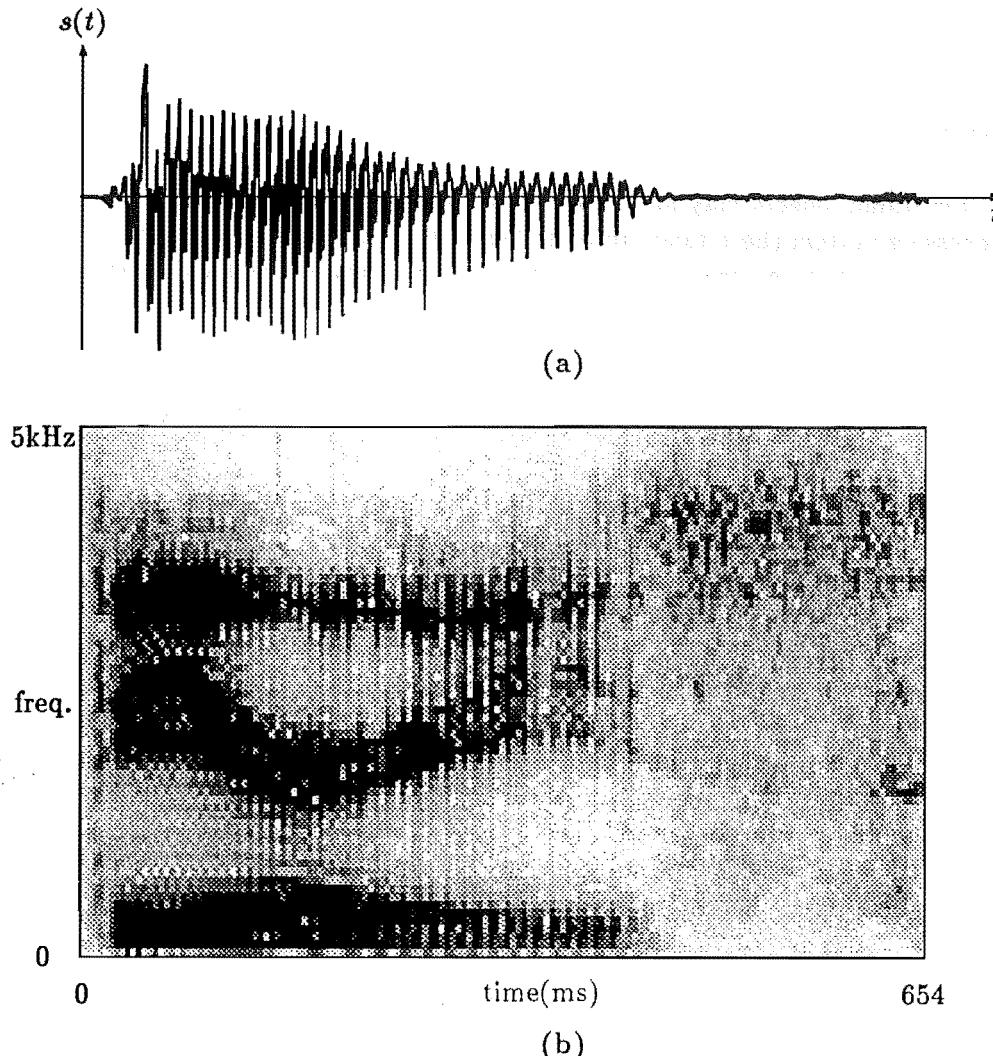
Figure 1.5. The spectral content of one frame of speech with the formants indicated.

down the vocal tract. The frequencies that propagate with minimal loss of acoustic energy are called resonances. When a vowel sound is uttered, its spectrum contains a small number of peaks (typically 3-4), each corresponding to a resonance. These resonances are called *formants* (see Fig. 1.5) and are labelled  $F_1, F_2, F_3, F_4$ , with the smaller numbers applying to lower frequencies. The pitch is labelled  $F_0$  and corresponds to the spacing of harmonics in the spectrum (see Fig. 1.5) (Fant, 1973, p5). As different sounds are produced the resonances within the vocal tract change and the formant positions alter.

It is sometimes useful to describe the width of a particular formant. The standard description is the difference between the higher and lower frequencies at which the formant spectral energy is at half its maximum power. This difference is called the *formant bandwidth*.

The spectrum depicted in Fig. 1.5 is of a single speech segment of 60 ms duration. However, it is often useful to examine the changes in spectra across a number of speech segments and the *spectrogram* (Fig. 1.6) is a common representation of this information (Fry, 1979, p111). The horizontal and vertical axes represent time and frequency respectively. A dark area in the spectrogram indicates a high energy level and conversely a light area indicates low energy. The patterns, or bands, formed by the dark areas represent the formants and the way they change throughout the utterance. The spectrogram shown in Fig. 1.6 illustrates that unvoiced sounds (see the /s/) have more high frequency energy than the vowel sounds, because unvoiced excitation energy spans a greater range of frequencies than glottal excitation. Fant (1973, §1) considers spectrograms to be particularly useful for analysing the structure of sounds.

The slow variation of the formant frequencies apparent in Fig. 1.6 is typical of the majority of speech sounds. Because they change slowly, and adequately represent the



**Figure 1.6.** The word “berries”: (a) the time waveform, (b) the spectrogram of the signal depicted in (a).

spectral content of a speech segment, it is possible to code speech efficiently in terms of formant positions. Coding techniques are described in more detail in Chapter 2.

#### 1.3.4 Emotional effects

A person’s emotional state is a variable that affects speech production. The term *neutral voice* describes a person speaking in a relaxed way under what can reasonably be taken to be normal conditions. Intuitively we expect a voice to change when someone becomes angry. Williams and Stevens (1972) studied a number of different emotional states by using trained actors to simulate particular emotions within the setting of a play. They report their results with particular emphasis on changes that occur in pitch and formants and the following paragraph summarizes their findings.

The emotions of anger, fear and sorrow all significantly alter a person’s voice. Anger causes the greatest change from neutral, with both the pitch and the first formant rising in frequency. Also, the extra emphasis with which certain words are spoken often increases irregularities in the glottal pulse repetition (called voicing irregularity). Fear, by contrast with anger, manifests itself in the voice by slowing the rate at which words are uttered. Both the pitch and formants remain similar to those of a neutral voice.

The emotion of sorrow lowers the pitch and the pitch range is narrowed compared with neutral voice. The articulation rate drops dramatically, by a factor of two compared with neutral speech. In some instances the vocal excitation reduces to noise and the voice becomes a whisper.

Emotional effects may have a significant affect on the voice if an automatic speech processing system (here taken to be either and word recognition or speaker recognition system) is analyzing the speech. For example, an angry person's voice may undergo such a drastic change from normal, that the automatic speech processing system performs incorrectly (which, of course, will just make the person more angry!).

#### 1.4 VOICE QUALITY

Most of us are very conscious of the ability to recognize people from their voices. Laver (1980, p1) describes voice quality as "the characteristic auditory colouring of an individual speaker's voice". It is a long-term cumulative measure, not directly related to how pleasant a voice sounds to the ear. "Imagine a cineradiographic film being taken of the vocal apparatus over, say, 30 seconds. If the individual frames of the film were superimposed on top of each other, a composite picture might emerge which would represent the long-term average configuration of the vocal organs." (see §2.8)(Laver, 1980, p13). This average configuration, is called the *articulatory setting*. The setting an individual adopts is dependent on the language and characteristic individual deviations from the language norm. Deviations from the articulatory setting are made to form the variety of sounds that make up speech.

The physical characteristics of individuals' vocal tracts affect the speech they utter. For example, individuals' different (finite) vocal tract lengths cause certain frequencies to propagate down the vocal tract with less attenuation than others. Other, less obvious, factors that affect speech include the way a person's mouth is held and the type of excitation the larynx produces.

Since there are a number of different articulatory settings, it is useful to define a neutral setting against which deviations can be measured. Laver (1980, p23) defines a neutral configuration of the vocal tract in the following manner.

- the lips are not protruded.
- the larynx is neither raised nor lowered.
- the supralaryngeal vocal tract is most nearly equal in cross-section along its full length.
- front oral articulations are performed with the blade of the tongue.
- the root of the tongue is neither advanced nor retracted.
- the muscles that connect the soft palate to the tongue, the side walls of the pharynx, and the larynx (faucal pillars) do not constrict the vocal tract.
- the pharyngeal constrictor muscles do not constrict the vocal tract.
- the jaw is neither closed nor unduly open.
- the velopharyngeal system causes audible nasality only when necessary for linguistic purposes.

Apart from the vocal setting, the setting of the vocal cords (phonatory setting) also affects a person's speech. The usual operation of the glottis is described by the term modal. It is characterized by the glottis opening and closing regularly and by an absence of speech distortion (i.e. the speech sounds normal). Modal is used to describe glottis vibrations within the range of frequencies commonly used for speaking and singing (Laver, 1980, p109).

Falsetto excitation occurs when the glottis opens and closes at a pitch approximately twice that of modal voice (Laver, 1980, p118). The glottis tends not to close completely after a glottal pulse, causing the sub-glottal air pressure to be lower than for modal voice.

A whisper voice is produced by the glottis being held in a triangular opening, similar to an inverted Y. The characteristic whisper sound is produced by turbulence as air flows past the vocal folds. The whispery voice can be combined with either the modal or falsetto excitations. The spectral content of whispered speech tends to be somewhat noise-like, but with definite concentrations of energy where the formants occur (Laver, 1980, p115).

The creak is yet another form of excitation. It is characterized by a particularly low frequency of vibration (say 40Hz) and sounds like the series of rapid taps produced when a stick is run along a railing (Catford, 1977, p98). Folds of membrane around the glottis combine with the vocal folds to produce 'massive' vibrators that only make small amplitude movements.

Sometimes, under tense situations, a person speaks with a harsh voice. It is typified by aperiodic vocal cord vibration which the listener hears as noise. Both the larynx and the pharynx are in a state of hypertension (Laver, 1980, p126).

In contrast to the vocal cords being highly strung to produce a harsh voice, breathiness occurs when the vocal cords are so relaxed that they fail to close properly at the end of each 'vibration' cycle. This results in a large volume of air being exhaled. The incomplete closure of the glottis causes the glottal excitation to be of lower amplitude. Therefore, most speech produced by a breathy voice is of lower amplitude compared with that produced by a modal voice. It is also, almost invariably, of low pitch (Laver, 1980, p133). Breathiness and whisper voices sound similar, making it tempting to define them as different aspects of a common phenomenon. Physiologically these sounds differ in the muscle groups each employs, resulting in different voice excitations (Laver, 1980, p134).

Although only a selection of possible variations in vocal tract and excitation configurations have been described here, it is important to realize that each configuration has the potential to make an individual's speech unique. Given that individuals have different vocal characteristics, it is reasonable to assume that each speech pressure waveform contains information pertaining to a particular individual.

From the published literature (Doddington, 1985), it appears that the characteristics of the speech we utter varies between individuals. Many schemes have been proposed for recognizing people from their voices alone based on properties of the spectral characteristics of voices. Current speaker recognition research is reviewed in Chapter 3.



## CHAPTER 2

---

### SPEECH ANALYSIS AND MODELLING

This chapter is concerned with methods of speech analysis and modelling. A number of approaches are described, and each provides insight into a different aspect of the speech signal.

For speaker recognition, a good model is one that successfully represents the speech signal to within a tolerable error, when employing a finite, and preferably small, number of parameters. Such a model effects a significant reduction in the number of data-points required to represent speech signals. However, the choice of model is critical because the recognition accuracy depends upon the accuracy with which the model represents information specific to individuals.

With the advent of digital computers, various signal processing and modelling techniques have been applied to speech (Fallside and Woods, 1985). Many signal processing techniques transform the speech signal into another domain either for computational convenience, or to illuminate aspects of the signal structure. Two of these transforms, the Fourier and the Z, are introduced in §2.1. A useful model of speech production, the source filter model, is presented in §2.2. An acoustic model of the vocal tract, also used in the source filter model, is described in §2.3. Section 2.4 specifies methods for analysing speech prosody. An analysis technique based on the acoustic model of the vocal tract, linear predictive coding (LPC), is discussed in §2.5. In §2.6 methods of estimating the spectrum of the speech signal are described and the relationship between predictor coefficients and the spectrum of the speech is established. Techniques for producing a spectrogram of an utterance (see §1.3.3) are also explained. Cepstral coefficients, which are another way of representing speech, are presented in §2.6.3. Section 2.7 describes how a multi-dimensional vector, such as a set of LPC coefficients, can be quantized using a technique called vector quantization. In §2.8 a speech processing technique called shift-and-add is discussed and its origin in astronomical image processing is reviewed. Section 2.9 describes measures of speech noise. Finally, §2.10 summarizes the main points in this chapter.

#### 2.1 SIGNAL ANALYSIS

This section introduces the Fourier and Z-transforms and their associated notation in §2.1.1 and §2.1.2 respectively. Theoretical and practical aspects of sampling signals are discussed in §2.1.3.

##### 2.1.1 The Fourier transform

Any signal, including speech signals, can be considered to be composed of many sinusoidal signals, each having a different frequency and phase. Such a representation is called a spectrum. The Fourier transform constitutes the formal connection between a signal and its spectrum. The one-dimensional direct Fourier transform is defined by (2.1), while (2.2) defines the one-dimensional inverse Fourier transform

(Bracewell, 1986, p7).

$$G(f) = \int_{-\infty}^{+\infty} g(t) e^{-j2\pi ft} dt \quad (2.1)$$

$$g(t) = \int_{-\infty}^{+\infty} G(f) e^{j2\pi ft} df. \quad (2.2)$$

An upper-case letter is used to represent a function of frequency and a lower-case letter indicates its transform in the time domain. The Fourier transform, or spectrum, of a signal  $g(t)$  is therefore written as  $G(f)$ . This terminology can be even more concisely expressed by  $G(f) = \mathcal{F}\{g(t)\}$  or  $G(f) \longleftrightarrow g(t)$ . The  $\longleftrightarrow$  notation conveniently emphasises that  $G(f)$  and  $g(t)$  constitute a Fourier transform pair. The Fourier transform possesses properties which make it extremely useful for signal processing. A selection of those relevant to speech processing are presented below.

### 2.1.1.1 Useful properties of the Fourier transform

When a signal,  $g(t)$ , is passed through a linear filter the filtered signal is modified according to the response of the filter. The output signal is specified by the convolution theorem which is stated as

$$q(t) = g(t) \odot h(t) = \int_{-\infty}^{+\infty} g(t') h(t - t') dt', \quad (2.3)$$

where  $\odot$  denotes the one-dimensional convolution operator and  $h(t)$  can be considered to be the filter response (Bracewell, 1986, p108). The integral in (2.3) constitutes the actual definition of convolution. The argument  $(t - t')$  of  $h(\bullet)$  indicates that the latter is reflected in the origin of time before being slid past  $g(t')$  and multiplied by it. The Fourier transform of (2.3) is a product (Bracewell, 1986, p110), so it follows that

$$g(t) \odot h(t) \longleftrightarrow G(f)H(f). \quad (2.4)$$

Similarly, multiplication in the time domain transforms to convolution in the frequency domain. Convolution is commutative, associative and distributive (Bracewell, 1986, §7).

Correlation is a measure of the similarity between two signals and is defined over all time by shifting one signal relative to the other, thus

$$g(t) \star h(t) = \int_{-\infty}^{+\infty} g(t') h(t + t') dt', \quad (2.5)$$

where  $\star$  is the one-dimensional correlation operator (Bates and McDonnell, 1986, §7). Auto-correlation is a special case of correlation. It is defined for the signal  $g(t)$  by

$$gg(t) = g^*(t) \star g(t) = \int_{-\infty}^{+\infty} g^*(t') g(t + t') dt', \quad (2.6)$$

where  $gg(t)$  is the symbolic representation invoked here for the autocorrelation of  $g(t)$ , and  $g^*(t)$  is the complex conjugate of  $g(t)$ . Equation (2.6) has the property that its Fourier transform is the power spectrum  $|G(f)|^2$ , i.e.,

$$g^*(t) \star g(t) \longleftrightarrow |G(f)|^2. \quad (2.7)$$

This is called the Weiner-Khinchine or autocorrelation theorem (Bracewell, 1986, p115).

It is often useful to calculate the energy in a signal. Rayleigh's theorem or Parseval's theorem (Bracewell, 1986, p112), sometimes called the energy conservation theorem (Bates and McDonnell, 1986, §6),

$$\int_{-\infty}^{+\infty} |g(t)|^2 dt = \int_{-\infty}^{+\infty} |G(f)|^2 df, \quad (2.8)$$

equates the energy in the time and frequency domains.

Note that the limits on the integrals in (2.1) through (2.8) are  $-\infty$  to  $+\infty$ . Such integrals can sometimes be evaluated analytically, but for the majority of practical situations the infinite limits merely emphasise that all the significant values (i.e. above the prevailing noise level) of signals or spectra are to be included in the numerical integrations.

A signal, whose spectrum contains energy at frequencies less than a particular maximum, or cut-off frequency,  $f_c$ , is said to be *bandlimited* to  $(-f_c, f_c)$  (Slepian, 1976). A simple example of a bandlimited signal is Woodward's (1953) sinc function,

$$\text{sinc}(2f_c t) = \frac{\sin(2\pi f_c t)}{2\pi f_c t}, \quad (2.9)$$

which has a spectrum  $(1/2f_c)\text{rect}(f/2f_c)$ , where

$$\text{rect}(f) = \begin{cases} 1 & \text{for } |f| < f_c \\ 0 & \text{for } |f| > f_c \end{cases}. \quad (2.10)$$

### 2.1.1.2 The discrete Fourier transform

The discrete Fourier transform (DFT)(Bates and McDonnell, 1986, §12) of a sampled signal (see §2.1.3) replaces the continuous integral of (2.1) by a summation. The one-dimensional discrete Fourier transform for a sequence of  $N$  samples is defined as

$$\begin{aligned} G[k] &= \sum_{n=0}^{N-1} g[n] W^{nk} \text{ with } W = e^{-j2\pi/N} \\ g[n] &= \sum_{k=0}^{N-1} G[k] W^{-nk} \text{ with } W = e^{-j2\pi/N}, \end{aligned} \quad (2.11)$$

where  $k$  and  $n$  are integers. A computationally efficient algorithm for evaluating the discrete Fourier transform is the Fast-Fourier-Transform (FFT) (Brigham, 1974; Bergland, 1969). The FFT reduces the number of operations required to calculate the Fourier transform by manipulating the odd and even samples of  $g[k]$  separately, making use of the cyclic nature of  $W^{nk}$ . For a one-dimensional sequence comprising  $N$  samples (where  $N$  is a power of 2) the FFT involves only  $2N \log_2 N$  complex operations, compared with  $N^2$  operations required to compute the spectrum by straightforward application of the DFT. Because of this computational efficiency, the FFT is the preferred algorithm for evaluating the discrete Fourier transform. The spectrum of  $g[n]$ , denoted  $G[k]$ , contains spectral components spaced at discrete frequencies in the range of  $-1/2T$  to  $1/2T$  (where  $T$  is the sampling period). The difference between the frequencies of adjacent spectral components is  $1/(NT)$  Hz.

It is important to recognize that all of the useful properties of the continuous Fourier transform described in §2.1.1.1 remain valid for both the DFT and the FFT (Oppenheim and Willsky, 1983, p336).

### 2.1.2 The Z-Transform

Practical speech processing applications manipulate a sequence of digital numbers obtained by sampling a speech signal at regular intervals. The *Z*-transform of such a sequence is defined here as

$$G(z) = \sum_{n=0}^{+\infty} g[n]z^{-n}. \quad (2.12)$$

It is standard to let

$$z = e^{j\theta}, \quad (2.13)$$

where  $\theta$  represents an angle between  $-\pi$  and  $\pi$  radians in the complex  $z$  plane. Substituting (2.13) into (2.12) and comparing with the definition of the DFT in (2.11), it is apparent that the DFT corresponds to the evaluation of the *Z*-transform at points around the unit circle (Oppenheim and Willsky, 1983, p630), i.e.,

$$G(z)|_{z=e^{j\theta}} = \mathcal{F}\{g[n]\}, \quad (2.14)$$

when  $\theta = 2\pi k/N$  and  $N$  is the number of samples in the DFT. The convention in the rest of this thesis is to express spectra as functions of  $e^{j\theta}$ , i.e.

$$g[n] \longleftrightarrow G(e^{j\theta}). \quad (2.15)$$

Digital filters, constructed using delays and multiplications, can be represented compactly in the *Z*-domain (see §2.5). The response of such a filter to an impulse applied at the input is called the *impulse response* and the Fourier transform of the impulse response is called the *frequency response*.

### 2.1.3 Sampling considerations

Sampling is the process of converting a continuous signal into a series of discrete numbers or samples. The recording, or storage, of the instantaneous amplitude of a continuous signal at precisely determined instants, is called *ideal sampling*. When the interval between successive instants is constant it is called the *sampling period* and is denoted by  $T$ . Conceptually, ideal sampling is useful because it allows distortions introduced by practical sampling to be neglected. Practical sampling conditions under which it is reasonable to assume ideal sampling, the sampling period and sampling function are discussed here. An ideal sampling function is defined first, followed by a practical sampling function.

The ideal function for characterizing a single sample (Bracewell, 1986, §5) is a unit area impulse, i.e.,

$$\begin{aligned} \delta(t) &= 0 \quad \text{for } t \neq 0 \\ \int_{-\infty}^{\infty} \delta(t) dt &= 1. \end{aligned} \quad (2.16)$$

Impulses spaced at sampling period intervals constitute an *ideal sampling function*, here called an *impulse train*. The impulse train and its spectrum are denoted by

$$\Delta(t) = \sum_{k=-\infty}^{\infty} \delta(t - kT) \longleftrightarrow \frac{1}{T} \sum_{k=-\infty}^{\infty} \delta(f - \frac{k}{T}). \quad (2.17)$$

A *sampled signal* is identified in this thesis by square brackets, replacing the standard parentheses that enclose the continuously varying parameters on which signals are

conventionally assumed to depend. Thus,  $g[k]$  denotes the  $k^{\text{th}}$  sample of the continuous signal  $g(t)$  and, of course,  $k$  is constrained to be an integer.

Samples can be either ideal or practical, where the latter term implies the sampling operation is of finite duration. The concept of ideal sampling assumes that a signal can be measured instantaneously. In practice, however, sampling takes a finite time  $T_{\text{ad}}$ , called the *aperture width* and the sampling circuitry has only a finite bandwidth. A *practical sampling function* which describes the effects of both finite bandwidth and the aperture width centred about  $t = 0$  is denoted here by  $\text{samp}(t)$ . The samples,  $g[k]$ , are obtained by convolving a signal with  $\text{samp}(t)$ , e.g.,

$$g[k] = \int_{-\infty}^{\infty} g(t') \text{samp}(kT - t') dt'. \quad (2.18)$$

Practical sampling causes a distortion in the estimates of the signal value. Although other effects (often called noise) also distort a signal, the focus here is on sampling distortion. To estimate this, it is first convenient to replace  $g(t')$  in (2.18) with the inverse Fourier transform of  $G(f)$ :

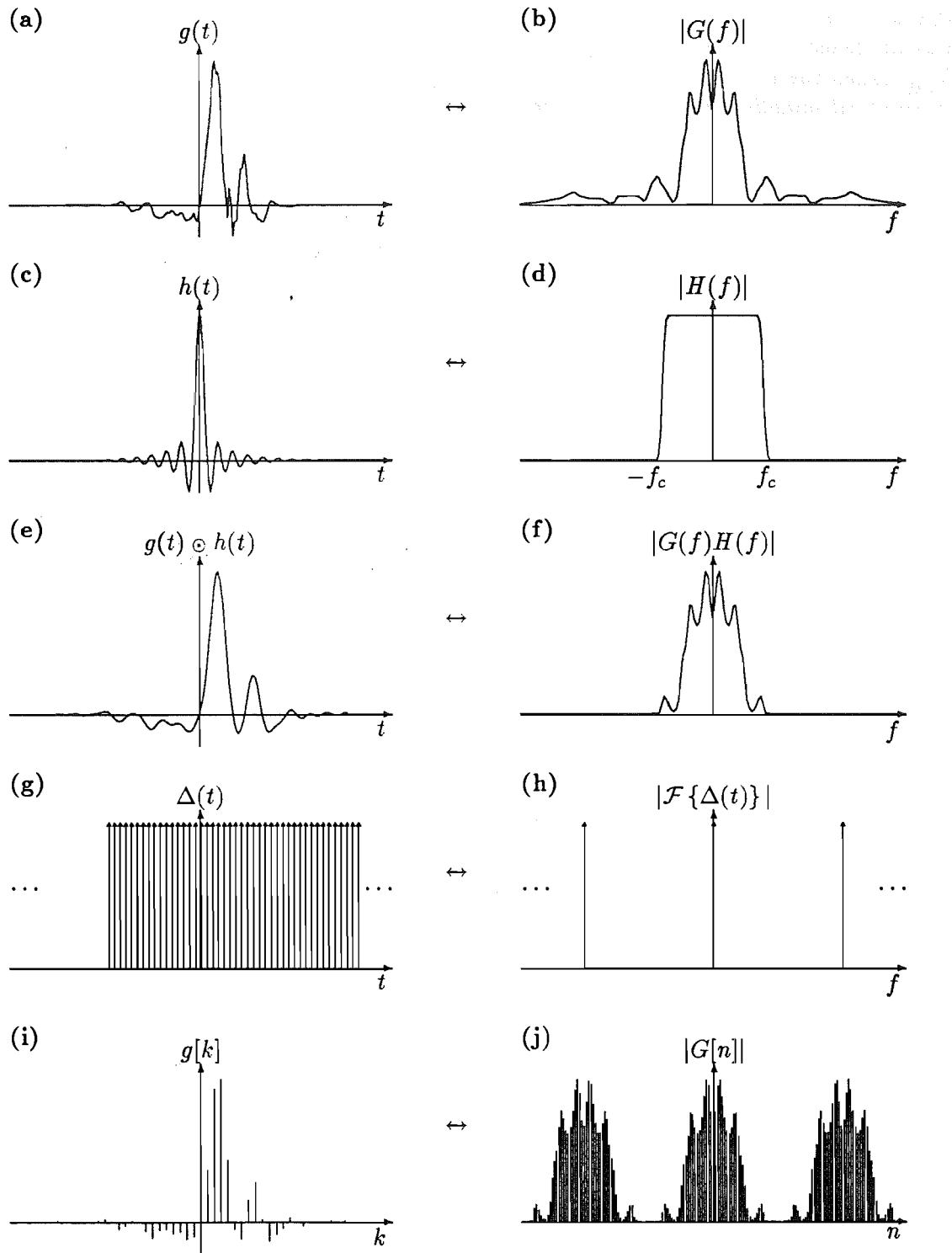
$$g[k] = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} G(f) e^{j2\pi f t'} \text{samp}(kT - t') dt' df. \quad (2.19)$$

Rewriting the exponent  $j2\pi f t'$  in (2.19) as  $-j2\pi f(kT - t' - kT)$  and recalling (2.1), allows the following rearrangement of (2.19).

$$g[k] = \int_{-\infty}^{\infty} G(f) \text{SAMP}(f) e^{j2\pi k T f} df, \quad (2.20)$$

where  $\text{samp}(t) \longleftrightarrow \text{SAMP}(f)$ . From (2.2) and (2.20) it is clear that  $g[k]$  suffers distortion, in the sense that it differs from  $g(kT)$ , if  $G(f)\text{SAMP}(f)$  is not identical to  $G(f)$ . This distortion is minimized when  $G(f)$  is bandlimited and the magnitude of  $\text{SAMP}(f)$  is close to unity out to the band limit, or cutoff frequency ( $f_c$  say), of  $G(f)$ . If  $\text{samp}(t)$  is an impulse,  $g[k]$  is not distorted at all and in practical sampling systems,  $\text{samp}(t)$  is designed to be as close to an impulse as possible. Typically, for the speech signals considered in this thesis the bandwidth of  $\text{samp}(t)$  is significantly wider than  $f_c$ , allowing the effect of sampling to be ignored.

The sampling theorem (Shannon, 1949) states that a bandlimited signal is uniquely determined by its samples, provided the sampling frequency is greater than  $2f_c$ . It is therefore critical to determine the highest frequency of interest before sampling a signal and then sample at more than twice that frequency. Fig. 2.1(a) and 2.1(b) depict a signal  $g(t)$  and its spectrum  $G(f)$  respectively. The impulse response of the filter in Fig. 2.1(c) is convolved with the signal to attenuate signal components above  $f_c$ , forming a bandlimited spectrum (see Fig. 2.1(f)). The ideal sampling function in Fig. 2.1(g) is multiplied by the filtered signal to produce a sampled representation of the signal. It is worth remarking that the repetitions apparent in the spectrum shown in Fig. 2.1(j) are due to the signal spectrum (Fig. 2.1(f)) being convolved with the spectrum of an ideal sampling function. Since the latter, shown in Fig. 2.1(h), consists of more than one impulse, Fig. 2.1(j) displays a sampled version of the spectrum shown in Fig. 2.1(f), repeated at the reciprocal of the sampling period. There is potential for spectral overlap in the sampled signal if the frequencies above  $f_c$  in the original signal (Fig. 2.1(b)) are not attenuated sufficiently. *Aliasing* is the term given to this spectral overlap and any information lost through this process is irrecoverable. The filter responsible for attenuating frequencies higher than  $f_c$  is often called an *anti-aliasing filter* (see Fig. 2.1(c) and (d)).



**Figure 2.1.** An illustration of the sampling theorem: (a) A signal  $g(t)$ , (b) its spectrum  $G(f)$ , (c) An anti-aliasing (low-pass) filter impulse response  $h(t)$ , (d) its frequency response  $H(f)$ , (e) the filtered (band-limited) signal  $g(t) \circ h(t)$ , (f) the spectrum of the filtered signal  $G(f)H(f)$ , (g) an ideal sampling signal  $\Delta(t)$ , (h) the spectrum of the ideal sampling function, (i) the sampled signal  $g[k] = g(kT)$ , (j) its spectrum  $G[n] = G(n/T)$ .

## 2.2 THE SOURCE FILTER MODEL

Conveying information to another person by spoken language involves producing a wide range of sounds. A good speech model represents each of these sounds accurately enough for them to be reproduced from a set of model parameters. Fig. 2.2 illustrates three different speech models. The source filter model shown in Fig. 2.2(a) is the simplest of these models, consisting of only a source and a filter. This model takes no account of the lips and simplifies the effect of the glottis. Furthermore, any interaction that may occur between the source and the filter is ignored. A more detailed model which incorporates these lip and glottal effects is depicted in Fig. 2.2(b) and is called the speech production model. The third model, shown in Fig. 2.2(c), is the articulatory speech synthesis model (Sondi and Schroeter, 1987) which models the nasal tract and glottal pulse in detail. The latter two models are contrasted with the source filter model to highlight the various assumptions invoked in the models and to emphasize improvements that might be made to the simplified model.

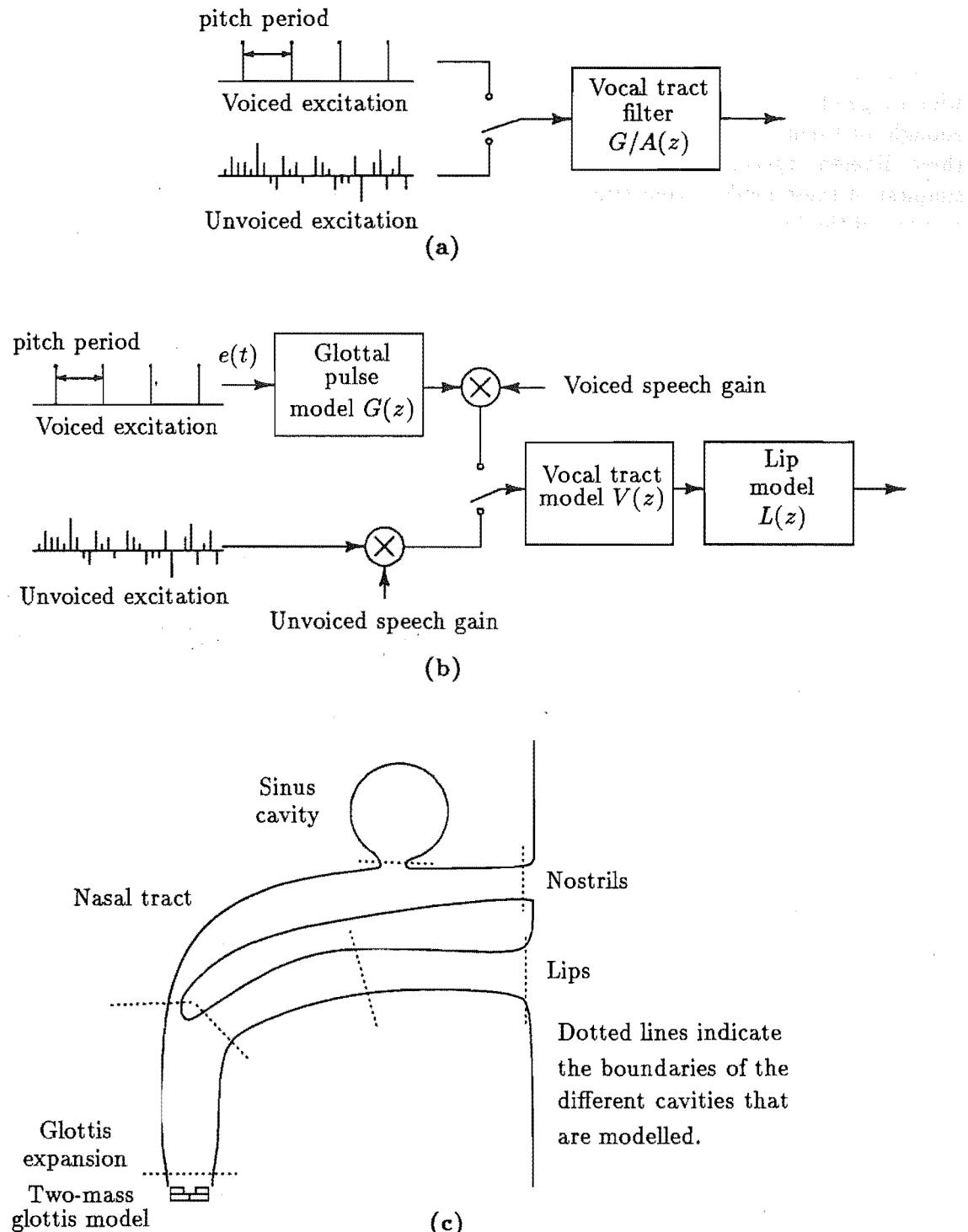
### 2.2.1 Source models

The source is the part of the source filter model that represents the excitation of the human vocal tract (described in §1.1). It is called the source because it provides the energy to excite the vocal tract filter. The output of the source model is the *excitation* to the vocal tract filter model.

Voiced and unvoiced sounds are different in nature because they are excited in a different manner, as explained in §1.2. Accurate production of voiced and unvoiced sounds therefore requires that appropriate source models be postulated. The source model for voiced sounds is periodic and characterizes the repetitive opening and closing of the vocal cords. Unvoiced sounds are excited by turbulence which is conveniently modelled by a noise source.

A regular train of impulses, occurring at pitch period intervals, constitutes a simple source model for voiced sounds. Each impulse stimulates all resonant frequencies of the vocal tract filter, but its spectral characteristics are different from those of an actual glottal pulse. This difference causes the output from the model's vocal filter to sound unnatural (Schroeder, 1985). The actual glottal pulse, as manifested in humans, has an amplitude spectrum which falls off roughly as  $1/f^2$  (where  $f$  is frequency)(Linggard, 1985, p90). The spectrum of a sawtooth shaped signal matches the glottal spectrum much better than does the flat spectrum of an impulse. The sawtooth excitation thus produces more natural sounding speech (*cf.* Linggard, 1985). Neither the impulse nor sawtooth excitations model the actual operation of the vocal cords. Although they are straightforward to implement, and produce speech of reasonable quality, such excitations are neither physically nor phonetically ideal.

Ishizaka and Flanagan (1972) account more realistically for the functioning of the vocal cords with, what they call, the two mass model. Each vocal cord is approximated by two coupled masses attached to damped springs. The coupled masses represent the vocal cord in two sections, the division being depthwise into an upper and lower portion. The two vocal cords are assumed to be bilaterally symmetric and therefore the movement of only one of them needs to be modelled. An estimate of the constant sub-glottal air pressure, as might be produced by the lungs, provides the excitation for the model. The model glottal flow (or volume velocity) is evaluated by solving differential equations which describe the pressures within the glottis and the motion of the vocal cord masses. The most interesting results obtained from this model illustrate the importance of interactions between the vocal tract shape and the glottal flow. The first formant resonance of the vocal tract is identified as having the greatest effect on



**Figure 2.2.** Models of speech production;(a) source filter model,(b) speech production model,(c) articulatory speech synthesis model (Sondi and Schroeter, 1987).

the glottal flow. For example, the glottal flow signal calculated for /a/ has ripples visible on the positive slope portion whereas /i/ exhibits no ripples at all. Although the ripples appear to be relatively small when compared with the total amplitude of the glottal flow, they do alter the spectrum of the vocal tract excitation. This implies that the source is not completely independent of the resonances of the vocal tract filter (Rothenburg, 1983).

Unvoiced excitation, a noise-like turbulence, is usually modelled by a sequence of random numbers, as depicted in Fig. 2.2(a). In the standard source filter model the unvoiced excitation is applied to the input of the vocal tract filter. However, the excitation would be better applied at the point within the vocal tract model where the greatest air turbulence occurs. This would more accurately model the physiological production of unvoiced sound. The first synthesis scheme devised to model such excitation is described by Flanagan and Cherry (1969), who associate a noise source with each section of the vocal tract. The vocal tract model consists of a number of tubular sections (see §2.3.2), with the air flow within each being determined during modelling. Noise excitation is automatically introduced at all constrictions where the conditions for turbulent flow are present. More recently, Sondi and Schroeter (1987) have modelled the turbulence at a vocal constriction by positioning a single noise source appropriately. However, these schemes are only useful for synthesizing speech with parameters that are set by hand. To the author's knowledge, algorithms have not yet been developed for estimating the position of a noise source within the vocal tract from the speech signal alone.

The source depicted in Fig. 2.2(a) models the excitation as being either purely voiced or purely unvoiced. However, some sounds, for example /v/, are excited by a combination of voiced and unvoiced excitation. In order to model the excitation for such sounds more realistically, Kwon and Goldberg (1984) calculate the voiced and unvoiced energy of a segment of speech and produce what they call 'mixed excitation'. The ratio of voiced to unvoiced excitation is estimated from the speech. They claim that speech analysed and then synthesized with their technique sounds slightly better than that produced using a binary voiced-unvoiced decision.

Another common excitation consists of many carefully positioned impulses placed within one pitch period. It has been given the highly descriptive name *multi-pulse* excitation (Atal, 1985, p101). The pulse positions and amplitudes are adjusted until the difference, or modelling error, between the model output and a given speech signal is minimized. The modelling error can also be reduced by increasing the number of pulses in the multi-pulse excitation. The error is weighted so that certain parts of the spectrum contribute less to the total error than others. Larger errors are acceptable in the parts containing the formants because they contain more energy than the parts between formants (Atal, 1985, p107). A weighting function is therefore chosen which de-emphasises the error contribution in the region of the formants. In practical application multi-pulse approximates both the voiced and unvoiced excitation, with only a few pulses (typically 8 pulses per 10ms) being sufficient for generating both voiced and unvoiced sounds with little audible distortion. Separate voiced and unvoiced source models are not required since the same pulse positioning algorithm is utilized for both excitations. However, the lack of pitch information necessitates coding the positions and times of all the impulses. Good quality speech synthesis using the multipulse technique has been reported by Kroon *et al.* (1986).

### 2.2.2 Filter models

In order to accurately model the response of the vocal tract it is important to choose the correct type of filter for the source filter model. A general digital filter which could conceivably be used to model the vocal tract is

$$V(z) = \frac{B(z)}{A(z)}, \quad (2.21)$$

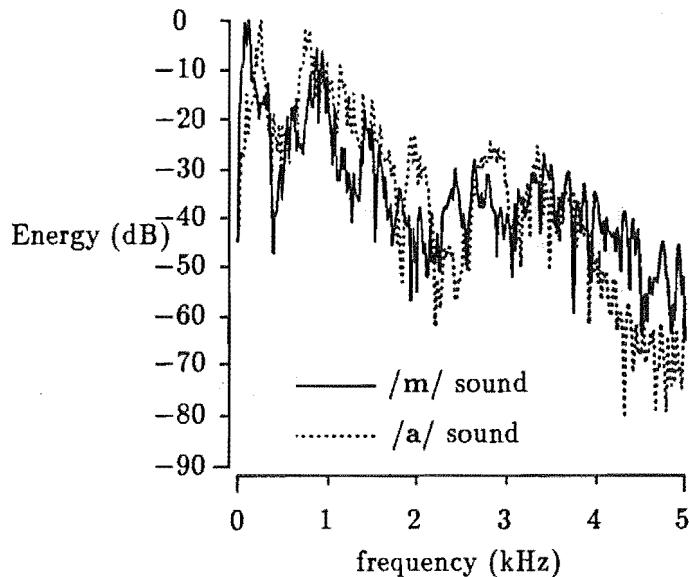


Figure 2.3. Spectrum of the sound /m/ and the sound /a/ taken from the word “mad”.

with  $A(z)$  and  $B(z)$  being polynomials defined by,

$$\begin{aligned} A(z) &= \sum_{i=0}^P a_i z^{-i} \quad (a_0 = 1), \\ B(z) &= \sum_{i=0}^M b_i z^{-i} \quad (b_0 = 1). \end{aligned} \quad (2.22)$$

As  $A(z)$  tends to zero,  $V(z)$  approaches infinity. The complex values of  $z$  for which  $A(z) = 0$  are called the *poles* of  $V(z)$ . Likewise, since  $V(z) = 0$  when  $B(z) = 0$ , the complex values of  $z$  for which  $B(z) = 0$  are called the *zeros* of  $V(z)$ .  $P$  and  $M$  define the number of poles and zeros respectively of  $V(z)$ .

The poles of the filter can be revealed explicitly by factorizing  $A(z)$ , i.e.,

$$A(z) = \prod_{i=0}^{P-1} (z - \xi_i), \quad (2.23)$$

where  $\xi_i$  is the  $i^{th}$  pole in the complex  $z$ -plane. Poles cause peaks in the filter frequency response because  $V(z)$  tends to infinity as  $z$  approaches a pole. In a vocal tract filter these ‘peaks’ correspond to formants in the speech spectrum (Schroeder, 1985). The bandwidth of a formant associated with a pole depends upon the pole’s distance from the unit circle. A pole located close to the unit circle projects a narrower bandwidth than a pole nearer the origin. It is important to realise that a formant on the unit circle is comprised of the projections from *all* the poles of the filter. The formant bandwidths of speech are modelled by positioning the vocal tract filter poles at appropriate distances from the unit circle (Linggard, 1985, p85). Similarly, the zeros of the filter are revealed by factorizing  $B(z)$ . These zeros model the absorption of energy in the vocal tract. This absorption attenuates the energy in the speech spectrum close to the real frequencies of the zeros. The resulting reduction in speech energy is observed as a ‘valley’ in the speech spectrum, as shown in Fig. 2.3.

One method of faithfully modelling speech is to choose a combination of poles and zeros and position them appropriately. Both speech production and perception influence the number of poles and zeros required in the vocal tract filter. For instance,

nasal sounds are produced when the velum is lowered, the nasal cavity opened and the oral cavity closed. However, the oral cavity still resonates and traps acoustic energy quite strongly at certain frequencies (Linggard, 1985, 47). The 'valley' in the spectrum of the nasal /m/ sound, shown in Fig. 2.3 at approximately 500 Hz, can be modelled by appropriately positioning zeros of the vocal tract filter.

Song and Un (1983) describe tests of four different algorithms for determining pole-zero models for speech. They propose a system that uses ten poles and ten zeros and report that nasal speech synthesized with the aid of the pole-zero model is of better quality than the same speech synthesized utilizing only poles. However, they also report that the improvement in the quality of speech comprising nasal and non-nasal sounds is not apparent to normal listeners.

It is important to take into account the relative ability of our ears to detect spectral poles and zeros when assessing the relative importance of poles and zeros in the vocal tract filter. Our hearing is known to be more sensitive to 'peaks' in the speech spectrum than to 'valleys' (or zeros) (Schroeder, 1984). Correspondingly, against a noisy background, it is easier to detect the presence of a signal comprising a single frequency tone, a sharp peak, than its absence. For the above reasons, the vocal tract filter utilized in this thesis contains only poles and is called an *all-pole* model. The polynomial  $B(z)$ , introduced in (2.22), is replaced by a constant which is set to unity for convenience. The effect of the zeros on the frequency response can be emulated by incorporating additional poles (Atal, 1985) which are positioned in such a manner that the 'valleys' between pole peaks reproduce as closely as possible the 'valleys' produced by the spectral zeros.

The modelling schemes illustrated by Figs. 2.2(a) and (b) approximate both the nasal and oral tracts by single filters that are isolated from the vocal tract source. However, as mentioned in §2.2.1, there is considerable interaction between the source and vocal tract components. Fig. 2.2(c) shows an articulatory speech synthesis model which has compartments that represent both the nasal and oral tracts. The response of each of these compartments is modelled, resulting in more accurate speech synthesis than that obtained from the simple source filter model of Fig. 2.2(a). Although such a model is physiologically accurate, its usefulness is restricted because an algorithm has not been developed for determining articulatory parameters directly from the speech signal (Sondi and Schroeter, 1987).

## 2.3 ACOUSTIC MODEL OF THE VOCAL TRACT

When we wish to utter a particular sound, we configure our vocal tracts in the appropriate manner. The spectral characteristics of the uttered sound depend upon the vocal tract shape (and also upon voicing and pitch, but this discussion concentrates on the vocal tract). The vocal tract can be modelled as a series of concatenated tubes of different sizes that correspond to the size of the vocal tract at various places. Articulators, such as the tongue, are considered to alter the sizes of the vocal tract tubes as different sounds are produced.

Certain simplifying assumptions about the tube model are outlined in §2.3.1 and, under these assumptions, propagation of a speech signal in a single tube section is discussed in §2.3.2. Propagation along a number of concatenated tubes is described in §2.3.3. This vocal tract model can be invoked to determine the reflections which occur at the boundaries between tubes. The fraction of the speech signal reflected at each boundary is defined by a *reflection coefficient*. Section 2.3.4 discusses computation of the cross-sectional area of vocal tract tubes from reflection coefficients.

### 2.3.1 Simplifying assumptions

By ignoring the effect of the nasal tract on speech production, modelling can be considerably simplified. Fortunately, nasal sounds constitute a small fraction of English speech sounds. Therefore, little modelling error results from neglecting the overall effect of the nasal tract on the speech signal. Such a model, that consists only of a vocal tract, does not model the nasal tract, so any nasal effects in the speech signal must be approximated by the vocal tract.

The vocal tract is modelled as a series of tubes, which are assumed to be lossless. It is also assumed that losses in the speech signal due to viscosity and heat conduction are negligible. Although certain synthesis models incorporate lossy tubes (Flanagan, 1972; Sondi and Schroeter, 1987), such models are not suitable for evaluating vocal tract parameters (see §2.2) because analysis required is too complicated.

The complete vocal tract model comprises a number of tubes of different cross-sectional area. Within each acoustic tube it is assumed that the acoustic variables (density, pressure, etc.) have constant amplitude on any given plane perpendicular to the direction of propagation of the speech signal (Kinsler *et al.*, 1982). These assumptions are reasonable when the transverse dimension of each tube is small compared with the wavelength of the speech (Wakita, 1973).

### 2.3.2 A single tube

The speech pressure and volume velocity signals are here denoted as functions of position within the tube and time by  $p(x, t)$  and  $u(x, t)$  respectively. For convenience, the subscript  $m$  is introduced to refer to separate tube sections of the complete vocal tract. Quantities relating to the  $m^{\text{th}}$  tube are identified by the subscript  $m$ , with  $m = 0$  being at the lips and  $m$  increasing towards the glottis. Because the cross-sectional area of the tube is constant along its length, both the pressure and volume velocity satisfy one-dimensional wave equations (Markel and Gray, 1976, p63):

$$\frac{\partial^2(\bullet)}{\partial x^2} = \frac{1}{c^2} \frac{\partial^2(\bullet)}{\partial t^2} \quad (2.24)$$

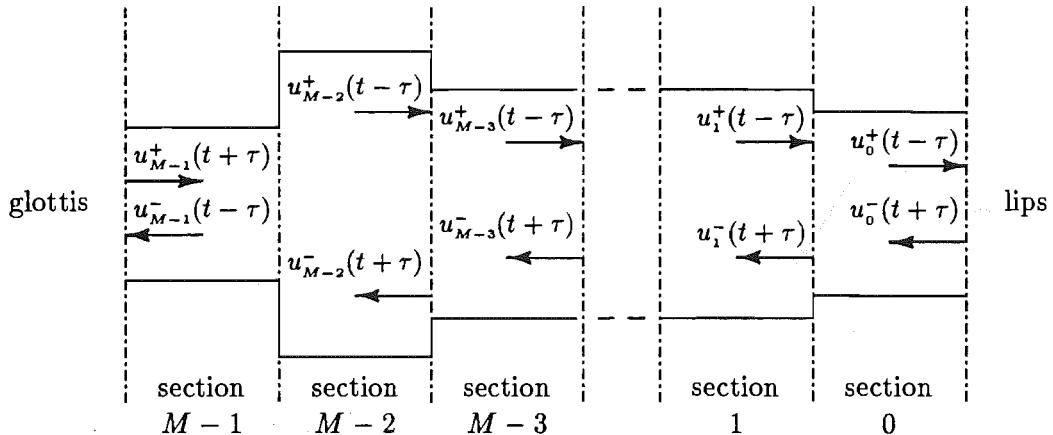
and

$$\begin{aligned} u_m(x, t) &= u_m^+(t - x/c) - u_m^-(t + x/c), \\ p_m(x, t) &= p_m^+(t - x/c) - p_m^-(t + x/c), \end{aligned} \quad (2.25)$$

where  $(\bullet)$  represents either  $u_m(x, t)$  or  $p_m(x, t)$ . Signals labelled with a  $+$  superscript travel from the glottis to the lips and are called forward travelling waves. Similarly, the  $-$  superscript identifies backward travelling signals which propagate from the lips to the glottis. The tube is of total length  $l$  and  $x$  is the distance along the tube axis from the centre of the tube. The time taken for the forward travelling signal to pass from the centre of a tube ( $x = 0$ ) to its end ( $x = l/2$ ) is  $l/2c$ , where  $c$  is the speed of sound in  $\text{ms}^{-1}$ . The forward travelling volume velocity signal, at the end of the tube (i.e.  $x = l/2$ ) is written as  $u_m^+(t - \tau)$  where  $\tau = l/2c$ . Fig. 2.4 shows the volume velocity at the edges of a tube for both the forward and backward travelling signals.

### 2.3.3 Lattice formulation

A complete vocal tract tube model consists of a number of constant diameter tubes concatenated together. Within these tubes the velocity signal propagates from the glottis to the lips. Reflections occur at interfaces between tubes of different diameters. The fraction of the volume velocity signal that is reflected at the end of the  $m^{\text{th}}$  tube is



**Figure 2.4.** The volume velocity at the boundaries of the tubes constituting the vocal tract model (from Markel and Gray 1976).

denoted by  $\mu_m$ , which is called the reflection coefficient. The component of a forward travelling signal that is reflected from a tube boundary becomes part of the backward travelling signal wave in the same tube. This is illustrated in Fig. 2.4.

By utilizing the reflection coefficients, and realizing that air is continuous within the vocal tract model, Wakita (1973) develops expressions for forward and backward travelling signals. A forward travelling signal in the  $m^{th}$  tube is composed of the forward travelling signal transmitted in the  $(m+1)^{th}$  tube plus the reflected component of the backward travelling signal in the  $m^{th}$  tube. An analogous statement holds for backward travelling signals. The forward and backward travelling signals at the righthand edge of the  $m^{th}$  tube are defined by Markel and Gray (1976, p66) as

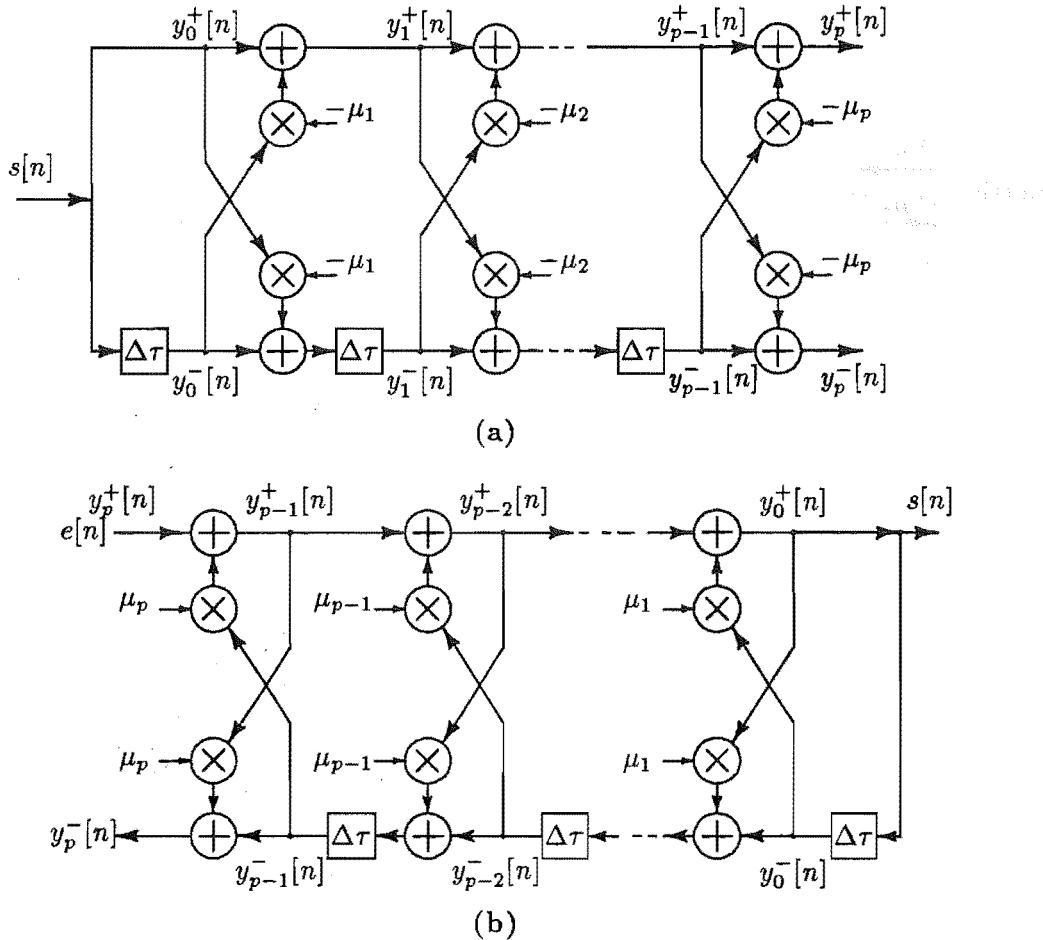
$$\begin{aligned} u_{m-1}^+(t+\tau) &= \mu_m u_{m-1}^-(t-\tau) + (1 + \mu_m) u_m^+(t-\tau) \\ u_m^-(t+\tau) &= (1 - \mu_m) u_{m-1}^-(t-\tau) - \mu_m u_m^+(t-\tau). \end{aligned} \quad (2.26)$$

In order to solve (2.26) it is necessary to specify vocal tract boundary conditions (i.e. the termination conditions at the glottis and mouth). Wakita (1973) outlines conditions which produce realistic vocal tract configurations. At the boundary of the vocal tract formed by the lips, it is assumed that the final vocal tube is connected to another tube of infinite cross-sectional area. This implies that the pressure signal goes to zero at the lips and that there is no radiated pressure wave. The reflection coefficient  $\mu_0$  is therefore set to be unity. Fortunately, this assumption produces almost identical results to those obtained with a more realistic description of the lip radiation (Markel and Gray, 1976, p68). At the other vocal tract boundary, formed by the glottis, the glottal excitation is taken to be a forward travelling volume velocity signal applied to the tube nearest the glottis (Markel and Gray, 1976, p71).

Rearranging the expressions for forward and backward travelling signals and transforming to the Z-domain gives (Markel and Gray, 1976, §4)

$$\begin{aligned} Y_m^+(z) &= Y_{m-1}^+(z) + \mu_m Y_{m-1}^-(z) \\ Y_m^-(z) &= z^{-1}(\mu_m Y_{m-1}^+(z) + Y_{m-1}^-(z)), \end{aligned} \quad (2.27)$$

where  $Y^+$  and  $Y^-$  represent the volume velocity along with terms containing  $(1 + \mu_m)$ , but in terms of a Lattice formulation as depicted in Fig. 2.5. The variables  $Y^+$  and  $Y^-$  describe the forward and backward propagating error signals in the lattice (Itakura and Saito, 1973).



**Figure 2.5.** A lattice arrangement of the forward and backward propagating error signals within the vocal tract model: (a) analysis, (b) resynthesis.

The role of the error signal is best demonstrated by considering the lattice analysis scheme shown in Fig. 2.5(a), where each individual lattice stage corresponds to a single tube of the vocal tract model. The positive and negative travelling errors are optimized as they travel through the lattice by choosing an appropriate value for  $\mu_m$  at each stage. These errors are minimized by predicting the value of the current speech sample from previous samples. The first stage of the analysis filter predicts the current sample from the last sample. The second stage adds prediction information from the second to last sample. Thus, two lattice stages predict the current sample from the two previous samples. Further lattice stages add to the order of the prediction, up to the total number of stages in the filter. The component of the speech that is not predicted is called the *prediction error* and the  $\mu_m$  are chosen so that this prediction error is minimized. The optimum values for  $\mu_m$  are determined by evaluating the partial correlation between the forward and backward travelling error signals (Markel and Gray, 1976, p41). The  $\mu_m$  are therefore sometimes referred to as PARCOR (from partial correlation) coefficients. The PARCOR equation is (Makhoul, 1977)

$$\mu_m = \frac{\sum_{n=-\infty}^{\infty} y_{m-1}^+[n]y_{m-1}^-[n]}{\left( \sum_{n=-\infty}^{\infty} (y_{m-1}^+[n])^2 + \sum_{n=-\infty}^{\infty} (y_{m-1}^-[n])^2 \right)^{1/2}}, \quad (2.28)$$

where the denominator of (2.28) is the geometric mean of the sum of the forward and backward travelling error signals in the  $(m-1)^{th}$  stage. The numerator of (2.28) contains the correlation between the forward and backward travelling error signals within the same section. The lattice structure allows updating of reflection coefficients on a sample by sample basis. This contrasts with the standard LPC approach which requires that a complete speech segment be collected before analysis can proceed (see §2.5).

A reflection occurring at any tube boundary is always of less energy than the incident signal, implying that the magnitude of the reflection coefficient must always be less than unity. Provided computation of vocal tract model coefficients yields  $\mu_m < 1$  for all  $m$ , the energy at the output of the vocal tract model tends rapidly to zero after the input vanishes (as is of course physically necessary). Since a stable output is produced when all the  $\mu_m < 1$ , such  $\mu_m$  constitute what is called a *stable* set of reflection coefficients. If, as a result of accumulated round-off error in the optimizations, one or more of the  $\mu_m$  becomes greater than unity, the output of vocal tract model ceases to be physically acceptable because its energy density can then exceed that of the input. The set of reflection coefficients is then said to be *unstable*. However, a useful property of employing (2.28) to compute  $\mu_m$  is that the stability of reflection coefficients is guaranteed (Makhoul, 1977). Other methods of computing the coefficients (see §2.5.3) do not guarantee stability.

Stability of reflection coefficients is guaranteed if they are calculated by linear interpolation between two other sets of stable reflection coefficients. This is because the coefficient interpolated between two coefficients that are both less than unity, is also always less than unity (Rabiner and Schafer, 1978, p446). This property is often utilized when averaging between sets of reflection coefficients, as might occur in the training of a vector quantizer (see §2.7.4.2).

#### 2.3.4 Area function

The relative cross-sectional areas of the vocal tract tubes can be calculated from reflection coefficients by invoking Wakita (1973)

$$\frac{\mathcal{A}_m}{\mathcal{A}_{m-1}} = \frac{1 - \mu_m}{1 + \mu_m}, \quad (2.29)$$

where  $\mathcal{A}_m$  is the cross-sectional area of the  $m^{th}$  tube. It is important to realize that (2.29) defines the ratio of the areas of adjacent tubes. Typically, the area of the first or last tube is normalized to unity and all other areas are evaluated relative to it. This normalized cross-sectional area of the model vocal tract, expressed as a function of position, is called the *area function* and is written  $\text{area}(x)$ .

The exact relationship between the calculated area function of a particular sound and the vocal tract shape utilized for producing the sound is critical if vocal tract shapes are to be inferred from area functions. Sondhi (1979) raises some interesting points in this regard. First, for a lossless tract which is modelled as being closed at the glottis and open at the lips, the area function is not unique. For instance, area functions  $\text{area}(x)$  and  $1/\text{area}(x)$  both have the same transfer function (Bonder, 1983). Second, the area function is critically dependent on the formant bandwidths. Sondhi (1979) illustrates this by plotting a set of four area functions for each of the vowels /a/ and /u/. Each member of each set is derived from four formants which have the same frequency and amplitude, but slightly different bandwidths. For the /a/ sound, the area functions have the same overall trend and worst case differences between them are as large as 50%. This sensitivity to formant bandwidth is highly significant because formant bandwidths are difficult to estimate accurately from the speech signal (Sondhi, 1979). Third, because

the energy density of the glottal source falls off rapidly at high frequencies (see §2.5.2), the transfer function of the vocal tract cannot be estimated reliably at frequencies higher than 3kHz. Wakita (1973) calculates area functions for a number of vowels and reports results that correspond closely to the expected vocal tract shape. These area functions are obtained after (approximately) removing the effects of the lips and glottis by pre-emphasizing the speech. Wakita also examines area functions for sounds that have not been pre-emphasized and finds that the area functions have a physically unrealistic shape. He therefore concludes that pre-emphasis is essential for determining realistic models of the vocal tract shape. Sondhi's (1979) explanation of these different results is that the shape of the final area function depends strongly on assumptions made about the glottal source and the lip radiation (pre-emphasis models these effects).

An alternative method of determining the vocal tract shape involves measuring the response of the vocal tract to an impulse applied at the lips. Sondhi and Gopinath (1971) describe a method that uses an impedance tube placed next to the subject's lips. The impedance tube contains both a source for exciting the vocal tract and a microphone for measuring the vocal tract response. To calculate the area function of a particular sound, subjects attempt to hold their mouths in positions which they would use to produce that sound. The source in the impedance tube excites the vocal tract and the area function is computed from measurements of the reflected signal. The disadvantage of this method is that the placement of the impedance tube against the lips inhibits the subject physically and is likely to cause the shape of the subject's vocal tract to differ significantly from the shape it would have in unconstrained speech. The authors report tests of the impedance device on a metal tube of known shape. The calculated tube shape, determined from measurements of the tube response, closely matched the actual shape of the tube.

Interestingly, although the approach developed by Wakita (1973) cannot lead to a unique connection between what is measured and the shape of the vocal tract, it usually generates what appear to be quite reasonable estimates of the vocal tract shape (Elder *et al.*, 1987; Watson *et al.*, 1991). From a practical point of view, it is irrelevant whether Wakita's shapes are non-unique if reasonably accurate vocal tract representations are generated from actual speech signals.

## 2.4 PROSODIC CHARACTERISTICS

As people speak they utter a range of sounds that are either voiced or unvoiced, with the voiced sounds having a particular pitch. Speech model parameters must be updated regularly so that these changes are recorded. Speech segmentation, as described in §2.4.1 is used to extract a portion of the total speech signal for analysis. Methods of intensity computation, pitch detection and voiced/unvoiced analysis are described respectively in §2.4.2, §2.4.3 and §2.4.4.

### 2.4.1 Speech segmentation

Speech comprises a succession of various phonemes (§1.2) of short duration, typically less than 0.1s. Each phoneme can be characterized by a set of parameters which describe properties of the sound. These parameters vary rapidly as different phonemes are produced, particularly for short phonemes such as /t/. To record the temporal variations of these parameters it is necessary to divide the speech into short *speech segments* or *frames* which are less than the phoneme length. The length of a speech segment is called the *segment length* and is typically 0.01-0.02s. Reference is often made in this thesis to 'short' or 'long' speech segments. The implication is that the lengths of these segments are chosen to be significantly longer (or shorter, as the case may be)

than the pitch period. When a speech record (of an utterance) is divided into a number of segments, the time difference between consecutive speech segments is called the *time step*. It is also convenient in certain contexts to state the length of a speech segment in samples. Whenever the sampling rate is given or known, the segment length and number of samples in a segment are used interchangeably.

### 2.4.2 Intensity

Recall from 1.3.1, that intensity is a measure of the speech signal that is related loosely to the perceived loudness. In applications such as speech analysis and synthesis, it does not matter that the intensity does not correspond to a defined loudness scale, because it is used to ensure that the correct loudness levels are preserved, rather than as an absolute measure.

The RMS intensity of a single frame of  $N$  speech samples, is expressed as

$$I = \sqrt{\frac{\sum_{i=0}^{N-1} s[i]^2}{N}} \quad (2.30)$$

where  $I$  is the RMS intensity. Note that the RMS intensity of a sequence of speech frames is denoted  $I[m]$ , where  $m$  is the frame number.

Throughout this thesis the RMS intensity is used to estimate the amplitude of the speech signal within a speech frame. Compared with the energy of the signal, which is the correlate of loudness used by many researchers (Rabiner and Schafer, 1978, §4), the RMS intensity has the advantage that the intensity value is of the same scale as the speech signal.

### 2.4.3 Pitch estimation

This section reviews a wide range of pitch estimation techniques and describes the particular method employed in the work reported in §4.4.2.3.

Recall from §1.2.2 that the pitch is related to the periodic vibration of a person's vocal cords. The pitch is difficult to estimate because the resonant cavities of the vocal tract distort the glottal excitation as it travels towards the lips. In addition, there are small perturbations in the period of the glottal excitation that complicate pitch detection methods.

Apart from the speech production process, external factors such as signal conditioning also affect the pitch procedure. For example, when speech has been bandpass filtered, as often occurs in the telephone system, the pitch, which is below the 200-300 Hz cutoff, must be detected from its harmonics.

Note that some researchers extend the pitch detection problem into the investigation of ways to determine the exact opening and closing times of the glottis (Cheng and O'Shaughnessy, 1989; Ananthapadmanabha and Yegnanarayana, 1979; Moulines and Di Francesco, 1990), but here the discussion is restricted to pitch detection methods.

Rabiner *et al.* (1976) divide pitch finding algorithms into the categories of time domain methods, frequency domain methods and hybrid methods that incorporate both time and frequency domain methods. The reader is referred to Rabiner *et al.* (1976) and McGonegal *et al.* (1977) for comparative tests of many of the algorithms described below.

#### 2.4.3.1 Time domain methods

Time domain methods use the time domain properties of the speech signal, such as the position of peaks or zero crossings, to estimate the pitch period. The detection of such

temporal features in the speech is usually a computationally straightforward procedure, so time domain techniques are well suited to real-time implementation (*cf.* Samouelian and Holmes, 1985).

One of the early, and well known, time domain methods is the parallel processing method proposed by Gold and Rabiner (1969). The name parallel-processing arises because the speech signal is processed simultaneously by several different pitch detectors. Rabiner and Schafer (1978, p136) summarize the basic principles of the scheme as follows:

1. The speech signal is processed to create a number of impulse trains which retain the periodicity of the original signal but discard features that are irrelevant to the pitch detection process.
2. Simple pitch detectors are used to estimate the period of each impulse train.
3. The estimates obtained from several of these simple pitch detectors are logically combined to infer the period of the speech waveform.

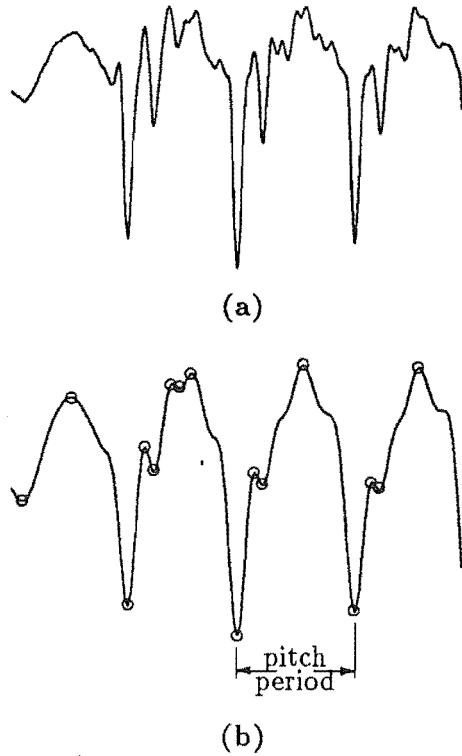
Several other pitch algorithms have evolved from Gold and Rabiner's algorithm (Sutherland *et al.*, 1988; Tucker and Bates, 1978). The pitch detector used for the work reported in §4.4.2.3 was developed by Brieseman (1984) and is a modification of Gold and Rabiner's algorithm and the peak tracking pitch detection algorithm developed by Tucker and Bates (1978).

Brieseman's pitch detector performs pattern matching amongst positive and negative peaks (extrema) in the speech signal. First the speech signal is filtered to remove any components with frequencies higher than 500 Hz. Extrema in the signal are located and their positions and amplitudes stored. Note that the extrema amplitude and position information represents a considerable compression of the original speech signal, making it feasible to store all the extrema from many past pitch periods for matching purposes. A matching algorithm is then invoked to match extrema amplitudes with previous extrema, and if enough extrema are matched (to within a specified threshold and having a similar time shift) the pitch period is taken to be the common time shift. Fig. 2.6 shows a segment of filtered speech, the extrema, and the computed pitch period. If the extrema match well across a single pitch period, they usually also match across two pitch periods. It is therefore important to choose the smallest possible pitch period for which the extrema are matched. The range of possible pitch values is limited by this algorithm to lie within the range of 75 to 500 Hz so that the amount of matching required is reduced.

An alternative time domain technique is to compute the difference between segments of the speech signal rather than locating and matching peaks. Ross *et al.* (1974) use such a method to form a difference signal between a delayed version of the speech signal and the original. Hence their technique is entitled the average magnitude difference function (AMDF) pitch extractor. The main advantage of the AMDF method is that it is computationally efficient, because no multiplications are required.

#### 2.4.3.2 Frequency domain methods

Frequency domain methods use spectral properties of the speech waveform to estimate the pitch. A typical example of such a method is the cepstral technique described by Rabiner and Schafer (1978, p314). The cepstrum of a section of speech has a peak due to pitch harmonics in the spectrum (for example, see Fig. 2.10(a) in the region of  $n = 85$ ). The pitch can be reliably estimated from the position of this peak. However, this method requires a considerable amount of computation to determine both the



**Figure 2.6.** An example of part of the pitch calculation algorithm: (a) the original speech, (b) the positions of extrema in the speech after components above 500 Hz have been removed.

spectrum and cepstrum (Rabiner *et al.*, 1976). Other pitch detection algorithms are therefore more attractive since they perform at least as well as the cepstral algorithm and have lower computational requirements.

Another example of a frequency domain pitch detection method is the comb filter technique (Moorer, 1974). This approach works by computing the inner product between the autocorrelation of the speech and a comb filter, for various comb spacings (Wise *et al.*, 1976). The spacing that has the largest inner product is directly related to the pitch period of the speech.

#### 2.4.3.3 Hybrid frequency domain and time domain methods

A typical time and frequency domain pitch detection algorithm performs signal conditioning on the speech signal in the frequency domain before applying a time domain technique to the conditioned signal to determine the pitch period.

An early published example of this type of pitch algorithm is the method proposed by Sondhi (1968) which uses the autocorrelation of spectrally flattened speech to estimate the pitch. Sondhi's spectral flattening technique uses band-pass filters and time delays to produce an effect that is similar to filtering by an inverse linear prediction filter (see §2.5) in that sharp peaks at pitch period spacing are made prominent. The autocorrelation of the spectrally flattened speech has a peak at the time shift that corresponds to the pitch period. Another method of pitch detection, based on a similar concept, is the simplified inverse filtering technique (SIFT) developed by Markel and Gray (1976, p206) (see Rabiner *et al.* (1976) for a detailed evaluation). The SIFT algorithm works by low pass filtering at 900 Hz a block of 40 ms of speech (10kHz sample rate), decimating the speech at a ratio of 5 to 1 and then passing the decimated speech through a 4th order inverse linear prediction filter (see §2.5) before performing autocorrelation analysis on the residual signal. The significant peaks in the residual

signal occur in the regions of glottal excitation within the original speech, so the main peak in the autocorrelation function can be used to indicate the time between the peaks in the residual signal. The SIFT algorithm accurately determines the position of the peak in the autocorrelation function by interpolating sample values around the peak.

#### 2.4.4 Voiced/unvoiced decision methods

An important aspect of speech analysis is the determination of whether a segment of speech is voiced or unvoiced. This is particularly true for any speech analysis algorithm, such as pitch detection, that is critically dependent on whether or not the vocal cords are vibrating. This section reviews several different voiced/unvoiced (V/UV) decision methods and specifies the particular methods used in this thesis for performing the V/UV decision.

V/UV classifiers are closely related to silence detectors (De Souza, 1983) and are sometimes called voiced/unvoiced/silence detectors (*cf.* Atal and Rabiner, 1976). They also often operate in conjunction with pitch detectors.

One approach to performing the V/UV decision is to adopt the pattern classification technique of Atal and Rabiner (1976). Parameters measured from a segment of speech are used to decide whether the particular segment of speech should be classified as voiced, unvoiced or silent. Atal and Rabiner (1976) employed as parameters the zero crossing rate, the speech energy, the correlation between adjacent samples, the first predictor from a 12-pole linear predictive coding (§2.5) analysis, and the energy in the prediction error (§2.5.3). A training set of manually classified speech data is used to determine the means and covariances of the specified parameters. Speech segments are then classified under the assumption that the parameters are distributed according to the multidimensional Gaussian probability density function.

Tests of a neural network V/UV classifier are reported by Bendiksen and Steiglitz (1990) who use a neural network with 6 input nodes, 10 internal nodes and 2 output nodes to implement a V/UV decision. In their system the 6 input features were the RMS energy of the speech, the RMS energy of the pre-emphasized speech, the first auto-correlation coefficient of the speech, the first auto-correlation coefficient of the pre-emphasized speech, the ratio of the speech energy above 4000 Hz to the energy below 2000 Hz and the product of the speech energy above 4000 Hz and below 2000 Hz. The neural network was trained with a total of 72 frames of speech taken from 2 speakers and tested with 479 frames of speech taken from 4 speakers. A total of 2 misclassifications were reported from all the test frames. Bendiksen and Steiglitz concluded that the network classifier performed as well as other V/UV classifiers based on pattern recognition methods.

For any sort of V/UV classifier, Siegel (1979) points out that it is important to ensure that the training segments of speech are taken from a range of sounds and speakers in order to ensure the classification algorithm works correctly on speech from people not included in the training set. However, training is more complicated with large numbers of training utterances, so a compromise is usually made between using large quantities of training data and inadequately describing the V/UV characteristics of speech. Siegel (1979) found that 179 training patterns and 6 features were enough to specify a good classifier. However, the methodology for evaluating the effect of varying the number of training patterns is not described in detail so it is difficult to know what significance to attach to this result.

A particularly simple V/UV decision method is to compare energy in different parts of the speech spectrum (Knorr, 1979). The energy contained in the speech signal above 5kHz is compared with that below 1kHz to determine whether the speech is voiced or unvoiced. The main features of this type of method are its simplicity and the ease with

which it can be implemented in real-time. Furthermore, Knorr (1979) reports that the classification accuracy is similar to that obtained by Atal and Rabiner (1976).

For the V/UV decisions required in this thesis (§4.4 and Chapter 5) two different V/UV decision algorithms are used. They are labelled here respectively as VUV1 and VUV2. The first algorithm, VUV1, is similar to that described by Knorr (1979). The RMS intensities of three different signals are compared to determine whether speech is voiced, unvoiced or silent. These signals are; the RMS intensity of the speech signal  $I_s[m]$ , the RMS intensity of the speech signal after it has been low-pass filtered to 200 Hz  $I_l[m]$  and the RMS intensity of the speech after the high frequencies have been emphasized by pre-emphasis (§2.5.2)  $I_h[m]$ . The following decision rule is then invoked to determine whether the frame is voiced, unvoiced or silent:

Segment  $m$  is

- Silent if  $I_s[m] < \zeta_s$ ,
- unvoiced if  $I_s[m] > \zeta_s$  and  $I_l[m] < 1.25I_h[m]$ ,
- voiced if  $I_s[m] > \zeta_s$  and  $I_l[m] > 1.25I_h[m]$ .

where  $\zeta_s$  is the silence threshold. The 1.25 scale factor was determined by trial and error and is included to improve the accuracy of the V/UV decision. Examples of the two signals  $I_l[m]$  and  $1.25I_h[m]$  which are compared to perform V/UV decision are plotted in Fig. 2.7(e). It is apparent that the energy in  $I_h[m]$  is significantly higher than that in  $I_l[m]$  for unvoiced sounds and vice-versa for voiced sounds.

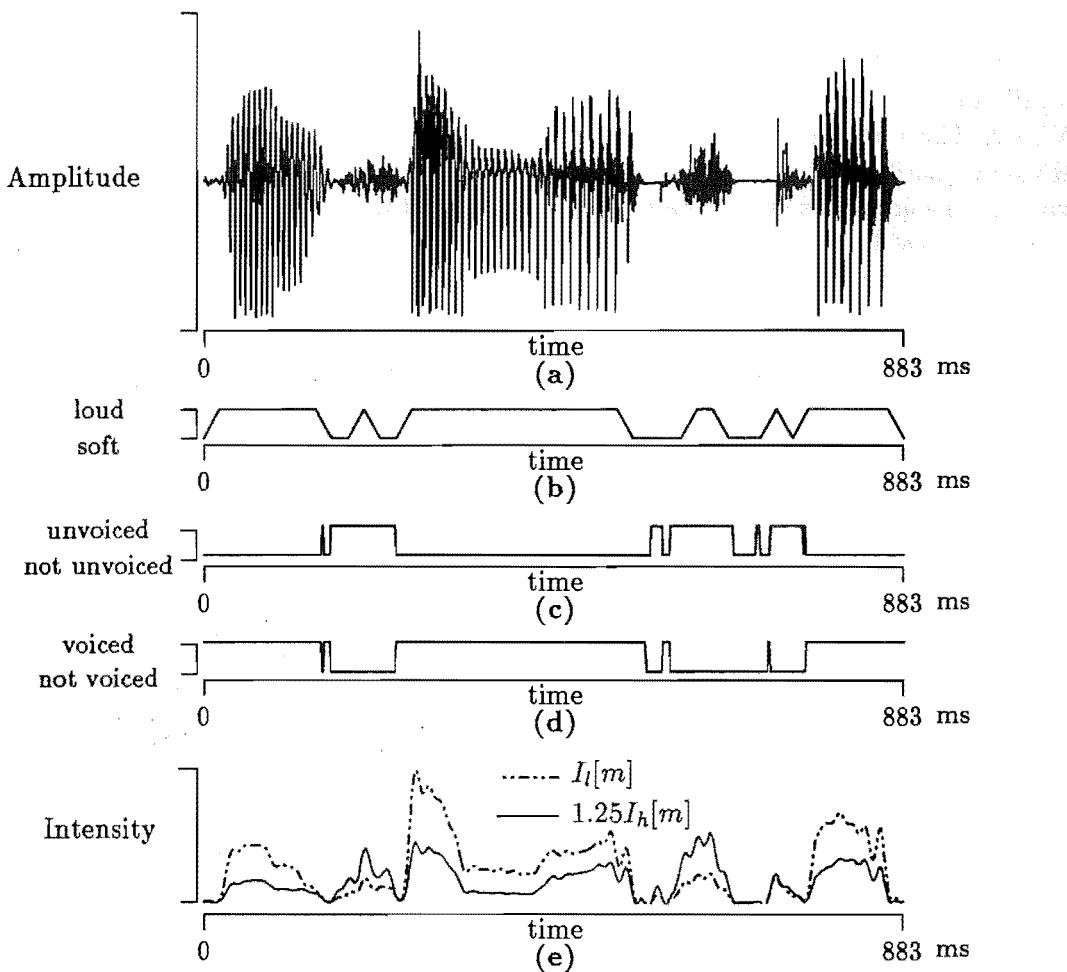
In an application such as speaker identification, a highly accurate V/UV decision is not necessary because the speech excitation is not important. Instead the aim is to extract those frames that contain significant information about a person's vocal characteristics. Here the claim is that if the energy of a speech frame is greater than a particular threshold, that frame contains potentially useful information for distinguishing speakers. The justification for excluding silent frames is that they do not contain any speaker information.

In the speaker identification experiments described in Chapter 5 a large number of utterances must be analysed, making it desirable to extract voiced frames of speech in a computationally efficient manner. The VUV2 algorithm uses the RMS intensity of each frame to indicate whether or not it is likely to be voiced. A threshold is selected relative to the maximum amplitude of the speech utterance and the RMS values of the speech throughout the utterance are compared against it. This algorithm does not perform the V/UV decision in a rigorous manner since, as shown in Fig. 2.7(b), certain loud unvoiced speech frames are erroneously considered to be voiced. However, the silent portions of the speech and those unvoiced frames that have little energy are excluded, thereby reducing the range of sounds to be represented by the speaker template.

## 2.5 LINEAR PREDICTIVE CODING OF SPEECH

Linear predictive coding (LPC) is a powerful modelling technique which has become well established in many areas of speech research (Rabiner and Schafer, 1978). It has been usefully applied in the fields of speech coding, word recognition, speaker recognition and geophysics. These applications all utilize the significant data reduction brought about by LPC to process, store, or transmit signals more efficiently.

An early speaking device which was a forerunner of LPC techniques is described in §2.5.1. The device was used to demonstrate that the information required to reproduce speech could be stored in a (relatively) small number of parameters, or coefficients. The way in which lip and glottal effects are accounted for in LPC is described in §2.5.2.



**Figure 2.7.** An example of the voiced/unvoiced decision for the phrase “When sunlight strike”: (a) the speech signal, (b) detection of the ‘loud’ portions of the speech, (c) unvoiced portions of the speech, (d) voiced portions of the speech, (e) the RMS intensity levels  $I_t[m]$  and  $1.25I_h[m]$  that are used to perform the voiced/unvoiced decision.

The process of determining LPC coefficients involves solving linear prediction equations in the manner outlined in §2.5.3. Section 2.5.4 shows that the LPC formulation corresponds to the lattice structure solution for reflection coefficients, as presented in §2.3.3.

### 2.5.1 An historical perspective

The first electrical device to analyse and reproduce speech is attributed to Dudley (1939). He called the process of reproducing speech ‘remaking’, although synthesis has since become the more commonly used term. His speech remaking device consisted of a number of filters, each of which determined the energy in a particular frequency band. The bands were specified to have widths (at the 3dB level) of between 200 and 300 Hz. By splitting the speech spectrum into ten bands and determining the energy in each, the vocal tract transfer function was estimated. To synthesize speech, the estimate of the power in each of the fixed frequency bands was modulated by one of two source signals. Dudley’s sources consisted of a ‘hiss-like’ noise signal for unvoiced speech and a ‘buzz-like’ periodic source for voiced speech. Systems that code speech in this manner derive their names from the words VOice CODER and are called *vocoders*. The signif-

icance of Dudley's work is that it demonstrated that much of the unprocessed speech signal is redundant (Schroeder, 1966). Further developments in voice coding proceeded from this assumption.

Pioneering vocoder developments, of the type reported by Dudley (1939), were constructed with analog electrical components. However, contemporary speech coders are implemented via algorithms which reside in digital computers. The LPC algorithm is one such technique. It is a method of estimating parameters for a low-order filter that models the vocal tract and was first applied to speech signals by Itakura and Saito (1968) and Atal and Hanauer (1971).

### 2.5.2 Lip and glottal effects

This section examines the way in which the characteristics of the glottal pulse and lip radiation are accounted for in LPC modelling. The effects of glottal pulses and the lip radiation can be incorporated into the standard source filter model to create a voice production model (Markel and Gray, 1976, p6) as previously indicated in Fig. 2.2(a) and (b). The excitation impulses  $e(t)$  are filtered with the aim of making them correspond more accurately to actual glottal pulses. The input to the vocal filter is then the excitation signal  $e(t)$  convolved with the glottal shaping filter with impulse response  $g(t)$ . The output of the vocal tract filter is convolved with a lip radiation model having impulse response  $l(t)$  to produce a synthetic speech signal  $s(t)$ . Such a model is represented compactly in the  $Z$ -domain as

$$S(z) = E(z)G(z)V(z)L(z). \quad (2.31)$$

The interface between the vocal tract and free space is at the lips. It is important to realistically model the effect of the lips because it defines how the speech sound actually comes out of the vocal tract. The radiated pressure signal can be quite accurately approximated by a 1st order differentiation (Fant, 1973, p7) of the sound within the vocal tract (Flanagan, 1972, p36), i.e.,

$$L(z) = 1 - z^{-1}. \quad (2.32)$$

Section 2.2.1 mentions that the spectrum of the typical human glottal excitation falls off approximately as  $1/f^2$ . This spectral characteristic is conveniently modelled as a two-pole low pass filter with a cutoff frequency of approximately 100 Hz (Markel and Gray, 1976, p7). Such a filter is of the form

$$G(z) = \frac{1}{(1 - e^{-cT}z^{-1})^2}. \quad (2.33)$$

where  $c$  is the speed of sound in air and  $T$  is the sampling period. Since  $cT$  is much less than unity,  $G(z)$  can be thought of as a filter having a -12dB/octave slope. Combining  $L(z)$  and  $G(z)$ , and observing that one of the terms in the denominator of (2.33) is effectively cancelled by  $L(z)$  (since  $cT$  is generally much less than unity) gives a filter

$$L(z)G(z) \approx \frac{1}{P(z)} = \frac{1}{1 - uz^{-1}}, \quad (2.34)$$

where  $u$  is close to unity. To separate the contributions from  $L(z)$  and  $G(z)$  from  $V(z)$  it is useful to rewrite (2.31)

$$S(z)P(z) = V(z)E(z), \quad (2.35)$$

where  $P(z) \approx \frac{1}{L(z)G(z)}$ .  $P(z)$  is called a pre-emphasis filter because it is applied to  $S(z)$  to give  $V(z)$ , which is then analysed to find the vocal tract parameters. Normally  $u$  is set to the value of 0.95 which implies a gain of 6dB/octave for a sampling rate of 10 kHz.

### 2.5.3 Linear prediction

The basic idea in linear prediction is that it is possible to predict the next sample in a sequence from a linear combination of the previous  $P$  samples. This corresponds to an all-pole system whose output is a linear combination of previous outputs and the current input (Makhoul, 1975). The coefficients used to predict the next sample are called prediction coefficients and are denoted by  $a_i$ . The linear prediction model for a sequence  $s[n]$  is defined by (Markel and Gray, 1976, p10),

$$s[n] = \sum_{i=1}^P -a_i s[n-i] + e[n], \quad (2.36)$$

where  $e[n]$  is the *residual* or *prediction error*. The  $a_i$ s can also be thought of as the coefficients of an all-pole filter which models the transfer function of the vocal tract, as intimated in §2.2.2. The vocal tract, and therefore the  $a_i$ s, are considered to be invariant throughout a short speech segment. Typically, lengths of 10ms to 25ms are found to be suitable for estimating vocal tract filter parameters (Witten, 1982, p126).

The total power  $\alpha$  in the prediction error over a speech segment defined between  $n = 0$  and  $n = N - 1$  is

$$\alpha = \sum_{n=0}^{N-1} e[n]^2 = \sum_{n=0}^{N-1} \left( s[n] + \sum_{i=1}^P a_i s[n-i] \right)^2. \quad (2.37)$$

This error power is minimized by setting its derivative with respect to each  $a_i$  to zero, i.e.

$$\frac{\partial \alpha}{\partial a_j} = -2 \sum_{n=0}^{N-1} s[n-j] \left( s[n] + \sum_{i=1}^P a_i s[n-i] \right), \quad (2.38)$$

and setting  $\partial \alpha / \partial a_j = 0$  gives the set of equations

$$\sum_{n=0}^{N-1} s[n] s[n-j] = \sum_{i=1}^P a_i \sum_{n=0}^{N-1} s[n-j] s[n-i] \quad \text{for } j = 1, 2, \dots, P. \quad (2.39)$$

Assumptions about the signal at the edge of the speech segment are required before the summation of the  $s[n-j]s[n-i]$  can be evaluated (Rabiner and Schafer, 1978, §8). One approach is to assume that the summations in (2.39) runs from  $-\infty$  to  $+\infty$ , but that the signal is zero outside the range of  $0 \leq n \leq N - 1$ . These conditions reduce (2.39) to the *autocorrelation method* (Markel and Gray, 1976, p14), viz,

$$(1) \text{ Set } r_m = \sum_{n=0}^{N-1-m} s[n] s[n+m] \\ (2) \text{ Solve } \sum_{i=1}^P a_i r_{|i-j|} = -r_j \quad \text{for } j = 1, 2, \dots, P \quad (2.40)$$

or, in matrix form

$$\begin{aligned} a_1 r_0 + a_2 r_1 + a_3 r_2 + \dots a_P r_{P-1} &= -r_1 \\ a_1 r_1 + a_2 r_0 + a_3 r_1 + \dots a_P r_{P-2} &= -r_2 \\ a_1 r_2 + a_2 r_1 + a_3 r_0 + \dots a_P r_{P-3} &= -r_3 \\ &\dots = \dots \\ a_1 r_{P-1} + a_2 r_{P-2} + a_3 r_{P-3} + \dots a_P r_0 &= -r_P. \end{aligned} \quad (2.41)$$

The matrix formulation of the autocorrelation is symmetric and components along each diagonal are equal. These properties identify the autocorrelation matrix as a Toeplitz matrix (Rabiner and Schafer, 1978, p403). Toeplitz matrices can be inverted

efficiently using a recursive technique such as that proposed by Durbin (see §2.5.3.1) (Makhoul, 1975).

The autocorrelation defined by (2.40) stipulates a summation which runs over a signal segment consisting of  $N$  samples. This is the same as multiplying a signal that extends from  $-\infty$  to  $+\infty$  by a rectangular window which is  $N$  samples long. However, a rectangular window causes large prediction errors at the beginning of a speech segment because samples are estimated from other samples which are defined to be zero. A similar situation arises just past the end of the segment, where samples that are zero are predicted from other nonzero samples. These large prediction errors are avoided if the speech samples reduce smoothly to zero at the edge of the defined speech segment (Rabiner *et al.*, 1977; Chandra and Lin, 1974). This can be achieved by multiplying the speech segment by windows such as the Hamming or Blackman windows (Harris, 1978), which both tend smoothly to zero at the ends.

The second approach to the summation in (2.39) is to fix the limits over which the summation is evaluated:

$$(1) \text{ Set } \phi_{ij} = \sum_{n=0}^{N-1} s[n-i]s[n-j] \\ (2) \text{ Solve } \sum_{i=1}^P a_i \phi_{ij} = -\phi_{0j} \quad \text{for } j = 1, 2, \dots, P. \quad (2.42)$$

Equation (2.42) represents the *covariance method*. In contrast to the autocorrelation method, the speech segment no longer requires tapering at the ends since the summation used to calculate  $\phi_{ij}$  incorporates samples outside the interval  $0 \leq n \leq N-1$ . Parameters calculated by solving (2.42) are still coefficients for the filter  $1/A(z)$ , but differ from those calculated via the autocorrelation method.

Calculating prediction coefficients using either the autocorrelation method or the covariance method requires a matrix inversion. Theoretically, the autocorrelation matrix is guaranteed to be invertible, however, it can occasionally become ill-conditioned due to computational roundoff (Rabiner and Schafer, 1978, 418). The covariance matrix is not guaranteed to be invertible (Makhoul, 1975).

### 2.5.3.1 Solution of LPC equations

Several methods have been developed for implementing the autocorrelation formulation. The Durbin-Levinson method is the most efficient of these methods and is specified as follows (Makhoul, 1975),

$$E_0 = r_0 \quad (2.43)$$

$$k_i = -\frac{r_i + \sum_{j=1}^{i-1} a_j^{(i-1)} r_{i-j}}{E_{i-1}} \quad (2.44)$$

$$a_i^{(i)} = k_i \quad (2.45)$$

$$a_j^{(i)} = a_j^{(i-1)} + k_i a_{i-j}^{(i-1)}, \quad 1 \leq j \leq i-1 \quad (2.46)$$

$$E_i = (1 - k_i^2) E_{i-1}. \quad (2.47)$$

The recursion equations (2.44) - (2.47) are solved for  $i = 1, 2, \dots, P$ . Each step of the recursion increases the order of the prediction by one and recomputes the predictor coefficients for the increased model order. On completion of the recursion the predictor coefficients are defined by  $a_k^P$ ,  $1 \leq k \leq P$ . Note that the sets of  $k_i$  and  $a_i$  defined in the recursion are employed in the speaker identification experiments reported in §5.3.

The equations which define the covariance method can be stated in matrix notation as

$$\Phi \mathbf{A} = \Phi, \quad (2.48)$$

where  $\Phi$  is a positive definite symmetric matrix with  $(i,j)^{th}$  element  $\phi_{ij}$ , and  $\mathbf{A}$  and  $\boldsymbol{\phi}$  are column vectors with elements  $a_i$  and  $-\phi_{0j}$  respectively. Because  $\Phi$  is symmetric and positive definite, Cholesky decomposition can be invoked to solve for  $\mathbf{A}$  (Rabiner and Schafer, 1978, p407).

The computational requirements of the autocorrelation and covariance methods are dominated by the calculation of the autocorrelation coefficients  $r_i$  or the  $\phi_{ij}$  which both require  $O(PN)$  operations (multiplications). Solving for the  $a_i$ s requires  $O(P^2)$  operations for the autocorrelation method and  $O(P^3)$  for the covariance method (Rabiner and Schafer, 1978, p418).

#### 2.5.4 Lattice filtering

The lattice models constitute an important link between the LPC predictor coefficients and the vocal tract acoustic model described in §2.3.2. The coefficients  $k_i$  calculated as intermediate results in the Durbin-Levinson recursive procedure correspond to the reflection coefficients of the lattice filter (see §2.3.3), as derived from the vocal tract tube model (Rabiner and Schafer, 1978, p415). LPC coefficients calculated using the autocorrelation method are therefore directly related to the vocal tract tube model.

Section 2.3.3 describes the relationship between the vocal tract model and the lattice filter. In this section the solution for the elements of the lattice filter is shown to be based on the same relationship as the autocorrelation method of LPC. Forward and backward prediction errors are defined in terms of the speech signal and then in terms of each other. Methods for calculating reflection coefficients that minimize both of these prediction errors are then presented.

The lattice filter can be derived as a direct consequence of the Durbin-Levinson algorithm outlined in the previous section. As the Durbin-Levinson algorithm recurses, the order of the prediction filter increases. The coefficients of this prediction filter can be configured as an  $i^{th}$  order inverse filter,

$$A^{(i)}(z) = 1 - \sum_{k=1}^i a_k^{(i)} z^{-k}, \quad (2.49)$$

where  $i$  is the iteration number of the Durbin-Levinson algorithm and  $a_k$  is the  $k^{th}$  filter coefficient. The filter is applied to a segment of speech  $s_l[n] = s[l+n]w[n]$ , where  $w[n]$  is a window and the speech segment  $s_l[n]$  begins at sample number  $l$ , resulting in a prediction error  $y_l^{+(i)}[n]$  defined by

$$y_l^{+(i)}[n] = s_l[n] - \sum_{k=1}^i a_k^{(i)} s_l[n-k]. \quad (2.50)$$

For clarity the subscript  $l$  is discarded, so the prediction error then becomes  $y^{+(i)}[n]$ . This is called the *forward prediction error* since the speech sample  $s[n]$  is predicted from the preceding  $i$  samples. The Z-transform of (2.50) is

$$Y^{+(i)}(z) = A^{(i)}(z)S(z). \quad (2.51)$$

The final prediction error of the LPC filter is obtained from (2.51) by allowing the recursion to run to  $P$ .

A *backward prediction error* is defined as

$$y^{-(i)}[n] = s[n-i] - \sum_{k=1}^i a_k^{(i)} s[n+k-i], \quad (2.52)$$

where  $s[n-i]$  is predicted from the  $i$  input samples that follow  $s[n-i]$ . By substituting (2.46) into (2.49), and utilizing the  $Z$ -transform of (2.52), the forward prediction error sequence can be expressed as

$$y^{+(i)}[n] = y^{+(i-1)}[n] + \mu_i y^{-(i-1)}[n-1], \quad (2.53)$$

and similarly, the backward prediction error becomes

$$y^{-(i)}[n] = y^{-(i-1)}[n-1] + \mu_i y^{+(i-1)}[n], \quad (2.54)$$

where the  $\mu_i$  are the reflection coefficients. Note that (2.53) and (2.54), which are derived from the Durbin-Levinson algorithm, correspond to the inverse  $Z$ -transform of (2.27), derived from the vocal tract tube model (Rabiner and Schafer, 1978). They also describe exactly the lattice filter derived from the vocal tract tube model and depicted in Fig. 2.5.

In order to solve for the reflection coefficients using a filter of the lattice form it is necessary to formulate an equation that defines reflection coefficients minimizing both the forward and backward prediction errors. One approach is to use the PARCOR equation defined by (2.28). An alternative approach minimises the sum of the squared forward-prediction errors and the squared backward prediction errors. This is the so-called Burg method and involves recursive computation of (Barnwell, 1980),

$$\mu_i^B = \frac{2 \sum_{n=0}^{N-1} y^{+(i-1)}[n] y^{-(i-1)}[n-1]}{\sum_{n=0}^{N-1} (y^{-(i-1)}[n-1])^2 + \sum_{n=0}^{N-1} (y^{+(i-1)}[n])^2}. \quad (2.55)$$

Burg's method and the Durbin-Levinson method of LPC have different windowing requirements. A window is required in the autocorrelation method to limit the range of the summation used to compute the autocorrelation coefficients. However, windowing is unnecessary for the Burg lattice formulation (Barnwell, 1980) since the autocorrelation is not one of the inputs to the Burg algorithm. Barnwell (1980) compares how well the speech spectrum is modelled by coefficients calculated using the unwindowed Burg method and the windowed autocorrelation method. He finds that the Burg technique consistently gives better spectral estimates, where the criterion is the quality of synthetic speech resynthesized from the estimates. In the listening tests it was judged that the Burg technique maintained a higher synthetic speech quality when the segment length was reduced. The main form of spectral distortion is that the spectral peaks are broadened when short windows are used with the autocorrelation method (Barnwell, 1980).

Although the lattice techniques are based on the same error criterion as the Durbin-Levinson solution of the autocorrelation matrix, they are not globally optimal. The error is minimized stage by stage in the lattice filter and updated for every new sample, whereas Durbin-Levinson's method uses the autocorrelation of the complete speech segment. A lattice method that updates the reflection coefficients using (2.28), and the Durbin-Levinson autocorrelation method, will produce the same coefficients for a stationary signal (in the sense defined in §2.7.1.1), but because speech signals are not strictly stationary, even for 10 ms, the two methods generally give different results.

In this thesis the autocorrelation technique is used to compute the  $\mu_i$  since it is more computationally efficient than the Burg method, which requires  $O(5NP)$  operations (Rabiner and Schafer, 1978, p418) to compute one set of  $P$  reflection coefficients (for  $N$  samples).

## 2.6 SPECTRAL ESTIMATION

Many of the speech analysis techniques invoked in this thesis use spectral estimation in some form. It is important, therefore, to review the limitations of various spectral estimation techniques in terms of their spectral resolution, accuracy and other relevant short-comings. This section discusses several different methods of spectral analysis and their application to speech signals. Section 2.6.1 describes certain limitations of the FFT for estimating the spectral content of a signal, while §2.6.2 defines the way in which the LPC filter models the spectrum of a speech signal. The cepstrum of a speech signal and its properties are discussed in §2.6.3.

### 2.6.1 Fast Fourier Transform

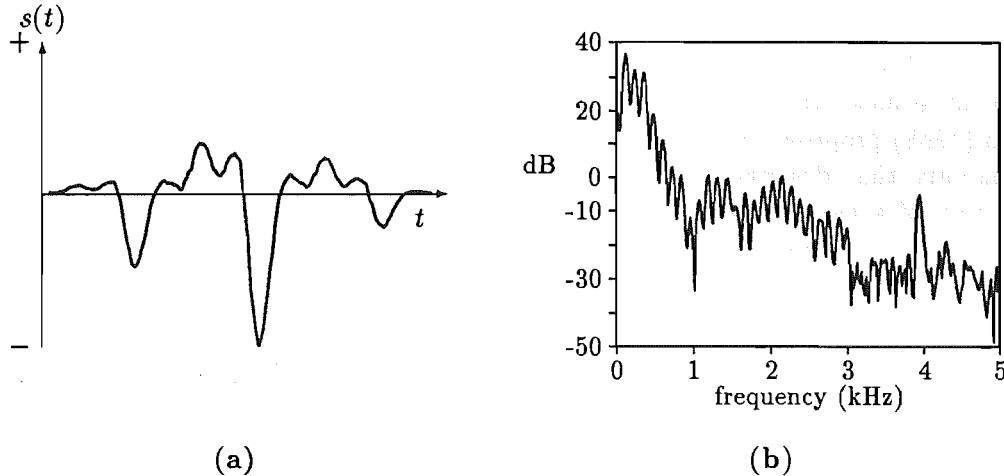
The FFT, as introduced in §2.1.1, is a computationally efficient technique that produces reasonable spectral estimates for diverse classes of sampled signals. However, spectral estimates obtained via the FFT are subject to unavoidable limitations, which means that care must be taken when interpreting spectra.

The most prominent limitation of the FFT is the frequency resolution which, in Hertz, is the reciprocal of the duration of the sampled data (Kay and Marple, 1981). The only way that the frequency resolution can be improved is to take a longer sequence of sampled data. However, a longer data sequence precludes accurate identification of short-term changes in the spectral content of the signal with time. This trade-off between frequency resolution and time resolution of the time-varying spectrum must be accommodated when performing spectral analysis.

Another limitation of the FFT is that it operates on a finite number of samples. Conceptually the infinite duration digitized signal is multiplied by a window which excludes all samples outside a certain interval. The choice of windowing function is critical since the calculated spectrum is the ideal spectrum convolved with the spectrum of the windowing function (Fallside, 1985). The power of a signal at any given frequency, will be spread by the convolution into adjacent frequency regions. This phenomenon is called *leakage* since the power in one spectral component 'leaks' into adjacent spectral components. Judicious choice of a window function significantly reduces the effect of leakage (Harris, 1978), but at the expense of decreasing the spectral resolution due to the effective reduction in segment length and the associated broadening of the main lobe of the window's spectrum.

It is important to remember that noise present in a signal is also present in its spectrum, although the effect of the noise can be reduced if the spectra from several speech segments are averaged together. However, it is only valid to do this when the spectra to be averaged together have the same spectrum, so this method has limited application to speech.

Notwithstanding limitations in spectral resolution and the leakage due to windowing, useful information about formant positions can be extracted from the speech spectrum. Fig. 2.8(a) and (b) respectively show a windowed segment of voiced speech for the vowel sound /o/ and the log magnitude of its spectrum (computed via the FFT). The approximate position of the formants can be reasonably estimated from Fig. 2.8(b), but the exact position is obscured by a strong harmonic structure superimposed on top of the underlying formant structure. These harmonics are due to the periodic glottal excitation which, in the frequency domain, gives rise to components spaced at the pitch frequency. By the convolution theorem, the speech spectrum is composed of the glottal shaping, vocal tract, and pitch spectra multiplied together, hence the pitch  $f_p$  is observed in the resulting spectrum.



**Figure 2.8.** An example of the effect of voicing on the speech spectrum: (a) a windowed segment of the voiced sound /o/, (b) the spectrum of the speech depicted in (a).

### 2.6.2 Linear predictive coding

In this section the nature of the relationship between the spectrum of a segment of speech and the LPC prediction filter is defined. The approach is via the maximum likelihood method proposed by Itakura and Saito (1968).

The LPC filter can be considered to approximate the spectrum of the speech that it is modelling. Here the Z-transform of a speech segment is denoted by  $X(e^{j\theta})$ . The energy in the speech spectrum is matched to the energy in the model spectrum by choosing an appropriate value for  $\sigma$  such that

$$\int_{-\pi}^{\pi} |X(e^{j\theta})|^2 \frac{d\theta}{2\pi} = \int_{-\pi}^{\pi} \frac{\sigma^2}{|A(e^{j\theta})|^2} \frac{d\theta}{2\pi}, \quad (2.56)$$

where  $\sigma^2 = \alpha$ , the energy in the prediction error as defined by (2.37). It is useful at this point to state that on a log magnitude scale, both  $A(e^{j\theta})$  and  $1/A(e^{j\theta})$  have zero mean, provided all the zeroes of  $A(e^{j\theta})$  lie inside the unit circle (Markel and Gray, 1976, p130). It follows that the average value of the log spectrum of the model filter is

$$\int_{-\pi}^{\pi} \ln \left| \frac{\sigma}{A(e^{j\theta})} \right|^2 \frac{d\theta}{2\pi} = \ln(\sigma^2). \quad (2.57)$$

As the number of coefficients in the prediction filter  $A(z)$  is increased, the prediction error  $\alpha$  decreases. The autocorrelation sequences of  $X(e^{j\theta})$  and  $\sigma/A(e^{j\theta})$  are therefore identical when the number of prediction coefficients approaches infinity. Replacing  $\sigma/A(e^{j\theta})$  with  $X(e^{j\theta})$  in (2.57) gives

$$\alpha_\infty = \exp \int_{-\pi}^{\pi} \ln |X(e^{j\theta})|^2 \frac{d\theta}{2\pi}, \quad (2.58)$$

where  $\alpha_\infty$  represents the minimum squared error possible for predicting  $x[n]$ . In expressing the accuracy of the model, the mean square prediction error  $\alpha$  should therefore be compared with  $\alpha_\infty$ , since the minimum possible prediction error is not actually zero. It is not immediately obvious why  $\alpha_\infty$  should be non-zero. One explanation for this is that the filter  $\sigma/A(e^{j\theta})$  still requires an excitation in order to match the spectrum of  $X(e^{j\theta})$ . Makhoul (1975) reasons that the impulse response of an all-pole filter is

perfectly predictable, except for its initial value. It is the energy of this initial value that is represented by  $\alpha_\infty$ , giving rise to the name *one-step prediction error*.

The relationship between the LPC and speech spectra can be expressed by defining a ‘distance measure’ between a speech spectrum and a model spectrum. Itakura and Saito (1968) propose a statistical formulation of such a measure. They maximize the probability that describes how ‘likely’ the spectrum of a speech signal is, given the spectrum of a model filter. This method, called the *maximum likelihood method*, can be expressed as a minimization of the following integral,

$$I = \int_{-\pi}^{\pi} [e^{V(\theta)} - V(\theta) - 1] \frac{d\theta}{2\pi}, \quad (2.59)$$

where

$$V(\theta) = \ln |X(e^{j\theta})|^2 - \ln \left| \frac{\sigma}{A(e^{j\theta})} \right|^2 \quad (2.60)$$

is the difference between the log spectra of the speech and the model. The measure defined by  $I$  is sometimes called the Itakura-Saito (or IS) distortion measure. Expanding (2.59) by substituting  $V(\theta)$  gives

$$\begin{aligned} I &= \frac{1}{\sigma^2} \int_{-\pi}^{\pi} |X(e^{j\theta})A(e^{j\theta})|^2 \frac{d\theta}{2\pi} + \int_{-\pi}^{\pi} \ln \left| \frac{\sigma}{A(e^{j\theta})} \right|^2 \frac{d\theta}{2\pi} - \\ &\quad \int_{-\pi}^{\pi} \ln |X(e^{j\theta})|^2 \frac{d\theta}{2\pi} - 1, \end{aligned} \quad (2.61)$$

which, by recalling that  $E(e^{j\theta}) = X(e^{j\theta})A(e^{j\theta})$ , and utilizing (2.58) and (2.57), results in

$$I = \frac{1}{\sigma^2} \int_{-\pi}^{\pi} |E(e^{j\theta})|^2 \frac{d\theta}{2\pi} + \ln(\sigma^2) - \ln(\alpha_\infty) - 1. \quad (2.62)$$

Minimizing  $I$  is the same as minimizing the average energy in  $E(e^{j\theta})$  since the last three terms in (2.62) are independent of the prediction coefficients (Markel and Gray, 1976, p135).

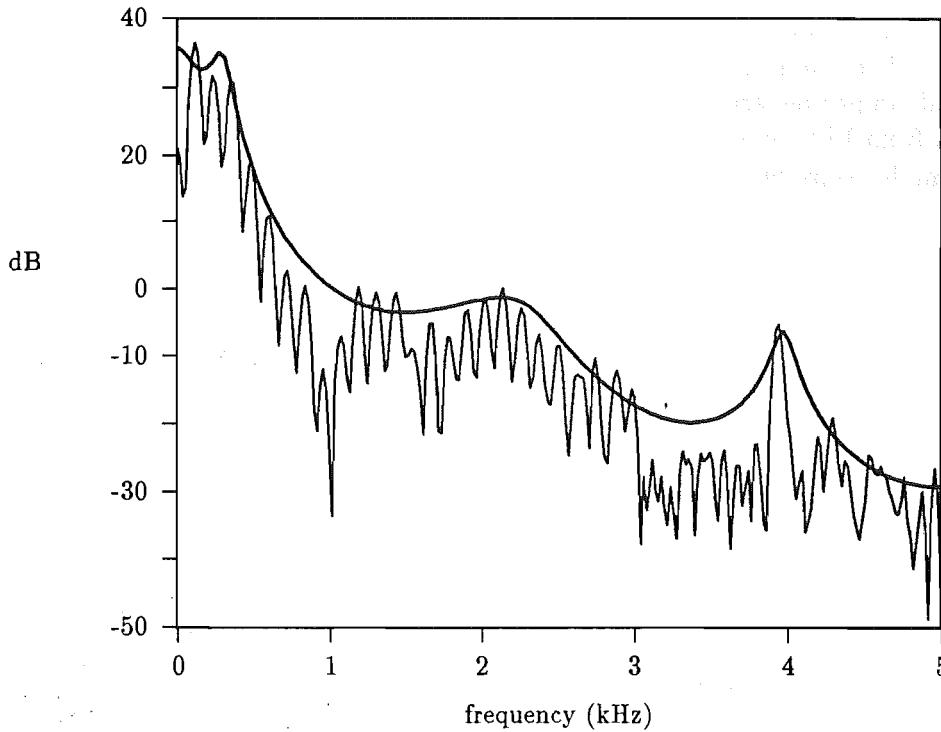
The energy in  $E(e^{j\theta})$  represents the prediction error power, which is the criterion minimized in the autocorrelation method. Therefore, the autocorrelation method of LPC can be interpreted as minimizing a non-uniformly weighted spectral error between the model spectrum and the signal spectrum. Fig. 2.9 shows the spectrum of a segment of speech and also the spectrum of the LPC filter.

Because of the non-linear formulation of  $I$ , peaks and valleys in the speech spectrum are treated differently. When  $V(\theta) \gg 1$ , the integrand of (2.59) approximates  $e^{V(\theta)}$ . Very large contributions to the total error are therefore produced when the log model spectrum is much less than the log speech spectrum. However, when  $V(\theta) \ll 1$ , the integrand in (2.59) is approximately  $V(\theta)$ . The log model spectrum is therefore only penalized ‘lightly’ wherever it is greater than the log spectrum of the speech, but is penalized more ‘heavily’ where it is less than the log speech spectrum. This property implies that the poles of the speech spectrum are modelled more accurately than the zeros.

### 2.6.2.1 Formants

The positions of the formants are important in characterizing the phonetic identity of a sound (Fant, 1973). Consequently, many techniques have been developed for determining formant positions.

This section introduces the use of LPC techniques to estimate the formant positions. The use of LPCs to estimate formant positions has several advantages. The amount



**Figure 2.9.** An example of the spectrum determined from LPCs. The signal is the same as that utilized in Fig. 2.8

of data required to represent the speech spectrum is reduced significantly, while maintaining an accurate representation of the positions of peaks in the speech spectrum (§2.6.2). Also, as illustrated in Fig. 2.9, the ‘pitch ripple’ is removed from the speech spectrum, enabling formant positions to be identified more readily.

Once the LPC filter  $1/A(e^{j\theta})$  is determined, the roots of  $A(e^{j\theta})$  can be computed and utilized to determine the formants and their associated bandwidths. In an LPC filter of order  $P$ , the zeros of  $A(e^{j\theta})$  occur in conjugate pairs which are denoted here as  $z_1, z_1^*, z_2, z_2^*, \dots, z_{P/2}, z_{P/2}^*$  (for even  $P$ ). The formant frequency and two-sided bandwidth associated with the pole  $z_k$  is given by (Atal and Hanauer, 1971)

$$F_k = \frac{1}{2\pi T} \operatorname{Im} \{ \ln z_k \} \quad (2.63)$$

and

$$B_k = \frac{1}{\pi T} \operatorname{Re} \left\{ \frac{1}{\ln z_k} \right\}. \quad (2.64)$$

Pole positions can be computed using a root solving program, however, this is a computationally expensive procedure. An alternative approach, outlined by Markel and Gray (1976, p167), evaluates  $1/A(e^{j\theta})$  at a number of discrete points around the unit circle. Parabolic interpolation together with a peak picking algorithm is then employed to locate the formant positions and their bandwidths.

The positions of poles located deep within the unit circle can be refined by evaluating  $1/A(e^{j\theta})$  on concentric circles within the unit circle (Markel and Gray, 1976; Duncan and Jack, 1988). This technique is called ‘off-axis’ spectral estimation or pole-focussing and allows the positions of broad-bandwidth poles (those far from the unit circle) to be determined more accurately.

### 2.6.3 Cepstral analysis

Cepstral analysis is a technique for obtaining coefficients that describe the log spectrum of a signal. The complex cepstrum of a signal is defined in this section, and some of its useful properties are discussed. In addition, a method for calculating cepstral coefficients from LPC prediction coefficients is presented.

The complex cepstrum  $\hat{x}[n]$  is defined to be the inverse Fourier transform of the log of the spectrum  $X(e^{j\theta})$  of a signal, viz

$$\hat{x}[n] = \int_{-\pi}^{\pi} [\ln |X(e^{j\theta})| + j \arg[X(e^{j\theta})]] e^{j\theta n} \frac{d\theta}{2\pi}, \quad (2.65)$$

where both the magnitude and phase of  $X(e^{j\theta})$  are incorporated into the calculation of  $\hat{x}[n]$ . Oppenheim and Schafer (1975, p503) show that when  $X(e^{j\theta})$  has no poles or zeros outside the unit circle,  $\hat{x}[n] = 0$  for all  $n < 0$ . A signal which satisfies this criterion is called a minimum phase sequence. However, if  $X(e^{j\theta})$  has all its poles and zeros outside the unit circle,  $\hat{x}[n] = 0$  for all  $n > 0$ , which corresponds to a maximum phase sequence.

Cepstral coefficients can be computed from the spectral magnitude and here they are denoted  $c[n]$ , to distinguish them from the complex cepstrum  $\hat{x}[n]$  computed from the complex spectrum. The  $c[n]$  are called the cepstrum, with the term ‘complex’ being omitted because only the magnitude of the spectrum is considered. The cepstrum is defined by

$$c[n] = \int_{-\pi}^{\pi} \ln |X(e^{j\theta})| e^{j\theta n} \frac{d\theta}{2\pi}. \quad (2.66)$$

Since the cepstral domain represents the log spectrum, additions in the cepstral domain transform to multiplications in the frequency domain and convolutions in the time domain. Deconvolution operations can therefore be implemented in the cepstral domain by subtraction. It is this feature that makes cepstral analysis useful for the purpose of separating different components of the speech signal via cepstral analysis.

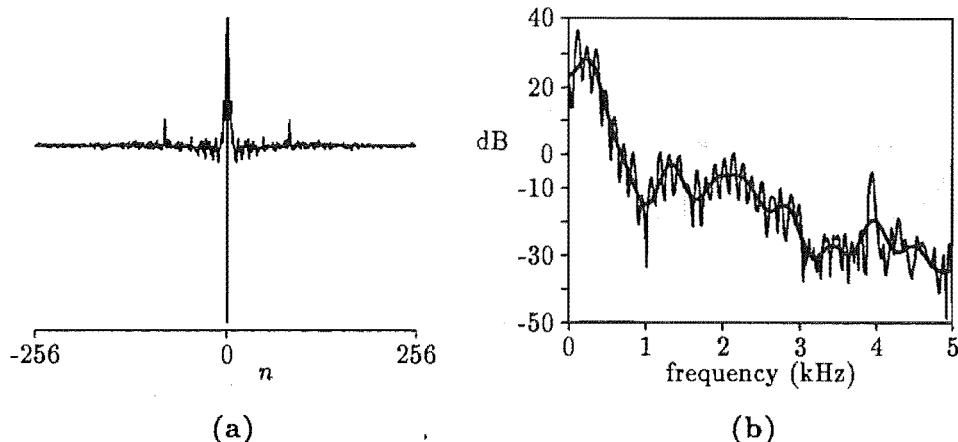
Ignoring the effect of the lips, a segment of speech can be modelled as

$$s[n] = (e[n] \odot g[n] \odot v[n])w[n], \quad (2.67)$$

where  $e[n]$ ,  $g[n]$  and  $v[n]$  represent the excitation (or pitch), glottal pulse model and vocal tract model respectively. A portion of the speech signal is defined to be non-zero by a window  $w[n]$ . The vocal tract response can be considered a minimum phase sequence because  $V(e^{j\theta})$  is a stable all-pole filter with all its poles inside the unit circle. However, the glottal excitation  $G(e^{j\theta})$  is modelled by a time-limited pulse whose Z-transform can be represented by zeros that occur both inside and outside the unit circle making it nonminimum phase and causing  $\hat{x}[n] \neq 0$  for  $n < 0$  (Oppenheim and Schafer, 1968).

The complex cepstrum can be invoked to separate the vocal tract response from the pitch synchronous contributions of the glottal pulse. In the cepstral domain, contributions from the pitch ( $e[n]$ ) are dominant for values of  $n$  round the pitch period and greater, while coefficients corresponding to the vocal tract and glottal pulse are dominant for  $n$  less than the pitch period (Oppenheim and Schafer, 1968). Furthermore the nonminimum phase contributions from the glottal excitation can be removed by setting  $\hat{x}[n] = 0$  for  $n < 0$ . These properties can be invoked to calculate a ‘smoothed’ spectral estimate of the vocal tract response.

Fig. 2.10(a) shows the cepstrum of a segment of speech. The peak at  $n_p = 85$  can be attributed to the effect of the pitch on the log spectrum. By setting  $c[n] = 0$  for



**Figure 2.10.** An example of cepstral analysis of a segment of speech: (a) the cepstrum of the voiced speech in Fig. 2.8 and (b) the cepstrally ‘smoothed’ spectrum.

values of  $n$  from  $n_p - 1$  to  $n = \infty$ , the pitch information can be removed and the smooth spectrum shown in Fig. 2.10(b) is obtained.

Both (2.65) and (2.66) utilize double Fourier transforms to calculate the cepstral coefficients from the speech signal. An alternative method is to calculate the cepstral coefficients from the spectral representation of the speech signal provided by the LPC prediction filter  $\sigma/A(e^{j\theta})$ . If all the poles of  $\sigma/A(e^{j\theta})$  are within the unit circle,  $\ln(|\sigma/A(e^{j\theta})|^2)$  can be expanded as a power series as (Markel and Gray, 1976, 230)

$$\ln(|\sigma/A(e^{j\theta})|^2) = \ln(\sigma^2) + \sum_{n=1}^{\infty} c[n]z^{-n}. \quad (2.68)$$

Note that  $c[0] = \ln(\sigma^2)$ , which is exactly the property defined by (2.57). Higher order cepstral coefficients are obtained by the following recursive relationships (Atal, 1974),

$$c_1 = a_1 \\ c_n = \sum_{k=1}^{p-1} (1 - k/n)a_k c_{n-k} + a_n, \quad \text{for } 1 < n \leq p \quad (2.69)$$

and

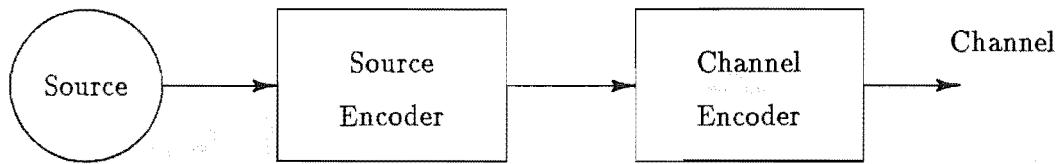
$$c_n = \sum_{k=1}^{p-1} (1 - k/n)a_k c_{n-k}, \quad \text{for } n > p$$

where  $p$  is the number of poles.

The cepstral coefficients calculated from prediction coefficients are different from those obtained by invoking Fourier transforms because the prediction coefficients constitute an approximation of the speech spectrum (as outlined in §2.6.2). However, cepstral coefficients can be computed very efficiently using this approach.

## 2.7 VECTOR QUANTIZATION

Vector quantization (VQ) is a compression technique which can be used to reduce the data rate required for speech transmission (Juang *et al.*, 1982; Makhoul *et al.*, 1985). It became a practical speech compression option after the development of LPC coding, in the early 70s, and the later development of a vector quantization training by Linde *et al.* (1980). It has been extensively applied in the low bit rate speech coding field



**Figure 2.11.** Information transmission across a channel.

where transmission rates as low as 800 bits/s have been achieved (Wong *et al.*, 1982). VQ is also often used in speaker recognition (Soong *et al.*, 1985; Soong *et al.*, 1987; Burton, 1987) and speech recognition systems (Rabiner *et al.*, 1983; Burton *et al.*, 1985), where VQ is utilized within a pattern classifier rather than a speech coder/decoder. It has even been applied to the problem of quantizing the range of vocal tract shapes that span the articulatory space (Larar *et al.*, 1988).

Section 2.7.1 introduces a general speech compression system and its constituent components. Section 2.7.2 highlights the advantages of vector quantization by way of two-dimensional VQ examples and briefly describes the measures that are used to characterize the performance of a VQ scheme. Many different distortion measures can be used in VQ systems and some of those commonly used for quantizing speech are defined in §2.7.3. Section 2.7.4 outlines methods for calculating the codevectors which define a VQ system and discusses options for storing and accessing them. These techniques are utilized in the speaker recognition experiments reported in Chapter 5 to construct templates of individual's voices.

### 2.7.1 Introduction

The basis for data reduction using VQ is best explained by assuming that speech can be represented as information emanating from a source with defined characteristics. The information source model is described in §2.7.1.1 and is restated in §2.7.1.3 in terms of a speech signal.

#### 2.7.1.1 The information source

Information theory provides the terminology for describing an information source and the amount of distortion introduced when the information is coded in some manner. In general, information emanating from a source is encoded in the manner depicted in Fig. 2.11. The source encoding stage shown in Fig. 2.11 introduces distortion of the original information. For example, if speech is considered to be a continuous amplitude pressure waveform, the process of digitizing such a waveform will introduce distortion. In speech coding the challenge is to design encoders that encode the source information efficiently and only distort the original information in an acceptably small way.

To begin with, it is convenient to assume that the source is continuous and can take on any value within a defined range. The probability of the source producing a particular value is defined by the source probability density function (pdf). In practice a source output can never be measured in a completely continuous fashion, but reasonable estimates of the continuous case can be approximated if the resolution of the quantizer is high. For modelling purposes a useful model for the source pdf is the Gaussian, since it is well suited to mathematical manipulation and adequately approximates the pdf of

many systems. The Gaussian pdf of a source with zero mean is defined by,

$$p(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-x^2/2\sigma^2}, \quad (2.70)$$

where  $\sigma^2$  is the variance of the source and  $p(x)$  is the probability of observing  $x$ .

Instead of a continuously varying output, some sources have a discrete output and produce symbols from a symbol space  $\{s_1, s_2, \dots, s_K\}$ , where a symbol  $s_k$  occurs with probability  $P_s(s_k)$  with the subscript  $s$  indicating that the probability  $P_s$  applies to members of the sample space  $\{s_k\}$ . Such a source that produces discrete symbols is called a discrete source (Gallager, 1968, p71). The expected value  $E\{s_k\}$  of the source output is a weighted average over all the symbols. It is defined as

$$E\{s_k\} = \sum_{k=1}^K s_k P_s(s_k). \quad (2.71)$$

The underlying law governing the generation of symbols and the complete set of all possible symbols is called the *process* (Robinson, 1980, p163). In many practical situations it is not possible to specify an exact model for the process that generates the observed output, but nevertheless it is often feasible to construct models that seem to match the observed behaviour quite closely. An example of this is speech modelling, where the source output is the result of a complicated physical process that is only approximated by a source model.

Although the probability distribution defines the expected distribution of symbols, it does not describe the process that generates those symbols. Robinson (1980, p163) states that “a process is termed *deterministic* if it does not contain any features of randomness; otherwise it is termed *stochastic*”. He also points out it is acceptable to use the term *random process* interchangeably with stochastic process. A random process whose statistical properties are invariant with time is called a *stationary* random process (Gallager, 1968, p163). If a signal is produced by a stationary random process, one segment of the signal, recorded at a particular time, will have essentially the same statistics as another segment of the same signal observed at any other time-period. Additionally, if the long time average over any selected subset of the source output is equal to the ensemble average of an infinite sequence of source outputs, the process is said to be *ergodic* (Gallager, 1968, p59).

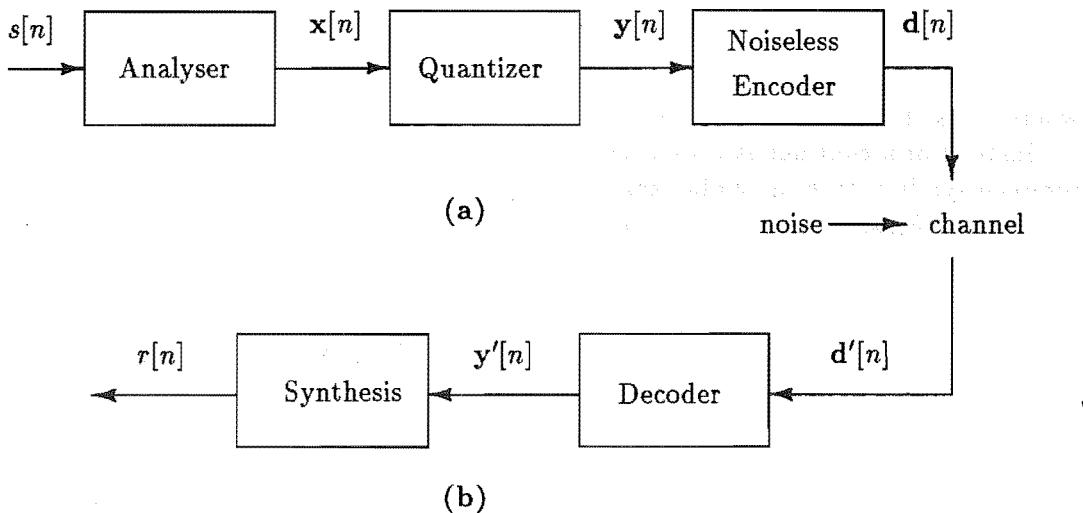
One particularly useful source model is the *discrete memoryless source*, where each symbol is statistically independent (Gallager, 1968, p38). Independence implies that there is no memory in the model, so each successive output is independent of the previous outputs. This type of source is particularly useful for testing vector quantization algorithms since the source output is straightforward to generate.

### 2.7.1.2 Preliminary notation

Much of the terminology introduced here relates to the speech transmission system depicted in Fig. 2.12. The speech signal  $s[n]$  is analysed to produce an unquantized,  $N$ -dimensional vector  $\mathbf{x} = [x_1 \ x_2 \ \dots \ x_N]^T$  where the vector components  $x_i$  are real-valued, continuous amplitude variables. The VQ task involves mapping the vector  $\mathbf{x}$  onto another real-valued, discrete-amplitude,  $N$ -dimensional vector  $\mathbf{y}$ , i.e.

$$\mathbf{y} = q(\mathbf{x}), \quad (2.72)$$

where  $q(\cdot)$  is the quantization operator and the vector  $\mathbf{y}$  is called the output vector. The finite set of values for  $\mathbf{y}$  is called the *codebook* and consists of  $\mathbf{Y} = \{\mathbf{y}_i; 1 \leq i \leq L\}$  where  $L$  is the total number of vectors in the codebook and each  $\mathbf{y}_i$  is a *codevector*.



**Figure 2.12.** A data compression system for speech coding: (a) speech is analysed and transmitted over a channel, (b) the received sequence of bits is decoded to reconstruct the speech signal.

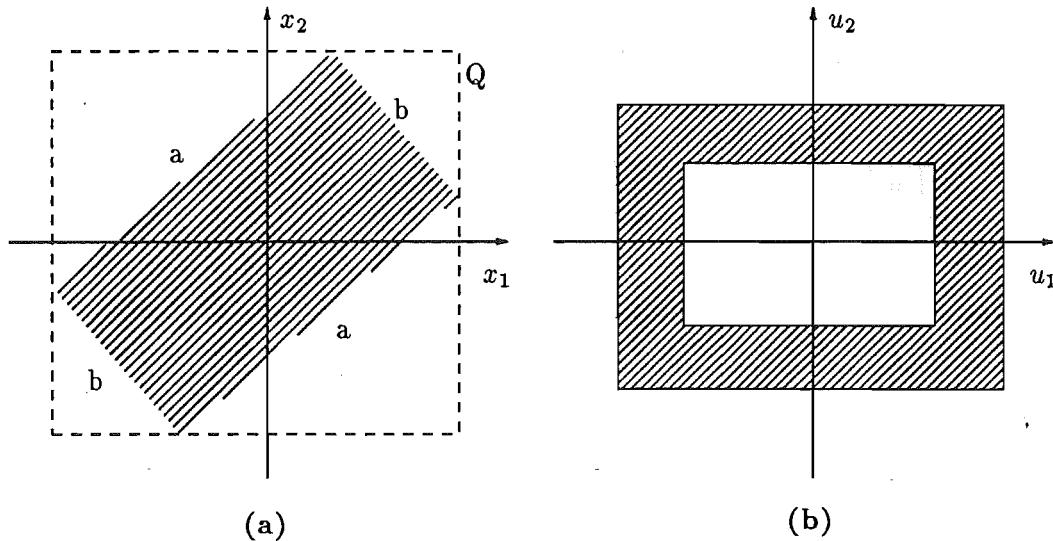
To design a VQ codebook, the  $N$ -dimensional space of the vector  $\mathbf{x}$  is partitioned into  $L$  regions, or cells, each denoted  $C_i$ . The quantizer assigns the code vector  $\mathbf{y}_i$  if  $\mathbf{x}$  lies in the region  $C_i$ , i.e.

$$q(\mathbf{x}) = \mathbf{y}_i \text{ if } \mathbf{x} \in C_i. \quad (2.73)$$

Determining which cell,  $C_i$ ,  $\mathbf{x}$  belongs to is called *partitioning* or *clustering* (Makhoul *et al.*, 1985). The group of vectors belonging to each of the cells  $\mathcal{C} = \{C_i; i = 1, \dots, L\}$  is called a *partition*. Provided a distance measure is specified, the cell shapes and the positions of the code vectors  $\mathbf{y}_i$  contain the same information because each can be uniquely determined from the other. To determine which cell contains a particular vector  $\mathbf{x}$  requires ‘distances’ to be calculated between the  $\mathbf{x}$  and each of the  $L$  code-vectors in the  $N$ -dimensional space. The quotation marks around the word distance are included to imply that often it is not the distance, in the Euclidean sense of the word, that is employed. For speech signals, perceptually relevant distortion measures are often invoked (Gray and Markel, 1976).

### 2.7.1.3 A quantizer for speech transmission

The basic components of a speech compression system are shown in Fig. 2.12. In the first stage of compression the input signal  $s[n]$  is analysed to produce a vector of unquantized parameters  $\mathbf{x}[n]$ . An example of the type of parameters produced at the output of the analyser are LPCs, calculated in the manner outlined in §2.5.3. The unquantized vector is then quantized into a vector  $\mathbf{y}[n]$  which is chosen from a *codebook* containing all the allowable *codevectors*. The codevector is in turn encoded as a sequence of bits  $d[n]$  before transmission on a channel. Errors in the channel are accounted for by defining the channel output to be  $d'[n]$  which will differ from  $d[n]$  whenever a channel error occurs. The estimate of the quantized vector  $\mathbf{y}'[n]$  is then decoded from  $d'[n]$  and applied to a synthesis stage which implements the opposite of the analysis process.



**Figure 2.13.** The joint pdfs of the two random variables where shading indicates a uniform non-zero probability: (a)  $p(x_1, x_2)$  where  $x_1$  and  $x_2$  are correlated, (b)  $p(u_1, u_2)$  where  $u_1$  and  $u_2$  are uncorrelated but depend upon each other nonlinearly.

### 2.7.2 Advantages of vector quantization

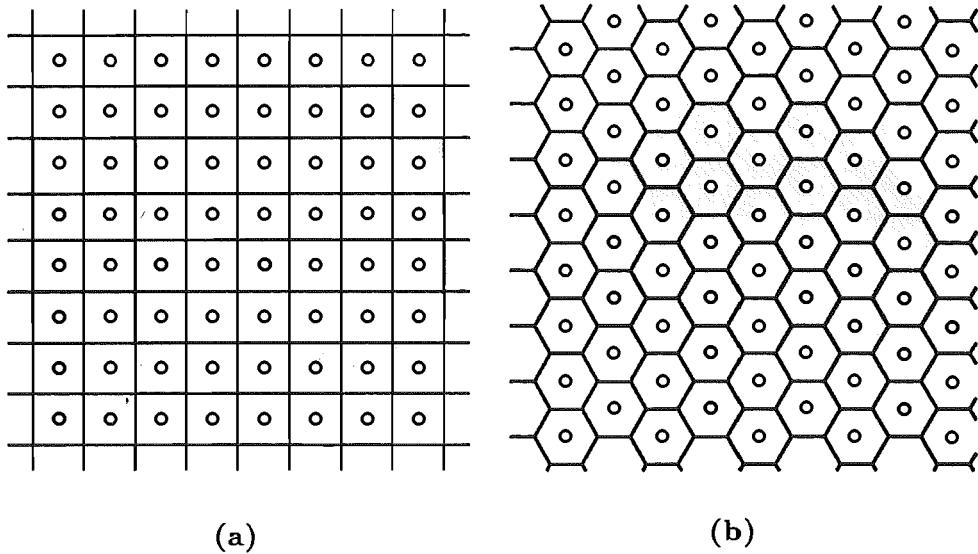
The advantage of VQ over scalar quantization is described here in terms of: (a) vector properties that can be exploited to increase coding efficiency and (b) the theoretical performance bounds of VQ.

#### 2.7.2.1 Vector properties

This section briefly describes four examples which illustrate properties of vector components that can be used to effect gains in coding efficiency. Makhoul *et al.* (1985) state that these four properties “when utilized appropriately result in optimum codebook design”. The properties are: linear dependency, nonlinear dependency, cell shape and the pdf (see also Lookabaugh and Gray (1989))

The first property, linear dependency, occurs when two or more vector components are correlated or dependent. Fig. 2.13(a) shows the joint pdf of two correlated random variables  $x_1$  and  $x_2$ . If both  $x_1$  and  $x_2$  are quantized using uniform scalar quantization, the complete area  $Q$  is divided up into square cells of equal size and spacing. However, certain combinations of  $x_1, x_2$  do not occur because the pdf is zero within much of  $Q$ . It is inefficient, in terms of coding, to assign cells to regions of zero probability. Linear dependency can be removed by redefining the axes so that each axis is parallel to either the boundary labelled 'a' or the boundary labelled 'b'. Maintaining a defined level of distortion and assuming the joint pdf has sides  $a = 2b$ , the transformed vector can be transmitted with a saving of 1.17 bits per vector (Makhoul *et al.*, 1985). Although the actual data rate reduction in any specific case depends on the particular joint pdfs involved, the principle that the bit rate can be reduced by ensuring the vector components are uncorrelated holds for all distributions.

Rotating the coordinate system removes all linear dependencies(or correlations) but nonlinear dependencies will remain after rotation. Fig. 2.13(b) illustrates such a nonlinear dependency for the two vector components  $u_1$  and  $u_2$ . Assigning codebook cells to the zero probability area within the probability distribution wastes bits, so an



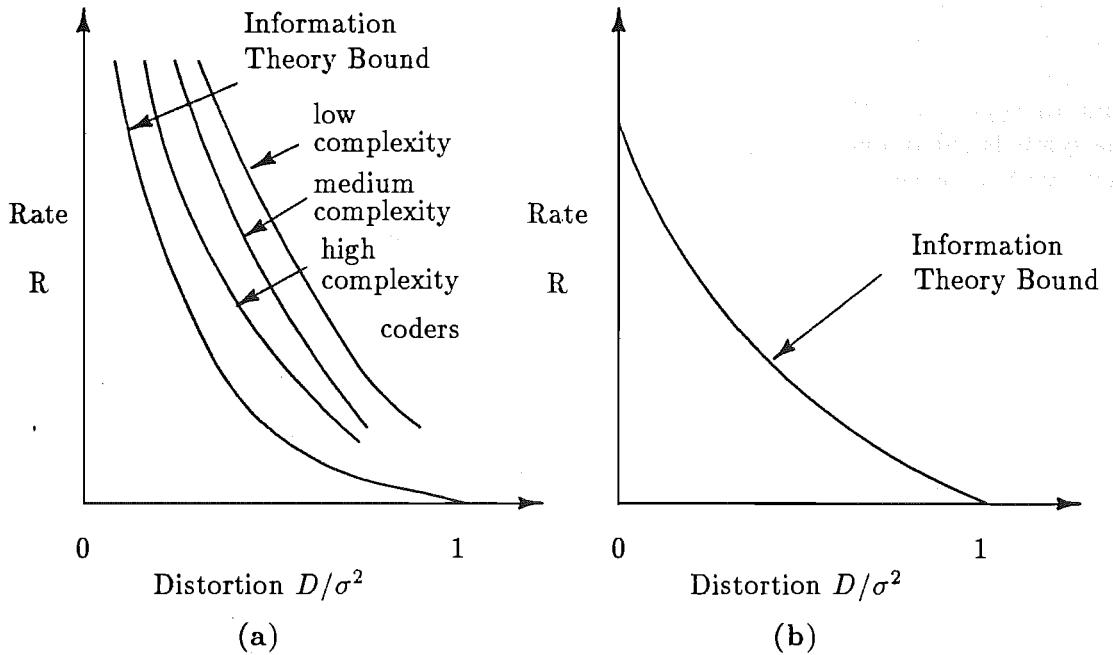
**Figure 2.14.** Arrangements for packing cells in a two dimensional space (from Makhoul *et al.* 1985):  
 (a) Square shaped cells of area  $A_S$ , (b) Hexagon shaped cells of area  $A_H$ .

efficient vector quantizer would only assign cells within the hatched region.

The two aforementioned examples assume the same regular quantization of both vector components, causing cells associated with each codevector to be square. However, it is plausible that other cell shapes can represent the vectors with a smaller total distortion. VQ allows various cell shapes to be chosen, allowing a departure from the standard N-dimensional cubic cells for N-dimensional vectors. The effect of choosing a different cell shape is illustrated in two dimensions by choosing hexagonal shaped cells instead of square cells. In the particular example shown in Fig. 2.14 it is assumed that the cells have the same area so  $A_S = A_H$ . The bit rate of the two quantizers is therefore identical since the same number of cells is required to cover a given area (neglecting the edges). However, the average distortion of the two quantizers differs. The ratio of the distortion from the hexagonal quantizer to that of the square quantizer is calculated by Makhoul *et al.* (1985) to be 0.962. The hexagonal quantizer therefore requires fewer cells than the square quantizer to cover the same area with a given distortion.

The final aspect of VQ design identified by Makhoul *et al.* (1985) is the pdf shape. When the pdf is uniform it is reasonable to choose a uniform cell shape. However, a non-uniform pdf is represented much more accurately by cells with a variety of shapes. This is because the overall distortion can be reduced by representing vectors with a high probability of occurrence more accurately than vectors with a low probability of occurrence. Correspondingly, large cells are assigned to regions of low probability and small cells to regions of high probability.

The VQ training algorithm used in this thesis (§2.7.4.3, §4.2.2) makes use of the above properties to effect a reduction in the storage required for vectors that characterize a person's speech. The VQ training algorithm adapts the positions and sizes of cells in a non-linear fashion and therefore takes into account the distribution of the training vectors (see §2.7.4).



**Figure 2.15.** Qualitative sketches of the rate distortion functions for: (a) a continuous-amplitude source, (b) an discrete-amplitude source. After Jayant and Noll (1984)

### 2.7.2.2 Theoretical performance

The theoretical performance of a vector quantizer is predicted by information theory. A useful measure of VQ performance is the rate, expressed in bits per source sample, required to reconstruct the source information given that a specified average distortion  $D$  can be tolerated (Berger, 1971, p7). This is called the *rate-distortion function* and is denoted  $R(D)$ . The actual transmission rate,  $R$ , has to obey

$$R \geq R(D) \quad \text{for a distortion } D. \quad (2.74)$$

The inverse of  $R(D)$  is  $D(R)$ , the *distortion-rate function* which defines the distortion for a given rate.

Before discussing  $R(D)$  in more detail it is necessary to define the quantity of information emanating from a source. Information is measured in bits and the information rate is specified in bits/vector or bits/symbol. The maximum rate required to transmit a vector  $\mathbf{y}$ , from a set containing a total of  $L$  codevectors is

$$R = \log_2 L. \quad (2.75)$$

The minimum information rate required to transmit these vectors (assuming a memoryless source) with zero distortion is given by the entropy of  $\mathbf{y}$  (Shannon, 1948),

$$H(\mathbf{y}) = - \sum_{i=1}^L p(\mathbf{y}_i) \log_2 p(\mathbf{y}_i). \quad (2.76)$$

Figs. 2.15(a) and (b) show that as the distortion increases the rate decreases monotonically. A memory-less zero-mean Gaussian source with variance  $\sigma^2$  would typically produce the type of rate-distortion curves depicted in Fig. 2.15. For such a source, a rate of zero corresponds to an average distortion of  $\sigma^2$  (which is the average distortion

associated with assuming that the source vector is zero), whereas a rate equal to the entropy produces zero distortion in the ideal case.

Typically, symbols  $y_i$  will occur within a sequence of symbols with different probabilities  $p(y_i)$ . It is therefore inefficient to assign the same number of bits to each of the symbols (Shannon, 1948). The most efficient number of bits required to represent each symbol (or vector) is given by

$$B_i = -\log_2 p(y_i) \text{ bits per vector,} \quad (2.77)$$

so that vectors that occur often are represented with fewer bits than vectors which occur less often. This is called *entropy coding* and Huffman (1973) defines a straightforward procedure for generating codes which approximately obey (2.77). Since  $B_i$  varies for vectors of different probability, the vector codes are of different lengths, which gives rise to the name *variable-length entropy encoding*. For speech transmission it is important to remember that a continuous output of decoded speech is desirable, so if a variable length code is utilized, a certain amount of buffering delay must be incorporated into the decoder to ensure output continuity (Jayant and Noll, 1984, p149).

### 2.7.3 Distortion measures

This section details the types of distortion measures that can be used for evaluating the 'distance' between vectors. The term 'distortion' is used here in the sense defined by Gray *et al.* (1980). The distortion is a positive number which represents the cost or distortion resulting when an input vector is represented by a particular quantized output vector. For the purposes of this description the distortion between two frames of speech data is denoted  $d(x, y)$ . It is desirable for the distortion measure to satisfy at least the first two, and preferably all, of the following properties.

1.  $d(x, y) > 0$  for  $x \neq y$  and  $d(x, x) = 0$ , i.e. the distance between  $x$  and  $y$  is positive except when  $x = y$  for which the distance is zero.
2.  $d(x, y)$  should have a physically meaningful interpretation in the frequency domain so that the distance measure relates to the spectral properties of the speech.
3. It should be possible to efficiently evaluate  $d(x, y)$ .
4.  $d(x, y) = d(y, x)$ , i.e. the measure is symmetric.

The above properties are identified as being important by Gray and Markel (1976) in their study of distance measures. Generally measures that have all of the above properties are called distance measures while those measures which are not necessarily symmetric are called distortion measures (Gray *et al.*, 1980). Property 2 implies that the distortion measure should be perceptually meaningful so that small and large distortions correspond respectively to good and poor speech qualities. The distortion measure must also be amenable to mathematical analysis as a precursor to the design of practical algorithms. Finally, it must be computationally efficient since it will be evaluated many times in any practical application. The properties of various distortion measures are described later in this section.

The most common distortion measure is the mean-square error (MSE) or Euclidean distance, viz

$$d_2(x, y) = \frac{1}{N}(x - y)^T(x - y), \quad (2.78)$$

where  $x$  and  $y$  are two  $N$ -dimensional vectors. The MSE is a popular distance measure because it is straightforward to compute, requiring only  $N$  subtractions and multiplications.

A weighted MSE can be defined which allows contributions to the distortion from the various components of a vector to be assigned different weightings,

$$d_W(\mathbf{x}, \mathbf{y}) = \frac{1}{N}(\mathbf{x} - \mathbf{y})^T W(\mathbf{x} - \mathbf{y}), \quad (2.79)$$

where  $W$  is a positive definite weighting matrix. If  $W$  is set to be the identity matrix  $d_W$  simplifies to the MSE. In addition, if  $W$  is symmetric, it can be factored and the vectors  $\mathbf{x}$  and  $\mathbf{y}$  transformed into a new set of vectors  $\tilde{\mathbf{x}}$  and  $\tilde{\mathbf{y}}$ , i.e.

$$\begin{aligned} W &= P^T P \\ \tilde{\mathbf{x}} &= P\mathbf{x} \\ \tilde{\mathbf{y}} &= P\mathbf{y}. \end{aligned} \quad (2.80)$$

Substituting  $P^T P$  for  $W$  in (2.79) and simplifying gives

$$d_W(\mathbf{x}, \mathbf{y}) = d_2(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}). \quad (2.81)$$

This particular distance measure has the interesting property that it is invariant under any arbitrary nonsingular linear transform of the vector space. A linear transform, such as the one dimensional Fourier transform, will not have any effect on the distortion (Atal, 1974).

$d_2$  and  $d_W$  are general measures that do not incorporate any information about the type of vector being quantized. Several distortion measures have been defined that measure the distortion between the coefficients which represent an LPC filter model and a segment of speech.

The first of the speech based distortion measures was defined by Itakura and Saito (1968). This is the same distortion measure which is minimized when the speech spectrum is approximated by an LPC filter. Refer to §2.6.2 for a more detailed description of this approximation. Support for the Itakura-Saito distortion measure stems from the fact that speech which has been synthesized by invoking LPC techniques sounds reasonable, implying the Itakura-Saito distortion measure is subjectively accurate. For calculation purposes the Itakura-Saito distortion measure can be expressed in the form (Buzo *et al.*, 1980)

$$d_{IS}(|X(e^{j\theta})|^2, |\sigma/A(e^{j\theta})|^2) = \frac{\alpha}{\sigma^2} + \ln(\sigma^2) - \ln(\alpha_\infty) - 1, \quad (2.82)$$

which follows directly from (2.62) if the integral containing  $|E(e^{j\theta})|$  is replaced by the residual  $\alpha$ . There are a number of different approaches to evaluating (2.82) (Gray, 1984; Gray *et al.*, 1980; Buzo *et al.*, 1980). An efficient method is proposed by Buzo *et al.* (1980) who use the Itakura-Saito distortion measure to find the closest match between a number of spectra, rather than the actual value of the distortion. If only a closest match is required then  $\ln(\alpha_\infty)$  and the  $-1$  term in (2.82) can be omitted since they are constant for all values of  $|X(e^{j\theta})|^2$  and (2.82) reduces to

$$d(|X(e^{j\theta})|^2, |\sigma/A(e^{j\theta})|^2) \approx \frac{\alpha}{\sigma^2} + \ln(\sigma^2), \quad (2.83)$$

where  $\sigma^2$  is the gain of the LPC filter as defined in §2.6.2. The residual  $\alpha$  represents the energy in the error signal after  $x[n]$  is passed through the inverse filter  $A(e^{j\theta})$ . This can be expressed in the time domain as (Buzo *et al.*, 1980)

$$\alpha = r_0^a r_0^x + 2 \sum_{n=1}^P r_n^a r_n^x, \quad (2.84)$$

where

$$\begin{aligned} r_n^a &= \sum_{k=0}^{P-n} a_k a_{k+n} \\ r_n^x &= \sum_{k=0}^{N-n} x[k] x[k+n], \end{aligned} \quad (2.85)$$

are the components of the vectors  $\mathbf{r}^a$  and  $\mathbf{r}^x$  respectively. The Itakura-Saito distortion measure is not a symmetrical measure (see §2.6.2) which imposes the restriction that before evaluating the distortion between two signals, a decision must be made about which way around to have them. In a recognition context it is usual for one signal to be the reference, or template, and another test signal to be compared against it. Typically  $\mathbf{r}^a$  is chosen to correspond to the reference and  $\mathbf{r}^x$  to correspond to the test. It is reasonable for recognition schemes and vector quantizers to operate in this configuration, since a test segment, or even a complete word is matched against a collection of reference templates, so the lack of symmetry in the distortion measure does not affect its applicability.

The Itakura-Saito distortion measure incorporates the gain  $\sigma^2$  of the LPC filter into the distortion measure. This means that two spectra that are represented by identical filter coefficients  $a_i$ , but have different gains, will produce a significant Itakura-Saito distortion. The LPC filter gain is dependent on the loudness of a persons voice, but in many recognition situations the spectral parameters of the voice, rather than the loudness, are considered to be more important. To remove this gain dependency, Itakura (1975) defines a gain-optimized version of the Itakura-Saito distortion measure which can be expressed as,

$$d_I(|X(e^{j\theta})|^2, |\sigma/A(e^{j\theta})|^2) = \min_{\lambda \geq 0} d_{IS}(|X(e^{j\theta})|^2, |\lambda\sigma/A(e^{j\theta})|^2). \quad (2.86)$$

If  $\sigma^x/A^x(e^{j\theta})$  is the autoregressive model of  $X(e^{j\theta})$ ,

$$d_I(|X(e^{j\theta})|^2, |\sigma/A(e^{j\theta})|^2) = \ln \left\{ \int_{-\pi}^{\pi} \left| \frac{A(e^{j\theta})}{A^x(e^{j\theta})} \right| \frac{d\theta}{2\pi} \right\}. \quad (2.87)$$

This is sometimes referred to as the log likelihood ratio because the bracketed term is a likelihood ratio under the assumption that the data source is Gaussian and the analysis window is much greater than the inverse filter length (Gray and Markel, 1976; Itakura, 1975). Note that  $d_I$  is related to  $d_{IS}$  through the following relationship (Gray *et al.*, 1980),

$$d_{IS}(|X(e^{j\theta})|^2, |\sigma/A(e^{j\theta})|^2) = \left( \frac{\sigma^2}{\sigma_x^2} \right) e^{d_I(|X(e^{j\theta})|^2, |\sigma/A(e^{j\theta})|^2)} - \ln \left( \frac{\sigma^2}{\sigma_x^2} \right) - 1. \quad (2.88)$$

For practical calculation purposes (2.87) can be expressed in terms of autocorrelation coefficients and prediction coefficients by invoking the autocorrelation matching property (Markel and Gray, 1976, p31) and the Toeplitz property of the autocorrelation matrix  $\mathbf{R}$  (Soong and Sondhi, 1988), viz

$$d_I(|X(e^{j\theta})|^2, |\sigma/A(e^{j\theta})|^2) = \ln \left( \frac{\mathbf{a} \mathbf{R}^x \mathbf{a}^T}{\mathbf{a}^x \mathbf{R}^x \mathbf{a}^{xT}} \right), \quad (2.89)$$

where  $\mathbf{a}^x$  is the LPC prediction coefficient vector of the sequence  $x[n]$  and  $\mathbf{R}^x$  is the matrix of autocorrelation coefficients for the same sequence. The vector  $\mathbf{a}$  contains the

LPC prediction coefficients which represent the impulse response of the filter  $A(e^{j\theta})$ . Equation (2.89) can be further simplified by realizing that the term  $\mathbf{a}^T \mathbf{R}^x \mathbf{a}^{T^T}$  is the same as the gain of the LPC filter  $(\sigma^x)^2$  (Rabiner and Levinson, 1981). Rewriting (2.89) as a summation gives,

$$d_I(|X(e^{j\theta})|^2, |\sigma/A(e^{j\theta})|^2) = \ln \left( \sum_{m=-P}^P \frac{r_m^x}{(\sigma^x)^2} r_m^a \right). \quad (2.90)$$

If  $A(e^{j\theta})$  is a reference vector in a recognition system, the coefficients  $r_i^a$  can be calculated immediately after determining the  $a_i$ s (as defined in (2.85)). The  $r_i^x$  autocorrelation coefficients of the sequence  $x[n]$ , are also required for the LPC analysis and so do not increase the computational overhead. In addition, the summation specified in (2.90) is symmetric about  $P = 0$  so it need only be evaluated from 1 to  $P$ . Computation of (2.90) therefore requires only  $P + 1$  multiplications and additions and one logarithm.

A necessary operation in the design of VQ codebooks is the calculation of the centroid of a cluster of vectors (§2.7.4.2). The gain-optimized Itakura-Saito distortion is not well formulated for this task, but another distortion measure that is closely related to both  $d_I$  and  $d_{IS}$  provides tractable solutions for centroid calculations. Called the model distortion (Gray *et al.*, 1980), it is written in a gain-normalized form as,

$$\begin{aligned} d_m^*(|X(e^{j\theta})|^2, |\sigma/A(e^{j\theta})|^2) &= \int_{-\pi}^{\pi} \left( 1 - \frac{A(e^{j\theta})}{A^x(e^{j\theta})} \right)^2 \frac{d\theta}{2\pi} \\ &= \int_{-\pi}^{\pi} \left| \frac{A(e^{j\theta})}{A^x(e^{j\theta})} \right|^2 \frac{d\theta}{2\pi} - 1 \\ &= e^{d_I(|X(e^{j\theta})|^2, |\sigma/A(e^{j\theta})|^2)} - 1, \end{aligned} \quad (2.91)$$

where the superscript \* indicates that the spectrums  $|X(e^{j\theta})|^2$  and  $|\sigma/A(e^{j\theta})|^2$  are gain normalized by dividing  $|X(e^{j\theta})|^2$  and  $|\sigma/A(e^{j\theta})|^2$  by their respective gains  $(\sigma^x)^2$  and  $\sigma^2$ . The two distortion measures  $d_m^*$  and  $d_I$  measure the same effect since  $d_m^*$  is expressed in (2.91) as a function of  $d_I$  only. In this sense they can be considered equivalent, although actual distortion values will differ. It is therefore acceptable to use  $d_m^*$  instead of  $d_I$  in certain circumstances (see §2.7.4.2). Furthermore, by comparing  $d_m^*$  with  $d_{IS}$ , as expressed in (2.88), one can see that gain normalizing (2.88) gives  $d_m^*$  (as defined in (2.91)) exactly.

#### 2.7.4 Quantizer design

The task of designing a vector quantizer is to determine a codebook which represents a particular distribution of vectors with minimum quantization distortion. One approach to designing such a codebook is to use a type of neural network, the Kohonen self-organizing feature map (Kohonen, 1990), and to train it to represent the distribution of training vectors. However, here the approach is to use the more usual iterative methods for training a VQ system developed by Linde *et al.* (1980). The reader is referred to Wu and Fallside (1991) for a comparison between the self-organizing feature map and the Linde, Buzo and Gray training method.

##### 2.7.4.1 Notation

Recall from §2.7.3 that the distortion between an input vector  $\mathbf{x}$  and the quantizer output  $\mathbf{y}$  is written  $d(\mathbf{x}, \mathbf{y})$ . The notation  $\mathbf{x}[n]$  is used to represent one of  $M$  unquantized

vectors. The overall average distortion due to the vector quantization of  $M$  vectors is,

$$D = \lim_{M \rightarrow \infty} \frac{1}{M} \sum_{n=1}^M d(\mathbf{x}[n], \mathbf{y}[n]). \quad (2.92)$$

If a vector process  $\mathbf{x}[n]$  is stationary and ergodic, the limit in (2.92) exists and tends towards the expectation  $E(d(\mathbf{x}, \mathbf{y}))$  (Makhoul *et al.*, 1985).

Partitioning of the vectors  $\mathbf{x}$  by the codebook  $\mathbf{Y}$  is written  $\mathcal{P}(\mathbf{Y})$  where  $\mathcal{P}(\mathbf{Y})$  is defined as

$$\mathcal{P}(\mathbf{Y}) = \{C_i ; i = 1, 2, \dots, L\} \quad \mathbf{x} \in C_i \text{ if } d(\mathbf{x}, \mathbf{y}_i) \leq d(\mathbf{x}, \mathbf{y}_j) \text{ for all } j, \quad (2.93)$$

where the notation  $\mathbf{x} \in C_i$  means that  $\mathbf{x}$  is an element of cell  $C_i$  under the conditions specified. The vectors in all  $L$  cells are defined in this manner. Summarizing (2.93) in words, a partition that is optimum for a given codebook  $\mathbf{Y}$  is determined by associating each  $\mathbf{x}$  with the cell corresponding to the minimum distortion or nearest-neighbour codeword.

An  $N$  level quantizer is said to be *globally optimal* if its expected distortion is less than that of all other quantizers. Like the globally optimal quantizer, a *locally optimal* quantizer has the property that the distortion increases for slight changes in the codebook. However, the distortion is only a local minimum so better codebooks may exist. Although it is preferable to determine a globally optimal quantizer, in practice it is more straightforward to identify a locally optimal quantizer that is adequate for the quantization task at hand.

It is useful to introduce notation for describing the total distortion between a set of training vectors and the current codebook so that the progress of a training algorithm can be monitored. Here the use of curly braces, for example  $D(\{\mathbf{X}, \mathbf{Y}\})$ , indicates that the distortion measure is the sum of the distortion between each training vector  $\mathbf{x}[n]$  and its closest match  $\mathbf{y}[n]$ . This can be expressed in the following two ways,

$$\begin{aligned} D(\{\mathbf{X}, \mathbf{Y}\}) &= \frac{1}{M} \sum_{n=1}^M d(\mathbf{x}[n], \mathbf{y}[n]) \\ &= \sum_{i=1}^L \sum_{j : \mathbf{x}[j] \in C_i}^{||C_i||} d(\mathbf{x}[j], \mathbf{y}_i), \end{aligned} \quad (2.94)$$

where  $||C_i||$  is the number of vectors in cell  $C_i$ , and the notation  $j : \mathbf{x}[j] \in C_i$  means  $j$  such that the vector  $\mathbf{x}[j]$  lies in the cell  $C_i$ .

A number of the distortion measures described in this section are between spectra rather than a vector  $\mathbf{x}$  or  $\mathbf{y}$ . In the same way that  $d(\mathbf{x}, \mathbf{y})$  denotes the distance between the vector  $\mathbf{x}$  and the vector  $\mathbf{y}$ ,  $d(|X(e^{j\theta})|^2, |G(e^{j\theta})|^2)$  denotes the difference between the spectrum  $|X(e^{j\theta})|^2$  and another spectrum  $|G(e^{j\theta})|^2$ . It is convenient to represent  $|G(e^{j\theta})|^2$  by the LPC filter  $|\sigma/A(e^{j\theta})|^2$ , where the coefficients and gain for the filter  $|\sigma/A(e^{j\theta})|^2$  are computed to match the spectrum  $|G(e^{j\theta})|^2$ .

#### 2.7.4.2 Centroid calculation

Each codevector has associated with it a partition of training vectors which, by definition, are ‘closest’ to the codevector. The centroid of the partition is the minimum distortion codevector for the partition. Ideally, the codevector should be in exactly the same position within the  $N$ -dimensional vector space as the centroid of the partition. This relationship between partitions, code vectors and centroids is utilized in the iterative algorithm described in §2.7.4.3 to improve the quantizer. The method selected

for determining the minimum distortion centroid vector of a cluster of vectors depends upon the type of distortion measure used.

For the MSE criterion, the centroid of the vectors lying within cell  $C_i$  is given by the arithmetic mean

$$\text{cent}C_i = \frac{1}{\|C_i\|} \sum_{j: \mathbf{x}[j] \in C_i} \mathbf{x}[j] \quad (2.95)$$

of the vectors. The number of training vectors in the cell  $C_i$  is denoted  $\|C_i\|$ .

The justification of the centroid calculation method for the Itakura-Saito distortion measure is described in Gray *et al.* (1980) and is included here for completeness. Assume that for some given distortion measure  $d$ ,  $\hat{f}(\theta) = \hat{\sigma}^2 / |\hat{A}(e^{j\theta})|^2$  describes the centroid of a cluster containing  $n$  vectors of the form  $f_i(\theta) = \sigma_i^2 / |A_i(e^{j\theta})|^2$ . For the Itakura-Saito distortion measure the total distance  $D_{IS}(\hat{f})$  between the centroid and each of the vectors is given by

$$\begin{aligned} D_{IS}(\hat{f}) &= \frac{1}{n} \sum_i \left( \int_{-\pi}^{\pi} \left( \frac{f_i(\theta)}{\hat{f}(\theta)} \right) \frac{d\theta}{2\pi} - \ln \left( \frac{\sigma_i^2}{\hat{\sigma}^2} \right) - 1 \right) \\ &= \int_{-\pi}^{\pi} \left( \frac{\bar{f}(\theta)}{\hat{f}(\theta)} \right) \frac{d\theta}{2\pi} - \frac{1}{n} \sum_i \ln \left( \frac{\sigma_i^2}{\hat{\sigma}^2} \right) - 1, \end{aligned}$$

where  $\bar{f}(\theta)$  is the arithmetic mean of all the  $f_i(\theta)$ . Furthermore, if  $\bar{\sigma}^2$  is the gain of  $\bar{f}(\theta)$  then

$$\begin{aligned} D_{IS}(\hat{f}) &= \int_{-\pi}^{\pi} \frac{\bar{f}(\theta)}{\hat{f}(\theta)} \frac{d\theta}{2\pi} - \ln \left( \frac{\bar{\sigma}^2}{\hat{\sigma}^2} \right) - 1 + \ln(\bar{\sigma}^2) - \frac{1}{n} \sum_i \ln(\sigma_i^2) \\ &= d_{IS}(\bar{f}, \hat{f}) + \ln(\bar{\sigma}^2) - \frac{1}{n} \sum_i \ln(\sigma_i^2), \end{aligned} \quad (2.96)$$

which is minimized when  $\hat{f}$  equals  $\bar{f}$ , since the gain  $\bar{\sigma}^2$  and  $\sigma_i^2$  terms depend upon the cluster itself so are independent of  $\hat{f}$ . The  $f_i(\theta)$  and the autocorrelation coefficients  $\mathbf{r}^x[i]$  are related by the Fourier transform. It is therefore acceptable to compute  $\hat{\sigma}^2 / |\hat{A}(e^{j\theta})|^2$  (which corresponds to  $\hat{f}$ ) by averaging autocorrelation coefficients and performing LPC in the usual manner.

The gain-optimized Itakura distortion between a centroid  $\hat{\sigma} / \hat{A}(e^{j\theta})$  and a cluster of  $n$  vectors has a logarithm incorporated within the summation (Gray *et al.*, 1980),

$$\begin{aligned} D_I \left( \frac{\hat{\sigma}}{\hat{A}(e^{j\theta})} \right) &= \frac{1}{n} \sum_i d_I(|\sigma_i / A_i(e^{j\theta})|^2, |\hat{\sigma} / \hat{A}(e^{j\theta})|^2) \\ &= \frac{1}{n} \sum_i \left\{ \ln \int_{-\pi}^{\pi} \left| \frac{\hat{A}(e^{j\theta})}{A_i(e^{j\theta})} \right| \frac{d\theta}{2\pi} \right\}, \end{aligned} \quad (2.97)$$

which is minimized when  $\hat{A}(e^{j\theta})$  is the geometric mean of the  $A_i(e^{j\theta})$ . The geometric mean of the  $A_i(e^{j\theta})$  is computationally intractable, but it can be approximated by the arithmetic mean, which constitutes an upper bound (in a distortion sense) on the geometric mean (Gray *et al.*, 1980). This is equivalent to minimizing the normalized model distortion  $d_m^*$  as defined by (2.91). The total distortion between a cluster of vectors and  $\hat{f}$  when using  $d_m^*$  is

$$\begin{aligned} D_m^*(\hat{f}) &= \frac{1}{n} \sum_i d_m^*(f_i, \hat{f}) \\ &= \frac{1}{n} \sum_i \frac{1}{\sigma^2} \int_{-\pi}^{\pi} f_i(\theta) |\hat{A}(e^{j\theta})|^2 \frac{d\theta}{2\pi} \\ &= d_m^*(\bar{f}, \hat{f}), \end{aligned} \quad (2.98)$$

where

$$\bar{f} = \frac{1}{n} \sum_i \frac{f_i(\theta)}{\sigma_i^2}. \quad (2.99)$$

Recall from §2.7.3 that  $d_m^*$  is equivalent to  $d_I$  in that they are both based on the gain optimized Itakura-Saito distance measure. They therefore give the same subjective speech quality. The use of  $d_m^*$ , however, provides a more tractable method for determining centroids.

#### 2.7.4.3 Iterative codebook design

This section describes the locally optimal iterative vector quantizer design algorithm proposed by Linde *et al.* (1980). The Linde, Buzo and Gray algorithm (LBG), as it is often called, has become a standard method for designing VQs with a constrained number of codevectors.

Identification of a locally optimal codebook requires a practical method for updating a set of cells  $C$  in such a manner that the total codebook distortion is reduced. The expected distortion from using codebook  $\mathbf{Y}$  to quantize a training set  $\mathbf{X}$  can be written as

$$D(\{\mathbf{Y}, \mathcal{P}(\mathbf{Y})\}) = E(\min_{\mathbf{y} \in \mathbf{Y}} d(\mathbf{X}, \mathbf{y}_i)), \quad (2.100)$$

where the partition  $\mathcal{P}(\mathbf{Y})$  is determined in the manner defined by (2.93). Because the partition in (2.100) has been constructed so that the total distortion for the given codebook  $\mathbf{Y}$  is minimized, any other partition will produce at least as great a distortion. For example, comparing any partition  $C$  with  $\mathcal{P}(\mathbf{Y})$  gives

$$D(\{\mathbf{Y}, C\}) \geq D(\{\mathbf{Y}, \mathcal{P}(\mathbf{Y})\}), \quad (2.101)$$

because the best possible partition of the codebook  $\mathbf{Y}$  is  $\mathcal{P}(\mathbf{Y})$ .

Even though the best possible partition of the codebook  $\mathbf{Y}$  is  $\mathcal{P}(\mathbf{Y})$ , it does not necessarily follow that the codebook  $\mathbf{Y}$  is the best possible codebook for  $\mathcal{P}(\mathbf{Y})$ . This is best illustrated by considering the situation where a large number of vectors in the training set lie to one side of the cell and the overall distortion for the cell can be reduced by moving the codevector to new location at the 'centre' of the group of training vectors. It is therefore necessary to determine the set of  $\mathbf{Y}$  centroid vectors that minimizes the distortion for a given partition. Assume that a known partition  $C = \{C_i; i = 1, \dots, L\}$  describes a quantizer and the distribution of  $\mathbf{x}$  is such that each of the  $C_i$  cells contains one or more vectors. For each cell  $C_i$  there exists a minimum distortion centroid vector (denoted here  $\text{cent}C_i$ ) defined by

$$E(d(\mathbf{x}, \text{cent}C_i) | \mathbf{x} \in C_i) = \min_{\mathbf{y}_i} E(d(\mathbf{x}, \mathbf{y}_i) | \mathbf{x} \in C_i). \quad (2.102)$$

The codebook which yields the least distortion for a fixed partition  $C$  is therefore written  $\mathbf{Y} = \{\text{cent}C_i; i = 1, \dots, L\}$ .

An iterative algorithm which improves a given quantizer by alternately partitioning and identifying codebooks is now developed. The iteration index is denoted by the subscript  $m$  and the initial codebook by  $\mathbf{Y}_0$ . For this particular algorithm the initial values of  $\mathbf{Y}_0$  are defined to be a set of  $L$  codevectors that are uniformly spaced in the  $N$ -dimensional sample space. These initial codevectors are then iteratively adjusted by Algorithm 2.1.

#### Algorithm 2.1

Step 1: Initialization: Let  $L$  equal the number of codevectors and  $M$  be the number of vectors in the training sequence. Let  $\mathbf{Y}_0$  be a set of  $L$  uniformly distributed  $N$ -dimensional codevectors and  $\mathbf{X}$  a set of  $M$  training vectors. Set  $m = 0$  and  $D_{-1} = \infty$ .

Step 2: Partitioning: Given a codebook  $\mathbf{Y}_m$ , find its minimum distortion partition  $\mathcal{P}(\mathbf{Y}_m)$  by invoking (2.93) for all vectors  $\mathbf{x}$  of  $\mathbf{X}$ . Compute the average distortion for the resulting partition from  $D_m = D(\{\mathbf{Y}_m, \mathcal{P}(\mathbf{Y}_m)\})$  using (2.100).

Step 3: Termination test: Check the decrease in distortion. If  $(D_{m-1} - D_m)/D_m \leq \epsilon$  (where  $\epsilon$  is 0.001) call  $\mathbf{Y}_m$  the final codebook and exit the algorithm. Otherwise continue.

Step 4: Codebook Updating: Find the optimal codebook from the latest partition by computing

$$\mathbf{Y}_{m+1} = \text{cent}\mathcal{P}(\mathbf{Y}_m). \quad (2.103)$$

Go to step (2).

Linde *et al.* (1980) test Algorithm 2.1 using a source with known characteristics, in this case a zero-mean, unit variance memoryless Gaussian source. Optimum distortion values for this source can be determined theoretically and are reported by Max (1960). For the scalar case with the number of output levels set to  $L = 2, 3, 4, 6$  and 8 and using 10 000 samples per quantizer output, no more than 20 iterations of Algorithm 2.1 were required to reduce the distortion to within 1% of the optimal values reported by Max (1960).

Linde *et al.* (1980) also report tests of Algorithm 2.1 for designing quantizers to work with blocks of samples. Note that a block containing 2 samples is treated as a single vector having 2 dimensions, and so on for the other block sizes. They report convergence in fewer than 50 iterations of Algorithm 2.1 for quantizing a 100 000 sample sequence into equal length blocks containing 1, 2, 3, 4, 5 and 6 samples for a rate of one bit per sample. The performance of Algorithm 2.1 for the block quantization task is depicted in Fig. 2.16. The block quantizer outperforms the scalar quantizer, but is still significantly worse than the rate distortion bound  $D(R) = 2^{-2R}$ , which equals 0.25 for  $R = 1$  bit per sample. However, in principle,  $D(R)$  is only approached in the limit as  $N \rightarrow \infty$ . Yamada *et al.* (1980) formulated a lower distortion bound for an  $L$ -level  $N$ -dimensional quantizer when  $L$  is large. The distortion bound tends to the rate-distortion bound as  $N \rightarrow \infty$  and, as indicated in Fig. 2.16, the training algorithm is within 6% of this optimal bound (Linde *et al.*, 1980).

Iterative codebook design techniques of the type described by Algorithm 2.1 provide methods for calculating better codebooks given an initial codebook of a specific size, in this case  $\mathbf{Y}_0$ . However, the method for choosing an initial codebook of the required size is not specified. Provided a locally optimal codebook of half the required size is known, a new codebook of the required size can be obtained by splitting each of the codevectors. Linde *et al.* (1980) call this “initial guess by splitting” method since the new codebook is produced by splitting each  $\mathbf{y}_i$  into two close vectors  $\mathbf{y}_i + \boldsymbol{\varepsilon}$  and  $\mathbf{y}_i - \boldsymbol{\varepsilon}$ , where  $\boldsymbol{\varepsilon}$  is a small perturbation vector. Note that the actual direction of the vector  $\boldsymbol{\varepsilon}$  is not critical since the newly ‘split’ vectors are iteratively moved to a more optimal (lower quantization error) position. The magnitude of  $\boldsymbol{\varepsilon}$  should be chosen to be small compared with the magnitude of  $\mathbf{y}_i$  and here all the components of  $\boldsymbol{\varepsilon}$  have small positive values (§4.2.2.2 contains details of the value of  $\boldsymbol{\varepsilon}$  used in the computation of speaker codebooks). The number of centroid vectors doubles at each split and the final codebook is constrained to  $2^R$  vectors, where  $R$  is the number of splits that have

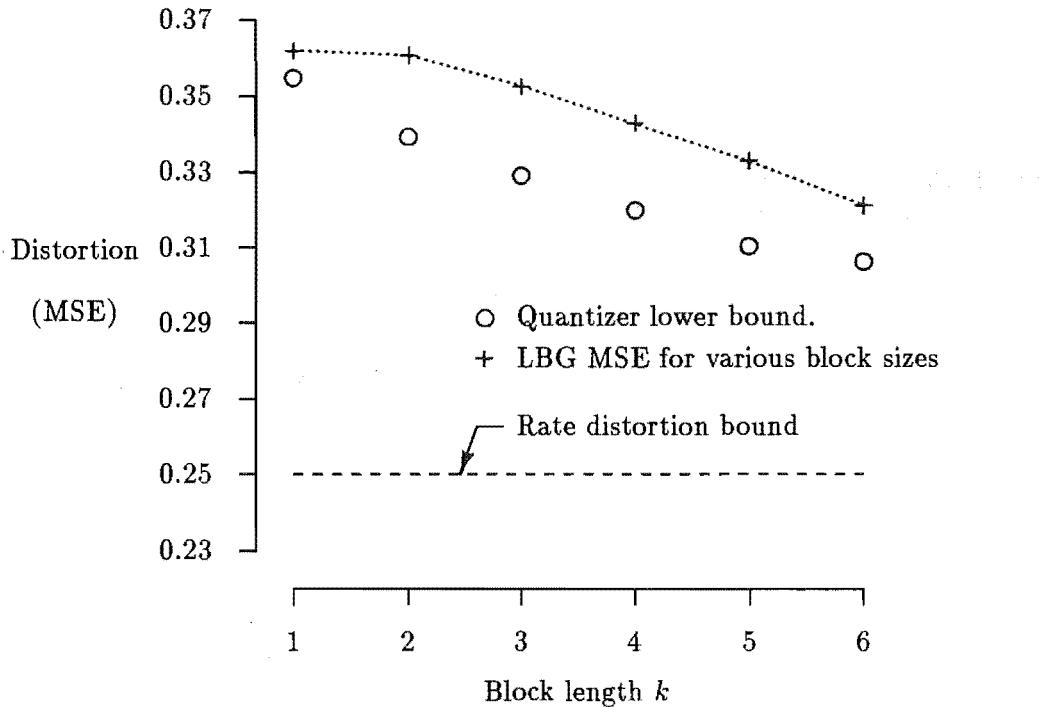


Figure 2.16. Distortion produced by block quantization of a Gaussian source at a rate of 1 bit per symbol (after Linde *et al.* (1980)).

occurred.  $R$  is often referred to as the *codebook rate* since it defines the number of bits required to uniquely identify a vector within a codebook. So as to record the total number of vectors in  $\mathbf{Y}$ , the number of vectors is inserted in parenthesis after the  $\mathbf{Y}$ . In a speaker recognition context the sequence of vectors  $\mathbf{x}[j]; j = 0, \dots, M - 1$  used to obtain speaker codebooks is computed from the training utterances belonging to a particular speaker. The following algorithm, based on that defined by Linde *et al.* (1980), is invoked to determine the set of codevectors  $\mathbf{Y}$  that represents a particular speaker's speech. Tests of the algorithm implementation on speech signals are reported in §4.2.2.3.

### Algorithm 2.2

Step 1: Initialisation: Fix the largest number of codevectors desired to be  $2^R$ , where  $R$  is an integer. Set  $M$  to the number of vectors in the training sequence and  $L$ , the number of codevectors, to 1. Define  $C_0 = \{\mathbf{x}[j]; j = 0, \dots, M - 1\}$  and  $\mathbf{Y}(1) = \text{cent}C_0$ , the centroid of the entire training sequence.

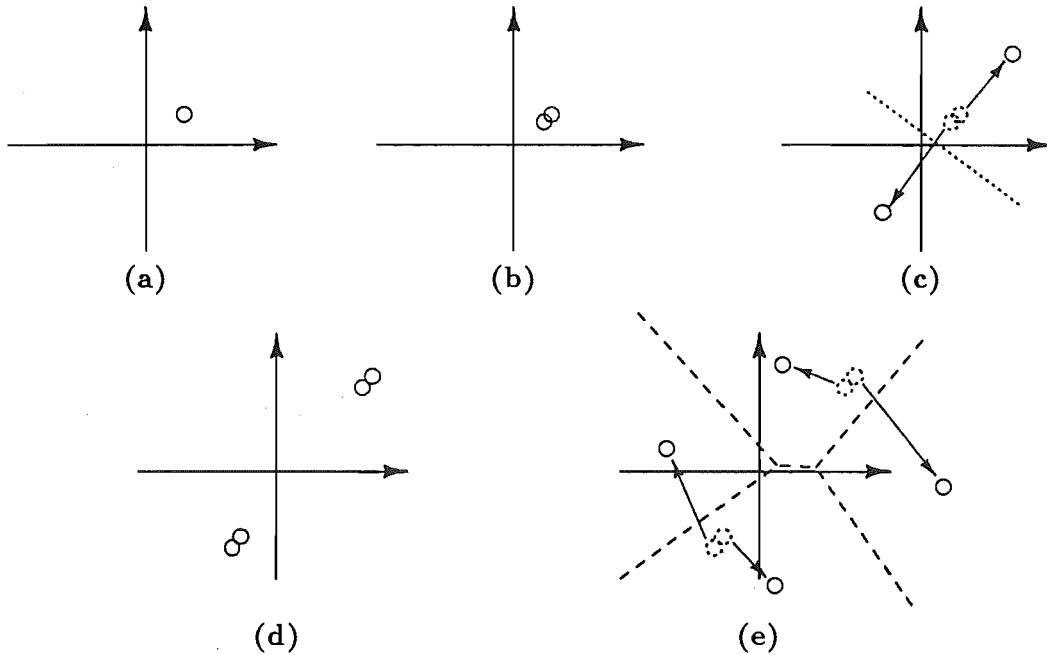
Step 2: Splitting: Given  $\mathbf{Y}(L) = \{\mathbf{y}[j]; j = 0, \dots, L\}$ , split each codebook vector into  $\mathbf{y}_i + \boldsymbol{\epsilon}$  and  $\mathbf{y}_i - \boldsymbol{\epsilon}$ . Set  $\mathbf{Y}_m(2L) = \{\mathbf{y}_i + \boldsymbol{\epsilon}, \mathbf{y}_i - \boldsymbol{\epsilon}, i = 1, \dots, L\}$  and replace  $L$  by  $2L$ .

Step 3: Reset variables: Set  $m = 0$  and  $D_{-1} = \infty$ .

Step 4: Partitioning: Find the optimum partition for the codebook  $\mathbf{Y}_m(L), \mathcal{P}(\mathbf{Y}_m(L))$  using (2.93). Compute the resulting distortion

$$D_m = D(\{\mathbf{Y}_m(L), \mathcal{P}(\mathbf{Y}_m(L))\}). \quad (2.104)$$

Step 5: Termination Test: If  $(D_{(m-1)} - D_m)/D_m \leq \epsilon = 0.005$ , go to step 7. Otherwise continue.



**Figure 2.17.** An example of splitting centroids to form new codebooks: (a) a single centroid of the entire training sequence, (b) the single codevector is split into two codevectors, (c) good positions for the two centroids are obtained by iteratively updating their positions, (d) the two centroids are split, (e) the four centroids are iteratively refined to give the final positions of the four centroids and cell boundaries are indicated with dotted lines.

Step 6: Update Codebook: find the next codebook  $\mathbf{Y}_{m+1}(L) = \text{cent} \mathcal{P}(\mathbf{Y}_m(L))$ , the centroids of the partitions for the codebook  $\mathbf{Y}_m(L)$ . Replace  $m$  by  $m + 1$  and go to step 4.

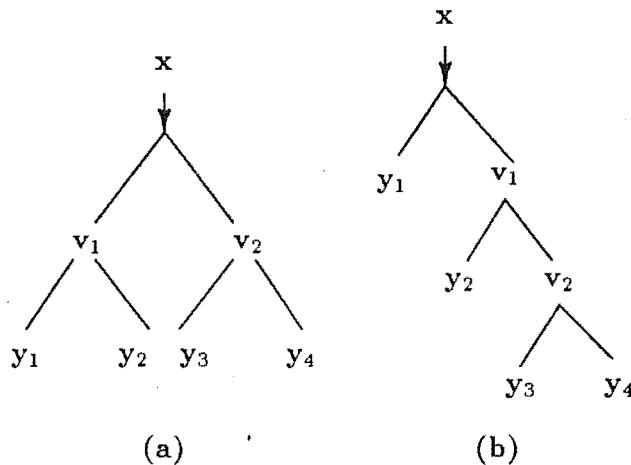
Step 7: Final Rate Test: Set  $\mathbf{Y}(L) = \mathbf{Y}_m(L)$ . If  $L < 2^R$  go to step 2, otherwise halt with the final quantizer  $\mathbf{Y}_m(L)$ .

Fig. 2.17 shows a two-dimensional example of the VQ splitting process. A single centroid, as shown in Fig. 2.17(a) is split into two centroids which are separated by  $2\epsilon$ . These two centroids are adjusted by partitioning (step 4) and then updating the codebook (step 6). Once centroid adjustments no longer reduce the distortion significantly (i.e.  $< \epsilon$ ), new centroids are split off the current ones, as depicted in Fig. 2.17(c), and the process continues.

#### 2.7.4.4 Aspects of codebook storage and codebook searching

Once a codebook has been designed it can be used to quantize any sequence of input vectors  $\mathbf{x}[n]$ . This involves searching the codebook to identify a codevector to associate with each input vector. The type of search employed depends on the codebook structure and the relationship between the stored codevectors. Codebook design involves choosing an appropriate codebook structure for the task at hand, taking into account the storage requirements and the computational complexity of the search algorithm. In general, the computational effort required to quantize a single vector can be reduced at the expense of additional codebook storage.

For the purposes of comparing the storage requirements of different codebooks a memory unit is defined as the amount of memory required to store a single real number. Recall that the number of dimensions in the unquantized vector is denoted  $N$  and the number of codevectors in the codebook is  $L$ .



**Figure 2.18.** Tree structures for vector quantizer codebooks: (a) a binary tree which is searched by comparing the intermediate  $v_i$  vectors with the input vector  $x$ , (b) a nonuniform tree.

## Full search codebook

One method of searching a codebook is to compute the distortion between the input vector and every vector in the codebook and choose the codevector with the minimum distortion. This method is called *full search* vector quantization. If the distortion calculation between an  $N$ -dimensional vector and a codevector requires  $N$  operations, and a codebook contains  $L$  code vectors, the number of operations to quantize a single vector is  $NL$ . The codevectors of a full search codebook are stored consecutively in memory, in no particular order. The total storage requirement for a full search codebook is therefore  $NL$ .

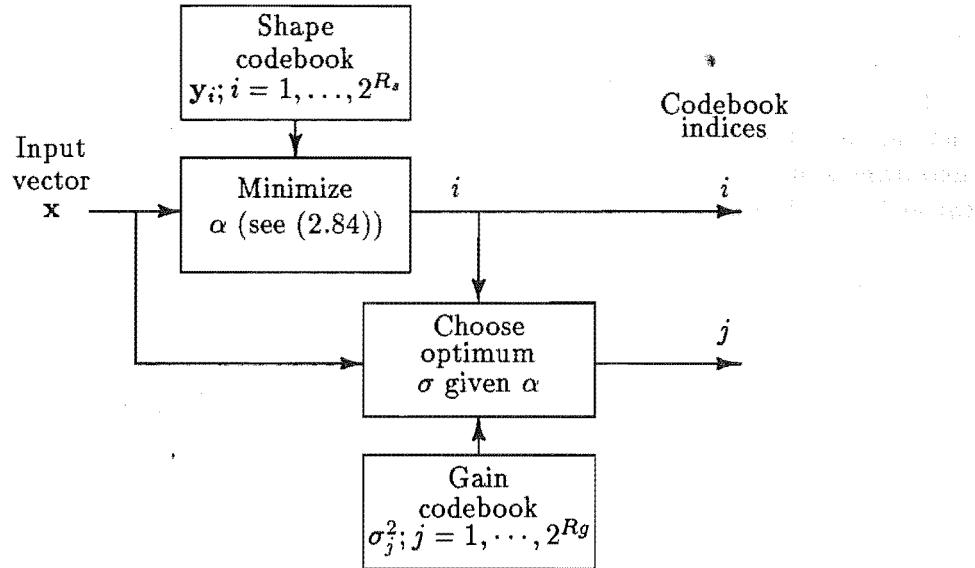
## Tree-search codebook

An alternative codebook structure is the uniform binary tree structure depicted in Fig. 2.18(a). The codebook is split into levels and the uppermost level is used to split the set of codevectors into two. Each successive level continues this splitting process until each set contains only a single codevector.

The computational requirements for searching a binary tree are significantly less than for a full search. The total number of operations is  $2N \log_2 L$  which only increases linearly with the number of bits. However, compared with a full search codebook, the total storage requirements are almost doubled to  $2N(L - 2)$  memory units, because, in addition to the codevectors, vectors representing the intermediate branches in the tree must be stored. Furthermore, the best match codevector is not guaranteed to be selected since the entire set of codevectors is not searched for the lowest distortion codevector.

Buzo *et al.* (1980) report comparisons between full-search and binary-search vector quantizations systems applied to speech coding. At a rate of 10 bits per speech frame, which corresponds to 1024 vectors in the codebook, the spectral distortion for the binary-search method was approximately 0.6 dB higher than the full-search method. However, the authors remark that this difference in spectral error is not significant when the computational reduction from 1024 distortion calculations to 20 distortion calculations is taken into consideration.

The binary tree depicted in Fig. 2.18(a) is uniform in the sense that each of the nodes splits off into two lower nodes. At some stage in the iterative training process it is possible that one of the clusters will have few vectors within it. To further split



**Figure 2.19.** Gain/shape VQ with a gain rate of  $R_g$  and a (spectral) shape rate of  $R_s$ . The spectrum of the input speech is initially compared against normalized spectral models. A filter is selected and the required gain is then determined via a scalar quantizer. Note that the ordering of the gain codebook from the smallest to largest removes the need for a full search of the gain codebook.

up such a cluster is wasteful of codevectors since the overall distortion will not be significantly reduced. Forming a VQ based on a nonuniform tree structure alleviates this difficulty because the cluster that distorts the input vectors most (or contains the most training vectors) is always the next to be subdivided. In this way the tree ‘grows’ in the direction where the most significant distortion reductions can be achieved (see Fig. 2.18(b)).

### Product codes

Sometimes the parameters that are to be quantized consist of the product of two or more identifiably different components. If these components are quantized separately, each with their own codebook, the product of the component codebooks is called a product code (Makhoul *et al.*, 1985; Gray, 1984).

An example of components of a product code is the gain of an LPC filter and the filter coefficients (Buzo *et al.*, 1980). This method of VQ is often called gain/shape VQ. A block diagram of gain/shape VQ system employing the Itakura-Saito distortion measure is shown in Fig. 2.19. The steps in performing quantization using the gain/shape VQ depicted in Fig. 2.19 are sequential since the LPC filter component is quantized first and the value of the quantized vector is used in the quantization of the gain component. Buzo *et al.* (1980) argue that separate gain/shape quantization allows smaller codebooks, although the total distortion increases slightly. If  $A(e^{j\theta})$  is represented by  $R_s$  bits and  $\sigma$  by  $R_g$  bits, the codebook size can be reduced from  $2^{R_s+R_g}$  for a full codebook to  $2^{R_s} + 2^{R_g}$  by separating parameters into different codebooks.

## 2.8 SHIFT-AND-ADD

This section introduces a processing technique called shift-and-add (SAA) which forms the basis of much of my research work. SAA is used to obtain an estimate of the long-term average glottal response, or LTAGR. Thorpe (1990) describes detailed investigations into the nature of the signal that is obtained by applying SAA to voiced

speech. Section 2.8.1 outlines the historical background of the technique, while §2.8.2 describes its application to speech. Section 2.8.3 describes the computational details of the SAA algorithm. The effect of pre-emphasis on the LTAGR is described in §2.8.4 and the use of a threshold to perform the voiced/unvoiced decision within the SAA algorithm is discussed in §2.8.5. The relationship of the LTAGR of synthetic speech to the artificial glottal pulse used to produce the speech is described in §2.8.6.

### 2.8.1 Historical background

Shift-and-add (SAA) processing was first postulated as a technique for improving the observable detail in astronomical images (Bates and Cady, 1980). Due to the optical transmission properties of the atmosphere such images are distorted versions of the celestial (or true) object. The atmospheric distortion is usefully modelled as a convolution of the true object with a wide band blurring function that is essentially invariant over short intervals of time but varies randomly over long intervals (Bates and Cady, 1980). SAA is one of a group of techniques that counteracts the image distortion by processing a sequence of short-time exposures (Bates and McDonnell, 1986). Each exposure or speckle image is distorted with a statistically independent blurring function. The ensemble of speckle images are combined so that the blurring tends to cancel, while the image is reinforced. SAA is based on the assumption that if the true object contains a dominant point, the brightest point of any speckle image is more likely to correspond to that point than to any other part of the object. Since convolution can be viewed as the superposition of many amplitude scaled and shifted copies of the true object (speckles), it is appropriate to shift the speckle image so that the brightest point of the speckle image lies at the origin (Bates, 1982). All the speckle images are similarly processed and then added together, causing portions of the image that are aligned at the origin to reinforce, while other randomly distributed lower amplitude ‘speckles’ tend to be cancelled. In experiments using simulated speckle images (Sinton, 1986, §5), and later actual astronomical data (Davey, 1989, §4.8.1), SAA was shown to faithfully reconstruct an image and reduce the atmospheric blurring to a background ‘fog’.

### 2.8.2 SAA processing of speech

Interaction between researchers at the University of Canterbury working in the fields of astronomical image processing and speech processing led to SAA being applied to speech by Brieseman *et al.* (1987). The description of SAA in §2.8.1 can be applied to speech by making a correspondence between the celestial object and the glottal excitation, and the atmospheric distortions and the vocal tract ‘distortion’ filter. Similarly, choosing the brightest point of a speckle image as a reference point corresponds to the assumption that the largest magnitude within a pitch period of voiced speech is most likely to represent the largest peak in the glottal pulse (Thorpe and Bates, 19XX; Davey and Thorpe, 1987).

Using the source-filter model of speech introduced in §2.2, each pitch period of speech can be sub-divided into its glottal excitation and a vocal tract filter. Here the  $m^{\text{th}}$  glottal pulse is represented by  $g_m(t)$ , which is constrained in the following manner,

$$\begin{aligned} g_m(t) &= 0 \text{ for } t < -\tau/2 \text{ and } t > \tau/2 \\ |g_m(0)| &\geq |g_m(t)| \text{ for } -\tau/2 < t < \tau/2. \end{aligned} \quad (2.105)$$

Note that constraining the extent of  $g_m(t)$  to be  $\pm\tau/2$  is an arbitrary choice that is computationally expedient. The contribution from the excitation signal and the vocal tract response during the  $m^{\text{th}}$  pitch period are denoted respectively by  $g_m(t)$  and  $v_m(t)$ .

The  $m^{th}$  pitch period of a speech record  $s_m(t)$  is expressed as

$$s_m(t) = g_m(t - T_{em}) \odot v_m(t) + c_m(t), \quad (2.106)$$

where, during the  $m^{th}$  pitch period,  $T_{em}$  is the instant at which the excitation is at its maximum. The range of  $m$  is 1 to  $M$ , the total number of pitch periods in the speech record. The contamination term  $c_m(t)$  includes any overlap in the pitch periods, recording noise, and interactions between the vocal tract and the glottal source. Recall from §2.4.1 that, for modelling purposes, the vocal tract can be considered stationary for at least a single pitch period. Therefore,  $v_m(t)$  can be expected to represent the vocal tract response for the entire  $m^{th}$  pitch period.

After a speech record has been partitioned into segments  $s_m(t)$ , SAA is invoked to estimate the long-term average glottal excitation  $g(t)$ . For each  $m$ , the instant at which  $|s_m(t)|$  is maximum is assigned to  $T_m$ . Each  $s_m(t)$  is then shifted so that the maximum occurs at the time origin and if the largest peak is negative, the sign of  $s_m(t)$  is reversed. All of the  $M$   $s_m(t)$  so processed, are averaged,

$$s_{saa}(t) = \frac{1}{M} \sum_{m=1}^M \text{sgn}(s_m(T_m)) s_m(t + T_m) \quad (2.107)$$

$$= < \text{sgn}(s_m(T_m)) s_m(t + T_m) >_m, \quad (2.108)$$

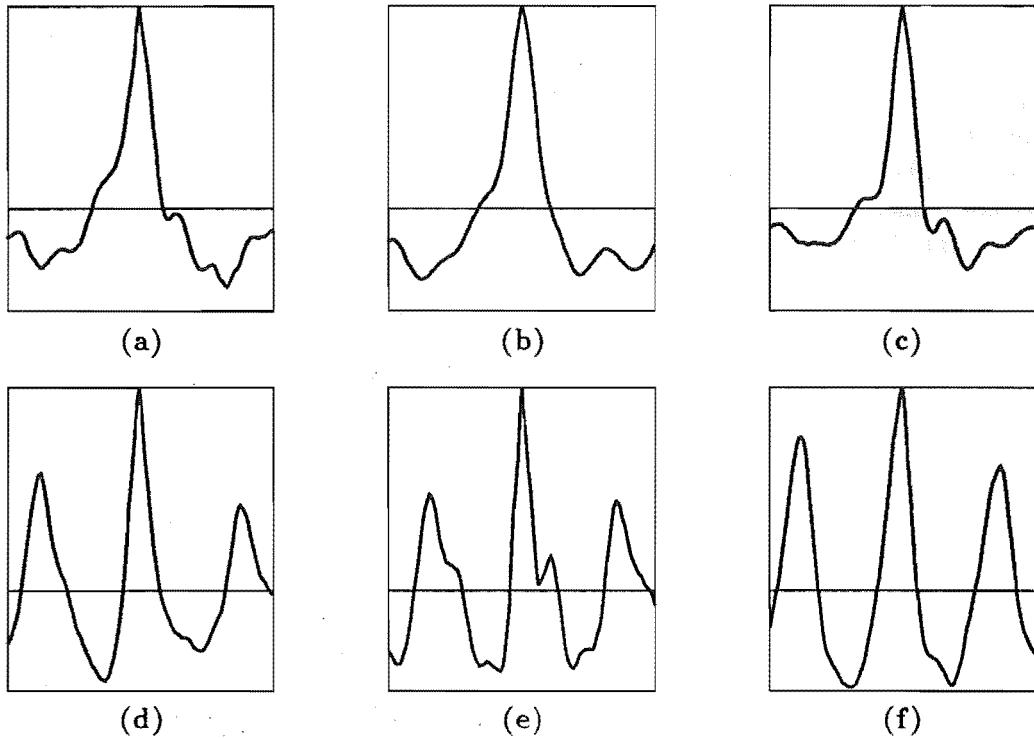
where  $< \cdot >_m$  denotes an average over  $m$ . In general,  $T_m$  is not the same as  $T_{em}$  because  $T_{em}$  is the instant that the glottal pulse is at its maximum and one would expect the instant of the maximum amplitude in the speech signal to differ because of the effect of the vocal tract filter.

The relationship between  $s_{saa}(t)$  and  $g_m(t)$  depends upon the variations in  $g(t)$  and  $v_m(t)$  recorded in  $s_m(t)$ . Brieseman *et al.* (1987), in a more rigorous formulation of SAA, represent  $g_m(t)$  and  $v_m(t)$  in terms of contributions that persist throughout an utterance and variable parts. SAA assumes that the vocal tract varies significantly (quasi randomly) and that a major component of the glottal pulse can be considered invariant under different vocal tract conditions (Thorpe, 1990, §4.2.2.1). Ananthapadmanabha and Fant (1982) show that the shape of the glottal pulse is not constant, since variations in the first formant cause slight variation in the skewness of the glottal flow. Kiozumi *et al.* (1985) extend the analysis of the glottal flow across all the formants and find that higher order formants also affect the ‘skewness’ of the glottal flow pulses, but the first formant has the dominant effect. Although the glottal flow is shown to be skewed by varying amounts, depending upon the formants, the variation in the overall shape of the glottal flow is quite small. The variation of the  $v_m(t)$  is therefore considered to dominate variations recorded in  $s_m(t)$ . However, the average vocal tract response is not an impulse, so it is not averaged out of the convolution expressed in (2.106). The signal  $s_{saa}(t)$  therefore represents the average of the  $g_m(t)$  convolved with the average of the vocal tract response (Davey and Thorpe, 1987) and for this reason is called the long-term average glottal response (LTAGR) (Brieseman *et al.*, 1987).

Thorpe (1990, §4.3.3) describes a simulation where vocal tract filters are excited by a fixed ‘glottal pulse’ and demonstrates that SAA processing does indeed recover an estimate of the ‘glottal pulse’ convolved with the average response of the vocal tract filter.

### 2.8.3 Computation details of the SAA algorithm

The SAA algorithm, while not being computationally intensive, requires that voiced speech be identified from the rest of the speech record. This voiced/unvoiced decision can be most easily implemented using one of the algorithms described in §2.4.4.

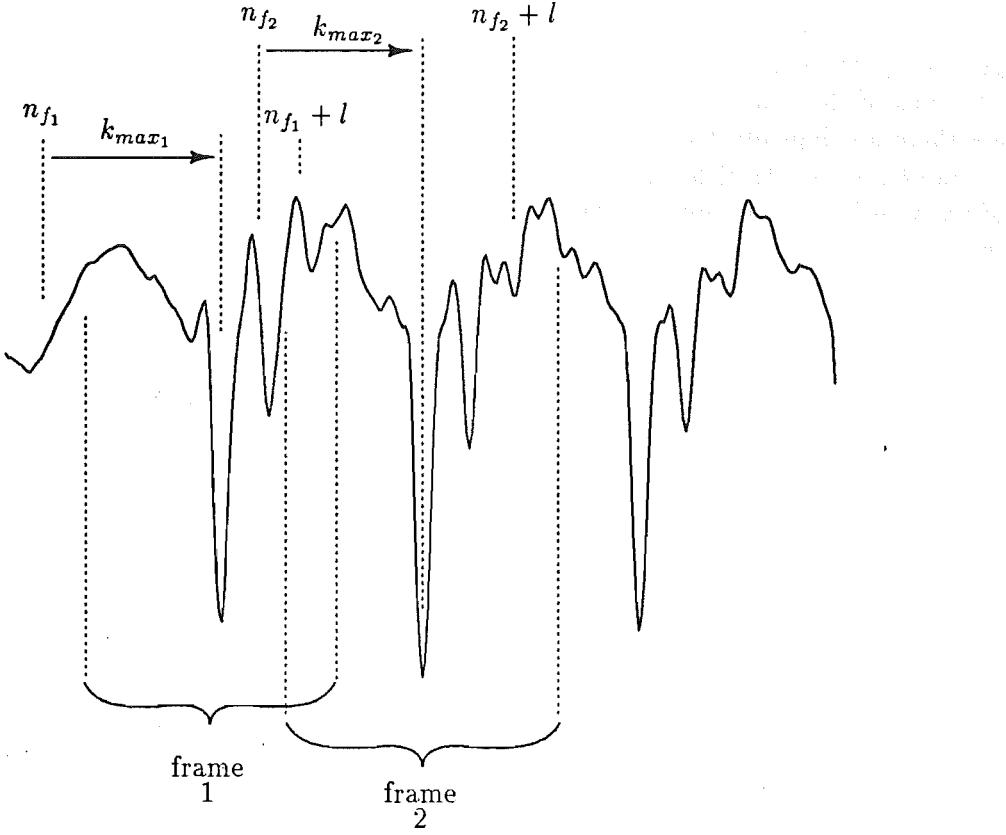


**Figure 2.20.** LTAGR signals from the phrase “*When sunlight strikes raindrops in the air, they act like a prism and form a rainbow*”. Speakers and sex: (a) AE, male, (b) WT, male, (c) BM, male, (d) DR, female, (e) KG, female and (f) TC, female.

The SAA of the voiced portion of the phrase “*When sunlight strikes raindrops in the air, they act like a prism and form a rainbow*” is shown in Fig. 2.20 for six different speakers. For ease of display the maximum amplitudes of the LTAGR signals are normalized to unity, and this convention is followed throughout the thesis. The horizontal line running through the centre of the plots represents the d.c. level. Furthermore, unless specifically mentioned, the duration of  $s_{saa}(t)$  is 12.8ms in all the examples presented here. This duration has been found to be acceptable for both male and female speech (Thorpe, 1990). The original phrases were recorded at a sampling rate of 10kHz, so 12.8ms corresponds to 128 samples. All LTAGR signals in this thesis are derived from speech signals that have been sampled at 10kHz.

The SAA computation, as posed in (2.107), indicates that the speech signal should be divided into segments, each containing a single pitch period, before the actual SAA averaging occurs. These two steps can be combined to give a more computationally efficient algorithm. This is defined in Algorithm 2.3 and illustrated by Fig. 2.21.

Variables introduced in Algorithm 2.3 are: a sequence of samples of voiced speech,  $s_v[n]$ , the position (in samples) of the start of a speech frame,  $n_f$ , the number of samples in the SAA frame,  $l$ , the amount that the SAA frame is moved each iteration of the SAA algorithm,  $n_s$ , the position of the maximum absolute amplitude within the SAA frame,  $k_{max}$ , and a threshold used to test whether a frame is of suitable amplitude to be added to the SAA signal,  $s_{thres}$ . The significance of  $s_{thres}$  is explained later in this section.



**Figure 2.21.** An example of SAA processing showing how frames are selected. The subscripts on the  $k_{max}$  and  $n_f$  variables indicate the iteration number of algorithm 2.3.

### Algorithm 2.3

Step 1: Initialization: Set the frame index  $n_f$  to 0. Set  $l$  to be the length of each frame in samples and let  $n_s$  be the number of samples that the frame is moved after each processing operation. Set  $s_{thres}$  to a reasonable level (typically 10% of the maximum amplitude) and zero  $s_{saa}[n]$ .

Step 2: Shift: Find

$$k_{max} = \arg \max_k |s_v[n_f + k]|; k = 1, \dots, l \quad (2.109)$$

and record the index  $k_{max}$  of the maximum value.

Step 3: Add: If  $|s_v[n_f + k_{max}]| > s_{thres}$  then  
add a normalized version of the frame centred on  $s_v[n_f + k_{max}]$  to  $s_{saa}$ , viz

$$s_{saa}[n] = s_{saa}[n] + s_v[n_f + k_{max} + n]/s_v[n_f + k_{max}]; n = -l/2, \dots, l/2. \quad (2.110)$$

Step 4: Next frame: Calculate the next frame position

$$n_f = n_f + k_{max} - l/2 + n_s. \quad (2.111)$$

Step 5: Termination: If  $n_f$  is past the end of the speech record  $s_v[n]$  then halt with  $s_{saa}[n]$ .

Observe in Fig. 2.20 that the males ((a)-(c)) and females ((d)-(f)) have different SAAs. This is because females tend to have a shorter pitch period than males, which

often causes the selected speech frame  $s_v[n_f + k_{max}]$  ( $-l/2 < n < l/2$ ) to contain the previous peak in the speech signal, the current peak, and also the next peak. The exact positions of the peaks to the side of the central peak in the LTAGR of a female voice are therefore dependent on the average pitch of the utterance.

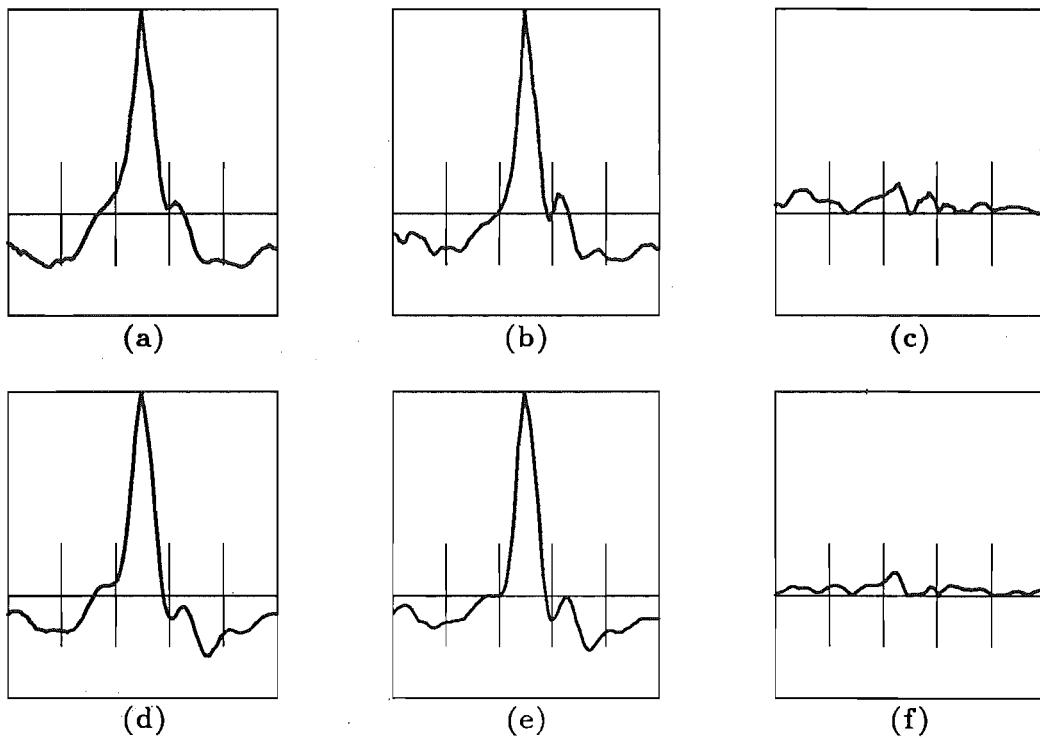
In (2.110) of Algorithm 2.3 each frame of speech is normalized to a maximum value of unity before being added to the SAA signal. Normalization makes the assumption that the glottal response is equally well recorded in both quiet and loud portions of the speech utterance. However, the alternative is not to normalize at all and instead to add the sample values of the located SAA frame to  $s_{saa}[n]$ . The main difference between LTAGRs from the two methods is that the normalized addition results in a peak that is slightly broader than that produced from SAA incorporating non-normalized frame addition. From Fig. 2.22(c) and (f) it is apparent that the largest difference is in the rising edge of the central LTAGR peak. The normalization of the peak of each LTAGR frame causes lower amplitude, broader, peaks to contribute significantly to the final LTAGR, whereas if each SAA frame is not normalized, the major contribution to the final LTAGR is from large amplitude peaks. Although the LTAGRs are taken from only two speakers, experimental experience shows that the aforementioned observations hold generally. The principal justification for normalizing is that various vocal tract shapes (or filters) attenuate the glottal pulse differently depending on the filter damping. Therefore, each of the frames should be normalized to prevent vocally ‘loud’ sounds dominating the SAA. As an aside, it is worthwhile noting that if each frame is normalized to unity before addition to the SAA signal, the maximum amplitude of  $s_{saa}[n]$  represents the number of SAA frames accumulated.

The threshold  $s_{thres}$  is incorporated into Algorithm 2.3 to remove a particular error that sometimes occurs when a frame with a small maximum amplitude is normalized. In the artificial situation depicted in Fig. 2.23, the original speech frame, starting at  $s_v[n_f]$ , has a small maximum value, but the shifted frame extends to  $s_v[n_f + k_{max} + l/2]$  and encompasses a significantly larger ‘false’ peak near the edge of the frame. Thus, normalizing the small value  $s_v[n_f + k_{max}]$  to unity causes the false peak to make a significant, and sometimes completely dominant, contribution to  $s_{saa}[n]$ . Such extraneous frames occur when the shifted frame  $s_v[n_f + k_{max} + n]$  ( $-l/2 < n < l/2$ ), labelled SAA frame in Fig. 2.23, does not have its maximum value at the centre, but can be excluded from the SAA summation by setting a threshold that  $|s_v[n_f + k_{max}]|$  must exceed before the frame is added to  $s_{saa}[n]$ . A threshold level of 10% of the maximum amplitude has been found to be satisfactory (Thorpe, 1990).

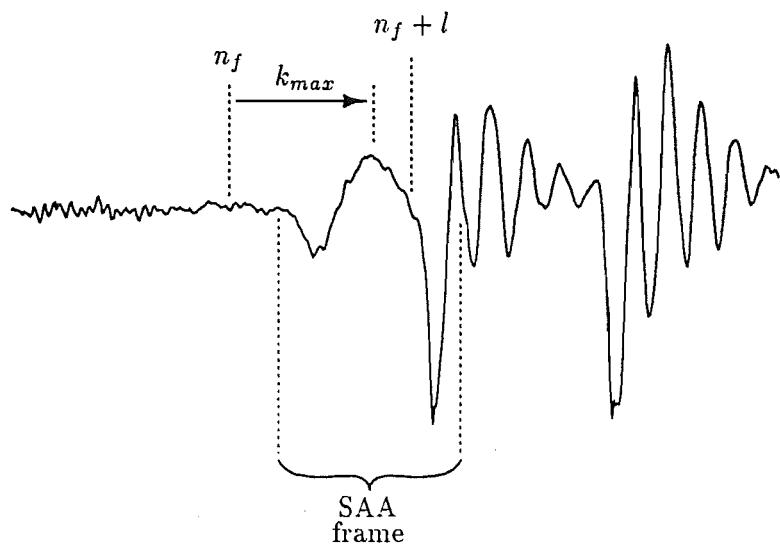
#### 2.8.4 The effect of pre-emphasis on the LTAGR

Thorpe (1990, §4.2.4.7) discusses the application of SAA to pre-emphasized speech and points out that pre-emphasis flattens the speech spectrum and causes each glottal ‘response’ in the speech signal to be more nearly impulsive, thereby reducing the likelihood of the SAA algorithm selecting erroneous peaks. Fig. 2.24 shows the effect of pre-emphasis on the LTAGR signal. The most obvious difference between the pre-emphasized LTAGR and a normal SAA is the ‘peakiness’ of the central peak. The narrowing of the central peak of the LTAGR is caused by the increased ‘impulsiveness’ of the speech signal.

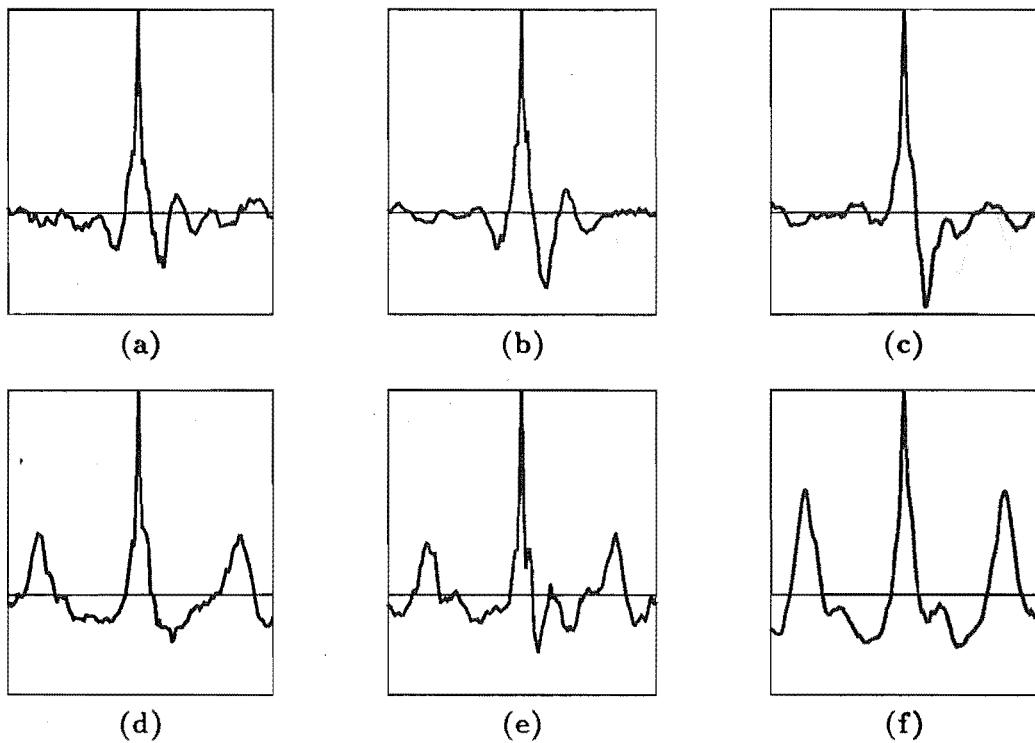
The voiced/unvoiced decision is more crucial when performing SAA on pre-emphasized speech since unvoiced sounds have most of their energy above 2 kHz and their amplitude relative to the voiced sounds is increased by pre-emphasis. Any pre-emphasized unvoiced speech incorporated into the SAA processing will therefore have a greater affect on the final LTAGR signal.



**Figure 2.22.** An example of the effect of normalizing on the LTAGR of voiced speech: (a) SAA where each frame is normalized before being added for speaker AE, (b) SAA where each frame is not normalized for speaker AE, (c)  $|(a) - (b)|$ , (d) SAA where each frame is normalized before being added for speaker BM, (e) SAA where each frame is not normalized for speaker BM, (f)  $|(d) - (e)|$ .



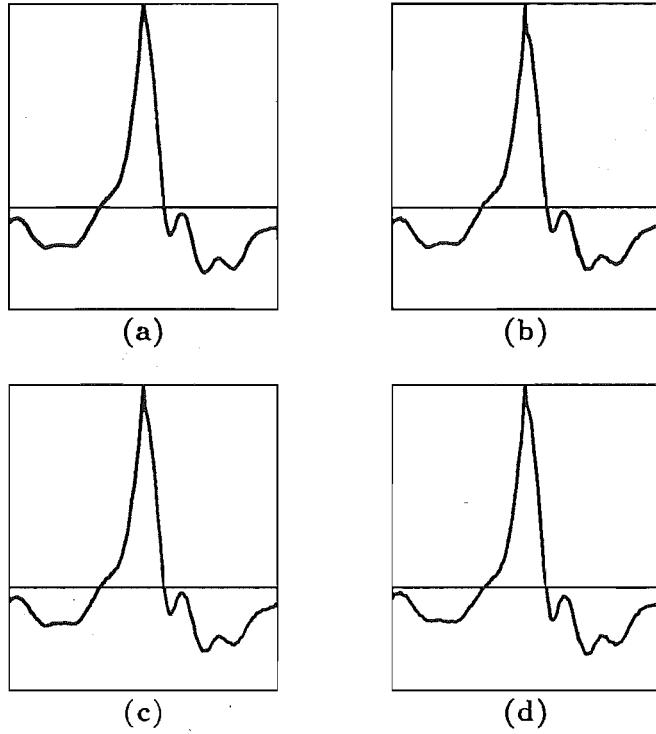
**Figure 2.23.** An artificial example of the selection of a SAA frame which has its major peak incorrectly positioned. Normalization of this type of frame within the SAA algorithm amplifies the off center peak, resulting in an erroneous LTAGR.



**Figure 2.24.** An example of the effect of pre-emphasis on the LTAGR of speakers for the same speakers and utterances as Fig. 2.20.

### 2.8.5 Incorporation of the voicing decision into SAA

A good estimate of the long-term average glottal response can be obtained only if the frames selected for addition to the accumulated SAA signal contain information about the glottal excitation. In the above description it is assumed that only voiced frames of speech are searched for instances at which the excitation of the vocal tract filter is most strongly present. Since speech contains voiced, unvoiced and silent periods, the voiced portions must be identified within the utterance. This voiced/unvoiced decision can be performed in a computationally efficient manner by comparing the maximum amplitude within each search frame against a predefined threshold  $\nu$ . If the maximum amplitude is greater than the threshold, the frame is assumed to be voiced, otherwise it is assumed to be unvoiced and is discarded. This simplified voiced/unvoiced decision is computationally efficient compared with other voiced/unvoiced schemes (see §2.4.4) and although certain unvoiced frames, such as those containing plosive sounds, are occasionally included in the SAA calculation, overall they have only a small effect on the final long-term average glottal response. Fig. 2.25 shows the effect of varying  $\nu$  between 0% and 25% of the maximum amplitude for the utterance “*When sunlight strikes raindrops in the air, they act like a prism and form a rainbow*”. It also shows the LTAGR obtained when using an accurate voiced/unvoiced decision algorithm that compares energies in different bands of the speech spectrum (VUV1 defined in §2.4.4). The LTAGRs depicted in Fig. 2.25 are all normalized to have a value of unity at the central peak. Their similarity indicates that the shape of the LTAGR is relatively independent of the threshold level chosen for the voiced/unvoiced decision. The central peaks of Fig. 2.25(b), (c) and (d) are more pointed than that of Fig. 2.25(a) due to unvoiced ‘spikes’ being occasionally included in the SAA accumulation. Comparing Fig. 2.25(a), (b), (c) and (d), one can observe that the amplitude of the spike is reduced



**Figure 2.25.** A comparison between LTAGRs obtained from the rainbow passage using different thresholds for performing the voiced/unvoiced decision: (a) LTAGR obtained from voiced speech only, (b) LTAGR obtained from analysing the entire utterance with threshold  $\nu = 0$ , (c)  $\nu = 0.10$ , (d)  $\nu = 0.25$ .

by having a larger voiced threshold level. Although Fig. 2.25 is obtained from applying SAA to a single utterance and a single speaker, experimental experience indicates that similar results are obtained for a wide range of speakers and utterances.

### 2.8.6 Relationship to the source filter model

This section explores the relationship of the LTAGR to the glottal excitation used to produce synthetic speech. Synthetic speech is used so that both the excitation and vocal tract filter are known.

Assuming that  $g(t)$  is a single glottal excitation and  $v_m(t)$  is the impulse response of the vocal tract filter for pitch period  $m$ , the synthetic speech for pitch period  $m$  is written (cf §2.8.1)

$$s_m(t) = g(t) \odot v_m(t), \quad (2.112)$$

and since convolution is associative,

$$\langle s_m(t) \rangle_m = g(t) \odot \langle v_m(t) \rangle_m. \quad (2.113)$$

Note that  $\langle s_m(t) \rangle_m$  is equivalent to  $\langle \text{sgn}(s_m(T_m))s_m(t + T_m) \rangle_m$  if  $T_m = 0$ , i.e., the position of the maximum peak in frame  $m$  of the synthetic speech is positioned at  $t = 0$ . Examples of  $\langle s_m(t) \rangle_m$  and  $s_{saa}(t)$  are presented here for synthetic speech.

The LPC filters used for the synthetic speech are computed from voiced, pre-emphasized segments of the utterance “When sunlight strikes raindrops in the air, they act like a prism and form a rainbow”. These filters are excited, to produce two

utterances of synthetic speech, using a sawtooth excitation for the first and a sine-squared excitation for the second. Fig. 2.26(a) and (b) show that  $g(t) \odot \langle v_m(t) \rangle_m$  corresponds closely to  $s_{saa}(t)$ . This correspondence supports the claim that the long-term average glottal response approximates the glottal excitation convolved with the average of the vocal tract impulse response. However, it is also obvious that  $s_{saa}(t)$  and  $\langle s_m(t) \rangle_m$  are not identical. The difference occurs because the vocal tract filter ( $v_m(t)$ ) alters the position of the maximum peak in the speech compared with the excitation. Fig. 2.27 shows the values of  $T_m$  used in the SAA algorithm to compute  $s_{saa}(t)$ . Each of the bars in Fig. 2.27(c) corresponds to the value of  $T_m$  for a particular segment,  $s_m(t)$ , of the synthetic speech. Although many of the  $T_m$  are zero, there are a significant number of non-zero values. The non-zero values indicate that the position of the maximum of the model glottal excitation is not the same as the maximum of the synthetic speech for speech segment  $s_m(t)$ . However, comparison between  $s_{saa}(t)$  and  $\langle s_m(t) \rangle$  in Fig. 2.26(a) and (b) shows they are still of similar shape.

The average vocal tract filter response is impulse like, so the main contribution to  $s_{saa}(t)$  is from the glottal excitation. Therefore, provided speech utterances contain a wide range of different sounds, the average vocal tract response can be expected to be impulse like, and the LTAGR will mainly contain contributions from the glottal excitation.

## 2.9 MEASURES OF SPEECH NOISE

Speech signals are always corrupted by noise, so it is important to assess the effects of noise on speaker identification accuracy. Here two separate noise measures are discussed that are computationally straightforward and can be used to generate speech with a defined amount of noise.

### 2.9.1 Signal-to-noise ratio

The simplest measure of the noise in a speech signal is the ratio of the average signal power to the average noise power. For a sampled signal  $s[n]$  with noise  $e[n]$  added, the signal-to-noise ratio is defined as

$$\text{SNR} = \frac{\sum_n s[n]^2}{\sum_n e[n]^2}. \quad (2.114)$$

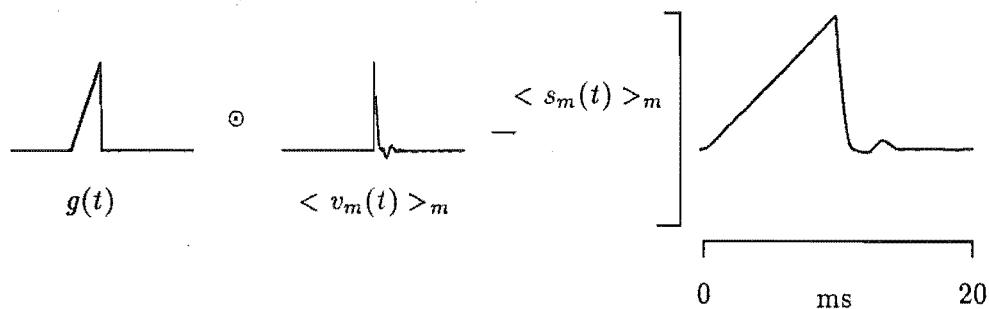
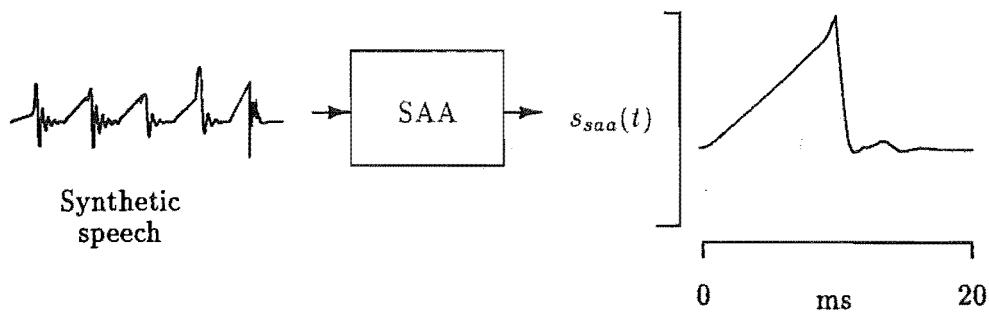
It is quite common to express the ratio in decibels (Stremler, 1982), viz

$$\text{SNR(dB)} = 10 \log_{10} \frac{\sum_n s[n]^2}{\sum_n e[n]^2}. \quad (2.115)$$

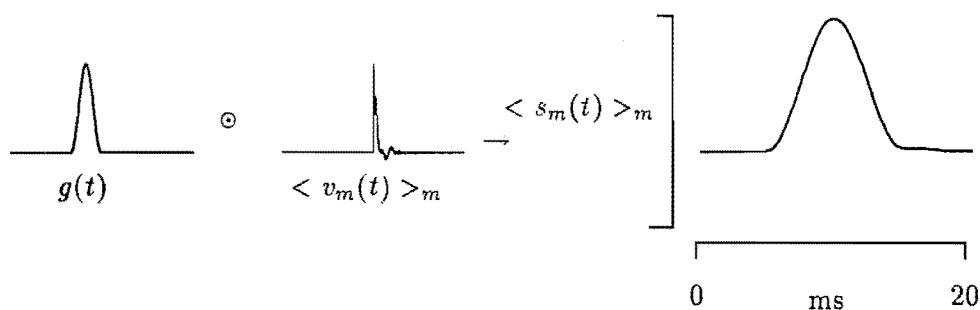
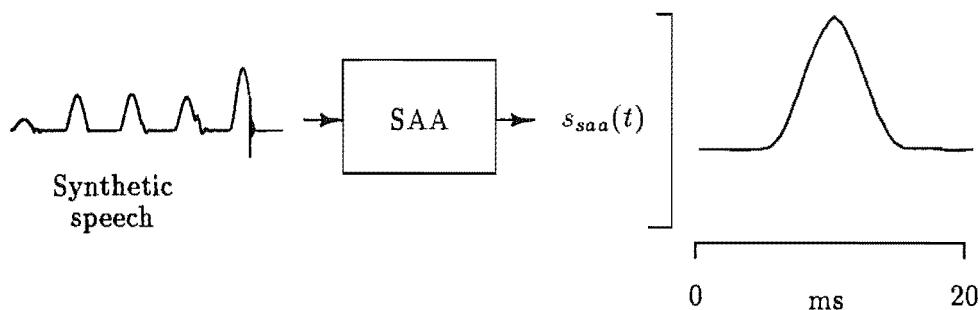
By generating noise  $e[n]$  of an appropriate amplitude and adding it to the speech signal  $s[n]$ , a speech signal with a known SNR is obtained.

While the SNR is useful for measuring noise levels, it is of limited usefulness for assessing the degradation of speech signals that have been coded in some manner. For example, two signals which are indistinguishable to the ear are,  $s(t)$  and  $-s(t)$ , but the difference between the two signals is  $2s(t)$  so compared with  $s(t)$ ,  $-s(t)$  has a signal-to-noise ratio of  $-6\text{dB}$ . This discrepancy is due to the ear's insensitivity to certain types of phase distortion (Schroeder, 1975).

The perceived degradation in speech quality is the most important quality measure. Consequently, it is best to use a noise measure that is directly related to the perceived speech quality.

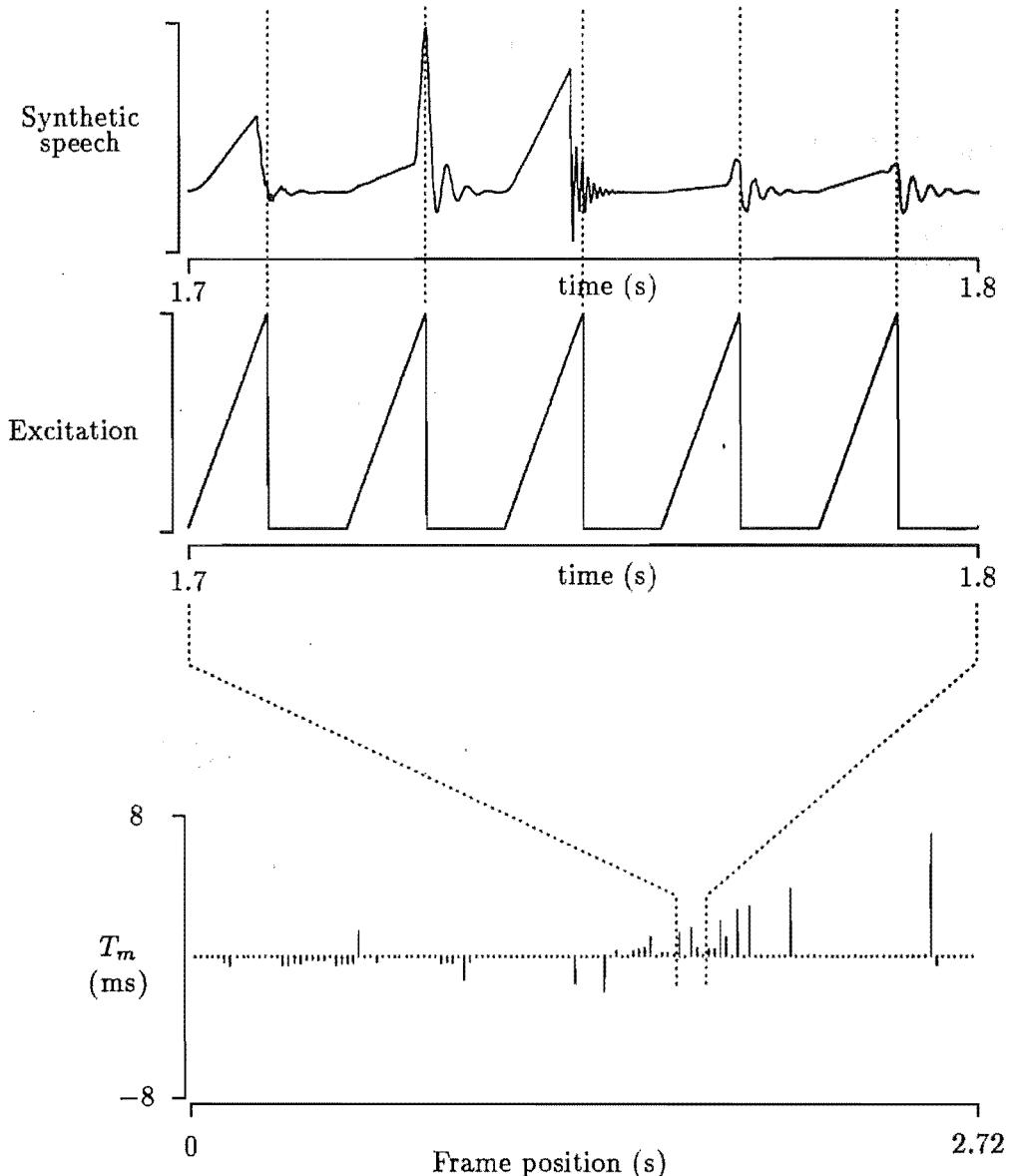


(a)



(b)

**Figure 2.26.** A comparison of the LTAGR obtained from synthetic speech,  $s_{saa}(t)$ , with a synthetic glottal pulse convolved with the average of the vocal tract filter responses,  $\langle s_m(t) \rangle_m$ . Synthetic speech generated from: (a) sawtooth excitation and (b) sine-squared excitations are depicted.



**Figure 2.27.** Details of the values of  $T_m$  used in the SAA algorithm. The five speech and excitation frames illustrate the way in which the positions of the maximum peaks in the speech signal are altered when the excitation is used to excite vocal tract filters.

### 2.9.2 Speech correlated noise

Speech degradation can also be determined by subjectively comparing the test speech signal with a ‘standard’ signal of high quality that has been contaminated with various amounts of noise. The method of generating the contamination is critical since the ‘standard’ signal should be perceptually similar to the contaminated signal under evaluation (Schroeder, 1968). One method of contaminating the ‘standard’ signal is to add a noise signal that is correlated with the amplitude of the speech signal. The contaminated signal  $r[n]$  is called a reference signal and is defined by

$$r[n] = s[n] + k(s[n]n_0[n]) \quad (2.116)$$

where  $s[n]$  is the ‘standard’ signal,  $k$  is a positive constant representing the amount of degradation and  $n_0[n]$  is a noise source of the type specified by Schroeder (1968) and IEEE (1969). Equation (2.116) forms the basis for the Modulated Noise Reference

Unit (MNRU) as specified by CCITT (Kitawaki and Nagabuchi, 1988) for adding speech correlated noise to a speech signal. A number of reference signals, with a known signal-to-noise ratio, can be created by choosing different values for  $k$ , thereby contaminating  $s[n]$  with varying amounts of noise. The quality of the reference signals is described by the signal-to-speech-correlated-noise ratio which is denoted  $Q(\text{dB})$  and expressed as

$$Q(\text{dB}) = 10 \log_{10} \frac{\sum_n s[n]^2}{\sum_n (k(s[n]n_0[n]))^2}. \quad (2.117)$$

Subjective comparisons between these reference signals and the trial speech signal allow the trial signal to be ranked within the reference quality range. The matching quality is expressed as the “opinion equivalent  $Q$  (dB)” (Kitawaki and Nagabuchi, 1988). Here the  $Q(\text{dB})$  measure is used to specify a noise level that is perceptually related and can be generated in a straightforward manner.

## 2.10 SUMMARY

This chapter introduces the speech analysis and modelling techniques that are used for performing speaker identification experiments. The main points are as follows:

- The concept of the source filter model for speech production is introduced and assumptions involved in using this model are discussed. The filter portion of the source filter model is related to a vocal tract tube model and it is shown that under certain conditions these two models can be considered to be equivalent.
- Algorithms for performing pitch detection and making voiced/unvoiced decisions are described. The particular pitch and voiced/unvoiced algorithms used in experiments reported in this thesis are presented.
- Linear predictive coding is introduced and the relationship between the spectrum of the speech signal and the spectrum of the LPC filter is described.
- Vector quantization is introduced and the advantage of VQ defined. Various distortion measures are discussed and their suitability for VQ evaluated. Details are given of algorithms that are invoked to train vector quantization codebooks.
- Shift-and-add, the algorithm used in this thesis to compute the long-term average glottal response is defined. Aspects of the operation of the SAA algorithm are discussed.
- Measures of the amount of noise present in a signal are presented as background for experiments reported in Chapter 5 that evaluate the effect of noise on the speaker identification accuracy.



## CHAPTER 3

---

### SPEAKER RECOGNITION FUNDAMENTALS

Often a person answers the telephone with a simple “hello” and we are able to identify who has answered. Although “hello” is a short utterance, in many instances it contains enough speaker-specific information for us to associate a particular person with the voice. Furthermore, without any formal training at all, people seem to be able to perform this recognition task. It is this innate ability of humans to recognize speakers that points to the possibility of automatic speaker recognition systems.

Section 3.1 introduces speaker recognition terminology and reviews various aspects of speaker recognition as reported in the literature. Assumptions about the parallels between fingerprints and voiceprints are discussed in §3.1.2. Experiments reported in the literature that determine the recognition accuracies obtained by people listening to recorded voices are reported in §3.2. Since all speaker recognition schemes rely on the extraction of features to describe each speaker, §3.3 discusses the broad classes of techniques used to process features for speaker recognition. The main problem in speaker recognition is choosing a set of features which allow individuals to be readily distinguished from each other. Statistical analysis is often invoked to determine the set of features that is likely to produce the best recognition performance. §3.4 discusses several of the more commonly utilized methods of feature analysis, paying particular attention to those techniques most suited to speaker recognition tasks. The features commonly selected for speaker recognition and the recognition accuracies of various methodologies are compared in §3.5. The effect of the quality of speakers’ test utterances in speaker recognition performance is examined in §3.6. The effects of mimicry, disguise, noise and voice variation with time are all discussed. Commercial speaker verification systems that operate in real-time are reviewed in §3.7. Section 3.8 contains a summary of the main points in this chapter.

#### 3.1 INTRODUCTION

This section introduces speaker recognition in general terms. Terminology for describing speaker recognition tasks is presented in §3.1.1 and error measures that are useful for describing the inaccuracies in a speaker recognition scheme are presented. Pioneering work in the field of speaker recognition took place in the 1960s and §3.1.2 outlines some of the debate that transpired over the relationship between fingerprints and voiceprints. Finally, §3.1.3 contains a brief description of practical applications of speaker recognition.

##### 3.1.1 Terminology

Speaker recognition tasks can be subdivided into two distinct categories. One category associates an unknown voice with a single individual from a known population of speakers, and is called *speaker identification*. The other category tests the voice of a person, who has claimed a particular identity, to confirm whether the voice matches

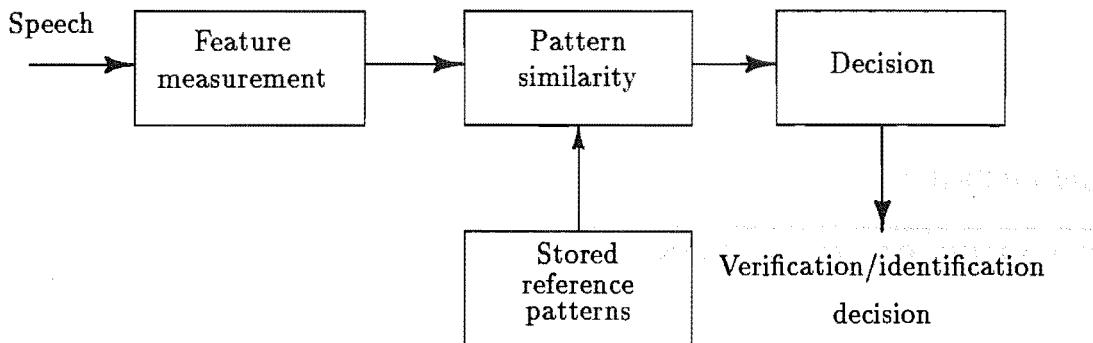


Figure 3.1. A general speaker recognition system.

the claimed identity. This is called *speaker verification* since the goal is to verify, or not, the speaker's claimed identity.

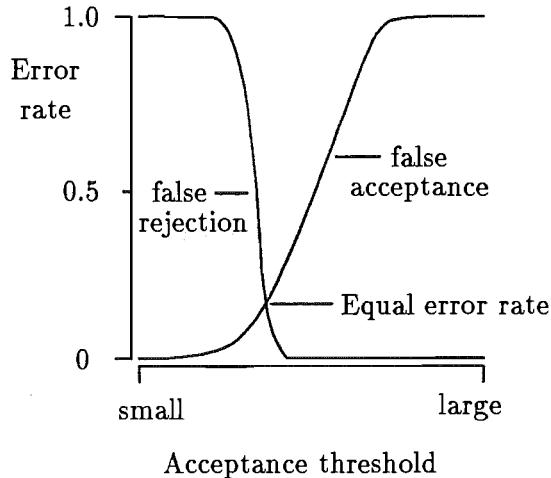
Speaker verification is predicated on an identity having already been claimed through an independent method, such as a personal identification number (PIN) entered via a keypad. The system can then check a person's 'voiceprint' against that stored for the entered PIN number to verify the identity of the person. If the person's claimed identity is verified, the claim is said to be *accepted*, otherwise it is *rejected*.

Aspects of the recognition procedure differ between speaker verification and speaker identification tasks. Speaker identification requires the test utterance to be compared with every template recorded in the reference database, and the closest match is then chosen as the identified match. Therefore, as the number of speakers increases the probability of incorrectly identifying a speaker increases. However, in a speaker verification task the test utterance is only compared with the template corresponding to the claimed identity, so the probability of error remains constant as the population of approved users is increased.

Speaker recognition systems can be further classified according to the way in which the spoken phrases are selected. When the recognition phrase is specified, and unchanging, the recognition system is said to be *text-dependent*. Instead of defining the text for every utterance to be used in the speaker recognition system, some recognition systems perform recognition on any test utterance. Such systems are called *text-independent*. An advantage of a text-independent system is that the user does not have to memorize, and later recite, a particular phrase. However, for this type of recognition the uttered phrase must be of sufficient duration to ensure that a phonetically balanced representation of the speaker's characteristics are obtained. These characteristics are usually described by long-term statistical features obtained from an utterance that is of 20-40 s duration.

The process of extracting speaker-specific information to store in a speaker recognition system is called *training*. Utterances that are used to train the speaker recognition system are called *training utterances*, while those used for testing purposes are called *test utterances*. The speaker-specific information, extracted during the training phase, is stored in *templates*, a single template representing speaker-specific information for one individual. Fig. 3.1 depicts a general recognition system. The trial utterance is input to the recognition system and features which constitute the test template are extracted from it. The next stage in the recognition task is to compare the test template with all of the prestored reference templates. The specific method of comparison invoked depends on the set of features being utilized and the final recognition result is defined by a decision rule, which specifies the method for selecting the closest match.

The number of template comparisons required for the identification and verification tasks governs the computation effort required for each of these recognition tasks. Since



**Figure 3.2.** A sketch of the dependence of the false acceptance and false rejection error probabilities on the acceptance threshold.

speaker verification requires test template comparisons against only a single template, the computation time is not increased when additional speakers are incorporated into the speaker database. However, when a speaker is added to a speaker identification system database, the recognition time increases by the time taken to compute a distance measure between the test utterance and a single template. From a computational and accuracy standpoint, speaker verification is more feasible than speaker identification when large numbers of speakers are involved.

The types of error that occur in speaker recognition depend upon the recognition task being performed. In speaker identification an error occurs when an individual is incorrectly recognized, and here this is called *false identification*. Experimental results are often reported in terms of the *error rate*, which is the total number of false identifications divided by the total number of identification trials. In speaker verification two distinctly separate errors can occur. A *false rejection* occurs when an approved speaker is rejected by the recognition system, while a *false acceptance* occurs when an imposter assumes an approved identity and is accepted by the verification system. Some researchers refer to the false rejection and false acceptance errors as type I and type II errors respectively (Naik, 1990). The speaker verification decision is performed as follows. The distance between a test template and the reference template of the claimed speaker is compared with a threshold, and if the distance is less than the threshold the speaker is accepted, otherwise the speaker is rejected. The probability of false rejection or false acceptance is a function of the acceptance threshold, as the sketch in Fig. 3.2 of the variation of the probability of error against threshold value shows. The probability of error at which the two error curves intersect is called the *equal error rate* (EER). The acceptance threshold which corresponds to the equal error rate is not necessarily the best threshold to use for verification since the sum of the false acceptance and false rejection rates may not be minimized at that threshold.

### 3.1.2 Fingerprints vs. voiceprints

In 1962, Kersta published an article entitled “voiceprint identification” which implied that voiceprints could be used to identify people in much the same way as fingerprints. This prompted many experiments and considerable debate over the reliability of voiceprints for identification and for legal purposes (Bolt *et al.*, 1970). A summary of the arguments and important results is presented here.

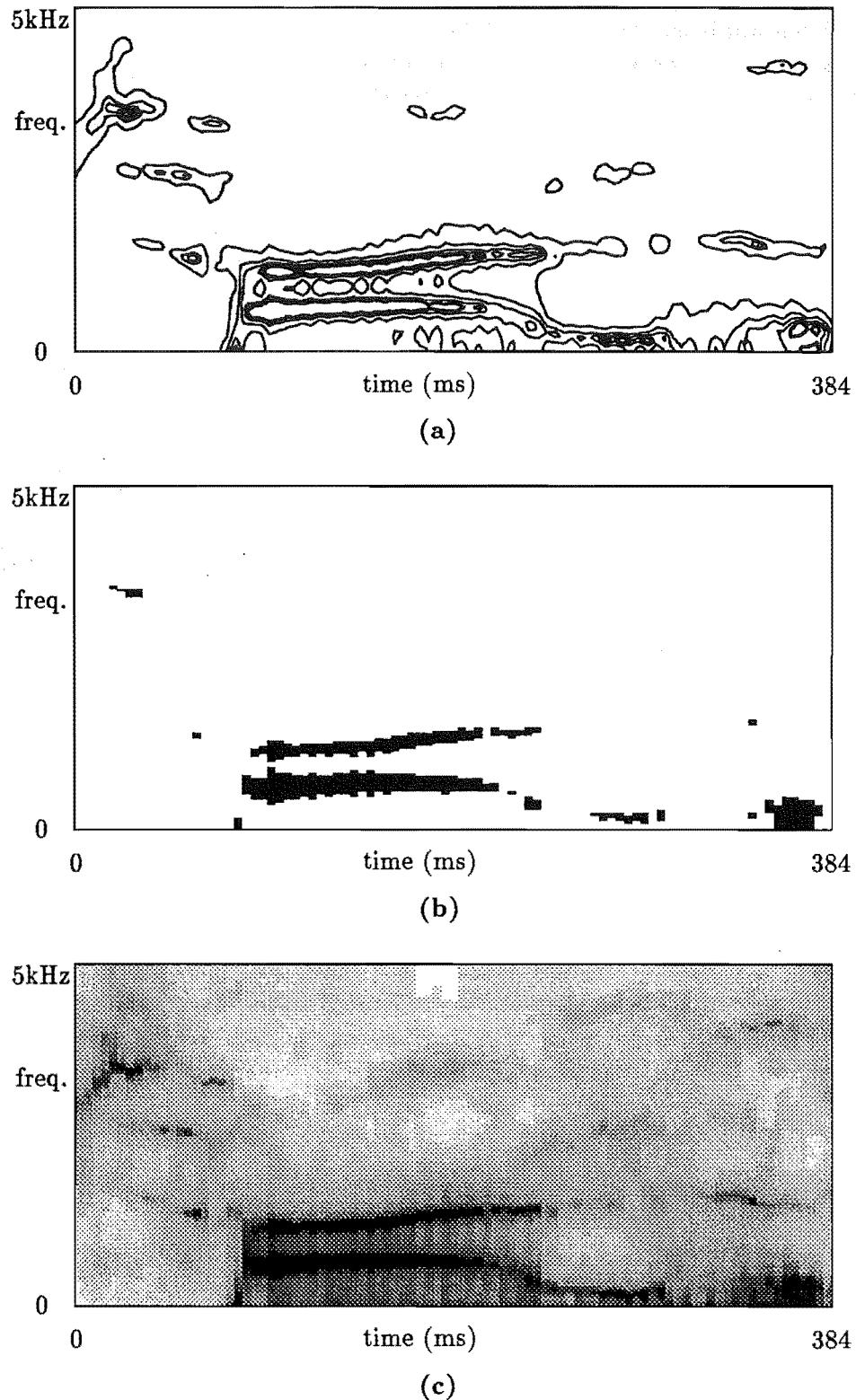
In the case of fingerprints, extensive research into their use for personal identification was performed by Francis Galton, a British geneticist and anthropologist, in 1888 (Bolt *et al.*, 1970). In 1894 the fingerprint was adopted by the British government as ultimate proof of a person's identity. Broad fingerprint classes are defined by gross ridge patterns, such as loops and whorls. These assist the filing and indexing of fingerprints, but for final identification, details such as bifurcations, terminations and interruptions are compared. Galton concluded in his study that the probability of false identification by fingerprint is approximately  $1/(64 \times 10^9)$ . Apart from the probability of incorrect identification, an important consideration for any method which attempts to identify humans is the effect of aging. Although fingerprints change size as a person ages, the relative position of features within the fingerprint have been found to remain constant. Aging does, however, affect the texture, or grain, of a fingerprint.

The rationale behind voiceprint identification is that one would expect physical characteristics of the vocal mechanism, such as the size of the vocal tract and the natural frequency (or pitch) at which the vocal cords vibrate, to vary between individuals. As well as the speech characteristics defined by a person's physical makeup, certain additional characteristics such as patterns of loudness and pitch variations are a consequence of a person's style of speaking. The learned, and individually characteristic speech style is formed within the confines of a particular speech accent. The prevalent accent during the formative years of speech acquisition (provided no accent retraining has been undertaken) is also a characteristic of an individual's voice.

Classically, the term voiceprint as defined by Kersta (1962) refers to either a spectrogram, or a contour plot of a spectrogram, as depicted in Fig. 3.3. Usually the voiceprint is taken of a single word uttered in isolation. Kersta (1962) refers to the spectrogram and contour spectrogram as bar voiceprints or contour voiceprints respectively. He points out that voiceprints display only one or two amplitude levels because they are drawn on paper as either black or white, whereas the contour representation contains more information about the spectral energy variations within speech.

Kersta (1962) reports a speaker identification experiment using voiceprints. Subjects, in this case female high school students aged 16-17, were given about 1 week's training in reading voiceprints and identifying voiceprint features. The students worked in pairs during the identification experiments because it was found that identification results were much better when consultation occurred between two students. Identification experiments utilized the words *the, to, and, me, on, is, you, i, it, and a* and each word was recorded four times by 12 speakers. Separate identification experiments using populations of 5, 9 and 12 speakers were performed using the 4 voiceprints of each utterance. It is difficult to deduce the exact experimental details from Kersta's paper, however it is apparent that the recognition experiment involved sorting the voiceprints into piles representing the individual speakers. The students may, or may not, have been told how many piles to sort the voiceprints into, or that there were the same number of utterances from each person. For the trials containing populations of 5, 9 and 12 speakers, the average accuracies were respectively 99.6%, 99.2% and 99%. The bar voiceprints were found to give higher identification accuracies than the contour voiceprints, which is in conflict with Kersta's earlier statement about the contour voiceprints containing more information than the bar voiceprints. Kersta also evaluated the effect of extracting words out of standard sentences by keeping the speaker population and test utterances constant and comparing the accuracies with those obtained from isolated words. He reports accuracies of 99.2% and 99% for the same word uttered in isolation and within a sentence respectively, indicating that utterances extracted from within a spoken sentence are slightly more difficult to recognize.

Several other researchers also evaluated the suitability of spectrograms for speaker



**Figure 3.3.** An example of: (a) a contour voiceprint, (b) a bar style voiceprint and (c) a modern gray level voiceprint for the word "the".

recognition. Since the experimental procedures differed significantly from Kersta's, each procedure is introduced briefly.

Stevens *et al.* (1968) used college students with no previous spectrogram reading experience to identify eight speakers. The utterances consisted of five repetitions of six two syllable words and three one syllable words which were recorded in two recording sessions spaced one week apart. A number of experimental sessions were conducted in which individual students were asked to match the test utterances, one at a time, to one of eight randomly selected reference spectrograms of the same utterance, spoken by each person, spread on the table in front of them. Error rates ranged from 18% for utterances consisting of many syllables to 50% for brief single syllable words.

Young and Campbell (1967) report a recognition experiment using highly trained observers, who all had training in speech pathology and audiology and were familiar with spectrographic analysis. The observers received additional training to familiarize them with the spectrograms of the speakers and the types of acoustic cues that might be useful for identification purposes. The short words *you*, *me* and *it* were used as recognition utterances. When these words were uttered in isolation the error rate was 1.6%, and in a context sentence the error rate increased to 62.7%.

It is interesting to compare Kersta's identification results with those of other researchers. The fact that Kersta's high recognition accuracies were not duplicated by other researchers highlights the effects of different experimental procedures. Kersta omits a description of his speakers and it is feasible that a cross-section of ages and sexes were used. This would introduce considerable variability between the speaker's voices, making his identification more reliable. Furthermore, the set of spectrograms employed by Kersta could assist the students in making their identifications if the number of piles (or people) to sort the spectrograms into was specified *a priori* and it was also known that there were an equal number of utterances by each individual.

It is mentioned elsewhere in this section that fingerprints remain essentially invariant with age. However, the same does not hold for voices. In a study by Endres *et al.* (1971) the variations in the voices of four males and two females over a period of 13-15 years are examined. The trend across all the subjects studied was that the average frequency of the formants decreased with age. In addition, the pitch frequency was found to decrease with age and the distribution of pitch frequencies utilized by a given speaker narrowed. Table 3.1 summarizes a comparison between fingerprints and voiceprints. It shows that voiceprints are inherently less reliable than fingerprints for identifying people. However, the convenience of voice for recognition purposes remains a strong incentive for developing voice recognition systems. Section 3.5 reports recent results that are much more accurate than those obtained by visually matching spectrograms.

### 3.1.3 Applications

Applications of automatic speaker recognition can be broadly classified according to whether speaker identification or speaker verification is performed.

The most common application of automatic speaker identification is in identifying a person from their voice in forensic applications. Although there are inaccuracies in voice identification, especially when compared with fingerprints, the ease of collection over telephone lines makes speaker identification a convenient method for identifying suspects.

Speaker verification is most often applied in a security context. A commonly cited example of such an application is telephone banking. As a precursor to a banking transaction, individuals would be required to utter a phrase which would serve as a 'key' into their accounts. If their claimed identities are verified, the banking transactions could proceed, otherwise they would be terminated. Another application of speaker

| Fingerprint patterns   | Voice patterns   |
|--|--|
| Patterns are inherent in anatomy, not changeable in kind, i.e., they cannot be changed from one pattern to another. Parts of a pattern, large or small, can only be obliterated.                         | Patterns are dependent only on anatomy and are changed only by the articulatory movements needed to realize the language code.   |
| Details of patterns:<br>(a) are permanent;<br>(b) are not affected by growth (aging merely changes the size or print grain);<br>(c) are not affected by habits (calluses merely change the print grain). | Details of patterns:<br>(a) are just as variable as the overall patterns;<br>(b) are affected by growth;<br>(c) are affected by habits (learning new dialects and voice qualities).  |
| Pattern similarity depends entirely on underlying anatomical structure.  | Pattern similarity depends primarily on acquired movement patterns used to produce language code and only partially on anatomical structure.   |
| Patterns result from a direct transfer from the skin of the finger to the surface touched by it.   | Patterns result from an analysis of voice sounds which, in turn are related only indirectly to the vocal anatomy of the speaker. Moreover, the transmission channel from speaker to spectrograph is vulnerable to acoustical and electrical distortions. |

Table 3.1. A comparison of aspects of fingerprint and voiceprint patterns (from Bolt *et al.* (1970)).

verification is in controlling access to a building for security purposes. A simple code, such as an employee number, could define the claimed speaker identity, and entry be decided by a speaker verification system. In an access system of this type it is not desirable to require individuals to utter a long phrase, since talking to a ‘door’ for 20 s or more would tend to frustrate anyone who actually wants to pass through it! In order to keep the speaker recognition accuracy high, while maintaining short utterances, text-dependence is often incorporated into such a speaker verification system.

Currently, few speaker recognition systems have been used in real life applications. Section 3.7 describes a number of practical systems that are either currently used for speaker verification or have the prerequisite hardware to perform speaker verification in a commercial environment.

One of the drawbacks of current speaker recognition systems is that they require keypad entry of a PIN number. It would be much better if the person using the system could just say “my name is John Brown” to assert their identity, thus making the complete transaction voice driven. With current technology (particularly banking) the PIN number is usually used for identity verification and a plastic bank card has information recorded on it that asserts the identity of the person using the banking facility. This is a convenient banking method and current speaker recognition machines do not offer significant advantages over these types of facilities. Until speaker recognition and speech recognition are successfully linked together, so that transactions can occur in a ‘natural speech’ environment, speaker recognition will remain somewhat of a novelty.

| Group                            | <i>N</i> | Percent Correct Identifications |
|----------------------------------|----------|---------------------------------|
| listeners who know speaker       | 10       | 98.0                            |
| listeners who don't know speaker | 47       | 39.8                            |
| foreign listeners                | 14       | 27.1                            |

Table 3.2. Speaker identification accuracy for different familiarities as determined by Hollien *et al.* (1982). Note that *N* is the number of listeners used for the experiment.

### 3.2 FACTORS THAT AFFECT RECOGNITION PERFORMANCE BY HUMAN LISTENERS

Humans demonstrate a natural ability to recognize a speaker. There has been considerable research into aspects of speaker recognition, particularly towards understanding what cues, or features, are important for recognizing people from their voices. This section summarizes aspects of human speaker recognition performance. Section 3.2.1 discusses the relationship between identification accuracy and familiarity with the person's voice. The contribution that the duration of the sample utterance makes to the identification accuracy is described in §3.2.2, while §3.2.3 discusses the importance of pitch in speaker recognition. Finally, §3.2.4 describes experiments reported in the literature to assess which voice characteristics are used by human listeners when they perform speaker identification.

#### 3.2.1 Familiarity

To recognize a speaker, the listener must have an impression or template of the speaker's voice. The more familiar the listener is with the speaker's voice the more accurately the listener remembers (or recalls) the speaker's personal speaking nuances. This is supported by experiments conducted by Hollien *et al.* (1982) and Schmidt-Nielsen and Stern (1985). Hollien *et al.* (1982) tested listeners on a population of 10 healthy male speakers who uttered phrases consisting of 50-58 words. The listeners were partitioned into three groups; (a) those who knew the speakers, (b) those who didn't know the speakers and (c) those who didn't know the speakers and didn't speak the language (foreign listeners). The experimental findings are summarized in Table 3.2. Before the experiment began 24 listeners were asked to rate their familiarity with the voices of 39 individuals on a scale of 0-6, with 0 representing 'totally unfamiliar' and 6 representing 'highly familiar'. The distinctiveness of each of the individual's voices was also rated on a scale of 0-6 by the listeners. Using speech of 29.8s average duration, recorded from a battleship game, listeners were required to identify 24 speakers and after each recognition record how confident they were of their choice. The confidence scale consisted of a 3-point scale with levels described by 'guessing', 'fairly sure' and 'very sure'. For the 0 familiarity score the recognition accuracy was 44.4%, whereas the highly familiar listeners scored 92.0% correct. The overall trend in the recognition accuracy as a function of familiarity led Schmidt-Nielsen and Stern (1985) to conclude that familiarity is highly correlated with recognition accuracy.

Schmidt-Nielsen and Stern (1985) make what appear to be contradictory claims about the relationships between speaker identification accuracy, the rated familiarity and rated distinctiveness. On the one hand, they state that the average rated distinctiveness for each speaker and the percentage of times that the speaker was correctly identified were correlated at a level of 0.40, which is statistically insignificant. On the other hand, they claim that the rated distinctiveness and rated familiarity are significantly correlated ( $r = 0.79, p < 0.01$ ) and that the percentage of times a speaker was

correctly identified was correlated with the familiarity at a level of ( $r = 0.592, p < 0.01$ ).  $p < 0.01$  implies that the probability of the two variables being uncorrelated is at the low level of 0.01, so the levels of correlation are quite significant. Comparison between the accuracies obtained for various familiarity and distinctiveness ratings show that as the familiarity rating increases from 0 to 4 the identification accuracy increases from 44.4% to 94.4%. The distinctiveness rating levels of 0 to 4 have identification accuracies of 84.4% and 92.5% respectively. From these accuracies it is apparent that the accuracy of distinctiveness level 0 is much higher than that of familiarity level 0. One problem with the distinctiveness level, as measured by Schmidt-Nielsen and Stern (1985), is that it requires prior knowledge of the person's voice, which implies some level of familiarity. It seems unreasonable, therefore, to compare results from level 0 of the familiarity scale with results from level 0 of the distinctiveness scale. It is this dependence of the distinctiveness ratings on familiarity that is overlooked by Schmidt-Nielsen and Stern (1985) and leads to inconsistencies in their results. This example highlights the importance of careful experimental procedures and a thorough understanding of relationships between the variables that are recorded.

### 3.2.2 Duration

The ability of listeners to identify a speaker also depends upon the duration of the test (or sample) utterance. Pollack *et al.* (1954) attempted to evaluate whether the duration of a speech sample affects the ability of listeners to identify speakers by performing tests with utterances containing the same amount of speaker specific information, but having different duration. Utterances of short duration, consisted of a single monosyllabic word, while utterances of long duration, consisted of the same utterance repeated three times. The exact duration of the monosyllabic word is not reported. Pollack *et al.* (1954) found no significant difference between the short and repeated utterances for speaker identification. They therefore concluded that the duration of an utterance is important only because more speaker specific information is added to the sample, not because listeners are particularly sensitive to the length of an utterance. A limitation of this study is that no attempt was made to determine the minimum duration of sample required for speaker specific information to be perceived by a listener. Bricker and Pruzansky (1966) reached a similar conclusion by way of an experiment that varied the number of syllables in the trial utterances. Again, listener's found that the duration was relatively unimportant, except that it allowed a wider sampling of the sorts of sounds that an individual utters. No additional information is extracted by a listener when utterances are replayed.

### 3.2.3 Pitch

The particular cues, or characteristics, that a listener invokes to recognize a person are difficult to isolate. Several researchers have cited pitch as being very important for voice recognition (Voiers, 1964; Murry and Singh, 1980; Atal, 1974). To investigate the effect of pitch, Van Lancker *et al.* (1985) performed an experiment in which test utterances were played backwards so as to remove phonetic information from the speech and to reverse temporal information such as loudness changes, pitch and pitch range, but leave them still recognizable. If the pitch and pitch contour are the most important features for recognizing voices, one would expect that when voices are presented in reverse the identification accuracy would be degraded uniformly across all speakers.

Van Lancker *et al.* (1985) tested this assertion in an experiment using utterances from 45 famous people and 94 listeners. They found that the degradation in identification accuracy was not uniform across all the speakers. Some of the familiar voices

were recognized nearly as well backward as forward. Interestingly, 20% of the voices were recognized better when presented in reverse, but most were recognized more accurately when presented normally (forward). This difference in identification accuracy led Van Lancker *et al.* (1985) to suggest that listeners use cues within utterances that are dependent upon the voice being listened to. In contrast with this, consistently poorer recognition accuracies for reversed speech were obtained in an earlier experiment by Bricker and Pruzansky (1966). Their sample of 10 speakers and 16 listeners was considerably smaller than the 45 speakers and 94 listeners examined by Van Lancker *et al.* (1985), so it is possible that the differences in the two sets of results can be attributed to the different sizes of their speaker and listener populations. However, if it is assumed that the sample of the total population taken by Van Lancker *et al.* (1985) is a reliable estimate of the entire population, Bricker and Pruzansky (1966) would only have an 11% chance of selecting a speaker population whose forward voice is consistently more recognisable than their backward voice. It is difficult to reconcile these two conflicting results, but the larger sample and listening population of Van Lancker *et al.* (1985) would tend to indicate that their results should be more reliable.

Although the pitch of a voice contains speaker dependent information, it is not the only speaker specific information in an utterance. This is demonstrated in an experiment by Coleman (1973) where the glottal excitation is replaced by an electro-larynx which produces a steady buzz of  $85\text{ Hz} \pm 3\text{ Hz}$ , thus removing the pitch variable from the speech completely. A total of twenty subjects, ten male and ten female, were trained in the use of the electro-larynx, and each subject then recorded four utterances. From these recordings 40 pairs of utterances were produced, 20 pairs of the same speaker, 10 pairs of a male speaker with a female speaker, 5 pairs of two different male speakers and 5 pairs of two different female speakers. Therefore, 50% of these paired utterances contained the same voice and 50% contained different voices. Twenty eight listeners were then required to decide whether two utterances (which constituted a pair) were spoken by the same person. The listeners correctly chose whether or not the utterances were spoken by the same person for 90% of the trials, indicating that the differences amongst individuals other than pitch and glottal pulse shape are readily detectable by the human ear.

Another method of speaking without generating any pitch information is to whisper. Pollack *et al.* (1954) used two separate groups comprised of four and eight male speakers to examine the identification accuracies for normal and whispered voices. They discovered that whispered utterances can be used for identification, but such utterances do not contain as much speaker specific information as 'normal' voices. For both the four and eight speaker experiments they concluded that a whispered utterance must be approximately three times the duration of a normal utterance to attain the same recognition accuracy (and, therefore, contain the same quantity of speaker specific information). From these experiments it is apparent that, although the pitch contains speaker specific information, other vocal tract information can also be used to identify speakers.

### 3.2.4 Perceptual factors

For the purpose of building an automatic system to perform speaker recognition it would be useful to estimate other perceptual factors that humans use to recognize voices. Voiers (1964) reports an experiment where 32 listeners described their perceptions of 16 voices by scoring several bipolar items (e.g. 'loud-soft') on a 7 category rating scale. A total of 49 items such as 'laboured-easy', 'low-high' and 'rich-thin' were scored in this manner. The aim was to determine what perceived auditory information (called the speaker effect) was used to establish the identity of a speaker. Listener biases and

listener idiosyncrasies constituted what was called the listener effect. Factor analysis (see §3.4.3 or Duntzman (1984, §7)) was performed on the experimental results to determine the significant factors that contributed most to the speaker effect and those that contributed most to the listener effect. For the speaker effect 4 factors were found to account for 88% of the total item variance. The first factor, which accounted for most of the variance, contained items such as 'clear-hazy', 'deliberate-careless' and 'beautiful-ugly' and was given the label 'clarity' or, perhaps, 'intelligibility' of the person's voice. The second factor was labelled 'roughness' since it contained large contributions from items such as 'rough-smooth' and 'scraping-gliding'. Voiers labelled the third and fourth speaker factors 'magnitude' and 'animation' respectively. In addition to the speaker effect, Voiers calculated the listener effect and found that 6 factors accounted for 57% of the variance. It is not immediately obvious how to apply Voier's results in an automatic system since it is difficult to relate the perceptual descriptions (for example 'happy-sad') to the sampled speech signal. However, the number of items examined, and found to be significant, does indicate that many descriptors should be extracted from the speech signal if the goal is to represent a speaker's voice characteristics.

### **3.3 CLASSES OF TECHNIQUES USEFUL FOR SPEAKER RECOGNITION**

This section describes techniques for processing parameters that characterize certain aspects of a voice. An utterance spoken by a particular individual, processed on a frame by frame basis, yields a sequence of parameters (or features) which are then processed using one of several techniques. Adopting the terminology of Furui (1981), these techniques are broadly classified into two groups; dynamic techniques and statistical techniques. Dynamic techniques, described in §3.3.1, utilize the way speech parameters change with time, while statistical techniques, described in §3.3.2, utilize long-time statistical information. Another technique, called vector quantization (VQ), (described in §2.7), can be invoked to reduce the storage requirements of speech parameters used in speaker recognition. Section 3.3.3 describes the use of VQ in comparing test and reference templates and the way in which VQ can be used to characterize an individual's voice characteristics.

#### **3.3.1 Dynamic techniques**

Dynamic techniques use features that are recorded in a temporal sequence throughout an utterance. Matching the features of a test utterance and template utterance involves computing the distance between the two time-registered sequences. However, spoken repetitions of the same phrase are typically of differing durations. The human listening mechanism and its associated recognition processing is not critically sensitive to small changes in speech duration, but computer-based recognition systems must use either duration insensitive models, or special matching procedures to reduce the effects of duration variations.

One simple method of accounting for changes in duration has been implemented by Atal (1972). He adjusted the length of the analysis frames by a constant amount throughout the utterance to ensure that each utterance was segmented into the same number of frames. Atal (1972) achieved considerable success (97% recognition accuracy) with this method when using the pitch contour to characterize a speaker's voice. Atal (1974) also utilized the same alignment method, but this time with a variety of LPC related parameters. In most instances it seems reasonable to vary the size of all the frames throughout the utterance, thus time-aligning two utterances of different duration. Although Atal varied the size of the speech frame between utterances, it

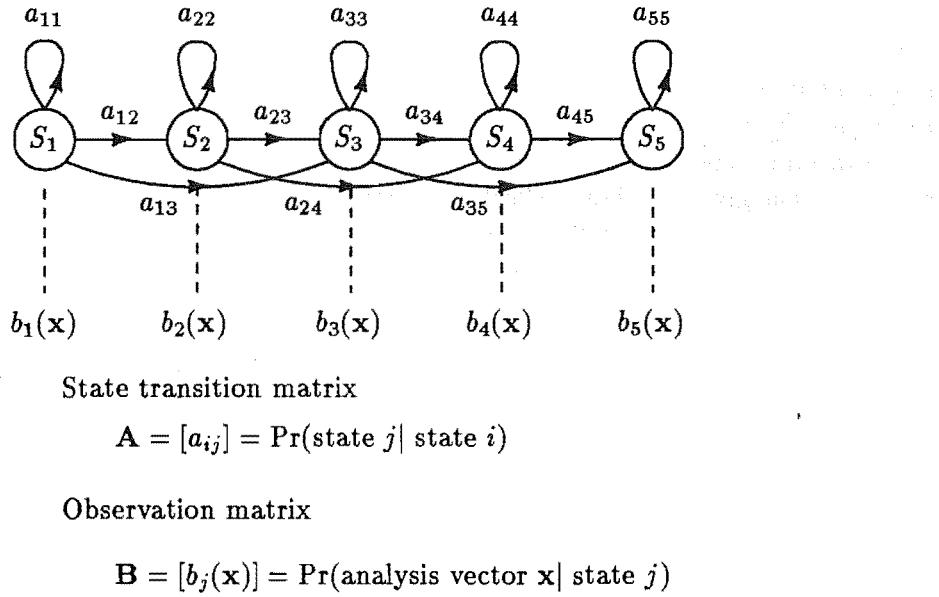


Figure 3.4. An example of a five state Hidden Markov Model.

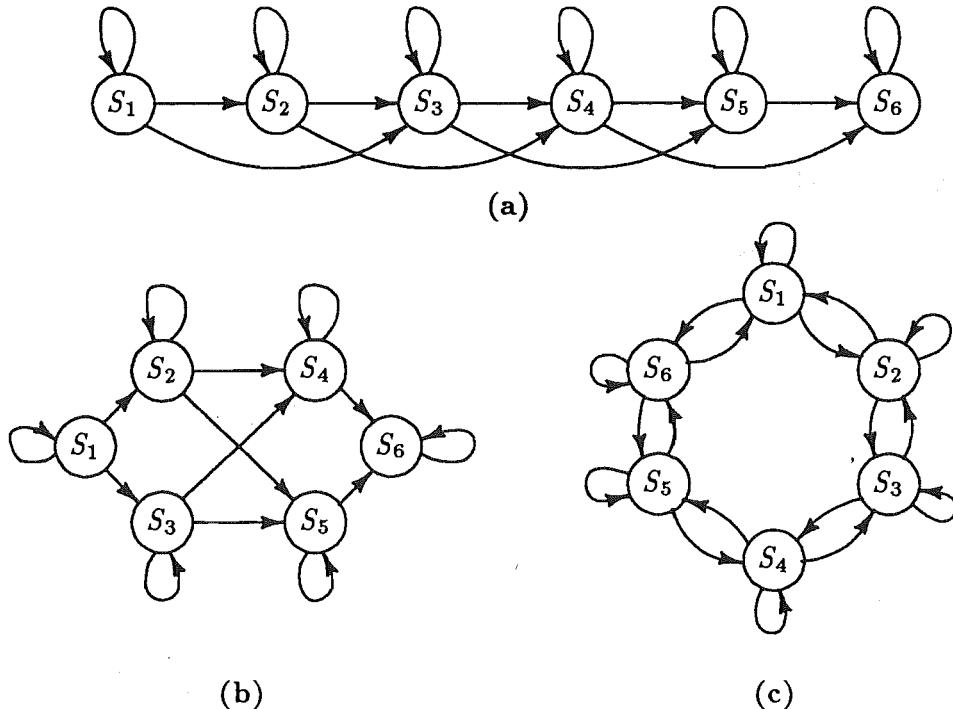
remained a constant size throughout any utterance, and can thus be considered to be a linear warping of the time scale. In an attempt to improve on the linear warping, more flexible alignment techniques and parameter representations have been introduced. One such technique represents a time ordered sequence of templates as a probabilistic model called the hidden Markov model. This model, and the corresponding matching method, is outlined in §3.3.1.1. An alternative time-alignment method for matching a sequence of speech parameters in a non-linear manner is called dynamic time warping (DTW). The principles of dynamic time warping are summarized in §3.3.1.2.

### 3.3.1.1 Hidden Markov Models

The time varying characteristics of a speech utterance can be represented by a probabilistic model such as the hidden Markov Model (HMM). The HMM models a doubly stochastic process with an underlying stochastic process that is *not* observable (it is hidden), but can be observed only through another set of stochastic processes that produce a sequence of observed symbols (Rabiner and Juang, 1986). The HMM represents the speech production mechanism as being in a finite number of states throughout an utterance. Each state is capable of producing an output which, in some HMMs, is restricted to one of a finite number of defined outputs - the output levels are quantized. An alternative to quantized outputs is to allow the output to take on any value from a continuous range (Rabiner *et al.*, 1985), but this is more complicated computationally.

Fig. 3.4 depicts a left-to-right HMM with five states. The left-to-right HMM imposes a temporal order on the HMM since states at the left occur before states at the right. Each state corresponds to a set of temporal events in the speech sound. The left-to-right constraint is important in word recognition applications and text-dependent speaker recognition since it is desirable to encode the order of acoustic events within the model.

The HMM is defined by the state transition matrix  $\mathbf{A}$  which describes the probabilities of transitioning to a particular state, given the current state, and the observation matrix  $\mathbf{B}$  which describes the probability of observing a particular set of speech characteristics for all the states (Rabiner, 1989). Training an HMM involves the determi-



**Figure 3.5.** Three different HMM structures (from Zheng and Yuang (1988)); (a) a left-to-right Markov chain, (b) a parallel Markov chain, (c) a circular Markov chain.

nation of values for the matrices **A** and **B** which best describe the observed training sequence. In discrete word recognition a single HMM represents the acoustic transitions for a single word. However, to increase recognition accuracy, speaker recognition systems typically utilize multiword phrases, making training more complicated. Multiword utterances can be modelled with HMMs by separately representing the individual words in the utterance and the silences between the words with their own HMMs.

Recognition by HMM requires computation of the probability that the observed sequence of acoustic features was generated by the model under evaluation. An entire phrase can be recognized by suitably combining probabilities for individual words within the phrase (Naik *et al.*, 1989). In a speaker identification system the person whose HMM scores the highest probability of producing the observed set of acoustic features is recognized as the speaker. Rabiner (1989) contains concise algorithm details for HMM training and recognition.

The computational requirements for training an HMM and for performing recognition differ significantly. Estimating the state transition matrix for the model which maximizes the probability of the observation sequence is a complicated problem. It can be solved only by iterative techniques, since there are no known analytical solutions (Rabiner and Juang, 1986). Typically, training involves many iterations of an algorithm called the Baum-Welch algorithm to improve estimates of both **A** and **B**. Training of the HMM is slow compared with recognition, since recognition uses the somewhat faster forward-backward procedure which, for an  $N$  state model and  $N_T$  frames in a test utterance, requires in the order of  $N^2 N_T$  operations (Rabiner and Juang, 1986).

The application of HMM to speaker recognition is a relatively recent development. Naik (1990) in his review of speaker recognition described HMM as a viable scheme for speaker verification, but did not give any recognition results. Zheng and Yuang (1988) carried out a comparative study between the identification performance of three dif-

ferent HMM configurations using 12th order LPCs vector quantized to 64 codewords. Depicted in Fig. 3.5 are the left-to-right chain, the parallel chain and the circular chain that they used in their verification experiments. They found that the identification accuracies for a population of 6 males and 4 females were 93.7%, 88.7% and 90% for the circular chain, left-to-right and parallel HMMs respectively. They concluded that the circular HMM is better suited to speaker identification than the other HMM configurations, since states can be ‘revisited’ as sounds are repeated in an utterance.

### 3.3.1.2 Dynamic time warping

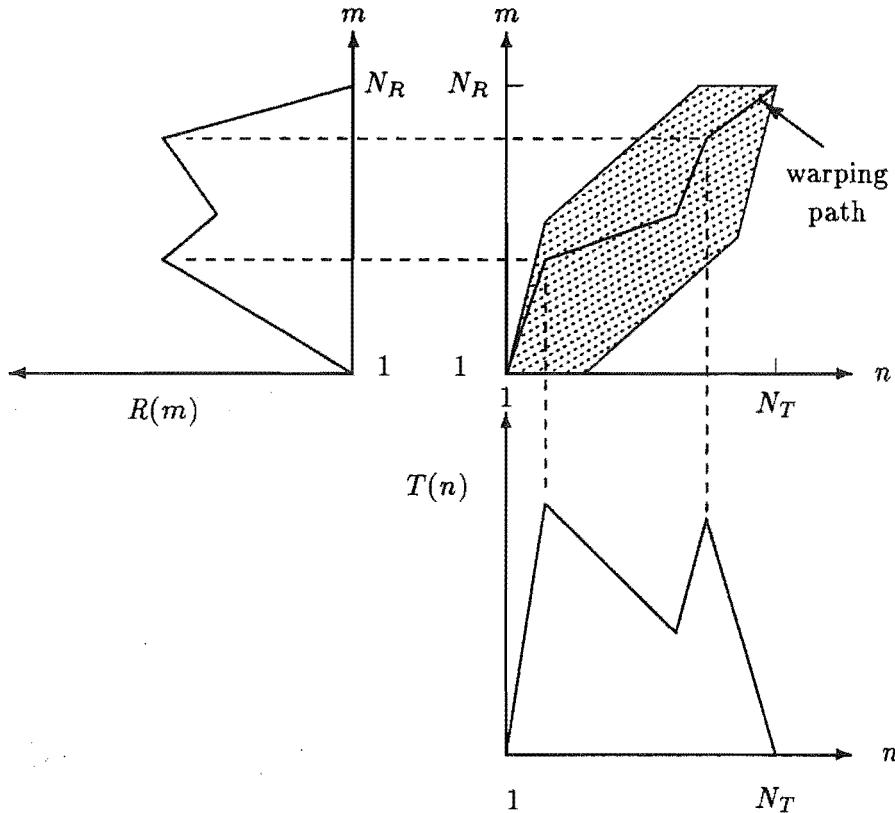
Dynamic time warping is a method of evaluating the distance between two time ordered sets of feature vectors. Vectors, such as LPCs, are matched using one of the distortion measures described in §2.7.3. The time registration is allowed to ‘compress’ or ‘expand’, causing non-linear warping of the time sequence of feature vectors. An example of a DTW match between a test and reference sequence consisting of a one-dimensional feature vector is shown in Fig. 3.6. The minimum warped distance is determined using a dynamic programming technique that ‘tracks’ the minimum distance path through a space containing distances between the sequence of template vectors and the sequence of test vectors. For the purposes of this explanation the test sequence is denoted  $T$  and the reference sequence is denoted  $R$ . The number of analysis frames in  $T$  and  $R$  is denoted by  $N_T$  and  $N_R$  respectively. DTW determines a warping function,  $m = w(n)$ , which maps the time (or frame number) axis,  $n$ , of the test template onto the time axis,  $m$ , of the reference template. To prevent the dynamic programming algorithm from producing an undesirable warping function, the warping function is restrained to lie within the bounds of the shaded area depicted in Fig. 3.6. The final warping function is derived from the solution to the following optimization problem,

$$D = \min_{w(n)} \left[ \sum_{n=1}^{N_T} d(T(n), R(w(n))) \right]. \quad (3.1)$$

One might expect that to locate the minimum warping path through the shaded region depicted in Fig. 3.6, the distances for all possible paths falling within the shaded region must be computed. However, in practice this is not the case, because the warping path can be determined by extrapolating the minimum error path from the first frame. In other words, the warping path,  $w(n)$ , is computed incrementally, with each increment being the minimum distance option from a number of possible path directions. The number of possible path directions is constrained so that only a few ‘reasonable’ directions for the warping path are examined.

The usefulness of DTW for speaker verification was demonstrated using pitch and intensity contours by Rosenberg (1976) (see also Rosenberg and Sambur (1975)). Utilizing only these two parameters, extracted from speech that was transmitted over standard telephone lines, an error rate of approximately 5% was obtained for a test population of over 100 males and females. Furui (1981) obtained verification error rates of less than 1% using DTW and cepstral coefficients computed from utterances transmitted over standard telephone lines.

The DTW example depicted in Fig. 3.6 indicates that the end points of the test and reference phrases are to be strictly aligned with each other and that warping can only occur between the endpoints. This is called constrained endpoint DTW and requires accurate location of the endpoints of the two words (or the phrases), otherwise the alignment is not valid. In practice, endpoint detection is a difficult problem (Rabiner and Sambur, 1975), however, the development of unconstrained endpoint DTW circumvents this difficulty by allowing the warping path to be broad at each end. Rabiner



**Figure 3.6.** An example of matching between a sequence of test vectors,  $T(n)$ , and a sequence of reference vectors,  $R(m)$ , using dynamic time warping.

et al. (1978) report improved recognition accuracies from this method compared with constrained endpoint DTW.

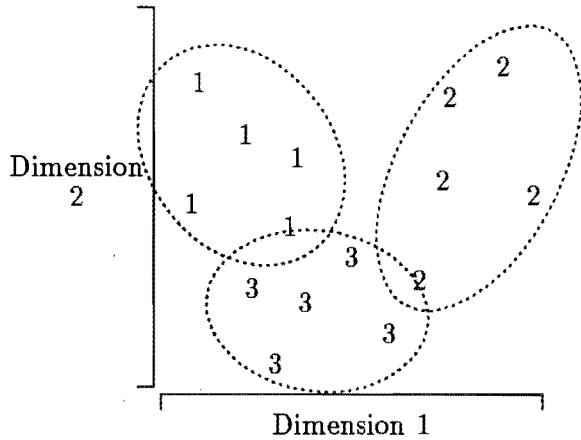
Although DTW and HMM are structured quite differently, DTW can be written in a statistical framework when LPCs are utilized to code speech and the maximum likelihood difference is invoked to measure the difference between the test and reference vectors. If the underlying state transition structure of the HMM is equiprobable, Juang (1984) shows that HMM is equivalent to performing DTW on all possible warping paths.

### 3.3.2 Statistical techniques

Statistical techniques utilize long-term characteristics of a person's voice (Furui and Itakura, 1973) and ignore the particular phonetic sequence of sounds that comprise an utterance. Examples of features having useful statistics are, a person's average pitch, the long-term average spectrum (LTAS) and the average reflection coefficients. Statistical measures, such as the mean, variance and covariance of these features, are often used to characterize a speaker and this is described in more detail in §3.4. Since there is no time registration and the statistics are computed from the entire utterance, statistical techniques are useful for text-independent speaker recognition.

Typically, many statistics are calculated for each utterance, but only a subset of them used for recognition purposes. For example, Furui (1981) extracts more than 80 statistical features from each word and selects the 'best' 20 elements for recognition. Methods for choosing the combination of statistical features that is likely to be most accurate for speaker recognition are described in §3.4.

Since the statistics are based on so-called 'long-term' averages, it is important to



**Figure 3.7.** An example of two-dimensional feature vectors for 5 utterances from 3 different speakers. This example is for illustration only, and does not correspond to acoustic features that might be extracted from an utterance.

quantify the effect utterance duration has on the stability of the features. The goal is to produce the type of separation depicted in Fig. 3.7, where the variation in the two-dimensional feature vector is considerably smaller between utterances spoken by the same speaker than between different speakers (terminology, along with exact mathematical descriptions for describing these variations is presented in §3.4). Markel *et al.* (1977) present results which examine the effect of increasing the number of speech frames, and therefore the utterance duration, on the variations of statistical features between and within four speakers. They find that the average standard deviation of the long-time average reflection coefficients decreases proportionally to  $N_v^{-1/3}$ , where  $N_v$  is the number of voiced speech frames used in the analysis and  $N_v$  ranges from 1 to 1000. In a later study in which 17 speakers were evaluated, Markel and Davis (1979) also find a significant decrease in the standard deviation of long-term features as the number of voiced frames increased. Using 1000 frames of speech they achieve recognition accuracies of 98.05% for speaker identification and an equal error rate of 4.25% for speaker verification. Although longer utterances allow a more statistically accurate representation of a person's speech, in an application it may be inconvenient for a person to speak into a recognition system for an extended period.

Computationally, statistical techniques use fewer operations than dynamic techniques, since matching is between the single vectors containing statistical features, rather than sequences of vectors. Furui (1981) found in his speaker recognition experiments that it is more efficient to utilize statistical techniques than dynamic techniques, since the computational effort is approximately one-tenth that required for dynamic techniques. However, as mentioned in the previous paragraph, statistical techniques require longer utterances than dynamic techniques, so the computational effort in determining the feature vectors is increased.

### 3.3.3 Vector quantization techniques

VQ can be utilized in a number of standard speaker recognition schemes, but is described separately from both the dynamic and statistical techniques because it does not fit neatly into either the dynamic or statistical class. For example, it is often incorporated into DTW or HMM schemes, but it can also be used in a manner more closely associated with statistical recognition techniques.

In a recognition scheme that uses DTW, vector quantization can be invoked as a

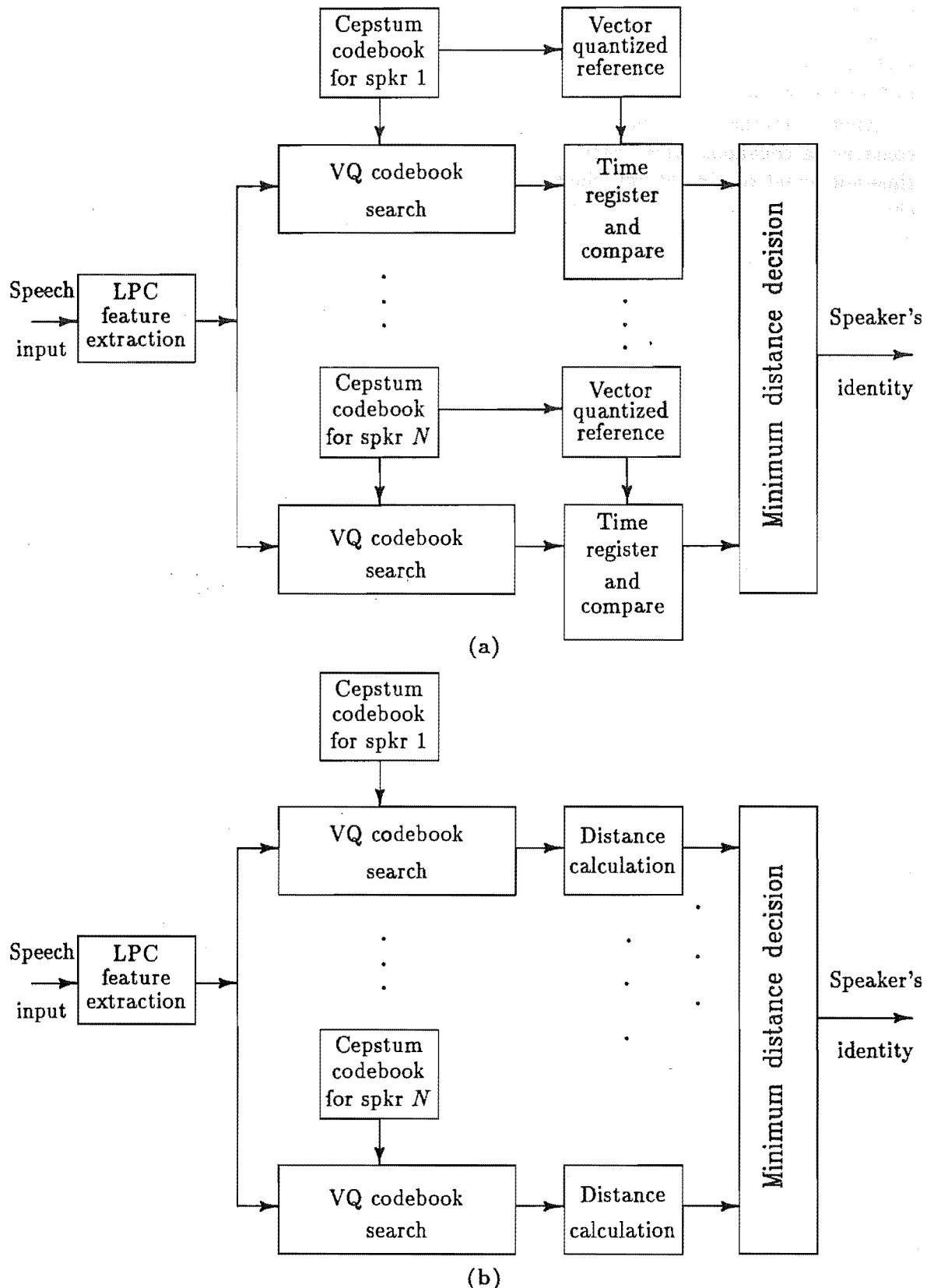
data reduction technique, thereby facilitating a reduction in the storage requirements of an individual's speech parameters. Fig. 3.8(a) depicts a VQ-based speaker identification system which uses a separate codebook for each speaker and DTW to determine the distance between a test utterance and a previously stored sequence of quantized vectors.

Another method of configuring a VQ recognition system, shown in Fig. 3.8(b), is to construct a codebook which represents a speaker's reference utterances, ignoring any time-alignment considerations. Such a VQ codebook represents an individual speaker's characteristic vectors within a multi-dimensional space as a number of codevectors. Of course, not every possible acoustic combination for an individual will be represented by their codebook, but a person's codebook should more accurately represent the types of sounds present in that person's utterances than the sounds in any person's utterances. Fig. 3.9 depicts this in a stylized manner, showing the way in which frames from a speaker's utterance (hopefully!) fall close to one of the centroids that are stored in the codebook describing that person's speech characteristics. Recognition proceeds by quantizing an unquantized sequence of test vectors and recording the quantizing error that results from using each individual's codebook. The quantization process can be described as follows. Each vector in the test sequence is matched against the 'closest' codevector in a codebook and the error summed over the entire utterance. The codebook associated with the smallest quantization error is assumed to correspond to the speaker who spoke the test utterance. Soong and Rosenberg (1988) find that removing time-alignment information from the feature set causes a reduction in the identification accuracy of 2.5% when the test utterance is a single spoken digit.

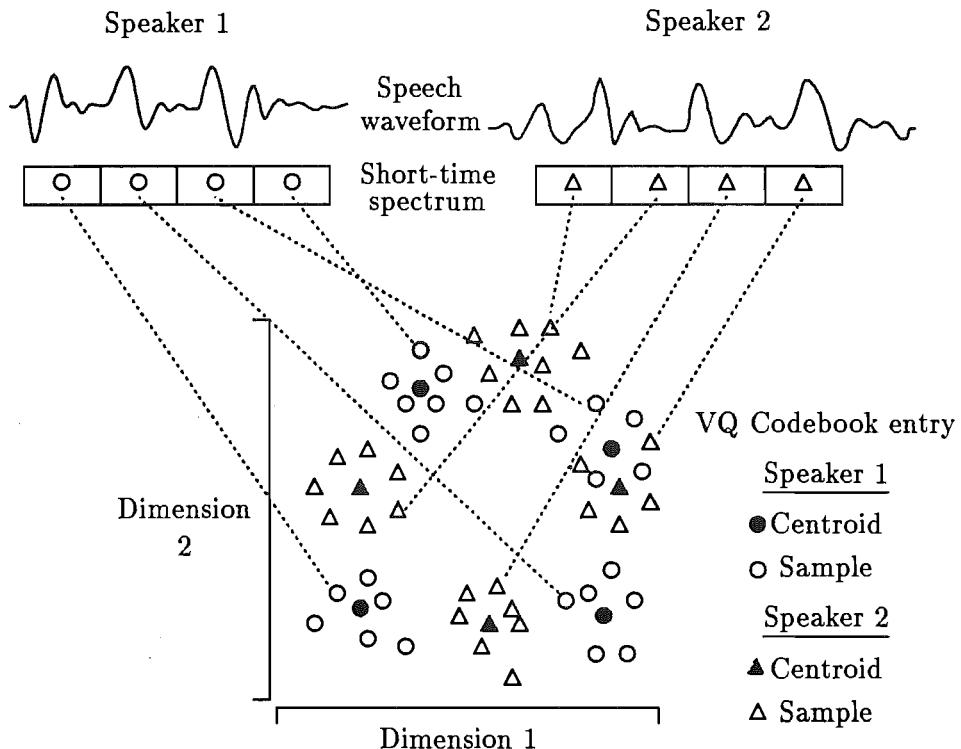
Soong *et al.* (1985) examined the effect of varying the size of the codebook and the length of the test utterance on the speaker identification accuracy. The utterances used consisted of the digits zero to nine recorded over telephone lines by 100 speakers. The recognition accuracy increased significantly as the codebook size was increased. For example, test utterances of ten digits and codebooks of 1, 2 and 64 codevectors gave identification error rates of 34%, 22% and 1.5% respectively. When 64 codevectors were used per individual and the number of digits in the test utterance was reduced, the identification error rate increased to 25% for one digit, compared with 1.5% for ten digits. The identification error rate is therefore reduced as the duration of the test utterance is increased and as the codebook size is increased.

Shirai *et al.* (1988) describe an interesting variation on the use of vector-quantized features for speaker identification. They used 100 words recorded from each of 200 speakers to construct a universal 256 vector codebook. Each speaker's template consisted of the frequency distribution against centroid number when 100 words were spoken and quantized. The identification procedure compared frequency distributions using a Euclidean distance. For the 200 speakers (97 female and 103 male), uttering 10 test words, the error rate was 12%. Shirai *et al.* (1988) omitted to apply the more standard method of using one VQ codebook per person, so it is not possible to ascertain whether or not identification accuracy is actually improved through using their technique. One obvious advantage is that since only the distribution of vector occurrences are compared across the speakers, there is no need to search through codebooks belonging to each speaker. Instead, a straightforward vector (the distribution) comparison suffices.

Although a VQ codebook can not strictly be considered a statistical description of a person's voice, it does contain representative vectors which minimize the long-time average difference between the reference phrases and the VQ codebook. In this respect, and also because there is no time-alignment information recorded in the codebook, the VQ codebook can be considered to be more closely associated with statistical rather than dynamic techniques. O'Shaughnessy (1986) argues that matching against a VQ



**Figure 3.8.** Speaker identification systems (based on Soong and Rosenberg (1988)) that use cepstral coefficients stored in VQ codebooks as speaker templates: (a) text-dependent system where references are stored as a sequence of indices to each speaker's codebook and matching is performed using DTW, (b) a text-independent recognition system (no time-alignment).



**Figure 3.9.** A conceptual diagram illustrating how the frames of analysed utterances are matched against codewords (centroids) for speaker recognition purposes (from Soong *et al.* (1987)).

codebook attempts to locate specific sounds in the test utterance, but avoids the complicated problem of time-alignment. Performing recognition using VQ codebooks can be considered an alternative to DTW and therefore does not fit neatly into either the dynamic or statistical techniques.

### 3.4 STATISTICAL METHODS FOR ASSESSING DISCRIMINATION ABILITY OF FEATURES

Typically, when designing a recognition system, a large number of features are extracted to characterize, in varying degrees, whatever it is that is to be identified. In a speaker identification context, the larger the number of features that characterize a speaker the better one might expect the recognition system to perform. However, the drawback of having a large number of features is the increased time required to compute the features and to compare them with the set of reference features. Furthermore, unless the features are to a certain degree independent from each other, adding extra ones will not necessarily increase the recognition accuracy. Most researchers, therefore, perform statistical analysis on the features extracted from the training data and discard those features that do not improve the recognition accuracy. In addition, weightings are calculated for the remaining features so that the discrimination between speakers is enhanced. This section is concerned with methods for determining which features will produce the best recognition performance.

Notation is now introduced based on that used by Bricker *et al.* (1971) for describing statistical techniques for speaker identification. Let  $i$  denote the speaker number, ranging from 1 to  $k$  and  $u$  denote the utterance number which ranges from 1 to  $n_i$  where  $n_i$  is the number of utterances for speaker  $i$ . The total number of features evaluated is  $p$ . A single *feature vector* corresponding to the  $i^{th}$  speaker uttering the  $u^{th}$  utterance

would be,

$$\mathbf{X}_{iu}^t = (x_{i1u}, x_{i2u}, x_{i3u}, \dots, x_{ipu}), \quad i = 1, 2, \dots, k, \quad u = 1, 2, \dots, n_i. \quad (3.2)$$

Typically each speaker in a speaker recognition system is represented by a single feature vector that contains feature averages, i.e.,

$$\begin{aligned}\bar{\mathbf{X}}_i^t &= \frac{1}{n_i} \sum_{u=1}^{n_i} \mathbf{X}_{iu}^t \\ &= (\bar{x}_{i1}, \bar{x}_{i2}, \bar{x}_{i3}, \dots, \bar{x}_{ip}).\end{aligned} \quad (3.3)$$

The distribution of individual utterance features about their means directly affects the expected recognition accuracy.

The average feature vector, taken across all the individual speaker averages is,

$$\begin{aligned}\bar{\mathbf{X}}^t &= \frac{1}{n} \sum_{i=1}^k n_i \bar{\mathbf{X}}_i^t, \quad \text{where } n = \sum_{i=1}^k n_i \\ &= (\bar{x}_1, \bar{x}_2, \bar{x}_3, \dots, \bar{x}_p).\end{aligned} \quad (3.4)$$

Ideally, the  $\bar{\mathbf{X}}_i^t$  should be well spread out in the  $p$ -dimensional space, since the further apart they are, the better separated the feature vectors corresponding to each individual. This has the effect of lowering the probability of recognition error. Conversely, it is desirable that the  $\mathbf{X}_{iu}^t$  be distributed close to the  $\bar{\mathbf{X}}_i^t$  ( $i$ , the speaker number, held constant) so that repeatability is maintained. Measures of these two distributions are the between speaker variance and the within speaker variance. The between speaker variance, often called the *interspeaker* variance, and specified for each feature separately is defined by,

$$b_j = \frac{1}{k-1} \sum_{i=1}^k (\bar{x}_{ij} - \bar{x}_j)^2, \quad (3.5)$$

where  $j$  is the feature number and can take on any value between 1 and  $p$ . The average (or pooled) within speaker variance, or the *intraspeaker* variance averaged over all speakers is defined by,

$$w_j = \frac{1}{n-k} \sum_{i=1}^k \sum_{u=1}^{n_i} (x_{iju} - \bar{x}_{ij})^2. \quad (3.6)$$

The above expressions of the interspeaker and intraspeaker variances for each feature are utilized in the following measures of the ‘usefulness’ of individual features.

### 3.4.1 F-ratio

One measure of the usefulness of a feature is its F-ratio. Pruzansky and Mathews (1964) define an F-ratio in terms of speaker features to be,

$$F = \frac{\text{variance of speaker means}}{\text{mean intraspeaker variance}}. \quad (3.7)$$

The F-ratio can be calculated for feature number  $j$  from (3.5) and (3.6), viz,

$$F_j = \frac{b_j}{w_j}. \quad (3.8)$$

This is straightforward to calculate and Mohn (1971) points out that it is a reasonable measure of feature worth, even in the absence of Gaussian distributed feature variables.

As intimated earlier, it is desirable that  $b_j$  be large and  $w_j$  small, which implies that the larger the value of  $F_j$  the better the  $j^{th}$  feature should perform in a speaker recognition system. However, O'Shaughnessy (1986) argues that choosing features with high F-ratios does not necessarily guarantee fewer recognition errors, since F-ratios tend to be high for features where the utterances of one or two speakers are very different from those of the rest. A drawback of F-ratio ranking of feature effectiveness is that, if features are not independent, choosing the best few (as measured by F-ratios) for recognition may result in a performance that is worse than that obtained by choosing a random selection of features (Mohn, 1971).

Markel and Davis (1979) examined the effects of the choice of population on F-ratios for gain, pitch and ten reflection coefficients. The aim was to ascertain whether a set of F-ratios calculated for one set of speakers could be used to approximate the F-ratios of another set of speakers. They found that F-ratios varied significantly between male and female populations. In addition, when the male population was arbitrarily divided into two equally sized subsets, significantly different F-ratios were obtained from each subset. Markel and Davis (1979) therefore concluded that their sample of 17 speakers was not large enough to produce consistent F-ratios representative of general sets of speakers as well as subsets, and that a larger database of perhaps more than 100 speakers would be required.

### 3.4.2 Discriminant analysis

The description of discriminant analysis presented here is based on the statistical techniques described by Bricker *et al.* (1971). Firstly, it is useful to define additional measures of the variations in the training data. The within speaker covariance matrix for speaker  $i$  is given by,

$$\mathbf{W}_i = \frac{1}{(n_i - 1)} \sum_{u=1}^{n_i} \{(\bar{\mathbf{X}}_{iu} - \bar{\mathbf{X}}_i)(\bar{\mathbf{X}}_{iu} - \bar{\mathbf{X}}_i)^t\}, \quad (3.9)$$

which defines the variation of feature values around the centroid of a single person's utterances. Averaging all the individual covariance matrices results in the pooled within speaker covariance matrix which is defined as,

$$\mathbf{W} = \frac{1}{(n - k)} \sum_{i=1}^k (n_i - 1) \mathbf{W}_i, \text{ where } n = \sum_{i=1}^k n_i. \quad (3.10)$$

The diagonal of  $\mathbf{W}$  contains the elements  $w_j$  as defined by (3.6). The between speaker covariance matrix is determined by evaluating,

$$\mathbf{B} = \frac{1}{(k - 1)} \sum_{i=1}^k \{(\bar{\mathbf{X}}_i - \bar{\mathbf{X}})(\bar{\mathbf{X}}_i - \bar{\mathbf{X}})^t\}. \quad (3.11)$$

Certain variance measures can be incorporated into distance measures to make use of the covariance information from the training data to weight the features being utilized in recognition. The weighted distance measure, introduced in §2.7.3, is restated here in a form that defines the distance between  $\mathbf{Z}$ , a test vector calculated from the test utterance, and the template (or mean) vector corresponding to speaker  $i$ , viz,

$$D_i = (\mathbf{Z} - \bar{\mathbf{X}}_i)^t \mathbf{M}_x (\mathbf{Z} - \bar{\mathbf{X}}_i). \quad (3.12)$$

The weighting matrix  $\mathbf{M}_x$  can be chosen in a number of different ways, as indicated by the subscript  $x$ . One solution is to choose,

$$\mathbf{M}_1 = \mathbf{W}_i^{-1}, \quad (3.13)$$

which sets the weighting matrix to be the inverse of the within speaker covariance matrix of the feature vectors for speaker  $i$ . The matrix,  $\mathbf{W}_i$ , must be invertable for each speaker, requiring the number of training utterances for each speaker to be greater than the number of features (Bricker *et al.*, 1971). Otherwise, the estimates of the within speaker variance contained  $\mathbf{W}_i$  are inaccurate. An alternative option is to set

$$\mathbf{M}_2 = \mathbf{W}^{-1}, \quad (3.14)$$

where  $\mathbf{W}$  is the pooled within speaker covariance matrix. Note that this is often called the Mahalanobis distance, which has its origins in statistical decision theory (O'Shaughnessy, 1986).  $\mathbf{W}$  is much more likely to be nonsingular than  $\mathbf{W}_i$ , since more sets of feature values are incorporated into the estimate of  $\mathbf{W}$  than  $\mathbf{W}_i$ . Bricker *et al.* (1971) recommend this particular weighting for use in speaker recognition systems.

In §3.4.1 the F-ratio is defined for an individual feature, whereas the following F-ratio depends upon all of the features and the vector  $\mathbf{a}$ , which defines a linear combination of the original feature variables. It uses the ratio of the between speakers covariance to the pooled within speaker covariance (Gnanadesikan, 1977, p85) and can be maximized by the appropriate choice of  $\mathbf{a}$ . This F-ratio is defined as

$$F_a = \frac{\mathbf{a}^t \mathbf{B} \mathbf{a}}{\mathbf{a}^t \mathbf{W} \mathbf{a}}. \quad (3.15)$$

$F_a$  can be maximized by the appropriate choice of  $\mathbf{a}$ . The solution for  $\mathbf{a}$  is obtained by calculating the set of eigenvectors which satisfy the following equation,

$$(\mathbf{W}^{-1} \mathbf{B} - \lambda \mathbf{I}) \mathbf{a} = 0. \quad (3.16)$$

The ordered eigenvalues  $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_r > 0$  of  $\mathbf{W}^{-1} \mathbf{B}$  and the corresponding eigenvectors  $\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_r$  can be used to define a transformed coordinate space which is called the *discriminant space*. The number,  $r$ , of positive eigenvalues of  $\mathbf{W}^{-1} \mathbf{B}$  is usually the smaller of  $(k - 1)$  (recall that  $k$  is the number of speakers) and  $p$  (recall that  $p$  is the number of features). Each  $\mathbf{a}_i$  can transform the original vector onto a coordinate in the discriminant space. In general, a subset of  $l$  vectors is chosen from the  $\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_r$ , because the higher order eigenvectors, with small eigenvalues, do not contribute significantly to the F-ratio. A suitable value for  $l$  can be determined by altering the number of eigenvectors and using the F-ratio to calculate the expected separation of the training data into individual speakers.

Once a subset of  $l$  eigenvectors has been selected they can be combined into a matrix  $\mathbf{A}_l^t$  which is an  $l \times p$  matrix, having the eigenvectors  $\mathbf{a}_1^t, \mathbf{a}_2^t, \dots, \mathbf{a}_l^t$  as rows. A vector of features, say  $\mathbf{y}$ , is transformed into the  $l$ -dimensional discriminant space by premultiplying it by  $\mathbf{A}_l^t$ , viz,

$$\mathbf{q} = \mathbf{A}_l^t \mathbf{y}. \quad (3.17)$$

The transformation from the original feature space to the discriminant space is incorporated into the distance measure specified by (3.12) by setting

$$\mathbf{M}_3 = \mathbf{A}_l \mathbf{A}_l^t. \quad (3.18)$$

Gnanadesikan (1977, p94) points out that using  $\mathbf{M}_2$  in the distance measure is identical to using  $\mathbf{M}_3$  with all  $r = p$  eigenvectors included. However, often  $r = (k - 1) < p$ , causing only an equivalence relationship to hold.

The advantage of using  $\mathbf{M}_2$  is its simplicity, since it avoids the computation of eigenvectors. However, computation of eigenvectors may highlight certain features that are essentially noise, leading to a reduction in dimensionality and a possible improvement in recognition performance (Gnanadesikan, 1977, p98).

### 3.4.2.1 Stepwise inclusion of features

Stepwise inclusion of features into a set of discriminating features aims to select an optimal set of features without including any features that have poor discriminating power.

A forward stepwise procedure begins by selecting the feature that has the most discriminating power and then testing the improvement in discrimination when each of the other variables is used, one at a time, to improve the discrimination. The feature that produces the largest improvement in discrimination is then added to the set of discriminating features. This procedure continues until either the remaining variables do not improve the discrimination by a sufficiently large margin to be considered useful, or all of the features have been added to the set of discriminating features.

A backward stepwise procedure works in the reverse direction by beginning with all of the variables in the selected feature set and discarding the feature that contributes the least to the discrimination. Forward and backward selection can also be combined by performing forward selection and then using backward selection to ascertain whether any of the previously included variables have become insignificant and should no longer be included in the selected set. This situation can arise when a variable shares discriminating information with another variable that is selected on a subsequent step.

Stepwise procedures require a measure of discrimination of a set of features in order to compare the usefulness of various feature combinations. Klecka (1980) reviews several different criterion for measuring discrimination, but here the description is restricted to Wilk's Lambda, since this is the discrimination measure used in the stepwise discriminant analysis of §4.3.3.2 and §5.4.2.1. Wilk's Lambda is defined to be

$$\Lambda = \prod_{i=1}^q \frac{1}{1 + \lambda_i}, \quad (3.19)$$

where  $q$  is the total number of functions and  $\lambda_i$  are the eigenvalues defined in (3.16). This can also be expressed as

$$\Lambda = \frac{|\mathbf{W}|}{|\mathbf{B} + \mathbf{W}|}, \quad (3.20)$$

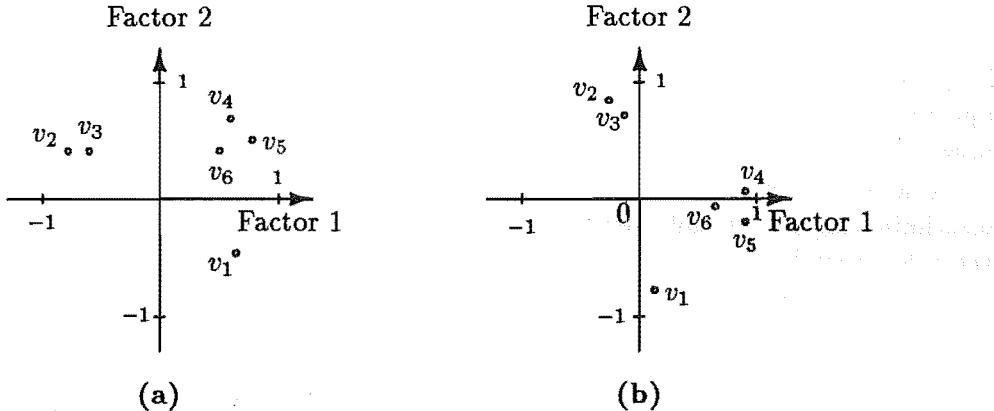
where  $\mathbf{B}$  and  $\mathbf{W}$  are defined in equations (3.11) and (3.10) (Lindeman *et al.*, 1980, p225).

Wilk's Lambda is an inverse statistic so the smaller the value of  $\Lambda$ , the better the discrimination. Therefore, the feature that produces the smallest  $\Lambda$  for a particular step is added to the set of selected features.

### 3.4.3 Factor analysis

Factor analysis is a generic term that describes procedures for analysing the correlations amongst variables (Cooley and Lohnes, 1971). The aim is to identify a small number of new variables (typically 2-4), called factors, that describe most of the variance recorded in the total set of variables. A commonly used method for identifying factors is called principal component analysis (PCA) (Comrey, 1973). PCA finds orthogonal factors by extracting the eigenvectors of the correlation matrix  $\mathbf{R}$ . The components of the correlation matrix  $\mathbf{R}$ , computed from the feature vector defined by (3.2) across all speakers ( $i = 1, 2, \dots, k$ ) and all utterances ( $u = 1, 2, \dots, n_i$ ) are expressed as (Gorsuch, 1983, p49)

$$r_{mn} = \frac{\sum_{i=1}^k \sum_{u=1}^{n_i} x_{im} u x_{in} u}{n \sigma_m \sigma_n}, \quad (3.21)$$



**Figure 3.10.** An example of applying factor rotation to two factors. (a) Unrotated factor axes for the six variables  $v_1, v_2, \dots, v_6$ , (b) Rotated factor axes for the six variables  $v_1, v_2, \dots, v_6$ . The contributions each variable makes to the factors is now more obvious than for the unrotated case.

where  $n = \sum_{i=1}^k n_i$ , and

$$\sigma_l = \sum_{i=1}^k \sum_{u=1}^{n_i} (x_{ilu} - \bar{x}_l)^2. \quad (3.22)$$

The eigenvectors of the correlation matrix are determined from the solution of

$$|\mathbf{R} - \lambda \mathbf{I}| = 0. \quad (3.23)$$

The matrix  $\mathbf{S}$ , containing the *factor structure*, is related to the eigenvectors by (Cooley and Lohnes, 1971, p106)

$$\mathbf{S} = \mathbf{V}\mathbf{L}^{\frac{1}{2}}, \quad (3.24)$$

where  $\mathbf{V}$  is a matrix of eigenvectors (the solution of (3.23)) organized in columns, and  $\mathbf{L}$  is a diagonal matrix with  $\lambda_j$  in the  $j^{th}$  position on the diagonal. The elements of each factor, recorded in the columns of  $\mathbf{S}$ , specify the strength with which a particular variable is present in the factor and are called the *factor loadings*. Note that the factor loadings can also be considered to represent the correlation between a factor and the variables.

The eigenvalue associated with a particular factor defines the proportion of the total variance accounted for by that factor. Factors with small eigenvalues describe little of the variance and are deemed unimportant. Such factors are discarded, thereby reducing the dimensionality of the feature space. Such a reduction in dimensionality allows combinations of factors to be plotted against each other, giving insight into the structure of the dataset. For example, if there are two or more separate clusters of variables within the dataset they are likely to be observable in the factor plots. The main function of factor analysis, therefore, is to provide insight into any underlying structures that might be present in a set of variables (or descriptors).

#### 3.4.4 Factor rotation

Factor rotation is employed to aid the interpretation and labelling of factors by rotating the factor axes so that the factor loadings become either large ( $\pm 1.0$ ) or small ( $0.0$ ), while maintaining an orthogonal relationship between the factors. Fig. 3.10 shows two unrotated factors and the same factors after a rotation. The advantage of such a rotation is that the relationships between the variables and factors become more obvious. For example, in Fig. 3.10(a), variable  $v_6$  is equally correlated with factor 1

and factor 2, since the loadings for both are approximately 0.5. However, after rotation the loading for factor 1 has increased to approximately 0.6, while the loading for factor 2 has been reduced to 0. This implies that  $v_6$  is correlated with factor 1, but not with factor 2. The rotation also highlights that variables  $v_1$ ,  $v_2$  and  $v_3$  are highly correlated with factor 2, while variables  $v_4$  and  $v_5$  are highly correlated with factor 1.

There are several different methods of rotating factors so that the correspondence between factors and variables appears more obvious. One method, called *varimax* (Cooley and Lohnes, 1971; Gorsuch, 1983), rotates factors so that the variance of the squared factor loadings across all the factors is maximized. This maximized variance is achieved by adjusting the factor loadings so that the loadings are either large ( $\pm 1.0$ ) or small (0.0).

Orthogonal rotations maintain an orthogonal relationship between factors. However, other rotations are sometimes used that allow minor correlations between factors. Such a rotation produces factors that are oblique. One popular oblique rotation is the *promax* rotation which adjusts factor loadings subject to a constraint on the amount of correlation allowed between factors.

Gorsuch (1983, §9.4.3) compared 19 different factor rotation procedures which perform either orthogonal or oblique rotations. He recommended that rotation using the varimax procedure should be performed prior to an oblique rotation using either promax or Harris-Kaiser criteria (Gorsuch, 1983, p205). An advantage of performing a two-stage rotation is that both the varimax solution and the oblique solution can be examined and compared. This two-stage rotation is used in the factor analysis performed in §4.3.3.1.

### 3.5 COMPARATIVE PERFORMANCE OF SPECIFIC TECHNIQUES AND FEATURES FOR SPEAKER RECOGNITION

This section is concerned with the performance of different techniques and sets of features for speaker recognition purposes. Results from papers that compare different features and recognition strategies are presented. In general, it is not possible to compare recognition accuracies across experiments since they are usually performed under very different experimental conditions (including speech databases). Instead, where possible, trends from experiments that use the same speech databases are described.

#### 3.5.1 Statistical vs dynamic techniques

In §3.3.1 and §3.3.2 both dynamic and statistical techniques for speech recognition are described. Since they represent speech parameters in a different format one would expect their recognition performances to be different. Furui (1981) compared their performances using log-area ratio coefficients. The selected statistics were: a subset of the mean, the standard deviation, the covariance matrix and the correlation matrix. The dynamic technique used DTW with unconstrained endpoints. He found that when the training period was longer than 3 months, the statistical technique was more accurate than the dynamic technique. However, if the training period was less than 3 months, the dynamic technique performed better than the statistical technique. Furui (1981) asserted that the statistical and dynamic techniques characterized speaker dependent information in an independent manner. By combining both techniques, the identification and verification accuracies were increased.

Another evaluation of the effect of incorporating the dynamic information into a recognition procedure is reported by Soong and Rosenberg (1988), who compare VQ-based speaker identification with and without DTW. The results of this aspect of their

|               |      |      |      |      |      |      |      |
|---------------|------|------|------|------|------|------|------|
| Transitional  | *    | *    |      |      | *    | *    | *    |
| Instantaneous |      |      | *    | *    | *    | *    | *    |
| Weighting     |      | *    |      | *    |      | *    | *    |
| DTW           |      |      |      |      |      |      | *    |
| Accuracy      | 75.5 | 82.6 | 84.3 | 88.3 | 89.2 | 92.8 | 95.3 |

**Table 3.3.** Speaker identification accuracies for instantaneous and transitional cepstral coefficients for a single digit test token (taken from Soong and Rosenberg (1988)).

| Parametric representation | Identification accuracy |
|---------------------------|-------------------------|
| Predictor coefficients    | 63.8%                   |
| Impulse response          | 60.6%                   |
| Autocorrelation function  | 59.7%                   |
| Area function             | 57.0%                   |
| Cepstrum function         | 70.3%                   |

**Table 3.4.** A summary of the average identification accuracies for a single speech frame for various parametric representations of speech segments (from Atal (1974)).

experiments are recorded in the final two columns of Table 3.3. They found that incorporating time registration and DTW matching into the identification procedure increased the recognition accuracy compared with the non-time-aligned VQ approach.

### 3.5.2 Transitional vs instantaneous features

Usually features extracted from speech are calculated on a frame-by-frame basis and no attempt is made to record the way in which the features vary between frames. However, recently there has been research into the information contained within the transition of features from one frame to the next. Soong and Rosenberg (1988) evaluated and compared the recognition performance of both instantaneous and transitional CEP coefficients for speaker identification. The reference information for each speaker was stored in the form of a VQ codebook, a separate codebook being used for each person's instantaneous and transitional features. Table 3.3 contains recognition results for both instantaneous and transitional cepstral coefficients. The weighting option indicates that coefficients from the inverse of the pooled intra-speaker covariance matrix were used to weight contributions to the distance from the cepstral coefficients. The instantaneous and transitional parameters did not perform as well individually as they did when combined. This implies that the instantaneous and transitional parameters contain independent speaker information.

### 3.5.3 Vocal tract features

One of the issues of speaker recognition is deciding which of the many sets of parameters derived from LPCs perform best for speaker recognition. Atal (1974) describes a comparison between: predictor coefficients, the vocal tract impulse response, the autocorrelation function, the area function and the cepstral function. From Table 3.4, a summary of Atal's results, it is obvious that cepstral coefficients are considerably more accurate than the other parametric representations of the speech signal. It is worth noting that Atal utilized a weighting matrix corresponding to  $M_2 = W^{-1}$  for calculating the distance between test vectors and reference vectors.

### 3.5.4 Glottal flow

Boves (1984) applied inverse filtering techniques to estimate the glottal flow spectrum. First, the vocal tract filter was calculated for the portion of a speech frame where the glottis was closed. The speech signal was then filtered by the inverse of the vocal tract response and the glottal flow spectrum computed. The energies in 13 different frequency bands were determined and a 'critical band spectra' constructed (Scharf, 1970). Glottal flow spectra were computed for 60 Dutch vowel sounds uttered by 5 speakers (Boves, 1984). Speaker identification accuracies of approximately 60% were achieved when 60 glottal flow spectra per person were matched against templates containing the mean glottal flow spectra. Boves (1984) attributed this poor recognition accuracy to the fact that the glottal flow spectra were derived from speech segments of less than 120 ms duration, which caused any short term variations in the vowel sounds to be significant.

Bove's results should be contrasted with those of Hollien and Majewski (1977) who performed speaker identification experiments using the long-term average spectra of 100 male (50 American and 50 Polish) speakers recorded under laboratory conditions (high quality recording equipment housed in a sound-proof room). Matching was performed using Euclidean distances between the long-term average spectra. Hollien and Majewski (1977) recorded 100% speaker identification accuracy for the 50 Poles and 96% accuracy for the 50 Americans. These results imply that there is characteristic information recorded in the long-term average spectrum of a person's speech.

### 3.5.5 The effect of removing glottal characteristics

In §2.2 the speech signal is modelled as the convolution of a glottal source with a vocal tract filter. This section reviews experiments which investigate the effect that removing glottal characteristics from the speech has on the stability of parameters estimated from the speech. Stable parameters are desirable since the task of speaker recognition is less difficult if a person's speech features do not vary too much between utterances. Results of speech identification experiments with and without the glottal characteristics removed are also reported.

Furui (1974) removed (approximately) the glottal characteristics by glottal inverse filtering. The glottal inverse filter is constructed from the long-term average spectrum of the speech and Furui (1974) states that the long-term average spectrum contains mainly contributions from glottal characteristics and the lip radiation. The assumption is that vocal tract responses are averaged across the various phonemes and so the main effect recorded in the long-term average spectrum is that of the glottis and lip radiation. This is not entirely true since the average of the vocal tract response is not an impulse, so vocal tract effects are present in the long-term average spectrum (this also holds for SAA, see §2.8.2). Nevertheless, for the same reasons as those outlined in §2.8.2, it is reasonable to assume that the long-term average spectrum contains mainly contributions from the glottal excitation.

Furui used PARCOR coefficients from normal and glottal inverse filtered speech to perform speaker identification. Glottal inverse filtering improved the speaker identification accuracy when the training sequences were recorded over 10 days (short-term training). Furui's test utterances consisted of either one or two words and he found that the recognition accuracy using the one word test utterance was improved by 4.0% (from 88.3%), while the two word test utterance showed a 5% improvement from 94.5%. However, removal of the glottal characteristics did not improve the identification accuracy when the reference templates were constructed from long-term (10 months) training data. Single word test utterances with the glottal characteristics removed performed

1% worse than the unfiltered speech and the two word test utterances gave exactly the same recognition accuracy. Furui asserted that this was due to the fact that the long-term variation in parameters was irregular, but he didn't specify the nature of these irregular parameter variations. He also stated that the long-term effect of the glottal spectrum on the parameters used to characterize a speaker was small. A similar trend was observed for speaker verification experiments.

### 3.5.6 The effectiveness of different phonemes

One might expect that the particular sounds, or phonemes, utilized in an utterance would affect the accuracy of speaker identification. Kashyap (1976) examines the interaction between speakers and their speech and concludes that some phonemes vary more significantly between speakers than others. The more accurately a particular phoneme can be recognized when uttered by different speakers, the less it varies between speakers and, therefore, the less useful it will be for speaker recognition. The opposite holds for phonemes that are unable to be accurately recognized across speakers. Kashyap (1976) finds that the phonemes /l/ and /e/ give the best speech recognition performance. It has been suggested that nasals, which use the invariant nasal passage, might be better suited to speaker recognition than vowel sounds (Glenn and Kleiner, 1968), but Kashyap's results indicate that the nasals /m/ and /n/ are not necessarily superior to vowels.

Matsumoto (1989) reports a more comprehensive set of experiments that utilized phonemes to identify ten male speakers by applying discriminant analysis (see §3.4.2) to determine the phoneme-dependent and phoneme-independent speaker information in a set of observation vectors. Each observation vector consisted of 20 cepstral coefficients and the pitch extracted from a 40 ms segment of voiced speech. In the training stage, phonemes were grouped into sets before performing discriminant analysis to optimize speaker discrimination. The speaker identification accuracy was shown to vary depending on how the sets of phonemes were partitioned. For example, nasal phonemes performed best when each phoneme was separated into its own subspace, but vowels performed best when the subspaces were overlapped so that certain phonemes appeared in more than one subspace. For 10 male speakers the text-independent identification accuracy was 90% and 100% for 0.5 s and 1.4 s of voiced speech respectively. Compared with text-independent identification, which is based on long-time statistical averaging, phoneme based identification represents a considerable saving in terms of the duration of speech required for accurate independent identification. Although these identification accuracies are excellent considering the shortness of the test utterances, only a small number of speakers were tested, and comparative results using standard techniques were omitted.

In general, the drawback with phoneme specific identification methods is that individual phonemes require labelling in the training phase, and this is a tedious process. This phoneme labelling procedure occurs implicitly when vector quantization is performed. Assuming that each phoneme occurs reasonably frequently in the training utterances, the final centroid vectors will be positioned so as to represent the average phoneme positions in the feature space.

### 3.5.7 Feature spacing and dimension

All of the features described in this chapter are determined from segments of speech extracted from speech utterances. It is therefore important to choose a segment size that will give reliable and consistent speech features and ensure high speaker recognition accuracy. Velus (1988) examined speaker verification error as a function of the segment

length and found that the equal error rate (EER) varied approximately 0.5% between segment lengths of 15 ms and 60 ms, with 30 ms producing the minimum EER.

As well as the segment duration, the number of LPC coefficients extracted from each speech segment critically affects the recognition accuracy. In general the accuracy with which a segment of speech is modelled is increased when more coefficients are used. However, there is a tradeoff between computational complexity and modelling accuracy. Velus (1988) reported that when using cepstral coefficients the EER for speaker identification decreased almost 0.7% (from 6.3%) when the model order was increased from 10 to 14, but decreased by only 0.3% when the model order was further increased to 20. As the model order became greater than 20, the improvement in verification accuracy was insignificant.

### **3.6 EFFECT OF VOICE DISTORTION ON SPEAKER RECOGNITION ACCURACY**

This section describes types of voice distortion that can affect the accuracy of speaker recognition. These can be broadly classified into distortion originating from the speaker and distortion induced by the transmission medium. Various distortions are discussed under the following headings; mimicry, disguise, noise and transmission distortion, voice variation with time and health.

#### **3.6.1 Mimicry**

Mimicry, the act of copying another person's voice, may occur in either a speaker identification or speaker verification context. Only limited research has been reported on the accuracy with which mimics can copy another's voice. This is probably because the ability to mimic people's voices is not a common skill, so mimicry research would require the services of professional mimics.

The most significant research into the effect of mimicry on speaker verification accuracy is described in Rosenberg (1973). Professional mimics were employed to copy the 8 speakers that constituted the set of accepted speakers. Subjective tests were in the form of paired comparisons, where a test and reference utterance were presented and a listener then decided whether or not the test utterance belonged to the reference speaker. The mimics had a 22% false acceptance rate, compared to 4% for natural sounding speakers. The success with which the mimics confused listeners, indicates that mimicry can be difficult to detect. It is worthwhile noting that when the same utterances were processed by an automatic verification system using pitch and amplitude profiles the false acceptance rate was 14% (Lummis, 1971). This indicates that computer-based speaker verification systems should combat mimicry at least as well as the average human listener, although Rosenberg (1973) found that the best human listeners outperformed the automatic system, with a false acceptance rate of only 3-4%.

#### **3.6.2 Disguise**

In a speaker recognition context, disguise occurs when people alter their voices in an attempt to conceal their identity. A typical disguise for a male speaker is to use a falsetto voice. In forensic applications there are two distinct types of voice disguise that can be expected to occur. One of these is when a speaker wants to speak in a voice that is understandable enough to convey an anonymous message, but is not concerned with how distorted it sounds. Typical examples of this type of disguise are bomb threats and ransom demands. Another type of disguise occurs when a person does not want the listener to know that a disguised voice is being used. An example of this situation is

mimicry, where the mimic might attempt to obtain access to classified information. If such mimicry is detected, access will, of course, be barred. Voice disguise is a problem in any type of speaker recognition problem, since recognition accuracy is significantly reduced for disguised voices.

An example of the degradation in speaker identification accuracy with disguised speech is reported by Reich and Duke (1979). As well as undisguised recordings from 40 adult speakers, 5 different disguised voices were recorded. The 5 different disguises were '70-80 years old', 'severely hoarse voice quality', 'severely hyper-nasal voice quality', 'extremely slow rate' and a disguise of the speaker's choice. All of the speakers spent time rehearsing the various disguises. The experiment involved listeners deciding whether pairs of utterances were both spoken by the same person. In this type of experiment, false identification errors (listener wrongly identifies two utterances as being the same person) or false elimination errors (listener wrongly says the utterances are different) could occur. Listeners could discriminate between speakers with 97% accuracy for undisguised speech, but this fell to 59%-81% for disguised speech, depending upon the particular disguise.

The effect of voice disguise on voiceprint identification is also considerable. Kersta (1962) claimed that voiceprints were unaffected by attempts at vocal disguise, however Reich *et al.* (1976) found that disguising speech increased the false identification errors significantly. For example, speaking at a slow rate lowered the recognition accuracy by 14%, while a disguise of the speaker's choice lowered the recognition accuracy by 35%. Reich *et al.* (1976) considered that the large variations in the formant structure affected by voice disguise constituted a formidable stumbling block for spectrographic examiners.

Although it is difficult to perform speaker identification on disguised voices, it seems that it might be possible to at least detect when a voice is disguised. Reich (1981) reports an experiment where listeners were able to detect a disguise in male voices with more than 90% accuracy. Furthermore, both naive (undergraduate students) and sophisticated (doctoral students and professors of Speech and Hearing Science) listeners identified disguise in voices with the same accuracy. Although Reich's experiment shows that disguise can be detected reliably, it does not evaluate the aural cues within the utterance that indicate that the voice is disguised. In this respect, the findings are not helpful for the design of practical speaker identification systems, although they do indicate that it should be possible to detect voice disguise in most instances.

### 3.6.3 Noise and transmission distortion

Noise and distortion are included here to account for differences in the speech signal between the uttered speech, and the signal that is heard by a person or recorded by an automatic system. Noise accounts for the random, extraneous events in the speech waveform that are not produced by the speaker, while distortion accounts for non-random alterations of the speech signal. A typical example of such a distortion is the anti-aliasing filter which removes high frequency components of the speech signal. Obviously, both noise and distortion are always present to some extent, particularly if the speech is encoded into a low bit rate format. In almost all practical situations there are significant levels of distortion (Krasner *et al.*, 1984), prompting an examination of the effects of distortion on recognition accuracy.

The effect of coding speech on speaker recognition accuracy has been examined by McGonegal *et al.* (1979) and Schmidt-Nielsen and Stern (1985). McGonegal *et al.* (1979) utilized both ADPCM and LPC coding to process speech that was recorded over a system having the nominal telephone bandwidth of 100-3200 Hz. The speech database consisted of 10 male and 10 female customers and 40 male and 40 female

imposters who were all recorded over a period of 2 months. They reported that human listeners could perform verification equally well for high quality speech and speech that was recorded over the telephone. However, the error rate increased when high quality speech was used for the reference utterances and test utterances were coded (and decoded) using ADPCM or LPC coding. This indicates that speaker verification by human listeners is sensitive to differences between the types of noise and distortion introduced to the test and reference utterances. The accuracy of automatic verification, using pitch and intensity, remained independent of the transmission system used for the test and reference utterances. Schmidt-Nielsen and Stern (1985) reported that identification accuracy was significantly lowered by the distorting effects of LPC-10 (2400 bits/s) coding. The identification accuracy of 24 speakers by listeners fell from 88% to 69% for LPC coded phrases.

Speaker recognition accuracy can be improved if the recognition system is designed to be insensitive to noise and transmission distortions. A partial solution to the problem of noise in transmission systems is proposed by Noda (1988). Noda utilized spectral parameters for speaker verification and warped those portions of the speech spectrum where the noise power was small compared with the speech power. He found that the warping he introduced to counter the effects of the noise increased speaker verification accuracy for clean (no noise added) utterances and also for noisy utterances with a 10 dB SNR (created by adding white noise). This indicates that by invoking procedures to counter noise in the speech signal, recognition accuracy can be increased, making recognition practical in the less than perfect transmission conditions that prevail in all communication networks.

One method of removing the effects of any constant linear distortion that occurs in the transmission channel is to apply an inverse filter determined from the mean spectrum of the speech signal. This is called *channel normalization* or *spectral equalization*. Birnbaum *et al.* (1986) describe a speaker verification system that performs channel normalization by subtracting the mean of a set of cepstral coefficients from the complete set of coefficients, thereby performing inverse filtering. Of course, the long-term spectral characteristics of the speech are also removed, but this does not reduce the verification accuracy. Furui (1981) performed a similar operation using an inverse filter constructed from the long-term average spectrum of the speech (the same technique as described in §3.5.5). Spectral equalization was shown to improve the speaker verification accuracy, especially when the test utterances were recorded five years after the training utterances.

The sensitivity of voice-personality to alterations in pitch, formant frequency and formant bandwidth was examined by Takagi and Kuwabara (1986). In their experiment the aforementioned features of a speech utterance were perturbed independently and three listeners decided whether the perturbed speech corresponded to a particular speaker (yes or no). Perceived voice personality was found to be very sensitive to changes in the formants. The individual characteristics of the voice were found to be almost completely removed when all formants were shifted more than 5%. The perceived voice-personality was less sensitive to changes in the lower three formants than the upper formants. Variation of the pitch of a person's voice was found to have little effect on the recognition accuracy, since 50% recognition accuracy was still achieved for pitch scalings of between 0.6 and 1.45.

### 3.6.4 Voice variation with time

This section reviews studies of the effects of voice variation on speaker templates and the identification accuracy. It has been well established that speaker recognition accuracies decrease with time due to changes in individuals' voices (Matsumoto, 1989; Furui, 1974;

Furui, 1981). The following paragraphs outline a number of different ways this reduction in recognition accuracy can be minimized.

Recognition accuracy is critically dependent on the template of features that is used to represent a speaker's speech characteristics. Obviously, the aim is to construct a template that represents the constant characteristics of a person's voice, rather than the small scale deviations that might occur on a day to day basis. Furui (1981) compared the identification accuracy of templates constructed from 9 or 12 utterances collected over 10 days (short-term) and 12 utterances collected over 10 months (long-term). One limitation of this experiment was that a different number of utterances were collected for short-term and long-term training, making it difficult to deduce whether the differences obtained from the two different templates should be attributed to differences in training duration or differences in the number of training utterances. Furui (1981) did not discuss this in his paper, so he must have considered the variation in the number of training utterances to be insignificant. He went on to demonstrate that templates constructed from utterances collected over a short term were not as accurate for identification as those collected over the long-term. For example, after four and a half years had passed between the training and recognition, 9 male speakers uttering two words could be identified with approximately 90% and 73% accuracy for long and short-term training respectively.

A drawback of long-term training is that speakers must be involved in the time consuming task of having their voices recorded. An alternative to collecting training data over an extended period is to utilize each test utterance to update the speaker template. Assuming that the recognition algorithm successfully locates a template that matches the test template, the reference template is then modified so that the distance between the test and reference templates is reduced. In this manner the reference template adapts to the changes in a speaker's voice. Texas Instruments utilized this technique in their speaker verification system and observed an overall decrease in verification errors with successive trials (Naik, 1990).

### 3.6.5 Health

Health can have a significant effect on the sound of a person's voice. Everyone is familiar with the vocal changes that occur during colds, coughs or sore throats. However, to the author's knowledge there has been no systematic investigation of the effects of these conditions on speaker recognition.

One factor that makes the investigation of the effects of colds, coughs and sore throats difficult is that they do not occur regularly. Furthermore, people having these conditions are often at home resting, and are therefore unable to participate in recording procedures.

Current speaker recognition systems do not account for the speaker's health, so one would not expect good recognition performance when people do not 'sound themselves'. Investigation into the changes in our voices that occur for various maladies would be required as a first step towards ensuring that speaker recognition systems perform accurately when people are 'sick'.

## 3.7 REAL-TIME SPEAKER VERIFICATION

This section reviews the performance of three speaker recognition systems. These systems are of interest because they constitute the practical application of techniques described elsewhere in this chapter. Note that only speaker verification systems are discussed here, since speaker identification requires matching against many templates and is therefore not well suited to real-time evaluation (see §3.1.1). One of the systems

reviewed in this section is available commercially and can therefore be considered to be mature technology.

As early as 1974 Texas Instruments developed and deployed a speaker verification system for controlling access to its computer center premises (Naik, 1990). The incoming speech was filtered by a 14-channel filter-bank representation and dynamic time warping (DTW) was utilized to compare test and reference utterances. The verification phrase was four words chosen from a set of sixteen (Doddington, 1985). A random combination of words was selected in order to foil any recording imposters wishing to gain access with prerecorded messages. The operational system had an overall rejection rate of 0.9% and an imposter acceptance rate of 0.7%. The verification system was housed in a booth that could hold several people and it was observed that the rejection rate was 0.5% when only one person was in the booth, compared with 1.8% for more than one person in the booth. The added distraction of having a second person in the booth must have affected the quality of the speech sample!

More recently, Texas Instruments (TI) have developed another verification system based on the spectral information contained in LPCs and using DTW as the matching technique (Naik, 1990). The recognition text consists of two phrases. The first phrase is a five digit code followed by a two word code phrase and the second phrase, a random sequence of 5 digits, is included to counter imposter access via tape recorded messages. Speech processing is performed on a specialized subsystem consisting of a TMS32010 and an analog I/O board. The system was evaluated on a population of 100 males and 100 females over a four month period and every user attempted to impersonate every other user at least once. The false rejection rate was 0.80% and 0.87% for males and females respectively and the false acceptance rate (for casual imposters) was 0.07% and 0.12% for males and females respectively.

In contrast with the types of systems developed by TI, AT&T have developed a speaker verification system specifically designed for operation over telephone lines (Birnbaum *et al.*, 1986). The identity claimed by a user is entered via the touch-tone keypad on a standard telephone. A speech synthesis chip is utilized to prompt the user at various stages of the verification procedure. The system uses CEP coefficients and DTW to implement text-dependent speaker verification. Channel normalization (§3.6.3) is performed to reduce the effects of different channel characteristics. Twenty one speakers evaluated this system, each making a minimum of 40 calls to the system over a ten day period. Sixteen of the speakers made local calls while the other five made long distance calls. In addition, a further 23 males and 22 females were recruited as imposters. At the verification threshold chosen for this trial population, the false rejection rate was 3.1% and there were no false acceptances. Although these results appear reasonable, the equal error rate (EER) was actually 1.9%, so the operational verification threshold selected had the effect of decreasing the false acceptance rate at the expense of the false rejection rate. Further examination of the cause of the false acceptances led to improved endpoint detection in the test utterance and a corresponding reduction of the EER to 0.4%. This is an impressive result considering the variations in transmission characteristics that can occur between telephone calls.

Another system that performs text-independent speaker verification has been designed and tested by Attili *et al.* (1988) at the Rensselaer Polytechnic Institute. It is based on a TMS32020 DSP (with ancillary hardware) and a host PC which houses the DSP hardware, stores speaker templates and provides the user interface. Only 2-3 s of unconstrained speech is required to perform verification, which is shorter than that required by most other text-independent verification schemes. The parameters utilized in verification are 12 PARCOR coefficients, 12 log area coefficients, 12 (LPC) cepstral coefficients and a normalized gain coefficient. During training, discriminant analysis

is performed on each individual speaker's characteristics in order to determine a new feature space which separates the individual speaker as much as possible from all the others. The system was tested on a population of ninety speakers using recordings of 70 s duration that were collected in a single session. The first 10 s of speech formed the training utterance while the remaining 60 s was used for testing the verification accuracy. For text-independent operation the false rejection and false acceptance error rates were 2.3% and 1.6% respectively. The system was also evaluated in text-dependent mode and the error rates reduced to 1.5% and 0.52% for false acceptance and false rejection respectively. Verification is fast, occurring in 75% of real-time, although training the system is considerably slower.

The development of accurate real-time speaker verification systems shows that speaker verification techniques are now well established and constitute a viable technology for security and access applications.

### 3.8 SUMMARY

This chapter introduces speaker recognition and describes factors that influence speaker recognition accuracy. The merits of several different speaker recognition techniques and features are discussed. The main points of the chapter are as follows:

- Factors that affect the recognition performance of human listeners are discussed. It is found that the more familiar a person is with the speaker, the more likely they are to be recognised correctly.
- Techniques for performing speaker recognition are divided into dynamic techniques, statistical techniques and vector quantization techniques. Features such as LPCs can be used in any of these techniques.
- Provided several training utterances are available for each speaker, interspeaker and intraspeaker variance can be used to assess the usefulness of features. Furthermore, contributions from various features can be weighted so as to enhance recognition accuracy and statistical techniques can be employed to assess the usefulness of individual features.
- Speaker identification results using many different techniques and features are reviewed. On populations of 100 speakers, vocal tract features can be expected to give identification error rates of less than 10%. Instantaneous and transitional parameters contain independent information. Text dependent recognition, where the sequence of features is recorded, is more accurate than text independent recognition.
- Speaker verification systems are available that work in real-time and have false rejection rates and imposter acceptance rates of less than 1.0%.

## CHAPTER 4

---

### FEATURES USED IN IDENTIFICATION EXPERIMENTS

This chapter describes details of those features abstracted for characterizing voices that are utilized in the recognition experiments presented in Chapter 5.

An important consideration in evaluating speaker identification systems is the database of speaker utterances. Details of the composition and recording protocol of the speech database are described in §4.1. Several of the features are quantized by using codebooks obtained with the Linde, Buzo and Gray (LBG) algorithm (as defined in §2.7.4.3). Section 4.2 describes the vocal tract features used and the implementation of the LBG algorithm used to form vector quantization codebooks. Details are presented of the procedure followed to ensure that the results concurred with those of LBG. In §4.3 important aspects governing the calculation and characterization of long-term average glottal responses are discussed. Real-time implementations of the shift-and-add (SAA) algorithm are described and several descriptors that characterize the shape of the long-term average glottal response (LTAGR) are defined in §4.3.2 and evaluated in §4.3.3. The sensitivity of the LTAGR to accent variation and speaker gender is examined in §4.3.3.1 and §4.3.3.2. Section 4.4 compares several methods of calculating the long-term average spectrum (LTAS), and §4.5 compares the spectral content of the LTAS and the LTAGR. The final section, §4.6, is a summary of the main results reported in this chapter.

#### 4.1 THE SPEECH DATABASE

This section describes the database utilized for speaker recognition experiments and the procedures followed to record it.

The group of words chosen for recognition experiments consisted of the digits zero to nine spoken with a small pause between digits, i.e. the words were discrete. There were two reasons for selecting this particular group of words: firstly, other researchers (Burton, 1987; Soong *et al.*, 1985) have used these words for speaker recognition; secondly, the isolated words were to be used in speech recognition experiments by my colleagues Tracy Clark (Clark *et al.*, 1990), Lim Ching Aun (Lim *et al.*, 1990) and John Kirkland (Kirkland and Garden, 1991). Twenty speakers are recorded in the speech database and a particular recording is referred to by the person's initials and a recording number. For example, AE3 is the third recording made by speaker AE (Andrew Elder). Table 4.1 contains personal information pertaining to each individual. Between sixteen and twenty recordings were obtained from each individual, but of these only the first fifteen were utilized for speaker recognition experiments unless there was an obvious recording mistake, such as missing the digit zero out of the zero to nine sequence.

A database recorded for use in speaker recognition experiments should include typical variations in an individual's voice. For this reason a maximum of two recordings were made each day, with a minimum separation of one hour between recordings. Furthermore, recordings were unsupervised and participants were free to record utterances at their own convenience. As a consequence of this somewhat unstructured recording

| Speaker number | Initials | Sex | Age group | Native country |
|----------------|----------|-----|-----------|----------------|
| 1              | AE       | M   | 20s       | New Zealand    |
| 2              | AM       | M   | 20s       | New Zealand    |
| 3              | AS       | M   | 20s       | Iran           |
| 4              | AW       | F   | 20s       | New Zealand    |
| 5              | BD       | M   | 20s       | New Zealand    |
| 6              | CA       | M   | 20s       | Malaysia       |
| 7              | CC       | M   | 20s       | New Zealand    |
| 8              | CP       | M   | 20s       | New Zealand    |
| 9              | CW       | F   | 20s       | New Zealand    |
| 10             | DB       | M   | 40s       | New Zealand    |
| 11             | DR       | F   | 30s       | New Zealand    |
| 12             | JK       | M   | 20s       | New Zealand    |
| 13             | MC       | M   | 20s       | New Zealand    |
| 14             | PK       | M   | 20s       | New Zealand    |
| 15             | RC       | M   | 40s       | South Africa   |
| 16             | RM       | M   | 20s       | New Zealand    |
| 17             | TC       | F   | 20s       | New Zealand    |
| 18             | TE       | M   | 20s       | New Zealand    |
| 19             | VS       | M   | 20s       | New Zealand    |
| 20             | WT       | M   | 20s       | New Zealand    |

Table 4.1. Personal information pertaining to people recorded in the speaker database.

timetable many participants regularly forgot to make recordings, which meant that the period over which utterances were recorded ranged up to four weeks - much greater than the eight working-day minimum. The time-dependent variations in the recorded voices are therefore larger than might be expected if recordings were made within a short time span.

All recordings utilized a hand-held microphone which participants kept a comfortable distance from the mouth. One of two different microphones was used at each recording, though both have a similar performance. One microphone was an AIWA CM-53, which has a flat frequency-response from 50-13000 Hz while the other was a Audio-Technica AT818II which is specified to have a flat frequency response from 50-15000 Hz. Recordings from nine people were made using the AIWA microphone in one series of sessions, and at a later date recordings of a further eleven people were made using the AT818II microphone. The microphone was connected to an AIWA F990 cassette tape-recorder, which recorded the speech signal on a low-noise tape employing Dolby-C noise-reduction. Measurements of the transfer function of the tape recorder by Thorpe (1990, p32) show it to have a flat frequency-response between 20-18000 Hz and a linear phase-response (to within 10°) between 100-5000 Hz.

Recordings were digitized by playing the tape recordings into an SX10 digital audio board (manufactured by Antex Electronics (Antex, 1990)), which stored the digitized utterances on the hard disk of a personal computer (PC). The sampling rate was 10 kHz and the sample resolution 16 bits. In order to detect any irregularities, each recording was monitored on headphones as it was 'played' into the SX10 board. Disk space was

conserved by restricting the amount of silence digitized at the beginnings and ends of utterances. The storage requirements for a single utterance ranged between 150 and 300 kBytes. The anti-aliasing filter incorporated into the SX10 digital audio-board (Antex, 1990) consists of a 7th order elliptic filter with a cut-off at 4.4 kHz. This anti-aliasing filter has an almost linear phase-response in the pass-band, but becomes less linear near and above the cutoff frequency (§5.5.2.3 evaluates the effect of phase distortion on the speaker identification performance of various features). After digitizing, the utterances were transferred to the Departmental VAX 3500 for further processing.

In order to check the integrity of the digitized database, all utterances in the database were plotted out in the format depicted in Fig. 4.1(a). This allowed the digitized recordings to be checked for incorrect truncation. In addition, if a digitized utterance was found to contain considerable silence at either end, a plot of the rms intensity, as depicted in Fig. 4.1(b), was used to select new beginning and end points, thereby removing surrounding silence from the utterance.

The final database labelling was checked aurally by playing back all the digitized utterances belonging to each individual and listening for any 'imposters'. This revealed that four utterances in the database had been incorrectly labelled when being transferred from the PC to the VAX, and these were redigitized. Although this procedure for checking the labelling relies on human speaker identification performance, in my experience it is a fairly straightforward task to identify an extraneous utterance within a sequence of utterances from a particular speaker.

#### 4.1.1 Observations on the speech database

Several aspects of the speech database are worth noting briefly. Firstly, within a single utterance there is sometimes considerable variation in loudness amongst the digits that constitute the utterance. The utterance depicted in Fig. 4.1 is a typical example of this type of variation in loudness: the first word is of considerably higher amplitude than the remaining words. Such variations can cause irregularities to occur within algorithms that utilize the maximum loudness to normalize the magnitude of certain parameters.

When aurally checking the digitized phrases, it was apparent that some people were quite tense for the first one or two recordings. This made their voice sound harsh, as described in §1.4. Although such tension was detectable by the ear, it did not seem to mask (aurally) the speaker's identity so no utterances were removed from the database on the grounds of 'tense' sounding voices.

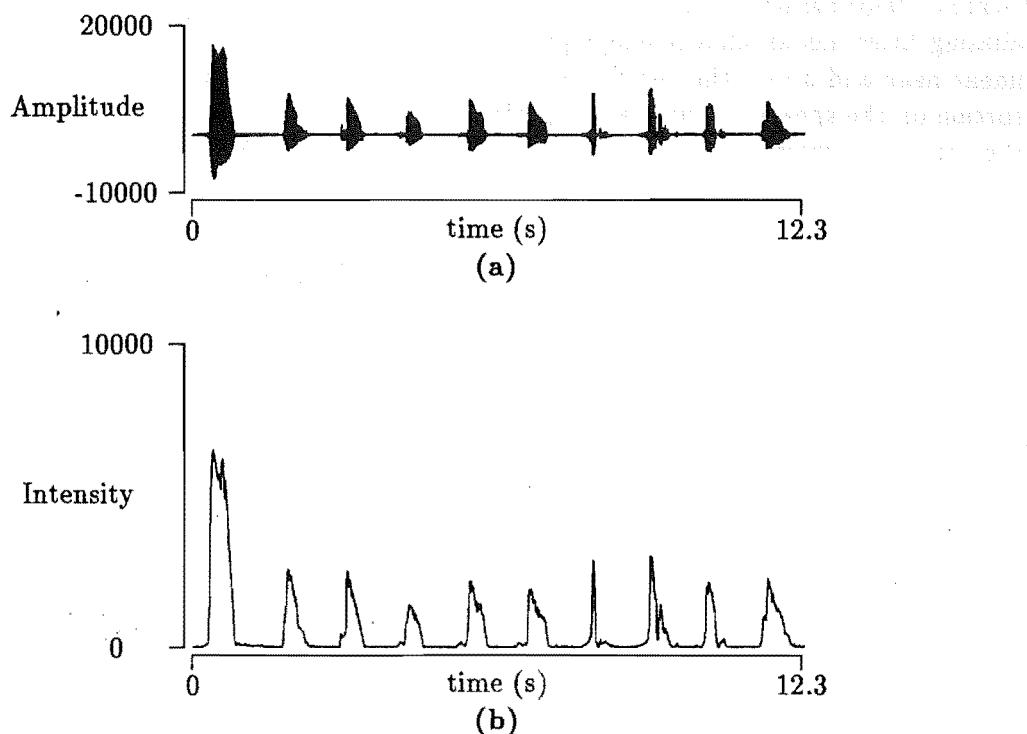
Another variable among the speakers was how they spoke the utterance, particularly the 'noise' level between words. Some speakers took noisy breaths between words, whereas others managed to speak the complete utterance without making any detectable breathing noises whatever. Figs. 4.2(a) and (b) respectively depict the speech signal recorded when no breath is recorded and when a breath is taken between words.

## 4.2 VOCAL TRACT FEATURES

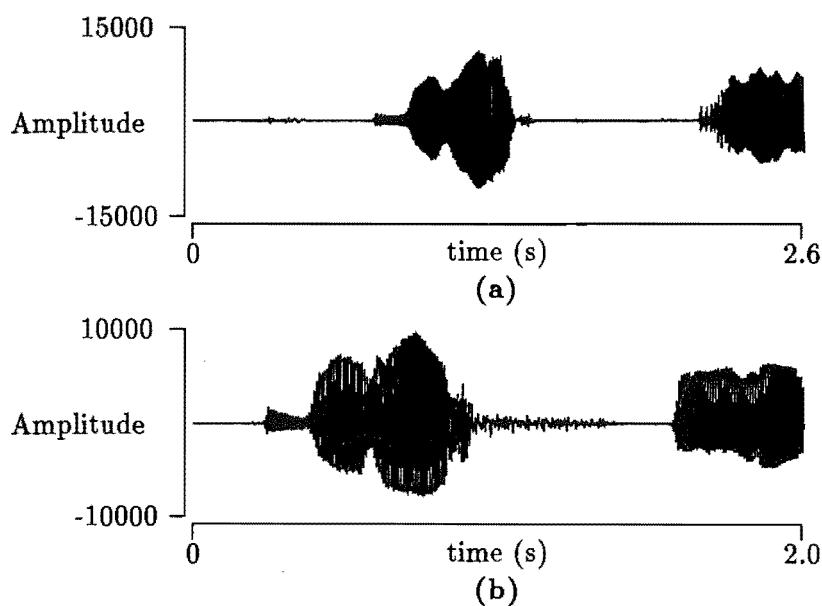
This section describes the computation of vocal tract features in §4.2.1 and the training of vector quantization codebooks containing vocal tract features in §4.2.2.

#### 4.2.1 Computation

The vocal tract features are selected from amongst those reported in Chapter 3 as useful for speaker recognition, and are listed in Table 4.2. These are also chosen because they



**Figure 4.1.** An example of the plotting format used to examine the integrity of the digitized recordings, highlighting the intensity variations that may occur as a person says different words: (a) the speech waveform of a entire utterance (JK6);(b) the rms intensity of the utterance depicted in (a).



**Figure 4.2.** An example of the differences in the recordings of: (a) silence between words (from AS11) and (b) a breath between words (from AS14).

| Coefficients        | Abbreviation |
|---------------------|--------------|
| Linear prediction   | LPC          |
| Partial correlation | PARCOR       |
| Cepstral            | CEP          |

**Table 4.2.** The abbreviations for the vocal tract parameters used for speaker recognition.

| Coefficients | Description of distortion measure    | Calculation method   |
|--------------|--------------------------------------|--|
| LPC          | Likelihood ratio<br>(see eqn (2.87)) | $d_1( X(e^{j\theta}) ^2,  \sigma/A(e^{j\theta}) ^2) = \ln \left( \sum_{i=-P}^P \frac{r_i^x}{\sigma^2} r_i^a \right)$ |
| CEP          | Euclidean<br>(see eqn (2.78))        | $d_2(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^M (x_i - y_i)^2$   |
| PARCOR       | Euclidean<br>(see eqn (2.78))        | $d_2(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^M (x_i - y_i)^2$   |

**Table 4.3.** The distortion measures for the different vocal tract parameters used for speaker recognition.

afford tractable methods for designing VQ codebooks (Gray *et al.*, 1980; Juang *et al.*, 1982).

Unless otherwise specified, the speech utterances are pre-emphasized by  $1 - z^{-1}$ . Vocal tract features are computed from adjacent speech frames that are 20 ms (200 samples) long. A Hamming window is applied to each speech frame before vocal tract features are computed. If voiced portions of an utterance are required, the VUV2 algorithm defined in §2.4.4 is used to extract voiced frames, unless otherwise specified.

PARCOR and LPC coefficients are computed using Durbin-Levinson's algorithm, as described in §2.5.3.1. CEP coefficients are computed from the LPCs using the recursive procedure presented in §2.6.3. Twelve coefficients (twelfth order model) are computed for each of the PARCOR, LPC and CEP features.

#### 4.2.2 Vector quantization of vocal tract features

This section describes the practical implementation and testing of the LBG VQ training procedure that is described mathematically in §2.7. Sections 4.2.2.1 and 4.2.2.2 describe the distance measures and centroid calculation methods that are used to train the VQ codebooks. Examples of VQ training and quantization errors are presented in §4.2.2.3.

##### 4.2.2.1 Distortion and centroid computation

The distortion measures applied to the different vocal tract features are detailed in Table 4.3. Each distortion measure requires a different method for calculating the centroid of a set of vectors. Recall (§2.7.4.3) that the set of training vectors can be described by  $\mathbf{x}[j]$ , where  $j$  varies from 1 to  $M$ . In addition,  $r_i^x[j]$  denotes the  $i$ th order autocorrelation coefficient of the  $j$ th vector of autocorrelation coefficients,  $\mathbf{r}^x[j]$ .  $\sigma^2[j]$  is the prediction error of the  $j$ th LPC filter.

Table 4.4 summarizes the centroid calculation methods (see §2.7.4.3) used here in the LBG vector quantization training algorithm. The equations describe the calculation of the centroid of the set of vectors lying within a cell  $C_i$  (see §2.7.1.2).

| Distortion measure | Centroid calculation method  |
|--------------------|--|
| $d_2$              | $\text{cent}_2(C_i) = \frac{1}{\ C_i\ } \sum_{j:x[j] \in C_i} x[j]$                    |
| $d_{LR}$           | $\text{cent}_{LR}(C_i) = \frac{1}{\ C_i\ } \sum_{j:r^x[j] \in C_i} r^x[j]/\sigma^2[j]$ |

Table 4.4. The centroid calculation method associated with each of the distortion measures.

#### 4.2.2.2 Centroid splitting

Central to the LBG algorithm is the concept of centroid splitting. The paragraphs below outline details of the splitting method used for the Euclidean distance, followed by the author's splitting method for the log-likelihood distortion measure.

The centroids of features that use the Euclidean distance are split by adding and subtracting a small, constant perturbation-vector  $\epsilon$  to each of the centroid vectors (as described by LBG). The constant is chosen to be small compared to the component values of the feature vectors, and here each element of  $\epsilon$  is equal to 0.05. Adding and subtracting  $\epsilon$  from each centroid doubles the number of centroid vectors. Recall from §2.7.4.3 that the direction of the split is not critical, since the aim of splitting is to create two new centroids that are near (in a distortion sense) the old centroid. In hindsight it would have been more logical to use a small proportion of the centroid vector to specify each component of the perturbation vector, since that would guarantee the magnitude of the two new vectors were in correct proportion to the magnitude of the original vector. However, since LBG used a constant perturbation vector and the algorithm converged successfully with the  $\epsilon$  specified, improvements in the splitting method were not investigated.

Centroid splitting for features that use the log-likelihood distortion measure is complicated because the centroid is stored as  $r^a[i]$ , which (as defined in (2.85)) is the autocorrelation of the linear prediction coefficients, in this case derived from  $\text{cent}_{LR}(C_i)$ . For the purposes of this explanation it is useful to define

$$r^c[i] = \text{cent}_{LR}(C_i), \quad (4.1)$$

where  $\text{cent}_{LR}(C_i)$  is the average of the gain-normalized autocorrelation coefficients associated with cell  $C_i$  (see Table 4.4) and  $r^c[i]$  denotes the autocorrelation coefficients of centroid  $i$ . The  $r^c[i]$  are used to compute the prediction coefficients for the centroid and  $r^a[i]$ . Experimental evaluation of centroid splitting by applying small random perturbations to the  $r^a[i]$  led to the conclusion that it is not possible to split a centroid consistently into two centroids that are 'close' in a  $d_{LR}$  sense, and so the training vectors cannot be partitioned sufficiently evenly. All of the training vectors tend to match to one centroid because there is no guarantee once  $r^a[i]$  has been perturbed in some random fashion, that the new  $r^a[i]$  is physically realizable (produced from stable prediction coefficients).

Another method of splitting the centroid is to convert it into a set of coefficients that are insensitive to minor perturbations, and then to perturb the alternative coefficients. Juang *et al.* (1982) converted the centroid to reflection coefficients and perturbed the reflection coefficients by multiplying them by 0.99 and 1.01 respectively. The paper contains no further details about how Juang's splitting was carried out. However, the reflection coefficients must be converted to prediction coefficients before  $r^a[i]$  is calculated.

The approach taken here is to split the centroid of the LPC prediction coefficients by adjusting each coefficient of  $r^c[i]$  by a uniformly distributed random quantity that is

defined to range between -0.0025% and 0.0025% of its original value. These ‘perturbed’ autocorrelation coefficients are then used to compute prediction coefficients. From these prediction coefficients the ‘split’  $r^a[i]$  are computed.

It is interesting to examine how this method differs from Juang’s in the number of internal conversions between parameter types and the additional storage requirements for keeping ‘extra’ copies of the centroid vector in another form.

The advantage of perturbing autocorrelation coefficients instead of reflection coefficients is that the conversion between reflection and prediction coefficients, as performed by Juang *et al.* (1982), is avoided. However, this method requires that the autocorrelation coefficients of the centroid,  $r^c[i]$ , should be stored along with the autocorrelation of the centroid prediction coefficients,  $r^a[i]$ , since it is impossible to compute the  $r^c[i]$  from the  $r^a[i]$ . Splitting centroids by using reflection coefficients has a similar requirement, since it is necessary to store the reflection coefficients for each centroid.

As part of the splitting process it is possible to have a cell with no training vectors in it. Such a cell does not reduce the aggregate quantization error and causes the iterative partitioning and centroid updating algorithm to work incorrectly. The approach taken here is to notify the user that an empty cell exists, and then to set the centroid associated with that cell to be a perturbed version of the centroid whose partition contains the most training vectors. Informal trials showed that for  $d_{LR}$  clustering of five training utterances, empty cells occur occasionally for 32 and 64 vector codebooks.

#### 4.2.2.3 Verification of the VQ training algorithm

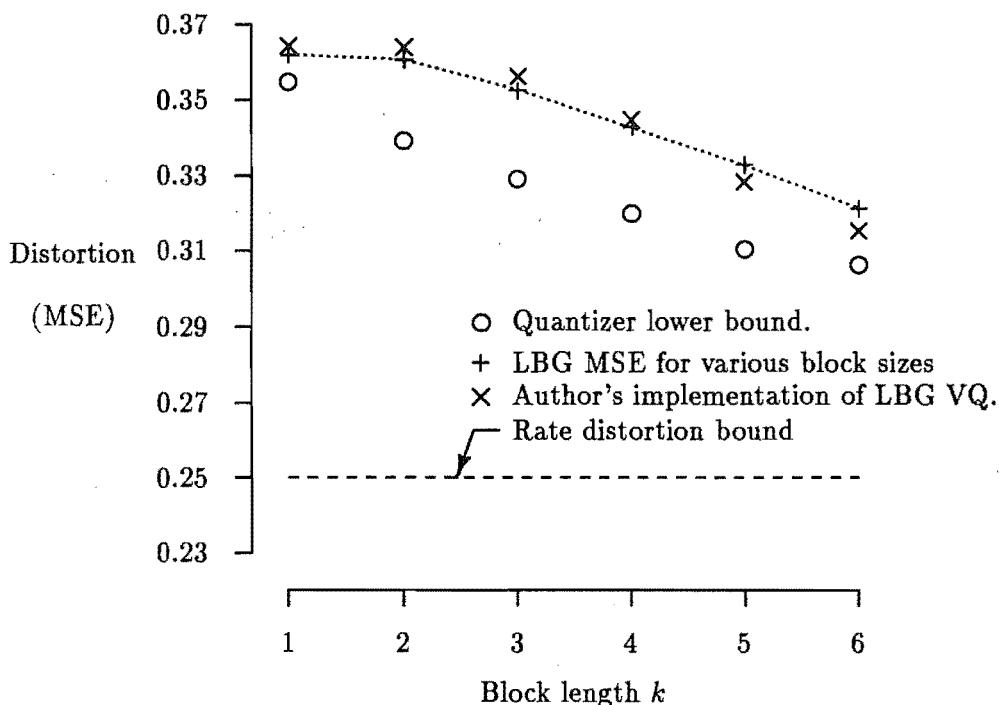
Much of the speaker identification work reported in §5.4 relies on the use of the LBG codebook training algorithm to calculate codevectors for each speaker. It is therefore vital that the implementation of the LBG training algorithm performs properly. The training algorithm is written in Fortran as an extension to SGPRC (Brieseman *et al.*, 1989), a signal processing package that runs on the Departmental VAX. Examples that employ the LBG training algorithm to design codebooks for several well-defined source distributions are presented in order to verify the operation of the implemented algorithm.

Max (1960) tabulates the mean-square error (MSE) caused by quantizing a memoryless Gaussian source with optimal scalar quantizers having different numbers of quantization levels. Since a scalar quantizer is a special case of a vector quantizer, it is valid to test the codebook training algorithm by examining the total distortion between the sequence of training samples and the final codebook. LBG tested their algorithm by performing VQ for several different numbers of quantization levels and here the author’s computer program is tested in the same manner. Table 4.5 contains the distortion figures for scalar quantization of 20,000 samples of a memoryless Gaussian source for 2, 4, 8, 16 and 32 levels by the quantizer determined by the author’s implementation of the LBG training algorithm. The optimal values determined by Max (1960) are also listed, and from the table it is apparent that the implementation of the LBG algorithm performs well, as it gives distortion values close to those obtained by Max.

LBG also test their VQ training algorithm by performing block quantization of a memoryless Gaussian source for block (or vector) lengths  $k$  equal to 1, 2, 3, 4, 5 and 6 samples and at a rate of 1 bit per sample. For a rate of 1 bit per sample the codebook size is  $2^k$ . The quantization distortions of the block quantizers are determined for the various block lengths and are illustrated in Fig. 4.3. The distortion levels match closely with those determined by Linde *et al.* (1980). An exact difference cannot be measured between the distortions reported here and those presented by LBG, since LBG do not give the exact distortion figures (LBG results in Fig. 4.3 are estimated from a figure in

| Number of quantization levels | Max's error | Author's error |
|-------------------------------|-------------|----------------|
| 2                             | 0.3634      | 0.3575         |
| 4                             | 0.1175      | 0.1174         |
| 8                             | 0.03454     | 0.03453        |
| 16                            | 0.009497    | 0.009400       |
| 32                            | 0.002499    | 0.002525       |

**Table 4.5.** A comparison between the MSE of applying two different quantizers to a memoryless Gaussian source. Max's error is the MSE associated with quantizing the source with the optimal quantizer, as computed from the Gaussian distribution and described in Max (1960). The author's error is the MSE after training a quantizer using 20,000 samples from a Gaussian source and the codebook training algorithm described by Linde *et al.* (1980).



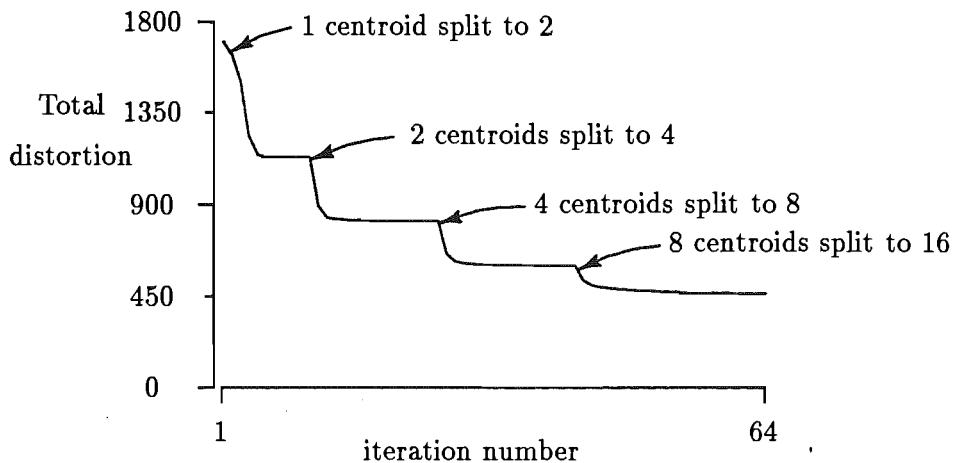
**Figure 4.3.** Distortion for block quantization of a memoryless Gaussian source at a rate of 1 bit/symbol. The author's quantization was performed on a sequence of 50 000 samples taken from a memoryless Gaussian source.

their paper). Nevertheless, the trends in the results indicate that the implementation of the LBG clustering algorithm used here to derive codebooks performs in a similar way to LBG's original algorithm.

Although it is difficult to prove that a complicated computer program performs correctly, it appears from the examples of quantizing a Gaussian source in various ways that it performs almost identically to that described by LBG. This makes the author confident that the implementation of the LBG clustering algorithm is correct.

#### 4.2.2.4 Examples of applying the VQ training algorithm to speech

In this section the application of the LBG training algorithm to speech is examined. The Gaussian source described in the previous section has a better defined sample



**Figure 4.4.** An example of the changes in total distortion (defined in (2.94)) as the LBG training algorithm progresses ( $\epsilon = 0.05$ ). The features used are 12 reflection coefficients taken from five utterances of the digits zero to nine by speaker JK.

distribution than would be expected from a sequence of PARCOR vectors representing frames of speech. For this reason it is important to check the performance of the LBG algorithm on actual speech.

The application of the LBG algorithm to speech is examined by recording the total quantization distortion at each iteration of the LBG algorithm. Fig. 4.4 shows the decrease in quantization error for PARCOR coefficients as the centroids are adjusted and split. A sudden drop in the quantization distortion occurs whenever the number of centroids is doubled, since a large number of centroids can represent the training vectors with less distortion than a smaller number of vectors. The other reductions in distortion occur as the centroids are iteratively moved to more ‘stable’, lower error positions within the vector space. Iterations continue until the difference between the current error and the previous error is less than  $\epsilon$ .

VQ training is complete when the sequence of training vectors has been divided up into the required number of cells and the iterations do not improve the error by more than  $\epsilon$ . As training progresses the distribution of the training vectors amongst the cells can be examined to check that there are no obvious abnormalities, such as cells with only one or two training vectors in them. The example distributions of Table 4.6 are the distribution of vectors among cells as VQ training progresses on five utterances of the digits zero to nine. The cell distributions just before the centroids are split (i.e. the centroid positions are considered ‘optimal’) are tabulated. It is apparent from Table 4.6 that there is considerable variation in the number of vectors in each cell, particularly when the codebook size is 16. However, the number of vectors in each cell does not highlight any obvious abnormalities in the algorithm, and is probably reasonable.

The implementation of the VQ training algorithm seems to work correctly for vectors extracted from real speech. Furthermore, the results reported in Chapter 5 reinforce the conclusion that codevectors determined by this algorithm are accurate representations of a person’s speech.

### 4.3 THE LONG-TERM AVERAGE GLOTTAL RESPONSE

This section describes details of the LTAGR calculation and representation. In §4.3.1 the implementation of real-time SAA algorithms for determining the LTAGR is described, and in §4.3.2 descriptors of the characteristics of the LTAGR are defined.

| Codebook size | Number of training vectors in each cell after iterations are completed. |      |     |     |
|---------------|---|------|-----|-----|
| 2             | 1396  | 1462 |     |     |
| 4             | 992   | 504  | 728 | 636 |
| 8             | 480   | 311  | 612 | 194 |
|               | 283   | 507  | 252 | 225 |
| 16            | 243   | 178  | 182 | 91  |
|               | 371   | 321  | 62  | 121 |
|               | 161   | 174  | 195 | 159 |
|               | 182   | 165  | 138 | 129 |

Table 4.6. The distribution of training vectors within cells as the LBG training algorithm progresses. The number of training vectors in each cell is recorded after the iterative updating of the centroids has converged for the specified number of codevectors.

Observation of the LTAGRs calculated using the real-time system (§4.3.1.1) led to the question whether or not the accent of a speaker has any identifiable effect on the LTAGR, and this is examined in §4.3.3.

#### 4.3.1 Real-time shift-and-add

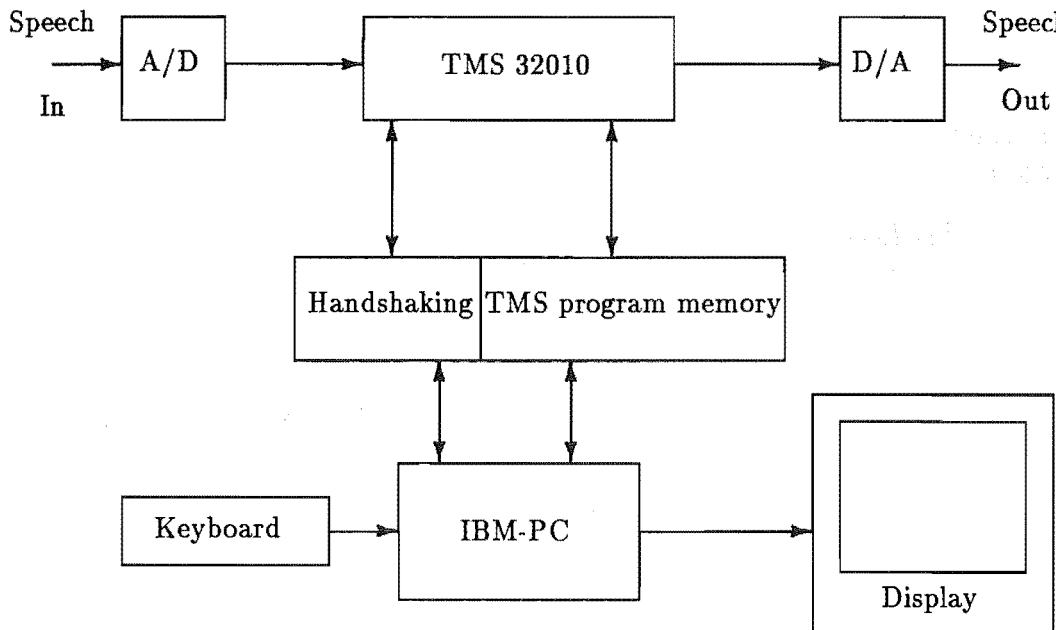
Real-time SAA was developed to facilitate the examination of LTAGRs. This led to the discovery that people tend to have distinctive LTAGRs, prompting research into the usefulness of the LTAGR for speaker identification. Another interesting observation arising from the development of the real-time SAA algorithm was that the voiced/unvoiced decision could be approximated by examining the amplitude of the largest peak in each consecutive speech frame. This is described in more detail in §4.3.1.2.

##### 4.3.1.1 Hardware

In order to perform signal processing operations at a rate fast enough to give the appearance of real-time operation, either a powerful computer or specialized hardware is required. The approach taken here is to use special purpose hardware in the form of a digital signal processor (DSP). The DSP hardware described here was originally designed by Turner (1986) and is constructed on a single card that plugs into one of the accessory slots of an IBM XT compatible PC. Five of these DSP systems were constructed for use in speech therapy research, and in order to simplify their construction the author redesigned aspects of Turner's hardware. The main simplification was the separation of the analog and digital sections on two wire-wrap boards. These DSP systems are currently operating successfully in the speech therapy project (Watson *et al.*, 1990).

Fig. 4.5 shows a block diagram of the real-time processing system. For a more detailed diagram of the DSP board see Elder *et al.* (1987). The DSP used is the TMS 32010 (hereafter referred to simply as 'TMS'), which performs integer operations only. Although this device has since been superseded by more powerful DSPs, it remains useful for performing simple signal processing tasks where it is important to keep costs low.

In the real-time system depicted in Fig. 4.5, the TMS is used for performing computationally intensive digital signal processing operations. The results of these computations are passed to the PC for display or other action. The input to the TMS is



**Figure 4.5.** Real-time signal processing hardware.

through a microphone and a 12 bit A/D converter. The PC accepts user input from the keyboard and controls the program run by the TMS.

Software for the real-time system is written in a combination of languages. On the PC a high-level language such as PASCAL or MODULA-2 is used. Callable library routines for both these languages have been written to allow programs to be loaded into the TMS and data transferred between the PC and TMS. Software on the TMS is written entirely in assembler code.

#### 4.3.1.2 Real-time shift-and-add implementations

This subsection discusses two real-time SAA algorithms that differ both in the way the peaks are detected and in the dynamic range of the accumulator used to store the long-term average glottal response. Much of the terminology used in this section is based on the description of the SAA algorithm in §2.8. For the purposes of this description the two implementations of SAA are called SAA1 and SAA2. Although it is apparent in the following description that SAA2 is superior to SAA1, SAA1 is described because it was the original algorithm used to investigate the differences in speakers' long-term average glottal responses (as calculated by SAA).

The SAA1 algorithm has a 16 bit accumulator and the SAA2 a 32 bit accumulator to store the current long-term average glottal response. If the amplitude of the incoming speech is at its maximum for the 12 bit A/D, the 16 bit accumulator would only be able to store 16 'pulses' before it overflowed. For this reason SAA1 divides each sample by 16 before it is added to the accumulator. Even with this division only 256 pulses of maximum amplitude can be added to the accumulator before it overflows. The 32 bit accumulator of SAA2 removes any likelihood of overflow. Without any scaling the accumulator has sufficient dynamic range to accommodate over one million samples of the maximum A/D output. Scaling, however, may be performed for other reasons, as described later in this section.

Both of the implementations of the SAA algorithm have the same interface to the PC and add speech frames to a SAA accumulator in a similar way, the only difference being the dynamic range of the accumulator, as outlined in the previous paragraph.

The TMS software that interfaces to the PC follows the sequence of steps specified below:

**Get parameters:** Get parameters from the PC that control the operation of the SAA algorithm. The parameters passed are:

- A threshold value that the maximum peak in a search frame must exceed before the frame is considered to be valid.
- The size of each SAA frame - specified in samples.
- The number of glottal pulse frames to collect and average.
- The frame step. Sets how far the next search frame is moved from the last identified glottal pulse.

**Wait for peak:** Loop until a peak is located by the interrupt routine (on the TMS) handling incoming samples.

**Add:** Add a SAA frame with the located peak in the centre to the accumulator for storing the long-term average glottal response.

**Check PC:** If the PC is ready, send the latest values in the accumulator (i.e. latest estimate of the long-term average glottal response) to the PC for display.

**Loop:** Loop back to the **Wait for peak**.

Note that the important parameters governing the operation of the SAA algorithm are set by the PC. The real-time SAA algorithm is therefore extremely flexible, since operating parameters can be set by the PC before the SAA algorithm starts to run.

The real-time SAA algorithms are organized to process speech samples while a person speaks. As each new sample is obtained from the A/D, it is placed in a circular buffer and examined to see whether it represents a peak in the input signal. The methods used for detecting peaks in the input speech differ between the two SAA implementations, as described in the next paragraph.

SAA1 uses a combination of a peak threshold and peak-finding algorithms to locate the position of the dominant peak within each frame of the speech signal. The peak threshold is used to ensure that the frame contains voiced speech. The peak threshold used by SAA1 is not constant. When a peak is located, the threshold is set to the peak value and as each successive sample is received the peak threshold is multiplied by 0.999, having the effect of reducing the threshold to 0.89 of the previous peak value after 100 samples. In order to exclude noise and unvoiced sounds from being included in the SAA calculation a lower bound threshold is specified by the PC. This use of a threshold that reduces at a fixed rate assumes that the amplitude of the maximum peak within the speech signal does not reduce more rapidly than the rate of reduction of the peak threshold. Since the SAA1 algorithm computes a long-term average and is reasonably insensitive, therefore, to the omission of a number of 'pulses', the occasional instance of the speech reducing too rapidly will not have a significant effect on the final long-term average glottal response.

The SAA1 algorithm works satisfactorily and demonstrates the usefulness of using a peak threshold to perform an approximate voiced/unvoiced decision. However, the algorithm is limited in that only positive peaks are detected by the peak searching algorithm, and the first peak after the threshold has been exceeded is selected for SAA frame alignment. Accordingly, the SAA1 algorithm is not the best possible implementation of SAA, and the deficiencies in the SAA1 led to the development of the SAA2 algorithm.

The SAA2 algorithm uses a peak-detection algorithm that determines the peak with the highest absolute amplitude in the SAA speech frame (the size of this frame is set by the PC and is typically 128 samples). The sign of the largest peak is recorded so that the speech frame can be added to the SAA accumulator and the central peak positively reinforced.

Input levels that exceed the dynamic range of the A/D are detected by monitoring the maximum and minimum sample values received from the A/D. The maximum and minimum sample values are not used in the SAA algorithm, but when passed to the PC for display, allow a user to monitor whether they are speaking too loudly.

The output of both the SAA1 and SAA2 algorithms is plotted on the PC screen. The display is updated as a person speaks. At first the shape of the averaged glottal response varies wildly as different sounds are spoken. Soon it stabilizes, as each successive speech frame has a relatively smaller effect on the values in the SAA accumulator. As frames of speech are added to the SAA accumulator, the range of amplitudes contained in the accumulator increases, so a new scale factor is determined for each set of SAA accumulator values before the LTGAR is plotted on the PC screen. The accumulator in the SAA2 algorithm consists of 32 bits, and all 32 bits are passed to the PC. The 32 bit number is converted to a REAL number before being scaled and plotted on the PC screen.

#### 4.3.1.3 Comments

The real-time SAA was used for informal speaker identification to evaluate the usefulness of the LTGAR. The software on the PC had the capability of matching a LTGAR (using Euclidean distance) with stored speaker templates that contained the LTGARs of the test population. Identification trials on a small number of speakers gave approximately 80% accuracy, prompting further research into the usefulness of the LTGAR for speaker identification.

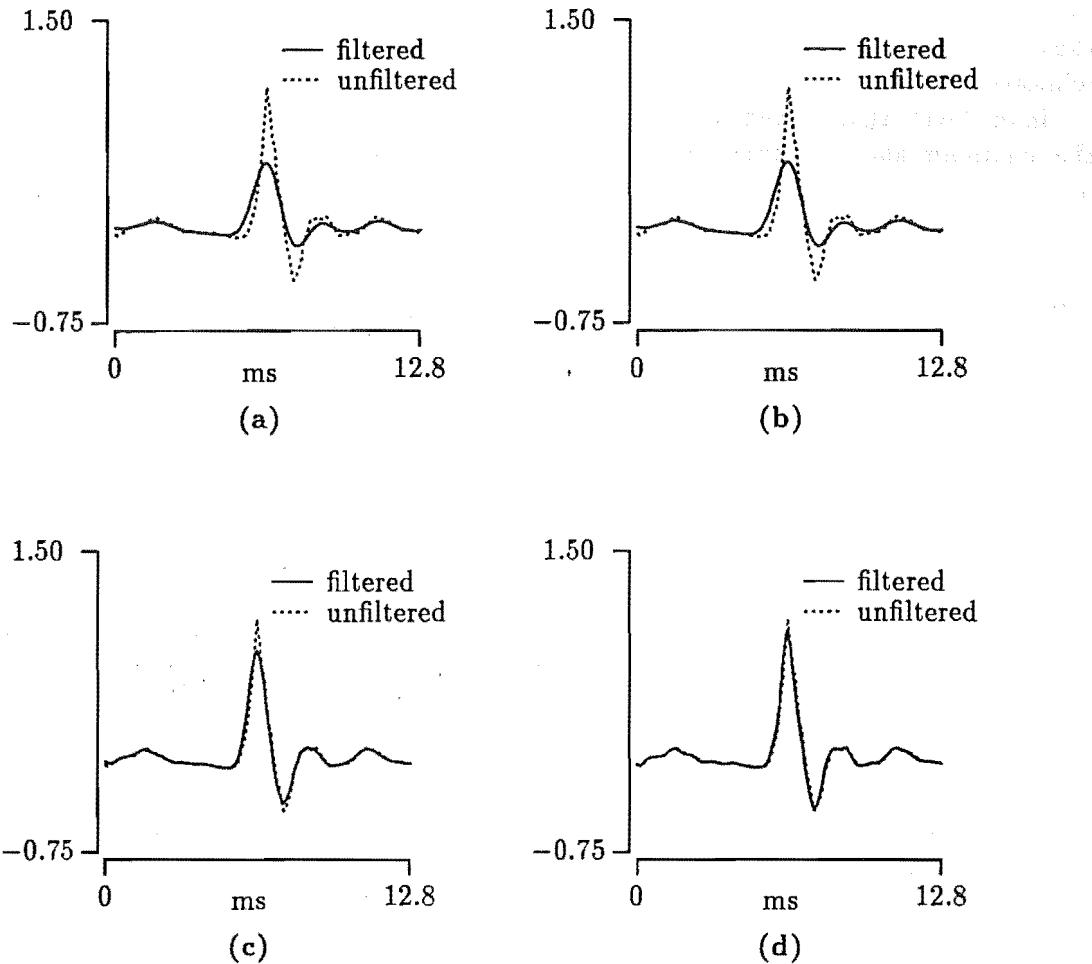
The real-time SAA algorithms worked well, but were not especially well suited to doing repeatable research, since the speech signal was not recorded simultaneously with a LTGAR(SAA) calculation. For this reason the real-time calculation of the LTGAR became a lower priority than the collection of the speech database described in §4.1, and real-time SAA analysis became a demonstration tool only. The digitized database of twenty speakers served as a constant speech source for the many different features and matching methods described in this thesis.

Plots of the resulting LTGARs, as determined by the two real-time algorithms, are not included, since there is no reference LTGAR to make comparisons with. Since SAA2 contains the more sophisticated algorithm, one would expect identical results to those obtained from SGPRC, provided the signals, after digitizing, are exactly the same. However, this test is impracticable owing to the different phase responses of the anti-aliasing filters on the SX10 and TMS32010 cards, coupled with the phase sensitivity of the SAA algorithm (see §5.5.2.3).

The two algorithms discussed here demonstrate that it is feasible to implement SAA in real time using a simple and cheap DSP. Furthermore, the algorithm SAA2 demonstrates that it is possible to use the peak searching and framing algorithm, as specified in Algorithm 2.3 and utilized in SGPRC.

#### 4.3.2 Descriptors of the long-term average glottal response

This section describes descriptors that are abstracted from a SAA estimate of the long-term average glottal response(LTGAR). These descriptors are defined so as to encode features of the LTGAR that are useful for characterizing its shape. They are used for



**Figure 4.6.** A comparison between the application of different smoothing filters to the LTAGR derived from the phrase PK1 and low-pass filtered at: (a) 10 Hz, (b) 100 Hz, (c) 1000 Hz and (d) 2000 Hz.

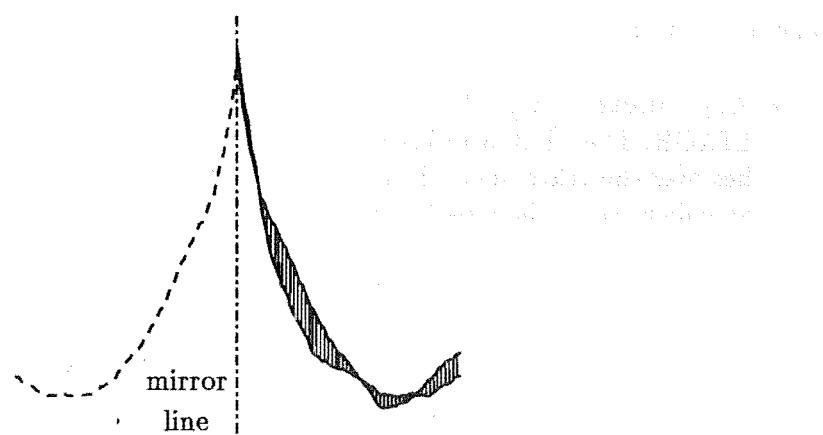
examining variation in the LTAGR with changes in accent and gender (§4.3.3) and also for identifying speakers (§5.4.2.1).

Some descriptors characterize information derived from ‘peaks’ in the LTAGR, necessitating an accurate definition of what constitutes a peak. In practice it is possible for a peak to be produced by a single sample having a significantly different value from the surrounding samples. However, here a peak is considered to be a visually obvious trend in the LTAGR, and therefore comprised of a number of samples. Small, rapid fluctuations are filtered out of the LTAGR by low-pass filtering the signal at a frequency of 1000Hz. The cut-off frequency of 1000 Hz was chosen after evaluating several different frequencies, as depicted in Fig. 4.6. A 1000 Hz cut-off is a compromise between low cut-off frequencies, which significantly distort the LTAGR shape, and high frequency cut-offs, which do not filter out all the high frequency fluctuations.

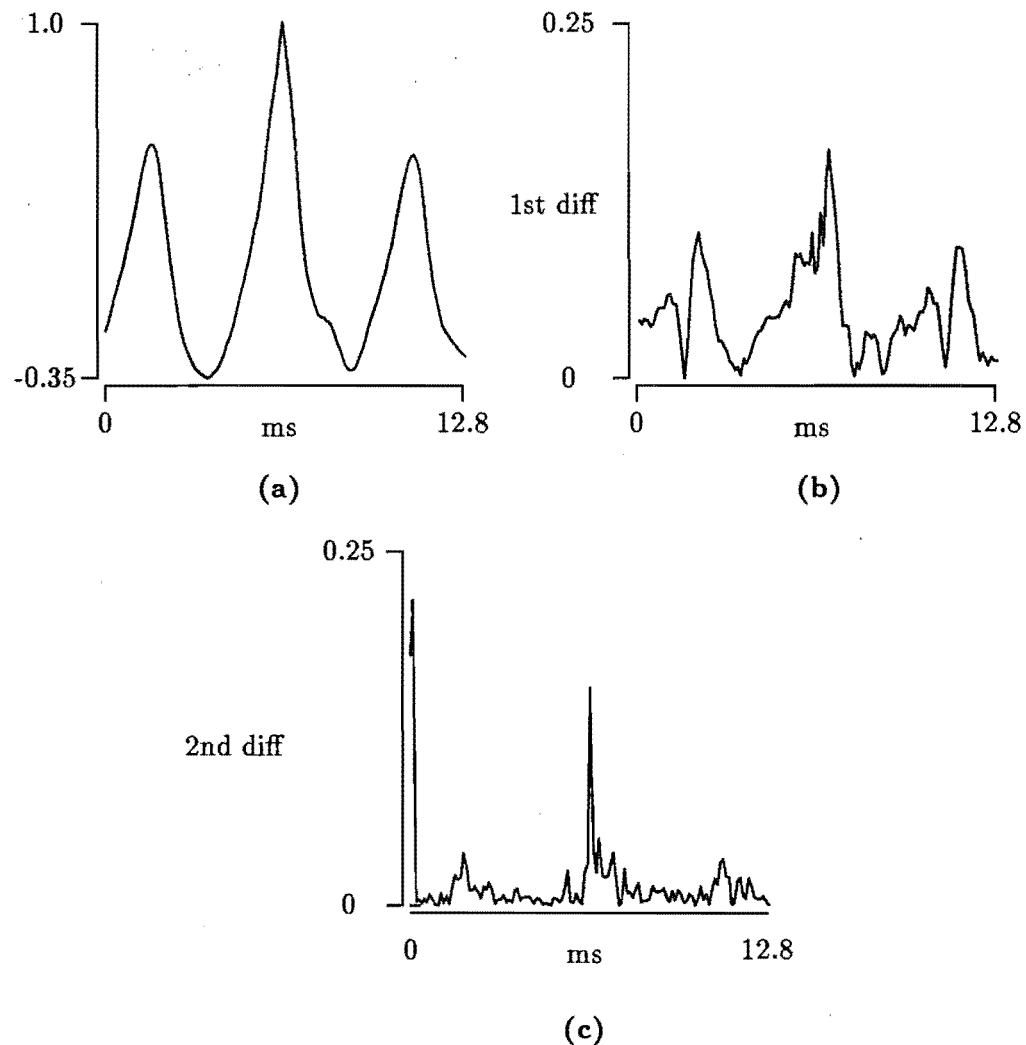
For the purposes of defining the descriptors presented here, the SAA estimate of the long-term average glottal response is defined by the sampled signal  $s_{saa}[n]$  where  $n$  ranges from 1 to 128. The main peak of the SAA signal is positioned at  $n = 64$ . All speech is sampled at 10 kHz so the duration of the SAA signal is 12.8 ms. Throughout this discussion of descriptors the terminology ‘left side’ and ‘right side’ is used to describe the ‘sides’ of the LTAGR taken about the central peak. The superscript  $l$  or  $r$

are used to indicate that the descriptor applies to the *left* or *right* side of the LTAGR. The following paragraphs define each of the LTAGR descriptors in turn:

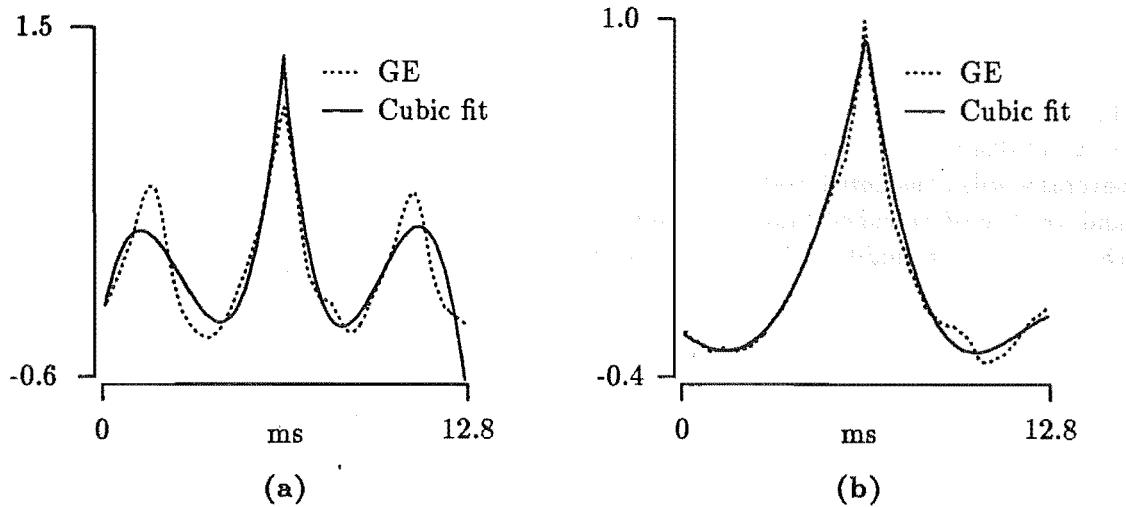
- **Asymmetry ( $A_s$ )**- Measures the difference between the left and right side of a LTAGR. The LTAGR is folded down its center line so that one side of the LTAGR lies over the other side. The area between the overlap of the two sides is evaluated as indicated by the crosshatching in Fig. 4.7.
- **Absolute amplitude(right) ( $A^r$ )** - Measures the average absolute amplitude of the right side of the LTAGR. This indicates whether the LTAGR falls from its main peak to zero or has samples of significant amplitude on the right side.
- **Absolute amplitude(left) ( $A^l$ )** - Corresponds to the absolute amplitude (right) above, but for the left side of the LTAGR.
- **Slope ( $S_1$ )**- The average of the absolute value of the first derivative of the LTAGR. A LTAGR with more than one significant peak, such as that illustrated in Fig. 4.8(a), has a higher average slope than a LTAGR with only a single dominant peak. Here the slope of a LTAGR is approximated by the first order difference, resulting in the signal depicted in Fig. 4.8(b).
- **Slope of slope ( $S_2$ )**- The average of the absolute value of the second derivative of the LTAGR, calculated from first order differences. Fig. 4.8(c) shows an example of the second derivative of a LTAGR.
- **Number of peaks ( $N_p$ )** - The number of peaks in the filtered LTAGR.
- **Left side peaks ( $N_p^l$ )** - The number of peaks in the left side of the filtered LTAGR.
- **Right side peaks ( $N_p^r$ )**- The number of peaks in the right side of the filtered LTAGR.
- **Left side cubic coefficients ( $C_{c0}^l, C_{c1}^l, C_{c2}^l, C_{c3}^l$ )** - The coefficients of a cubic polynomial fitted to the left side of the SAA signal. The cubic coefficients are calculated so that the error between the LTAGR and the cubic polynomial is minimized, in a least squares sense. Fig. 4.9 depicts both a LTAGR and its cubic approximation.
- **Right side cubic coefficients ( $C_{c0}^r, C_{c1}^r, C_{c2}^r, C_{c3}^r$ )** - The coefficients of a cubic polynomial fitted to the right side of the LTAGR.
- **Second highest peak ( $P_2$ )** - The amplitude of the second highest peak in the LTAGR.
- **Left side highest peak ( $P^l$ )** - The amplitude of the highest peak in the left side of the filtered LTAGR.
- **Right side highest peak ( $P^r$ )** - The amplitude of the highest peak in the right side of the filtered LTAGR.
- **Rise time of central peak ( $T_R$ )** - The time (measured in samples) taken for the LTAGR signal to rise from half the amplitude of the central peak to the maximum amplitude (see Fig. 4.10).



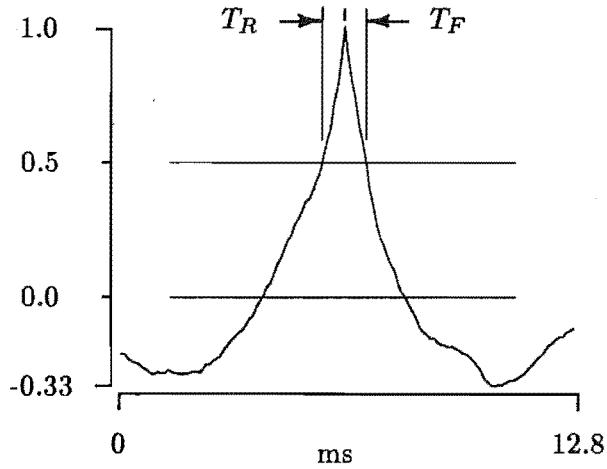
**Figure 4.7.** The shaded area between a reflected half of the long-term average glottal response and a non-reflected half is defined to represent the asymmetry measure for a particular LTAGR. In this example the LTAGR is derived from utterance AE1.



**Figure 4.8.** An illustration of the derivation of slope measures of the LTAGR: (a) original, (b) 1st difference and (c) 2nd difference.



**Figure 4.9.** An example of fitting cubics to the left and right side of the long-term average glottal response: (a) two separate cubics fitted to the left and right sides of the LTAGR of TC1, (b) two separate cubics fitted to the left and right sides of the LTAGR of AE1.



**Figure 4.10.** An illustration of the rise time,  $T_R$ , and the fall time  $T_F$  for the LTAGR.

- **Fall time of central peak ( $T_F$ )** - The time (measured in samples) taken for the SAA signal to fall from its maximum amplitude to half the maximum amplitude (see Fig. 4.10).

These descriptors are useful for reducing the number of dimensions of the LTAGR so that statistical analysis can be performed more easily. Statistical analysis examining the effects of accent and gender on the LTAGR descriptors is reported in §4.3.3. The usefulness of the various descriptors for speaker identification is assessed in §5.4.2.1.

#### 4.3.3 Variation of the long-term average glottal response with changes in accent and gender

In this section the variation in the LTAGR with changes in accent is examined using a database of speech collected from several different regions in the United States. American speech was selected because of the availability of a large digitized database on CD-ROM.

The particular database utilized is the prototype, December 1988, DARPA TIMIT

acoustic phonetic continuous speech database produced by Texas Instruments (TI), Stanford Research Institute (SRI) and Massachusetts Institute of Technology (MIT). The database contains a training set of utterances consisting of 4200 sentences spoken by 420 talkers. The SRI portion of the database consists of two sentences that contain several words considered useful for distinguishing between groups with different dialects, and are therefore called ‘dialect-shibboleth’ sentences (‘shibboleth’ means a test word that reveals nationality). The ‘dialect-shibboleth’ sentence, “She had your dark suit in greasy wash water all year” is spoken by 352 speakers and is used to evaluate the effect of accent and gender on the LTAGR.

All of the utterances recorded in the TIMIT database are digitized using 16 bit samples at a rate of 16 kHz (DARPA, 1988). Resampling to 10 kHz is performed, so a LTAGR of 128 samples corresponds to 12.8 ms of the speech signal. Four zero-valued samples are inserted between each sample and the signal low-pass filtered to 4.5 kHz using a 256 tap FIR filter. Decimation to a 10 kHz sampling rate is then performed by selecting every eighth sample.

In principle, comparison of LTAGRs of New Zealand speakers with those of American speakers is valid if the speech is sampled at the same rate. However, differences between the anti-aliasing filters of the TIMIT database and that used for the N.Z. database make it difficult to ascertain whether the differences in the LTAGR are due to the different recording conditions or different accents. Section 5.5.2.3 shows that the LTAGR is sensitive to phase distortion, so without detailed specifications of the phase response of the two recording systems it is not possible to remove the effects of the recording system from the LTAGR. Therefore, the evaluation of the effect of accent and sex on the LTAGR is performed entirely on speech extracted from the TIMIT database.

Table 4.7 lists the different regions recorded in the TIMIT database, and the number of utterances (ranging between 8 and 37) recorded for each region. The small number of utterances belonging to the Mobile Region is insufficient to perform classification experiments. However, this region is not as important as the others because these utterances are recorded from people that have moved round, and cannot be classified into one of the other accent regions.

#### 4.3.3.1 Factor analysis

Factor analysis, as described in §3.4.3, is applied to the LTAGR descriptors to examine which of the descriptors are strongly related, and to determine how many factors are required to represent the information contained in the LTAGR descriptors. Factors obtained from analysing the LTAGR descriptors using the statistical package SAS (SAS, 1985b) are listed in Table 4.8. The strength with which a particular descriptor is present in a factor is referred to as the *factor loading*. In order to identify common variables more clearly the factors have been rotated using the varimax rotation (Comrey, 1973, §7.4) (see §3.4.4), which rotates the factor axes to make the factor loadings either as large (1.0) or as small (0.0) as possible. The factor loadings for LTAGR descriptors abstracted from 352 American speakers are tabulated in Table 4.8.

Along with the factor loadings, Table 4.8 indicates the proportion of the total variance explained by each factor (as derived from the eigenvalues of the associated eigenvector). Factors 1, 2 and 3 explain 83.8% of the total variance; since the next largest factor (not shown in Table 4.8) contributes only 4%, only the first three factors are retained.

The first factor describes 67.7% of the total variance and is composed primarily of variables that describe the shape of the LTAGR. In particular, the cubic coefficients are highly correlated with factor 1. Furthermore, factor loadings for cubic coefficients that

| Region        | Sex    | Number |
|---------------|--------|--------|
| New England   | Male   | 21     |
| New England   | Female | 16     |
| Northern      | Male   | 34     |
| Northern      | Female | 18     |
| North Midland | Male   | 37     |
| North Midland | Female | 15     |
| South Midland | Male   | 33     |
| South Midland | Female | 19     |
| Southern      | Male   | 25     |
| Southern      | Female | 26     |
| New York City | Male   | 21     |
| New York City | Female | 11     |
| Western       | Male   | 34     |
| Western       | Female | 18     |
| Mobile        | Male   | 16     |
| Mobile        | Female | 8      |

Table 4.7. The regions used for classifying utterances in the TIMIT database and the number of utterances recorded from each region.

are positive for one side of the LTAGR are negative for the other side. For example, the loading of  $C_{c3}^l$  is large and positive whereas the loading for  $C_{c3}^r$  is large and negative. This is caused by the cubics fitted to the left and right sides of the LTAGR being almost mirror images of each other, reflected about the central peak. Other descriptors with large factor loadings represent different aspects of the shape of the LTAGR. Factor 1 therefore represents the general shape of the LTAGR. Factor 2 has large factor loadings for variables that describe the steepness of the central peak and the number of peaks on the right side of the central peak. The 3rd factor has most significant loadings for the  $N_p^l$  and  $N_p^r$  variables and is thus highly correlated with the number of peaks in the LTAGR.

The proportion of a descriptor's variation that is extracted by the factors is called the communality (Cooley and Lohnes, 1971, p109). A descriptor whose variation is well represented by the set of factors will have a communality close to 1.0. The communality of all the descriptors increases (up to a maximum of 1.0) as more factors are extracted from the dataset, since the variation of each descriptor is represented more accurately by having more factors (Gorsuch, 1983, p103). Table 4.9 contains the communality values for the three factors specified in Table 4.8. The descriptors with communalities less than 0.60 are shown in bold type. Since thirteen of the twenty-one descriptors have communalities greater than 0.9, clearly the variation of most of the descriptors is well represented by three factors. This is also implied by the fact that the first three factors account for 83.8% of the total variance in the LTAGR descriptors.

Scatter plots are used to examine the positions of observations along the factor axes. Scatter plots between paired combinations of factors 1, 2 and 3 (Fig. 4.11) indicate that variation of the descriptive values cannot be attributed to clusters occurring within the rotated factor space. Each LTAGR is represented by a single marker in the scatter diagram, with the marker's position along a particular factor axis specified by summing the contribution from each LTAGR descriptor multiplied by the correspond-

| Descriptor          | Factor 1      | Factor 2      | Factor 3     |
|---------------------|---------------|---------------|--------------|
| $C_{c3}^l$          | <b>0.969</b>  | 0.057         | 0.118        |
| $C_{c0}^r$          | <b>0.956</b>  | 0.057         | 0.125        |
| $C_{c2}^r$          | <b>0.948</b>  | 0.039         | 0.136        |
| $C_{c1}^l$          | <b>0.941</b>  | 0.042         | 0.186        |
| $A^r$               | <b>0.926</b>  | 0.262         | 0.030        |
| $A^l$               | <b>0.924</b>  | 0.259         | 0.063        |
| $P^r$               | <b>0.894</b>  | 0.319         | 0.088        |
| $P^l$               | <b>0.894</b>  | 0.319         | 0.088        |
| $S_1$               | <b>0.872</b>  | 0.427         | 0.090        |
| $A_s$               | <b>0.842</b>  | 0.182         | -0.053       |
| $P_2$               | <b>0.816</b>  | 0.349         | -0.014       |
| $S_2$               | <b>0.757</b>  | 0.459         | 0.030        |
| $C_{c0}^l$          | <b>-0.830</b> | -0.019        | -0.275       |
| $C_{c3}^r$          | <b>-0.942</b> | -0.030        | -0.141       |
| $C_{c1}^r$          | <b>-0.952</b> | -0.048        | -0.130       |
| $C_{c2}^l$          | <b>-0.965</b> | -0.049        | -0.141       |
| $N_p^r$             | 0.109         | <b>0.690</b>  | 0.309        |
| $T_F$               | -0.063        | <b>-0.648</b> | -0.175       |
| $T_R$               | -0.138        | <b>-0.701</b> | -0.038       |
| $N_p^l$             | 0.156         | 0.242         | <b>0.884</b> |
| $N_p$               | 0.171         | 0.542         | <b>0.801</b> |
| Variance Explained  | 67.7%         | 11.5%         | 4.6%         |
| Cumulative Variance | 67.7%         | 79.2%         | 83.8%        |

**Table 4.8.** The factor loadings obtained from performing principal component factor analysis on LTAGR descriptors of 352 speakers from the TIMIT database. The factor loadings for only the first three factors are shown here and have been rotated using the varimax rotation (see text).

ing factor loading for that factor. Each observation in the scatter diagram is labelled with a marker that depends on the individual's gender. This allows the variation in the descriptors with speaker gender to be observed. It is apparent from the scatter plots in Fig. 4.11 that both male and female LTAGR descriptors vary by a similar amount and that no distinct clusters have formed. Most of the observations are grouped in a single cluster comprising a mix of male and female speakers. Factor 3 should differentiate males from females, since it depends chiefly on the number of peaks on the right and left sides of the LTAGR and females tend to have additional peaks in their LTAGR compared with males (see §2.8.3). Therefore one would expect the larger values to be female and the smaller values to be male; however this is not the situation depicted in Fig. 4.11(b) and (c). Although small values (between -1 and -3) of factor 3 'tend' to be male, four females lie in the centre of the twenty-five or so males. At the other extreme of factor 3 large values might be expected to represent females, though this does not hold consistently. Again considering Fig. 4.11(c), one can observe that when factor 3 is large and factor 2 is approximately equal to -1 males predominate, whereas when

| Descriptor | Communality  | Descriptor | Communality  |
|------------|--------------|------------|--------------|
| $A_s$      | 0.744        | $C_{c3}^l$ | 0.956        |
| $A^r$      | 0.927        | $C_{c0}^r$ | 0.933        |
| $A^l$      | 0.925        | $C_{c1}^r$ | 0.926        |
| $S_1$      | 0.950        | $C_{c2}^r$ | 0.918        |
| $S_2$      | 0.784        | $C_{c3}^r$ | 0.909        |
| $N_p$      | 0.965        | $P_2$      | 0.788        |
| $N_p^l$    | 0.865        | $P^l$      | 0.909        |
| $N_p^r$    | <b>0.584</b> | $P^r$      | 0.909        |
| $C_{c0}^l$ | 0.765        | $T_R$      | <b>0.512</b> |
| $C_{c1}^l$ | 0.923        | $T_F$      | <b>0.455</b> |
| $C_{c2}^l$ | 0.954        |            |              |

**Table 4.9.** Communality estimates for the LTAGR descriptors using the factors defined in Table 4.8. Communalities less than 0.60 are in bold type.

factor 2 is large as well as factor 3 females predominate. This can be expressed more concisely by stating that when factor 2 + factor 3 > 3 the LTAGR tends to belong to a female. Apart from these regions in the scatter diagrams which have a predominance of a particular sex, most of the variance in the LTAGR from the trial of 221 males and 131 females seems to be unrelated to the sex of the speakers.

Correlations between a person's accent and the extracted factors are examined by selecting a marker according to the person's region and replotting the scatter plots using the key specified in Fig. 4.12(d). The scatter diagrams in Fig. 4.12(a),(b) and (c) do not show any significant correlation between the three most significant factors and the region that a speaker comes from.

The foregoing results show that the most significant factors contained in the LTAGRs are not directly related to either the sex or accent of the speaker. Possibly accent variations among speakers in the United States are insufficient to affect the variations in the LTAGR from other speaker differences. This bodes well for speaker identification because it implies that people with the same accent have LTAGRs that vary significantly and should therefore be differentiated enough to perform speaker identification.

#### 4.3.3.2 Discriminant analysis

Recall from §3.4.2 that the aim of discriminant analysis is to identify a discriminant function that classifies observations into appropriate groups (Cooley and Lohnes, 1971, §9). In contrast with factor analysis, which examines the relationships of descriptors across the whole set of observations, discriminant analysis determines the best method for classifying observations into a set of predefined groups. This is achieved by choosing a linear transformation of the descriptor coordinate system that minimizes the distance between observations belonging to the same group while maximizing distances between groups. The distance  $D(\mathbf{Z}, \bar{\mathbf{Y}}_i)$  in this discriminant space between an observation vector  $\mathbf{Z}$  and the mean  $\bar{\mathbf{Y}}_i$  of group  $i$  can be expressed as

$$D(\mathbf{Z}, \bar{\mathbf{Y}}_i) = (\mathbf{Z} - \bar{\mathbf{Y}}_i)' \mathbf{W}^{-1} (\mathbf{Z} - \bar{\mathbf{Y}}_i) \quad (4.2)$$

where  $\mathbf{W}$  is the pooled covariance matrix as defined in §3.4.2 (SAS, 1985b). Using the distance defined in (4.2) to classify each of the observations to the nearest group, a classification summary of all the observations can be obtained.

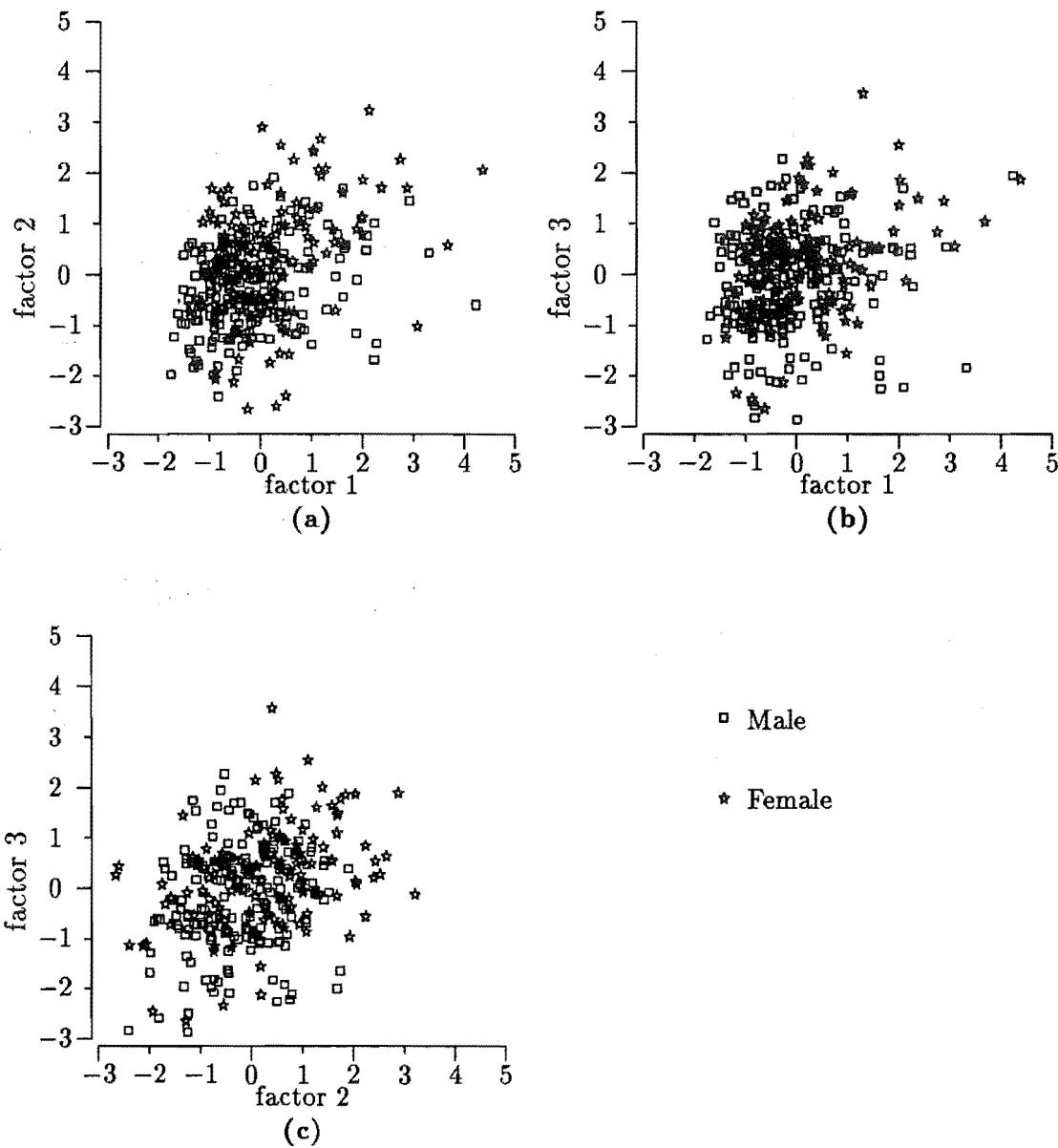
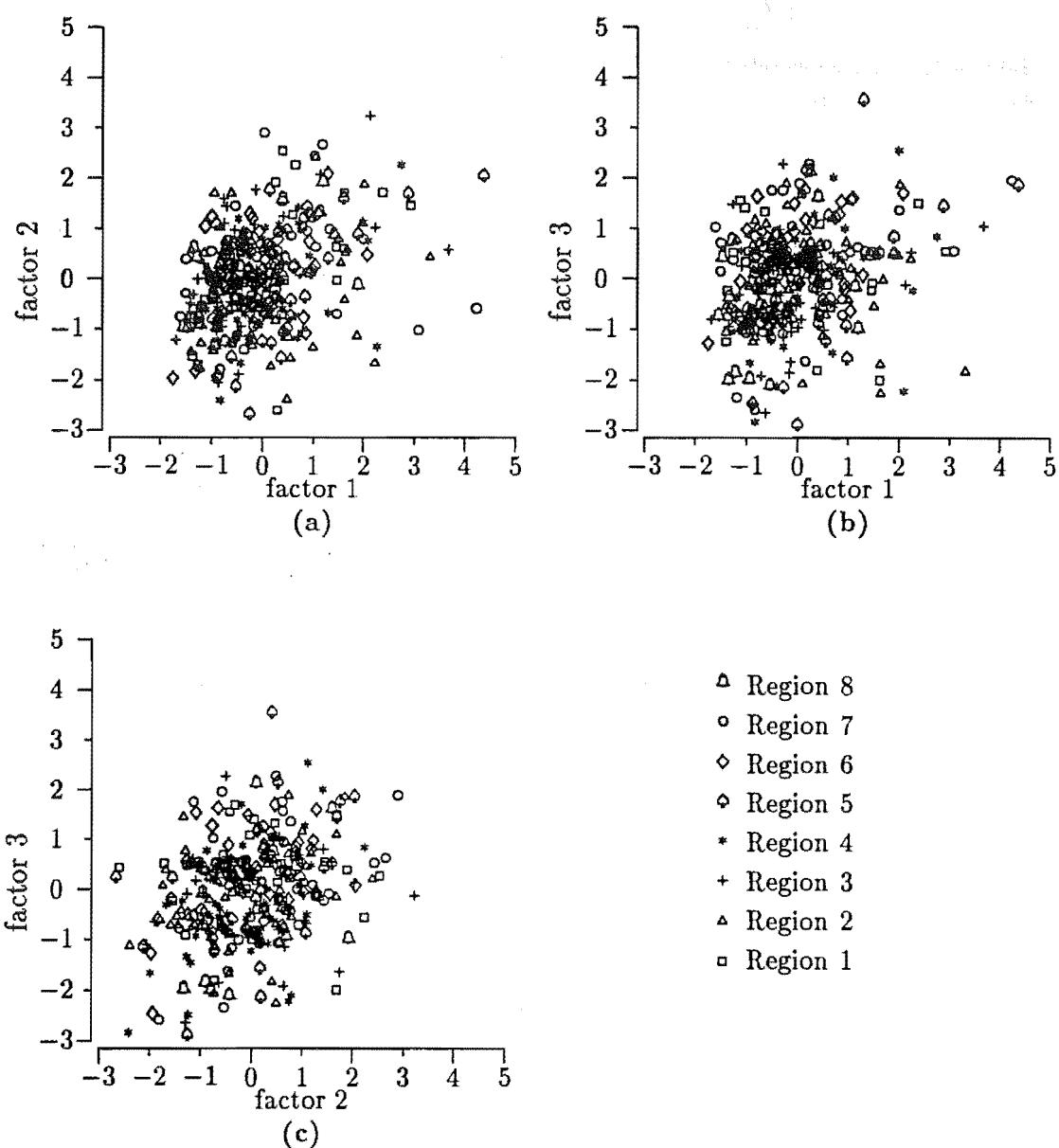


Figure 4.11. Scatter plots of factors 1, 2 and 3 labelled by sex.

Two separate accent classification experiments are reported here. The first experiment classifies speakers according to their accent region only, whereas the second experiment takes into account the sex of the speakers within each accent region.

Before performing discriminant analysis, stepwise discriminant analysis (see §3.4.2.1) is performed to find the subset of descriptors most useful for distinguishing between the accents. Only the descriptor  $N_p^l$  is selected, the other descriptors not contributing significantly to the discrimination. Table 4.10 contains the classification results from using this single descriptor. Clearly classification by accent alone is not feasible.

The next accent classification experiment separates the male and female speech from each region in the TIMIT database. This can be justified because the higher pitch of female voices tends to distinguish their LTAGRs markedly from male voices



**Figure 4.12.** Scatter plots of factors 1, 2 and 3 labelled by region.

| Error rate summary by region |      |      |      |      |      |      |      |      |
|------------------------------|------|------|------|------|------|------|------|------|
| Region                       | R1   | R2   | R3   | R4   | R5   | R6   | R7   | R8   |
| Error Rate                   | 1.00 | 1.00 | 1.00 | 0.44 | 1.00 | 0.28 | 1.00 | 1.00 |

**Table 4.10.** The error rates resulting from classifying LTAGRs by region using discriminant analysis. In the notation scheme used here R1 indicates Region 1, and so on for the other regions.

| LTAGR descriptors selected |            |            |            |
|----------------------------|------------|------------|------------|
| $C_{c0}^l$                 | $C_{c0}^l$ | $A_l$      | $C_{c1}^l$ |
| $C_{c2}^l$                 | $N_p$      | $C_{c0}^r$ | $A^r$      |
| $P^l$                      | $C_{c3}^r$ | $S_2$      | $N_p^l$    |

Table 4.11. Descriptors selected by using stepwise discriminant analysis to select descriptors that best distinguish between the accents (or regions) of the 352 American speakers from the TIMIT database.

| From Region | Classification Region |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |
|-------------|-----------------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
|             | R1M                   | R1F | R2M | R2F | R3M | R3F | R4M | R4F | R5M | R5F | R6M | R6F | R7M | R7F | R8M | R8F |
| R1M         | 6                     | 0   | 3   | 1   | 0   | 0   | 1   | 0   | 1   | 1   | 3   | 1   | 3   | 0   | 2   | 0   |
| R1F         | 0                     | 2   | 0   | 4   | 1   | 2   | 0   | 0   | 0   | 2   | 0   | 0   | 0   | 0   | 1   | 1   |
| R2M         | 2                     | 0   | 0   | 1   | 3   | 1   | 1   | 0   | 8   | 0   | 2   | 1   | 2   | 0   | 4   | 0   |
| R2F         | 0                     | 2   | 0   | 8   | 0   | 2   | 0   | 1   | 1   | 3   | 0   | 1   | 0   | 0   | 0   | 0   |
| R3M         | 2                     | 1   | 2   | 0   | 7   | 0   | 1   | 0   | 1   | 0   | 7   | 0   | 6   | 0   | 10  | 0   |
| R3F         | 0                     | 1   | 0   | 2   | 0   | 4   | 0   | 1   | 0   | 3   | 0   | 2   | 0   | 2   | 0   | 0   |
| R4M         | 3                     | 0   | 1   | 0   | 2   | 0   | 7   | 0   | 3   | 0   | 5   | 1   | 3   | 0   | 8   | 0   |
| R4F         | 0                     | 2   | 0   | 1   | 0   | 0   | 0   | 7   | 0   | 2   | 0   | 3   | 0   | 1   | 0   | 3   |
| R5M         | 0                     | 0   | 6   | 0   | 1   | 0   | 2   | 0   | 8   | 0   | 2   | 1   | 1   | 0   | 4   | 0   |
| R5F         | 0                     | 0   | 0   | 4   | 0   | 3   | 0   | 3   | 0   | 11  | 0   | 1   | 0   | 3   | 0   | 1   |
| R6M         | 1                     | 0   | 0   | 0   | 3   | 0   | 0   | 0   | 1   | 0   | 13  | 0   | 2   | 0   | 1   | 0   |
| R6F         | 0                     | 0   | 0   | 0   | 0   | 3   | 0   | 0   | 0   | 2   | 0   | 4   | 0   | 1   | 0   | 1   |
| R7M         | 4                     | 0   | 2   | 0   | 2   | 0   | 6   | 0   | 1   | 0   | 2   | 0   | 13  | 0   | 4   | 0   |
| R7F         | 0                     | 1   | 1   | 1   | 0   | 1   | 1   | 0   | 0   | 2   | 0   | 0   | 0   | 9   | 0   | 2   |
| R8M         | 2                     | 0   | 1   | 0   | 0   | 0   | 0   | 0   | 1   | 1   | 0   | 0   | 2   | 0   | 9   | 0   |
| R8F         | 1                     | 1   | 0   | 0   | 0   | 0   | 0   | 0   | 0   | 1   | 0   | 0   | 0   | 0   | 0   | 6   |

| Error rate summary by region (across rows of the above table) |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |
|---|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Region  | R1M | R1F | R2M | R2F | R3M | R3F | R4M | R4F | R5M | R5F | R6M | R6F | R7M | R7F | R8M | R8F |
| Error Rate  | .76 | .88 | .74 | .56 | .81 | .73 | .79 | .63 | .68 | .58 | .38 | .63 | .62 | .50 | .43 | .38 |

Table 4.12. A table of the confusion matrix resulting from classifying LTAGRs by region using discriminant analysis. In the notation scheme used here, R1F indicates Region 1, Female speech, and so on for the other regions.

(see §2.8.3). Stepwise discriminant analysis (§3.4.2.1) is again performed to determine the best LTAGR descriptors. Table 4.11 contains the twelve descriptors selected by applying stepwise discriminant analysis to all twenty-one LTAGR descriptors abstracted from the 352 American speakers. The nine descriptors not selected by the stepwise discriminant analysis do not contribute to the discriminatory power for identifying a person's accent region. In Table 4.12 is given the classification summary (or confusion matrix) for 352 utterances from the regions defined in Table 4.7, using the descriptors listed in Table 4.11. It is important to remember that, since the same set of descriptor observations are used for testing and training, the confusion matrix specifies how well the training data is separated and cannot be generalized to make statements about how accurate these descriptors might be for performing recognition. LTAGRs from many of the regions were classified with a lower error rate than the chance error rate of  $15/16=0.9375$ . However, none of the regions were classified without error, and in particular R1F and R3M have error rates greater than 0.80, approaching 'chance' classification error rate. Clearly the features abstracted from the LTAGR fail to classify American individuals into dialect groupings. It is not possible to determine from this analysis whether the inability to discriminate is because the descriptors do not represent the information in the LTAGR accurately enough or is because the LTAGR does not represent information about a person's accent accurately.

The speakers' accents are from a continuous accent range, so they would not all be expected to fall strictly into one of the specified regions. For example, a speaker from an area near the border of two regions could have an accent that falls somewhere between the average accent of the two regions.

The probability of an observation occurring from a region varies when the regions

| LTAGR descriptors selected |            |            |            |
|----------------------------|------------|------------|------------|
| $C_{c0}^l$                 | $C_{c0}^l$ | $C_{c1}^l$ | $A^l$      |
| $T_F$                      | $C_{c0}^r$ | $C_{c3}^r$ | $C_{c1}^r$ |
| $P^l$                      | $N_p^r$    | $S_2$      | $C_{c2}^l$ |
| $T_R$                      |            |            |            |

**Table 4.13.** Descriptors selected by using stepwise discriminant analysis to select descriptors that best distinguish the sex of the 352 American speakers from the TIMIT database.

| Sex    | Female | Male |
|--------|--------|------|
| Female | 124    | 7    |
| Male   | 7      | 214  |

**Table 4.14.** A table of the confusion matrix resulting from classifying LTAGRs by sex using discriminant analysis.

have significantly different populations. These probabilities of different regions (or groups in the general case) are called prior probabilities. The distance measure used to classify observations can be weighted by the prior probability to account for variations in the probability of observation (Klecka, 1980; SAS, 1985b), but here the prior probability is set to the constant value  $1/16=0.0625$ , which specifies each of the groups to be equally likely. This means that observations from regions with small numbers of subjects tend to get more matches than might be expected if prior probabilities are used (Lindeman *et al.*, 1980, §6.3.5), whereas regions with large numbers of subjects tend to get fewer matches. It is impracticable to assign prior probabilities to this analysis since the population distribution is unknown.

Examination of Table 4.12 shows that males tend to be confused with males from other regions and a similar trend is observed for females. Descriptors of the LTAGR were therefore analysed to ascertain whether they are likely to be useful for determining the sex of a speaker. Stepwise discriminant analysis was performed to select the descriptors that are most useful for distinguishing the sex of a speaker. Table 4.13 contains the thirteen descriptors selected as useful for classifying speakers by their sex. Table 4.14 shows the classification summary of LTAGRs from 131 females and 221 males using these descriptors. The small error rate of 0.040 indicates that the LTAGR is suitable for distinguishing between the sex of speakers from the United States.

The aforementioned gender classification error rate of 0.040 compares favourably with results reported by other researchers. Using cepstral features, Fussell (1991) performed gender classification on vowels, liquids, nasals, fricatives, stops. The lowest classification error rate for 420 speakers (290 male and 130 female) from the TIMIT database, using each of the above phonemes, was 0.060 for vowels. Childers *et al.* (1988) found that reflection coefficients, computed from six speech frames extracted from sustained vowels, gave an error rate of 0.000 for gender classification of 52 speakers (27 male and 25 female). They also found that classification using descriptors of the spectral characteristics of an LPC filter gave an error rate of 0.018 from only a single pitch period of speech. This is a low error rate, considering the duration of speech used, but the small number of speakers used in the experimental evaluation lowers the significance of the result. Although the duration of speech used for LTAGR computation must be longer than that required by the above methods, the computation of the LTAGR is straightforward and manual classification of phonemes (as required by both Fussell and Childers) is avoided.

The results of discriminant analysis by sex and by region indicate that the LTAGR is inadequate for distinguishing accents from different regions in the United States, but is useful for classifying people by sex.

## 4.4 LONG-TERM AVERAGE SPECTRUM

### 4.4.1 Introduction

The long-term average spectrum (LTAS) has been used by several researchers (§4.4.3) in applications in which an estimate of a person's average voice is required. The motivation for examining the use of the LTAS for speaker identification is that it is a similar feature to the LTAGR, since both the LTAS and LTAGR represent a long-time average of speech characteristics. The main difference between performing SAA and calculating the LTAS is that SAA averages in the time domain, whereas the LTAS averages in the frequency domain. Additionally, it would be useful for several applications if the spectrum of the LTAGR could be shown to correspond closely to the LTAS, since the LTAGR is considerably simpler to calculate than the LTAS (see §4.5.1.2). The remainder of this section describes experiments to assess the differences between several methods of LTAS computation (§4.4.2) and finally discusses certain other applications of the LTAS in §4.4.3 that could possibly benefit from using the LTAGR instead.

### 4.4.2 Comparison of various LTAS calculation methods

This section introduces a number of options available for calculating the LTAS. It is impractical to discuss all the variations simultaneously, so they will be discussed under the following headings: the effect of removing unvoiced speech, pitch synchronous spectral estimation and the effect of pre-emphasis. Note that, to highlight particular points, certain calculation methods may be mentioned more than once in the course of these comparisons.

#### 4.4.2.1 LTAS methods

The LTAS of an utterance is computed here in the following manner: frames of 128 speech samples are windowed with a Hamming window, their power spectra calculated and the spectra averaged over all frames. A number of different preprocessing techniques are applied in the examples presented. The aim of examining so many different computation methods is to determine the advantages and disadvantages of the different methods and to ascertain which of the standard LTAS calculation methods most closely match the spectrum of the LTAGR (see §4.5.1.1).

As was described in §2.5.2, the effect of the lip radiation and glottal source characteristics can be reduced by pre-emphasis of the signal before calculation of the spectra. For the purposes of pre-emphasizing the LTAS, the speech is passed through a filter having a transfer function of  $(1 - 0.95z^{-1})$ . This has the effect of increasing the magnitude of the second, third and higher order formants.

If the LTAS is to be used to deduce information about the glottis, as it might be in therapy, it is usual to utilize only the voiced portions of the speech signal. Both voiced and entire utterances are examined to highlight the effect that removal of unvoiced and silent portions has on the LTAS.

The final variation in calculating the LTAS, is to align the frames in a pitch synchronous manner with the duration of each frame set to a single pitch period. This is effectively the process used in aligning the glottal pulse in SAA. Averaging across frames is then similar to the averaging that occurs in SAA (§2.8.2), except that the

| Extension | Preprocessing technique                                    |
|-----------|--|
| blank     | no preprocessing and entire utterance                      |
| -V        | voiced speech only   |
| -E        | pre-emphasis   |
| -VE       | voiced pre-emphasized speech                               |
| -P        | pitch synchronous frame alignment                          |
| -PE       | pitch synchronous frame alignment of pre-emphasized speech |

Table 4.15. Extensions to the letters LTAS and their meaning.

contribution of the glottal ‘pulse’, as recorded in the speech signal, is averaged in the frequency domain instead of the time domain. One should expect that the LTAS calculated using this method to correspond most closely to that obtained by calculating the spectrum of the LTAGR signal.

To distinguish between the various methods of calculating the LTAS, the extensions defined in Table 4.15 are appended to the letters ‘LTAS’.

#### 4.4.2.2 Effects of removing unvoiced speech and silent periods from the speech

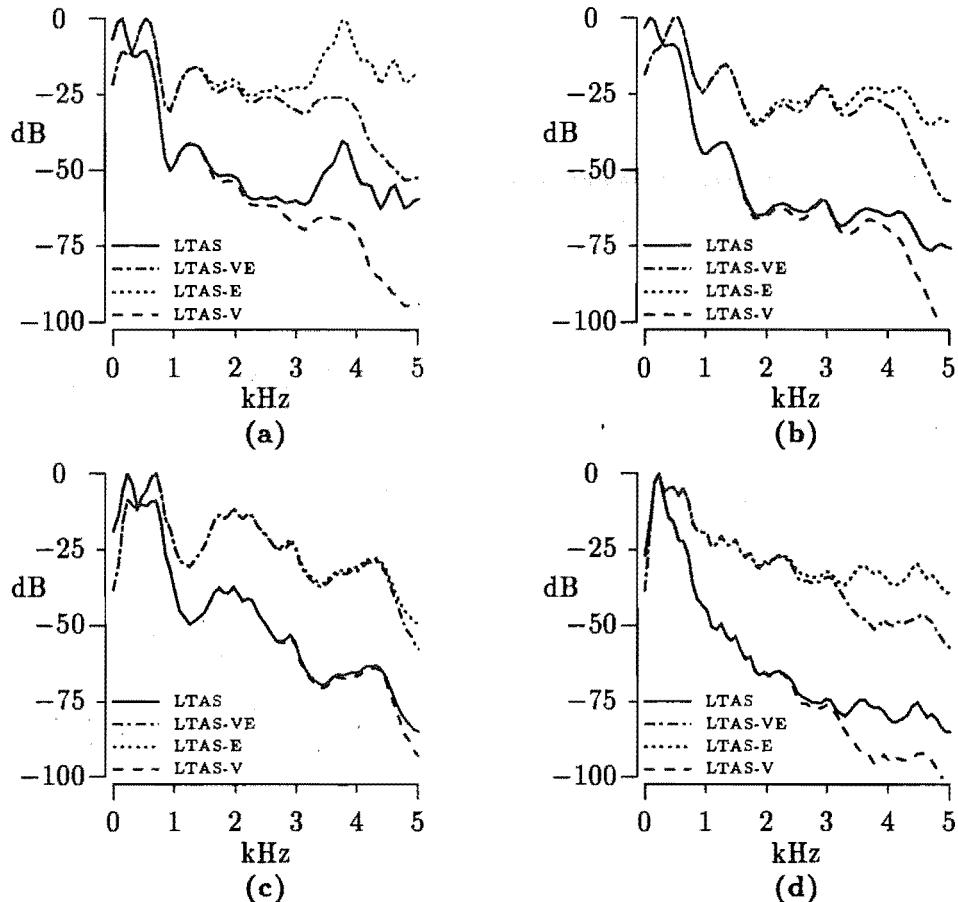
In general, for the LTAS examples presented here, distances between any two spectra are not uniform across the whole spectra. The parts of the spectrum where different methods give different results are identified by examining plots of spectra overlaid on the same graph. This type of comparison indicates whether one method of LTAS calculation contains information not represented by another method, prompting further examination of the information recorded using the different methods of LTAS computation.

Fig. 4.13 shows the LTAS calculated for voiced speech and the entire utterance for four different people saying the rainbow passage, “*When sunlight strikes raindrops in the air, they act like a prism and form a rainbow*”. The first point to note is that the effect of removing unvoiced and silent portions of speech is to reduce the high frequency energy (above 2 kHz) in the LTAS. This is to be expected since unvoiced sounds tend to have more energy above 2 kHz than below. Voiced sounds on the other hand are excited by a source whose spectral energy falls off at approximately  $1/f^2$ , and so contribute less spectral energy above about 3 kHz.

Examination of the LTAS of unemphasized speech depicted in Fig. 4.13 reveals that the maximum difference between the voiced and entire utterance in the band of 0-2 kHz is 2.11dB for speaker AE (see Fig. 4.13(a)). If interest in the LTAS is in the region of 0-2 kHz, it would seem unnecessary to implement a voicing decision.

The LTAS and LTAS-V spectra belonging to speaker KG (see Fig. 4.13(c)) differ by only 0.23 dB in the 0-2 kHz range, indicating that the difference between processing entire utterances and voiced speech is dependent on the speaker. The differences between speakers is further highlighted by examining the differences in the voiced and entire utterance LTAS above 2 kHz. Speakers AE, BM, KG and TC have maximum differences between LTAS of their voiced and entire utterances of 36, 26, 8 and 18 dB. Speaker KG stands out in having a very small difference between the two spectra. Both of the females have smaller differences than the males, which indicates that the relationship between female voiced sounds and unvoiced sounds tends to be different from that of males.

The effect of removing the unvoiced and silent portions of the speech and then



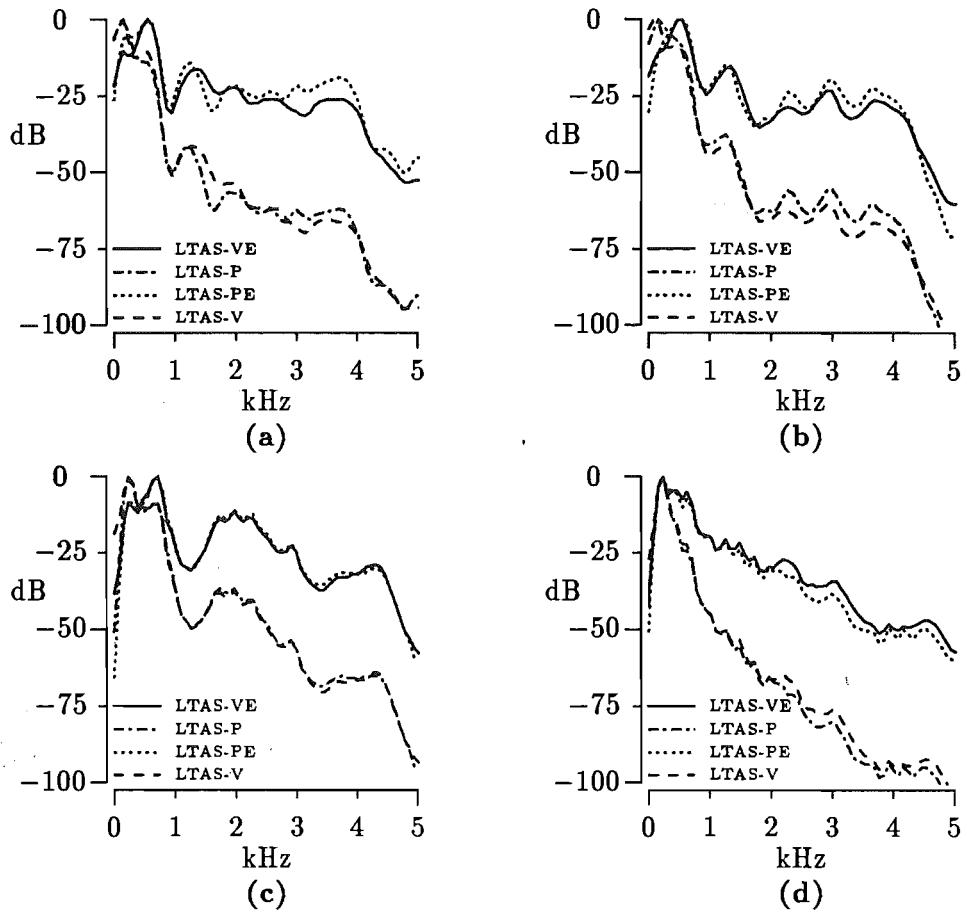
**Figure 4.13.** Depictions of the LTAS, LTAS-VE, LTAS-E and LTAS-V of the rainbow passage uttered by speakers;(a) AE (male), (b) BM (male), (c) KG (female),(d) TC (female).

performing pre-emphasis is also illustrated in Fig. 4.13. All the comments made above pertaining to unemphasized speech also hold for pre-emphasized speech. This is to be expected since pre-emphasis can be thought of as a smoothly varying gain that is applied across the whole spectrum and therefore does not significantly alter the differences between spectra.

#### 4.4.2.3 Pitch synchronous spectral estimation

It might be expected that the LTAS calculated in a pitch synchronous manner would more accurately represent information about the spectrum of a person's average glottal flow, since the peaks occurring in a person's speech are aligned to be in the centre of the speech frame. This section examines the effect of selecting speech frames in a pitch synchronous manner before determining the LTAS.

Before examining results taken from actual speech it is pertinent to discuss the differences that one might expect between pitch synchronous and asynchronous spectra. The difference between the two methods is in the alignment of each frame. This affects the spectral content of each frame, since windowing reduces the amplitude to zero at the frame edges. For example, in the non-aligned case, if a significant peak is at the edge of the frame, the windowing operation will cause its amplitude to be attenuated thus altering the spectral content of that frame. One might therefore expect the non-



**Figure 4.14.** The LTAS-VE, LTAS-P, LTAS-PE and LTAS-V for the raindrops passage uttered by speakers: (a) AE, (b) BM, (c) KG and (d) TC.

aligned LTAS to contain a less accurate representation of the spectrum of the average glottal flow. However, this adverse effect of windowing is minimized by overlapping speech frames so that the total spectral energy in the voiced speech is recorded. For the Hamming window this corresponds to an overlap of a third of the size of the speech frame. The non-aligned LTAS therefore records the average of *all* the spectral energy in the speech, whereas pitch synchronous LTAS records *most* of the spectral energy in the speech, since most of the energy is centred round the major peaks in the speech signal.

The LTAS of voiced speech and the pitch synchronous LTAS of voiced speech are depicted in Fig. 4.14. The differences between these two spectra is less than 10 dB for all frequencies except for dc (0 Hz) of speaker KG. The pattern of peaks and valleys described by the LTAS-V and LTAS-P is identical for speaker KG, whereas the other speakers have a small number (from one to three) of peaks or valleys that do not correspond to those of the LTAS-V. The additional excursions tend to be of small amplitude and do not significantly alter the general shape of the spectrum. An example of two small additional valleys in LTAS-P compared with LTAS-V is depicted in Fig. 4.14(a) at 4.2 and 4.7 kHz. At these frequencies the pitch synchronous spectrum has two extra valleys.

Differences between the LTAS-V and LTAS-P occur consistently for male speech in the low frequency region of the spectra. The portions of the spectra that describe

the pitch frequency and the first formant of the male voices AE and BM (depicted in Fig. 4.14(a) and (b)) have different patterns of peaks. The LTAS of the voiced speech shows two distinct peaks, one at about 80-130 Hz that is produced by the pitch, and another at about 400-500 Hz that corresponds to the first formant frequency. The LTAS-P has only a single peak at approximately the pitch period, which falls off smoothly to the first formant. It is apparent from the spectra depicted in Fig. 4.14 that, apart from the differences round the first formant, the pitch synchronous spectra do not contain significantly different information from the LTAS of voiced speech. The additional computational effort involved in determining the pitch and aligning the speech frames with the maximum peak in the speech waveform cannot therefore be justified.

Fig. 4.14 shows LTAS-VE calculated with and without pitch synchronism. The effect of pre-emphasis on LTAS-V and the pitch synchronous LTAS-P is to increase the amplitude of the higher frequencies. The differences between the LTAS-VE and LTAS-PE are similar to those outlined in the previous paragraphs, with the spectral differences being altered somewhat owing to pre-emphasis.

#### 4.4.2.4 The effect of pre-emphasis

Figs. 4.13 and 4.14 contain both pre-emphasized and unemphasized LTAS on the same graph. At first glance pre-emphasized and unemphasized speech appear to have the same general shape, differing by a gain factor. However the non-linear gain of the  $(1 - 0.95z^{-1})$  pre-emphasis filter across the spectrum causes the pitch frequency component to be attenuated compared with the first formant, whereas higher frequency components are amplified.

For the purposes of recording spectral information in a person's voice pre-emphasis before calculating the LTAS is of little importance. Provided the system used to calculate the LTAS has enough dynamic range to represent the low energies in the high frequency portion of the spectrum, the spectral weighting due to pre-emphasis can be applied after the LTAS has been calculated without loss of information. This outcome is based on the distributive property of the Fourier transform. In practice it can be performed by adding the log spectrum of the pre-emphasis filter to the log of the LTAS.

#### 4.4.3 Other applications of the LTAS

Löfqvist (1986) provided an overview of the application of LTAS to measurement of voice quality and put forward the opinion that "while long-term spectra are potentially useful in the clinic, their possibilities and limitations are not well understood". To gain a better understanding of the usefulness of the LTAS in clinical applications, several researchers have conducted experiments in which the LTAS is evaluated either as a tool in diagnosis or as an objective method of monitoring the progress of therapy.

Löfqvist (1986) compared the parametrized LTAS calculated from 37 normal and 36 clinical voices to determine whether the LTAS had potential for distinguishing between the two types of voice. The clinical voices contained cases of vocal fatigue, chronic laryngitis and vocal nodules. The aim was to examine whether the dominant differences among speakers could be attributed to an abnormality in their voices, or whether they were related to variations in people's voices that were independent of their clinical state. Löfqvist (1986) omitted to state whether or not all the clinical voices were female, so it is difficult to know whether the sex of the speaker had any influence on the reported results. Parameters based on the ratio of energies in the 0-1 kHz and 1-5 kHz speech bands indicated that there was considerable overlap between normal and clinical LTAS parameters, implying that the LTAS is probably unsuitable for patient

diagnosis. Kitzing (1986) supported this claim and added that interspeaker variations were large, which made comparisons between individuals difficult. Although large interspeaker variation is good for speaker identification, it is undesirable for defining an average 'characteristic' that corresponds to a particular clinical condition. However, Kitzing (1986) and Löfqvist (1986) both considered the LTAS to be useful for monitoring changes in a client's voice.

Researchers such as Wendler *et al.* (1986) and those cited in their paper, utilized the LTAS of speech to perform objective analysis of the voice and so improve voice diagnosis. The LTAS was used to classify voices according to the degree of hoarseness, roughness and breathiness. The LTAS was considered to be useful in helping a clinician make diagnostic decisions, but it was too inconsistent to be used as the only measure in performing a diagnosis.

The LTAS is also useful for providing an estimate of the glottal flow and lip radiation characteristics. Furui (1974) argued that by averaging the speech spectrum over a long time, the vocal tract response was averaged, leaving contributions from the glottal characteristic and lip radiation characteristic. This argument is almost identical to that presented in §2.8.2 to explain the averaging process that occurs within SAA. It is this similarity of the two techniques that prompts the comparison between them in §4.5.1.1. Furthermore, if the LTAGR and LTAS are shown to contain similar information, the LTAGR as calculated by SAA may be clinically useful in the same applications as the LTAS.

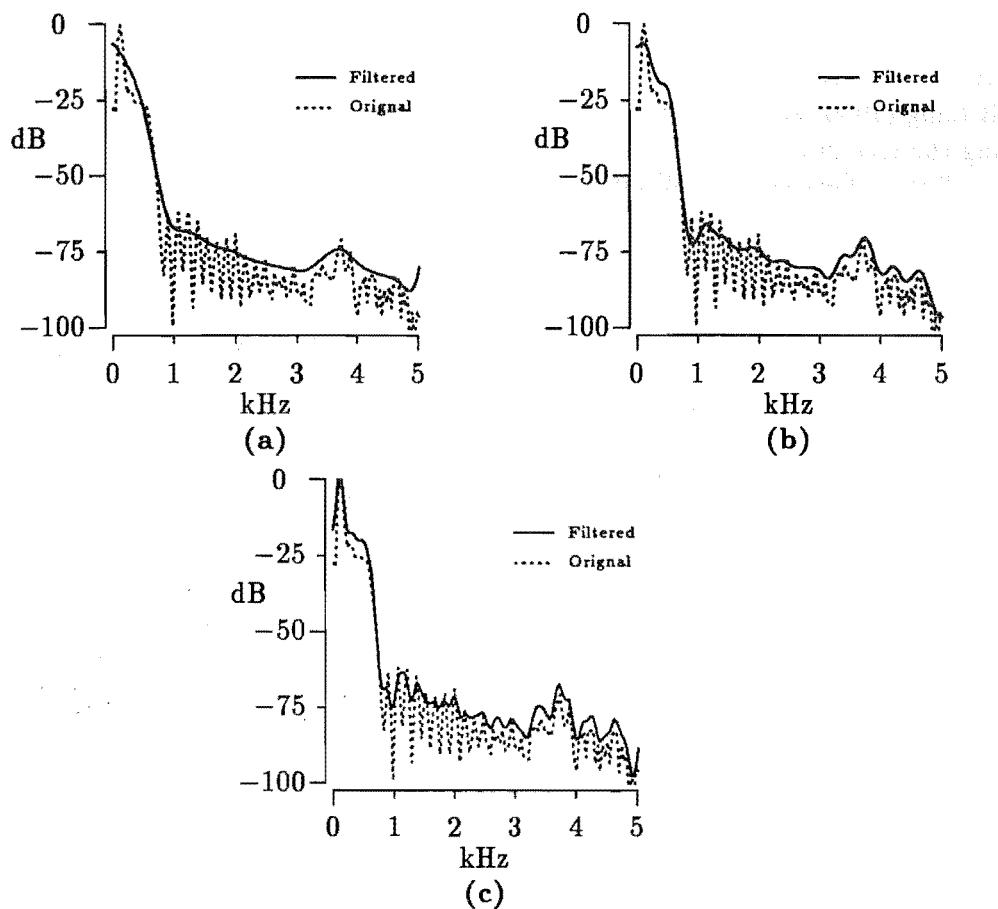
## 4.5 THE SPECTRUM OF THE LONG-TERM AVERAGE GLOTTAL RESPONSE

This section describes methods used for calculating the spectrum of the long-term average glottal response (LTAGR) and compares such spectra against LTAS-VE. The aim of this comparison is to determine whether the information recorded in the LTAS-VE can be extracted from the LTAGR, as determined by SAA. If the LTAS of speech can be shown to be approximated by the spectrum of the LTAGR, estimation of long-term speech characteristics by the LTAGR may be of practical use in clinical applications.

### 4.5.1 Calculation of the spectrum of the long-term average glottal response

The spectrum of the LTAGR is determined by zero packing the LTAGR (computed by the SAA algorithm) to 256 samples and then computing the power spectrum of the zero packed signal.

Fig. 4.15 shows that the power spectrum obtained by this method has a significant harmonic ripple on it, which tends to obscure the underlying formant structure. This ripple is caused by pitch harmonics within frames of the speech signal that are averaged together in the SAA algorithm. When comparing two spectra it is preferable, however, to examine the general trends and the ripple components of the log-spectrum are therefore attenuated by passing the log-spectrum through a 32 tap smoothing filter. The smoothing filter consists of a sinc function, which averages adjacent frequencies of the log spectrum when it is convolved with the log spectrum. The degree of smoothing is proportional to the width of the main lobe of the sinc function which is defined as the distance in hertz between the zero-crossings of the main lobe of the sinc function. Before being used as a filter, the sinc function is windowed by a four-term Blackman-Harris window to reduce artefacts. The smoothed log-spectra for several different sinc-based smoothing filters are depicted in Fig. 4.15 where it is apparent that the choice of smoothing filter significantly alters the SAA spectrum. A smoothing filter with a



**Figure 4.15.** Examples of filtering the spectrum of the LTAGR as determined by SAA for the phrase AERAIN1. The widths of the sinc-based smoothing filters are: (a) 3962.5 Hz, (b) 396.25 Hz and (c) 198.125 Hz.

width of 396.25 Hz was selected as most suitable for smoothing the spectrum, since it attenuates the ripple in the LTAGR spectrum without smearing its overall shape too much.

#### 4.5.1.1 Comparison of the spectrum of the LTAGR and the LTAS

The aim of comparing the spectrum of the LTAGR and the various LTAS is to ascertain whether they contain similar spectral features. The LTAS and LTAGR are compared by visually inspecting pairs of spectra. This gives a subjective indication of any spectral features that may be strongly present in one spectrum but absent in the other.

The LTAGR spectra are compared against the LTAS of voiced speech and pre-emphasized voiced speech for four different speakers. LTAS-VE and LTAS-V were chosen for comparison because, firstly, both are straightforward to compute, and secondly, LTAS-VE closely matches the LTAGR of pre-emphasized speech and LTAS-V the LTAGR of voiced speech.

Fig. 4.16 shows that the spectra of LTAS-V and LTAGR both fall rapidly to approximately -50 dB and after that fall off more slowly. The difference between the two spectra varies significantly amongst the four speakers. In general, between 1-3 kHz LTAS-V contains more energy than the LTAGR. On the whole the formant positions

for LTAS-V and LTAGR are the same, though the amplitudes vary.

In all the examples shown in Fig. 4.16 the initial fall off of the LTAGR (at about 1 kHz) is at least 10 dB greater than that of the LTAS-V. This fall-off in high frequency energy in the LTAGR can be attributed to the averaging that takes place in the time domain, which tends to smooth the LTAGR signal. Thorpe (1990, §4.3.1.2) points out that many of the high frequency components of the speech signal are not synchronized with the low frequency components, causing them to be smoothed out (Fujimura, 1968). In other words, since the SAA algorithm averages frames of the time waveform and records the phase of each component in that frame, the process of averaging many frames together to form a LTAGR causes certain components to be cancelled out. Components due to noise and vocal tract variation tend to be averaged out of the LTAGR. Complete cancellation does not occur, as evidenced by the spectrum computed after performing SAA on pre-emphasized speech having much higher energy in the 2-5 kHz frequency range than SAA on unemphasized speech. If the cancellation of high frequency components were complete, there would be no observable ‘formant-like’ structure at higher frequencies.

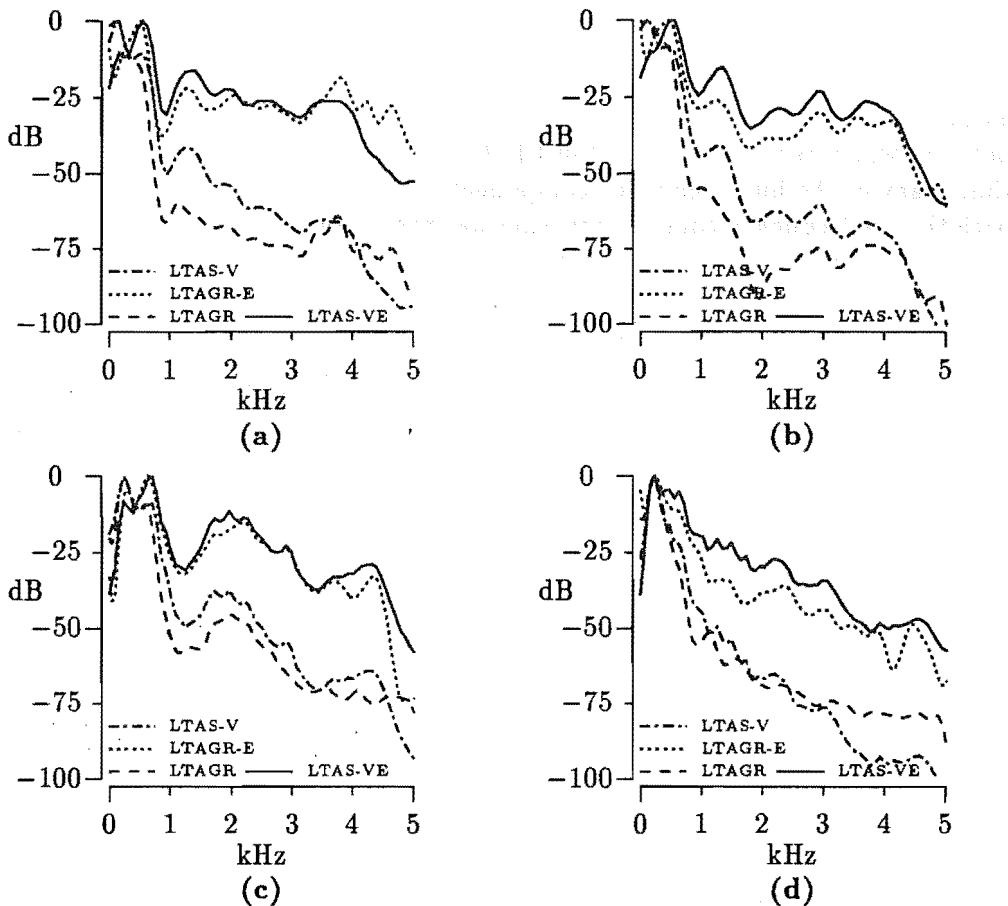
In the calculation of LTAS-V the power is summed up for each speech frame, so no cancellation whatever occurs. Smoothing of the spectrum occurs when a large number of spectra are averaged together, since varying contributions are averaged across all the frames.

It is important to point out that SAA performed on pre-emphasized speech is not based on the same frames as unemphasized speech, since the peaks in pre-emphasized speech occur at the steepest slopes of the unemphasized speech waveform. This shift in frame alignment means that LTAGR and LTAGR-E are not related by the transfer function of the pre-emphasis filter in the same way that LTAS-V and LTAS-VE are.

From the spectra of the pre-emphasized long-term average glottal response (LTAGR-E) plotted in Fig. 4.16, it is apparent that though the LTAGR-E and LTAS-VE spectra contain ‘formants’ at many of the same frequencies, the amplitudes of the formants are different. In addition, the LTAGR spectra sometimes contain peaks and valleys that are absent in the LTAS-VE. For example, the two peaks that occur at 4.25 and 4.7 kHz in the LTAGR-E spectra of Fig. 4.16(a) are completely absent in the LTAS-VE. In general, the approximation of LTAS-VE by the spectrum of the LTAGR-E is less accurate in the 4-5 kHz region. All of the speakers have additional harmonics in the 4-5 kHz region of the spectrum of the LTAGR-E that are absent in the LTAS-VE.

Further comparison of the LTAS-VE and LTAGR-E is by way of the differences between the two spectra. Fig. 4.17 shows the difference in dB between the LTAS-VE and the LTAGR-E. The range of the difference values varies significantly amongst the speakers. The relationship between the spectrum of the LTAGR-E and the LTAS-VE depends on the frequencies present in a person’s voice and how the LTAGR represents them for that particular person. The ‘peaks’ and ‘dips’ within Fig. 4.17 represent peaks or dips in the LTAS-VE that are recorded in the LTAGR-E, or vice versa. For example, at 4 kHz and 4.6 kHz in Fig. 4.17(c) two dips can be readily identified as departures from the LTAS-VE in Fig. 4.17(c). In general, the differences between LTAS-VE and LTAGR-E are less than 10 dB across most of the spectrum, and for speaker KG are significantly less.

Apart from these differences, the similarity of the LTAGR-E to the LTAS-VE indicates that the LTAGR-E and LTAS-VE represent similar aspects of the speech signal. Since the LTAGR-E is significantly easier to compute than the LTAS (see §4.5.1.2) and describes similar aspects of the speech signal, it may be suitable for the clinical assessment of speech disorders (see §4.4.3).



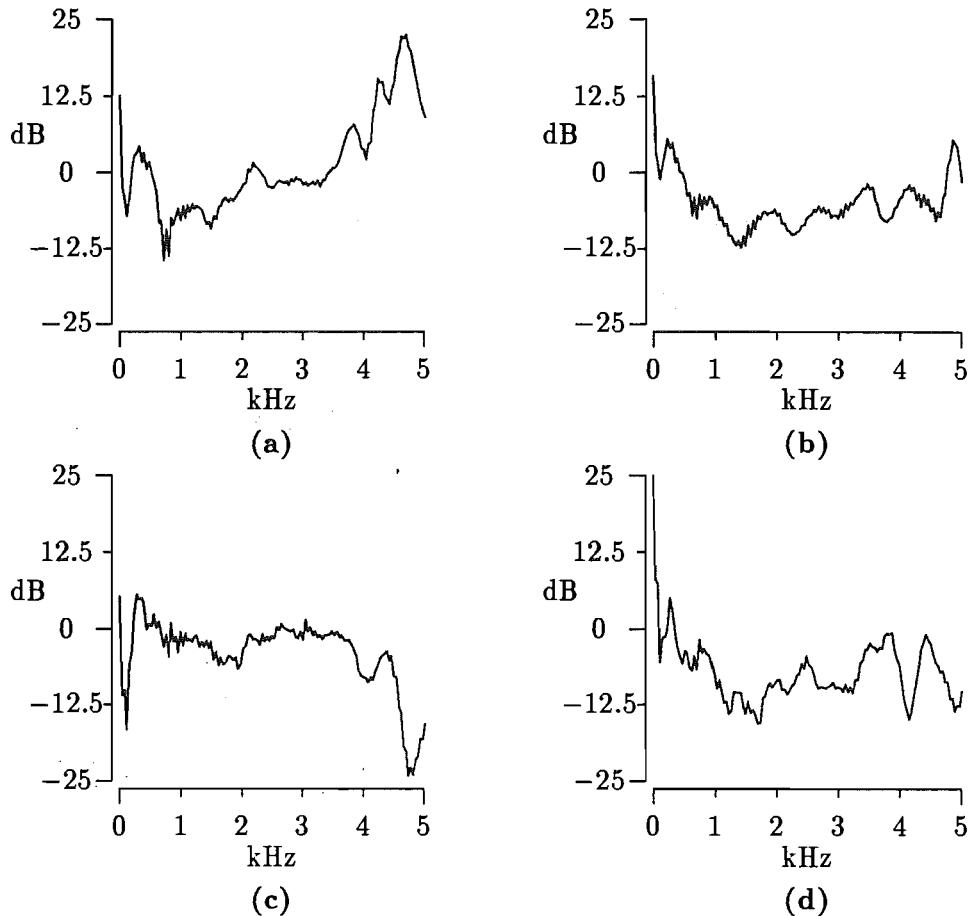
**Figure 4.16.** The LTAS-V, LTAS-VE and the filtered spectra of the LTAGR and LTAGR-E for the rainbow passage uttered by speakers (a) AE, (b) BM, (c) KG and (d) TC.

#### 4.5.1.2 Computational requirements

There is a significant difference in the computational resources required to determine an estimate of the LTAS with the LTAGR and the computation required to perform the many FFTs used in the standard LTAS approach. In order to estimate the number of operations required by these two methods various assumptions must be made about the number of frames that are processed per second and the voicing (or otherwise) of the sampled speech signal.

It is assumed that the speech being processed is voiced and the aim is to arrive at an approximate figure for the number of operations per second on voiced speech. It is also assumed that pre-emphasis has been performed on the voiced speech.

To calculate the number of operations used to determine the LTAGR, it is necessary to specify how many 'glottal' pulses occur in a second since this affects the frame rate. Here the pitch is assumed to be 100 Hz, implying a total of 100 frames per second. The number of additions recorded in Table 4.16 is therefore comprised of 12 800 additions per second for accumulating the glottal responses and 12 800 subtractions per second for locating the SAA frame. Once the LTAGR has been determined, one-off calculations are required to find the spectrum of the LTAGR and then smooth the spectrum. Calculation of the spectrum is by a 128 point FFT, which requires  $2 \times 128 \log_2 128 (= 1792)$  real multiplications and  $3 \times 128 \log_2 128 (= 2688)$  real additions



**Figure 4.17.** The difference between LTAS-VE and the filtered spectra of LTAGR-E (as depicted in Fig. 4.16) for the speakers (a) AE,(b) BM,(c) KG and (d) TC.

(Papoulis, 1980, p375).

The number of operations required to calculate the LTAS is more complicated to evaluate. In order to record the spectral content of the voice fully, each windowed speech frame is overlapped by 60 %. This means that each new frame moves along the speech record by  $128/3=43$  samples implying that, for speech sampled at 10kHz, one second of speech corresponds to 234 frames.

From Table 4.16 it is apparent that estimation of the spectral content of a person's voice via the LTAGR is computationally efficient compared with the more usual LTAS approach. The computational advantage is most pronounced for the number of multiplications/s required to determine the spectrum by the two different methods. The SAA algorithm does not require any multiplications at all on a per-second basis and only 6 016 for the final spectrum calculation. The LTAS approach on the other hand requires 85 560 multiplications/s. The difference in the number of additions and subtractions/s is not as large, but at 25 600 compared with 121 088, is significant nonetheless.

It is feasible to produce an aggregate measure of computational requirements by assigning a 'cost' to a multiplicative operation compared with an addition or subtraction, but this has been avoided since it depends on which computer the algorithms are run. Nevertheless, it is apparent from Table 4.16 that one would expect a 5-10 times reduction in computational requirements by using SAA to estimate the long-term spectrum of speech.

| Method | Units           | Operation     | Additions and subtractions | Multiplications |
|--------|-----------------|---------------|----------------------------|-----------------|
| SAA    | O/s             | peak location | 12 800                     | 0               |
|        |                 | summation     | 12 800                     | 0               |
|        | total O/s       |               | 25 600                     | 0               |
|        | O               | windowing     | 0                          | 128             |
|        |                 | FFT           | 2 688                      | 1 792           |
|        |                 | filter        | 4 096                      | 4 096           |
|        | total one-off O |               | 6 784                      | 6 016           |
| LTAS   | O/frame         | windowing     | 0                          | 128             |
|        |                 | FFT           | 2 688                      | 1 792           |
|        |                 | summation     | 128                        | 0               |
|        | total O/frame   |               | 2 816                      | 1 920           |
|        | total O/s       |               | 121 088                    | 85 560          |

Table 4.16. Comparison of the number of operations required to determine an estimate of the long-time average speech spectrum. Note that the letter 'O' in the above table stands for 'operations'.

## 4.6 SUMMARY

The purpose of this chapter is to introduce speech features that might be useful for speaker identification, to examine some of their properties and to set the scene for the experimental work reported in Chapter 5. Section 4.1 describes the database used for the speaker identification experiments described in Chapter 5. The features introduced in this chapter are listed below, and results of the analysis performed in this chapter are summarized.

- Tests on the vector quantization software that the author uses to construct speaker codebooks are reported in §4.2.2, and the operation of the software verified.
- Descriptors that characterize the LTAGR are defined in §4.3.2.
- Factor analysis (§4.3.3.1) of LTAGR descriptors of 352 male and female American speakers from eight different accent regions reveals that the first factor, which accounts for 67.7% of the variance, is strongly correlated with descriptors (such as cubic coefficients) that describe the general shape of the LTAGR. The next two factors account for 11.5% and 4% of the total variance and are most highly correlated with the number of peaks in the LTAGR. Scatter diagrams of the first three factors for all 352 speakers does not reveal any significant clustering, either by sex or by accent.
- Discriminant analysis (§4.3.3.2) of LTAGR descriptors of the 352 American speakers gives an average classification error rate for classification by region of 63.1 % whereas the classification error rate for classification by sex is 4.0%.
- Examination of various LTAS calculation methods in §4.4.2 reveals that the effect of removing unvoiced and silent portions from the speech becomes significant above 3 kHz. Pitch synchronous LTAS produces similar spectra to asynchronous techniques and is therefore abandoned. LTAS-VE is selected as the LTAS feature for use in identification experiments in Chapter 5.

- The spectrum of the LTAGR-E is found to be similar to the LTAS-VE in §4.5.1.1. The 5-10 times computational advantage of the LTAGR-E over the LTAS-VE (§4.5.1.2) may justify a preference for its use in applications that utilize the LTAS-VE.



## CHAPTER 5

---

### SPEAKER IDENTIFICATION EXPERIMENTS

This chapter reports the results of speaker identification experiments using the speech and recognition features described in Chapter 4. Various features for speaker recognition are subjected to a battery of tests to evaluate their performance under a variety of conditions. This is similar to the method adopted by Atal (1974), except that features other than LPCs are tested and the effect of noise and distortion is examined.

Section 5.1 introduces terminology for describing utterances and features. In §5.2 aspects of speaker template computation are defined. The section also describes analysis of the speaker templates to find the average intraspeaker and interspeaker distances. Section 5.3 describes statistical methods for interpreting results from the various speaker identification experiments. Speaker identification experiments using CEP, PARCOR, LPC, LTAGR and LTAS-VE features are reported in §5.4, along with the accuracy of certain combinations of the above features. The effect of noise and distortion on these features is examined in §5.5. Computational considerations are discussed in §5.6, and the main findings of the identification experiments are summarized in §5.7.

#### 5.1 TERMINOLOGY

Throughout this chapter, the following conventions are used for describing utterances. Utterances that are processed in their entirety, including silences and unvoiced portions, are called entire utterances. In many of the experiments reported here, only the voiced portions of utterances are used, and this is indicated by appending V to the feature abbreviation.

Where the speaker's template comprises a quantized feature, the number of vectors stored in the codebook is appended to the end of the feature abbreviation. For example, CEP4 implies cepstral coefficients and a codebook consisting of 4 codevectors.

Identification results are computed for VQ codebook sizes ( $L$ ) ranging from 2 to 128, with each increment in codebook size being a doubling of the number of vectors in the codebook. In order to make this range linear the codebook rate (see (2.75)), defined as  $R = \log_2 L$ , is used.

Since there are many different features, it is necessary to define notation to specify the feature utilized to measure a particular distance. The feature specifier is recorded in the subscript, for example,  $d_{CEP16}$  indicates that the distance is a measure between cepstral coefficients and a codebook of cepstral coefficient vectors containing 16 codevectors.

#### 5.2 TRAINING OF SPEAKER TEMPLATES

Speaker templates are formed from each individual's training utterances. The method used to calculate each template depends upon the format of the feature that is to be represented in the template. For example, the PARCOR and CEP templates are

formed using the LBG VQ training algorithm described in §2.7.4.3, while the LTAS-VE and LTAGR templates are the normalized averages of the LTAS-VE and LTAGR taken over five training utterances.

### 5.2.1 Normalization of the LTAGR and LTAS-VE

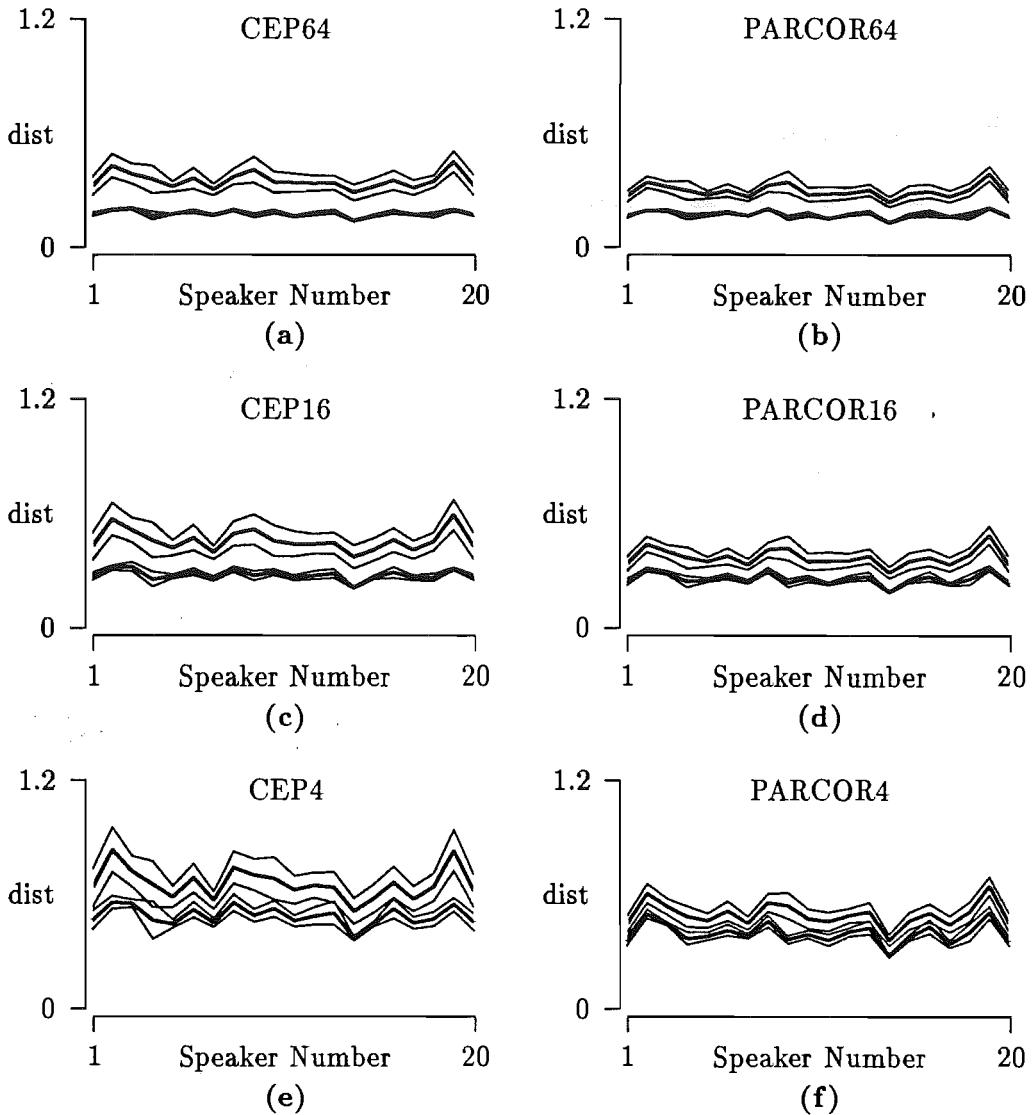
To ensure that contributions to the templates from each of the training utterances are of equal weight, the LTAGR and LTAS-VE are normalized before template averaging occurs. For the LTAGR, this is achieved by scaling the LTAGR so that the maximum amplitude is unity. This is an acceptable normalization procedure, since the speech being processed has no underlying dc component. The LTAS-VE is normalized by scaling the power spectrum so that the total energy in the spectrum is unity.

### 5.2.2 Analysis of speaker templates

One method of investigating the utility of a feature is to examine how well it separates a person from all the other people in the database. Here the training utterances are utilized to compare the average distance of a person's features from their own template (the intraspeaker distance) and the average distance from other people's templates (the interspeaker distance). In addition, following the analysis techniques of Soong and Rosenberg (1988), the correlation between intraspeaker distances derived from different features is examined.

The expected accuracy of a particular feature depends on the interspeaker distance, the intraspeaker distance and the standard deviations of both of these distances. A feature that works well for speaker identification should have an average interspeaker distance that is significantly larger than the intraspeaker distance across all the participating individuals. Fig. 5.1 shows the interspeaker and intraspeaker distances for PARCOR and CEP codebooks constructed from the entire speech utterance, while examples of interspeaker and intraspeaker distances taken from only voiced speech are shown in Fig. 5.2. Although the larger codebooks exhibit averaged interspeaker distances that are well separated from the intraspeaker distances, the interspeaker and intraspeaker distances are highly correlated. This is unexpected since an utterance that is not 'close' to the correct speaker's template should be 'close' to another speaker's template. However, the high correlation implies that an utterance that is 'close' to the correct template is also 'close' to all the other speaker's templates and vice versa. Table 5.1 tabulates correlations between the interspeaker and intraspeaker distances for individual speakers as calculated using Pearson's correlation formula (Eton, 1974). Templates and test utterances containing the entire speech utterance have a high correlation between the intraspeaker and interspeaker distances, whereas templates and test utterances calculated from only the voiced portions of the speech utterance are much less correlated, particularly for CEP coefficients.

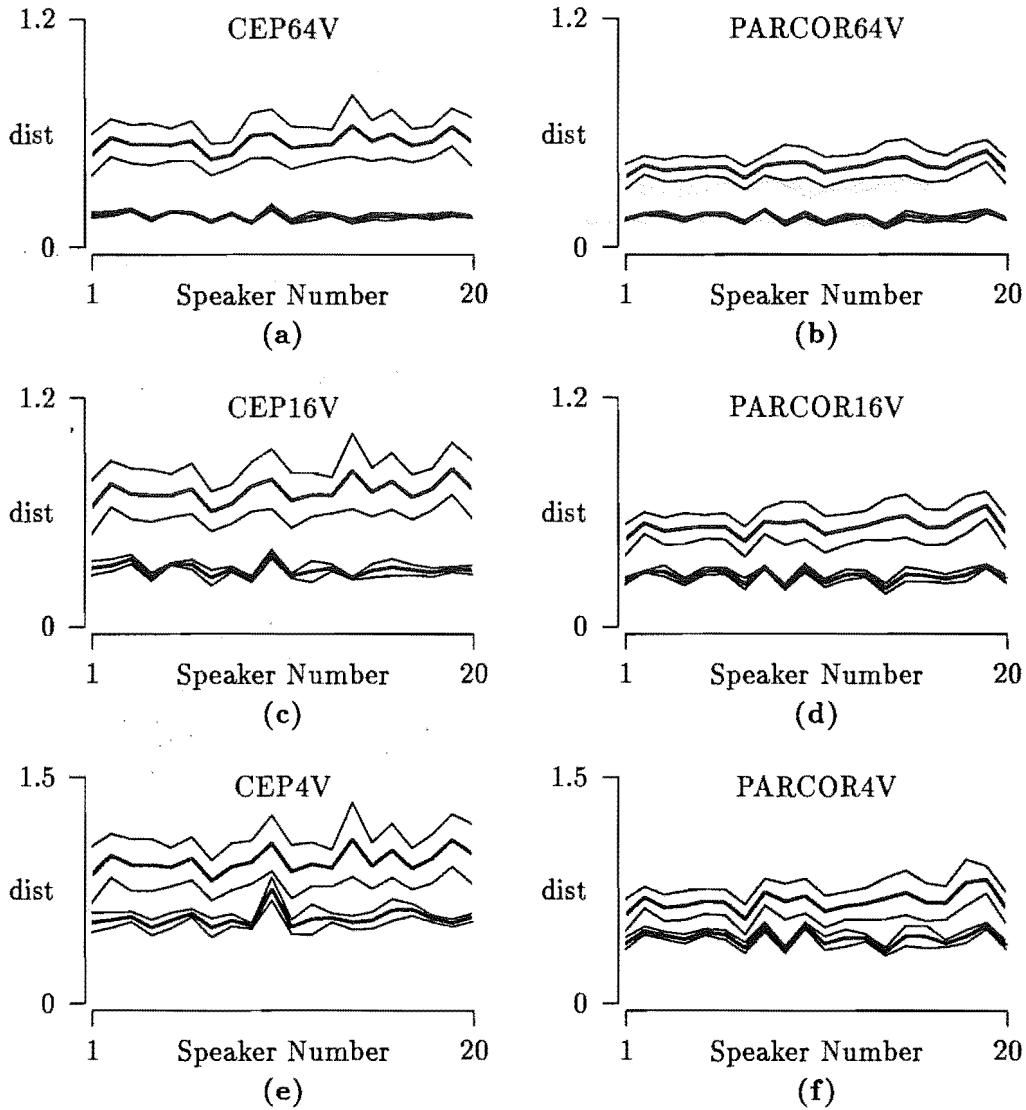
Increasing the number of vectors in the codebook does not significantly increase the average difference between the interspeaker and intraspeaker distances, nor the standard deviation of the interspeaker distance, but it does reduce the standard deviation of the intraspeaker distance. This can be deduced by observing the trend in the intraspeaker distance in Fig. 5.2 and Table 5.1 as the codebook size is increased. A comparison of the difference between the interspeaker and intraspeaker distances of the utterances with and without silence, indicates that the interspeaker distances and intraspeaker distances are separated more for voiced speech than for the entire utterance. This implies that the voiced speech should be more accurate for speaker identification.



**Figure 5.1.** The interspeaker and intraspeaker distances calculated between the training utterances and the speaker templates using the entire utterance. The thick lines represent the average distance and the thinner lines the standard deviation of the distance. The following features are examined: (a) CEP64, (b) PARCOR64, (c) CEP16, (d) PARCOR16, (e) CEP4, (f) PARCOR4.

| Feature  | <i>r</i> | Feature   | <i>r</i> | Feature | <i>r</i> | Feature | <i>r</i> |
|----------|----------|-----------|----------|---------|----------|---------|----------|
| PARCOR4  | 0.92     | PARCOR4V  | 0.59     | CEP4    | 0.90     | CEP4V   | 0.42     |
| PARCOR8  | 0.89     | PARCOR8V  | 0.51     | CEP8    | 0.87     | CEP8V   | 0.16     |
| PARCOR16 | 0.84     | PARCOR16V | 0.44     | CEP16   | 0.80     | CEP16V  | 0.15     |
| PARCOR32 | 0.78     | PARCOR32V | 0.38     | CEP32   | 0.76     | CEP32V  | 0.07     |
| PARCOR64 | 0.72     | PARCOR64V | 0.31     | CEP64   | 0.66     | CEP64V  | 0.04     |

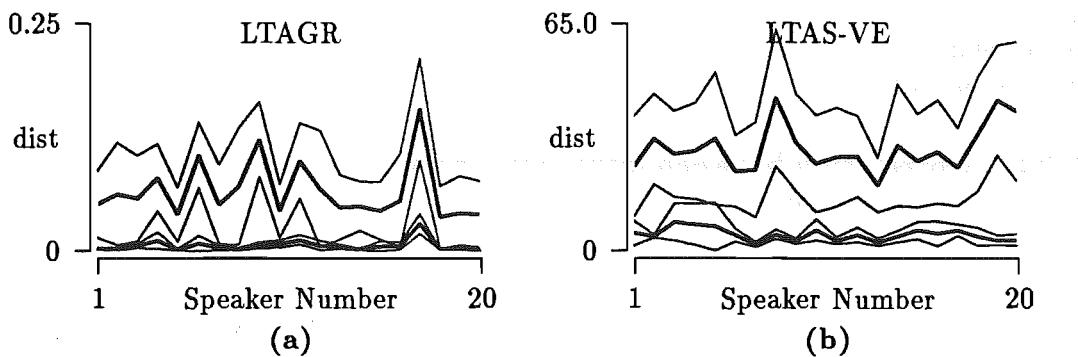
**Table 5.1.** Pearson's correlation coefficients evaluated between interspeaker and intraspeaker distances for various features. Recall that 'V' appended on the end of CEP and PARCOR indicates that features are calculated from only voiced frames of speech.



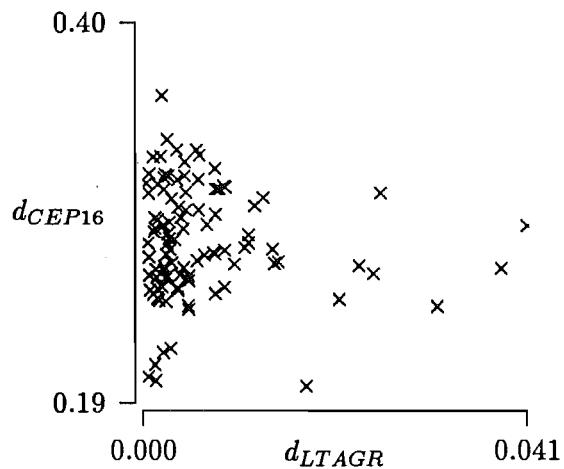
**Figure 5.2.** The interspeaker and intraspeaker distances calculated between the training utterances and the speaker reference templates. The thick lines represent the average distance and the thinner lines the standard deviation of the distance. The following features are extracted from voiced speech: (a) CEP64V, (b) PARCOR64V, (c) CEP16V, (d) PARCOR16V, (e) CEP4V, (f) PARCOR4V.

The distributions of interspeaker and intraspeaker distances for the LTAGR and LTAS-VE are depicted in Fig. 5.3. Both the long-time average features have small average intraspeaker distances compared with the interspeaker distances, but the relatively large standard deviation of the interspeaker distance implies that the large separation between the average interspeaker distance and the average intraspeaker distance will not necessarily result in more accurate recognition.

Often, after a number of features have been selected, the goal is to combine them in some manner that improves the overall recognition performance. In this situation it is important to quantify which of the features are independent. Soong and Rosenberg (1988) utilized the correlation between intraspeaker distances for the different features as a measure of whether a feature was redundant for speaker identification purposes. They asserted that, provided features contain speaker related information and are not highly correlated, combining them should increase speaker identification accuracy. Fig. 5.4 shows a scatter diagram of the intraspeaker distances  $d_{CEP16}$  and



**Figure 5.3.** The interspeaker and intraspeaker distances for each individual, calculated between the training utterances and each speaker reference template: (a) the LTAGR, (b) the LTAS-VE.



**Figure 5.4.** Scatter diagram showing the correlation between intraspeaker distances calculated using a long-term average glottal response and CEP16 (calculated correlation coefficient = 0.).

$d_{LTAGR}$ . The correlation coefficient for the scatter diagram is zero, indicating that the  $d_{LTAGR}$  and  $d_{CEP}$  should be complementary features for speaker recognition.

The correlations between a number of different features are summarized in Table 5.2. The intraspeaker distances  $d_{LTAGR}$  and  $d_{LTAS-VE}$  are more highly correlated with distances derived from voiced portions of the utterances than with unvoiced. This

| Feature   | LTAGR | LTAS-VE | CEP16 | PARCOR16 | CEP16V | PARCOR16V |
|-----------|-------|---------|-------|----------|--------|-----------|
| PARCOR16V | 0.16  | 0.10    | 0.29  | 0.48     | 0.58   | 1.00      |
| CEP16V    | 0.28  | 0.34    | 0.41  | 0.27     | 1.00   |           |
| PARCOR16  | 0.01  | -0.08   | 0.85  | 1.00     |        |           |
| CEP16     | 0.00  | 0.01    | 1.00  |          |        |           |
| LTAS-VE   | 0.30  | 1.00    |       |          |        |           |
| LTAGR     | 1.00  |         |       |          |        |           |

**Table 5.2.** Correlations between the intraspeaker distances of a number of different features.

is expected, because LTAGR and LTAS-VE are both calculated from voiced portions of the speech. The low correlation between the intraspeaker distances  $d_{CEP16}$  and  $d_{CEP16V}$  is caused by the silent portions within the utterance, and this is discussed in more detail in the next section. The same trend is apparent between the  $d_{PARCOR}$  and  $d_{PARCORV}$  distances.

### 5.2.3 Correlation between intraspeaker and interspeaker distances for entire speech

Fig. 5.1 and Table 5.1 indicate that the interspeaker and intraspeaker distances for the entire utterance are highly correlated. One would not expect the intraspeaker and interspeaker distances to be correlated at all, since there is no obvious reason why a training utterance that is a large distance away from its template should also be a large distance from templates belonging to other speakers. The correlations in Table 5.1 would seem to indicate that the unvoiced and silent portions of an utterance contribute most to the correlation between the intraspeaker and interspeaker distances.

The effect of the unvoiced and silent frames of an utterance on the total distance between a set of training vectors and a template is examined by calculating the intraspeaker and interspeaker distances for individual frames within an utterance. An example of this is depicted in Fig. 5.5 for the phrase AE1. On average, the intraspeaker distance should be smaller than the interspeaker distance if the template is to be a useful representation of a person's voice characteristics for performing speaker identification. This corresponds to most of the points plotted in Fig. 5.5 falling below the 45 degree dashed line. Comparison between Fig. 5.5(a), (b) and (c) shows that the unvoiced and silent portions of an utterance contribute a large number of small amplitude distances to the total distance. The many correlated small amplitude distances in Fig. 5.5(c) tend to dominate the fewer large distances in Fig. 5.5(b) causing the total distance to be correlated. Fig. 5.5(d) shows the effect of removing unvoiced and silent portions of speech from both the template training utterances and the test utterances.

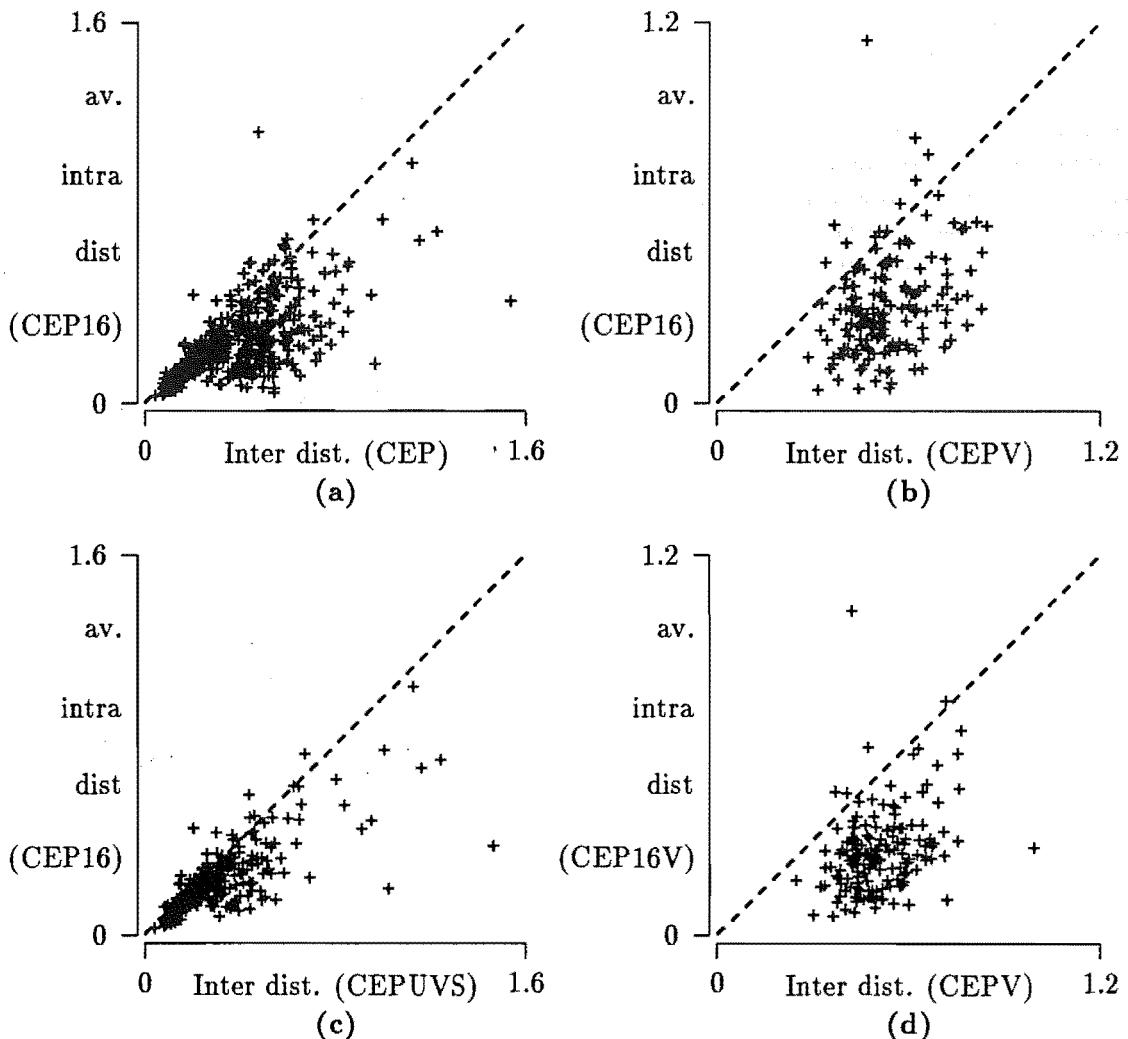
The speaker identification accuracy is best when the utterance AE1 is well separated from all the other speaker templates and all the points in the scatter diagram fall well below the 45 degree line. From Fig. 5.5 one would expect the speaker identification performance using a codebook of size 16 and voiced speech to be better than that obtained using the entire utterance.

## 5.3 STATISTICAL SIGNIFICANCE OF IDENTIFICATION RESULTS

This section addresses two problems concerning the interpretation of results from speaker identification experiments. The first problem is to estimate the confidence intervals to attach to results of identification experiments (§5.3.1 and §5.3.2), and the second problem (§5.3.3) is to select a suitable method for comparing experimental results from two identification systems.

### 5.3.1 A distribution for modelling the identification error rate

One of the difficulties in evaluating the performance of speaker identification systems is to know what significance to attach to the error rate observed for a particular experiment. In other words, is the number of errors recorded from a particular experiment an accurate representation of the performance of the system, and how wide is the spread of observed error rates from one experiment to the next for the same nominal average error rate. To address this problem it is necessary to assume a distribution for the error



**Figure 5.5.** Plots of the intraspeaker distance versus the interspeaker distance for a single utterance of zero to nine by speaker AE (utterance AE1). Each '+' corresponds to a single set of CEP coefficients calculated from a single frame within AE1. (a) Distances calculated between all the frames of AE1 and codebooks derived from entire utterances, (b) distances calculated between the voiced frames of AE1 and codebooks derived from entire utterances, (c) distances calculated between the unvoiced and silent frames of AE1 and codebooks derived from entire utterances, (d) distances calculated between the voiced frames of AE1 and codebooks derived from voiced frames.

rate so that it is possible to calculate the probability of observing a range of different results.

Each identification experiment is made up of a number of identification trials which consist of matching a test utterance against the speaker templates, performing identification, and recording whether or not the identification was correct. Since the identification outcomes are either in error, or not in error, they can be considered to be the outcomes of a binomial process which has associated with it a probability,  $p$ , of producing a recognition error (Johnson and Kotz, 1969). In the limit, an infinite number of identification trials would be required to obtain an exact estimate of  $p$ , the probability of error. Of course, the experimental results presented in this thesis are from experiments consisting of a finite number of trials so the number of errors observed only gives an estimate of  $p$ .

Several assumptions about the nature of the trials are implied by asserting that

the distribution of recognition errors can be usefully modelled by the binomial distribution. Experiments must consist of a number of *identical* trials, denoted  $N$ , with the probability of a particular outcome being the same for each of the trials. In the speaker identification trials reported here, successive trials on the same speaker would be expected to have the same probability of error, since the same text is used for each test. But there is no guarantee that trials on different speakers would have the same probability of error, so the conditions for binomially distributed outcomes are therefore not strictly satisfied. However, the aim is to determine an overall error rate across many speakers and it seems reasonable to assume that the average error rate, across many speakers, is approximately binomially distributed.

### 5.3.2 Binomial confidence limits

This section examines two different approaches for estimating confidence intervals for the underlying probability of error,  $p$ .

The most straightforward method of estimating a confidence interval is to assume Gaussian properties, and use values of the standard deviation,  $s$ . The standard deviation of a binomially distributed population is defined as

$$s = \sqrt{Np(1-p)}, \quad (5.1)$$

where  $N$  is the number of trials. The standard deviation values are utilized to plot error bars that extend  $\pm 2s$  from the error rates of the speaker identification experiments. The probability that an experimental outcome will fall outside the  $\pm 2s$  range is  $\alpha < 0.05$  (from the normal distribution). The quantity  $(1-\alpha)$  is called the confidence level, since it measures the probability that the experimental outcome falls within the specified range.  $\alpha$  is the probability the experimental outcome lies outside the specified range.

Blyth (1986) discusses an alternative approach, where values from the cumulative binomial distribution are computed to yield an exact solution for the confidence interval. Before discussing the method for computing exact confidence intervals, it is necessary to introduce some notation for describing binomial probabilities.

The probability of a particular experimental outcome is expressed in terms of the binomial distribution. If  $X$  represents the number of errors recorded in a speaker identification experiment, it follows that for the binomial distribution,

$$\Pr(X = x) = \binom{N}{x} p^x (1-p)^{N-x}, \quad (5.2)$$

where  $\binom{N}{x}$  is the total number of combinations that  $x$  items can be abstracted from  $N$  items without repetition (Kreyszig, 1979, p860).

In order to calculate a confidence interval for the underlying probability  $p$ , it is necessary to evaluate the probability that  $X \leq x$ . This is given by the cumulative binomial probability, which is written

$$\Pr(X \leq x) = \sum_{j=0}^x \binom{N}{j} p^j (1-p)^{N-j}. \quad (5.3)$$

Equation (5.3) can be used to compute confidence intervals by specifying a confidence level ( $1-\alpha$ ) and choosing a value  $p$  so that  $\Pr(X \leq x) = \alpha$ . This can be expressed concisely as

$$\Pr(X \leq x | p = p_0) = \alpha, \quad (5.4)$$

where  $p_0$  is the value of  $p$  that causes the probability to equal  $\alpha$ .  $\Pr(X \leq x)$  can be evaluated using the incomplete beta function (Blyth, 1986; Abramowitz and Stegun, 1965). Computation of the inverse of this function is required to determine  $p_0$

| $x$ | $p'$  | $p_0^u$ such that<br>$\Pr(X \leq x p = p_0^u) \leq \alpha$ | $x$ | $p'$  | $p_0^u$ such that<br>$\Pr(X \leq x p = p_0^u) \leq \alpha$ |
|-----|-------|--|-----|-------|--|
| 0   | 0.000 | 0.015  | 20  | 0.100 | 0.142  |
| 1   | 0.005 | 0.023  | 21  | 0.105 | 0.148  |
| 2   | 0.010 | 0.031  | 22  | 0.110 | 0.153  |
| 3   | 0.015 | 0.038  | 23  | 0.115 | 0.159  |
| 4   | 0.020 | 0.045  | 24  | 0.120 | 0.165  |
| 5   | 0.025 | 0.052  | 25  | 0.125 | 0.170  |
| 6   | 0.030 | 0.058  | 26  | 0.130 | 0.176  |
| 7   | 0.035 | 0.065  | 27  | 0.135 | 0.181  |
| 8   | 0.040 | 0.071  | 28  | 0.140 | 0.187  |
| 9   | 0.045 | 0.077  | 29  | 0.145 | 0.192  |
| 10  | 0.050 | 0.083  | 30  | 0.150 | 0.198  |
| 11  | 0.055 | 0.089  | 31  | 0.155 | 0.203  |
| 12  | 0.060 | 0.095  | 32  | 0.160 | 0.209  |
| 13  | 0.065 | 0.101  | 33  | 0.165 | 0.214  |
| 14  | 0.070 | 0.107  | 34  | 0.170 | 0.220  |
| 15  | 0.075 | 0.113  | 35  | 0.175 | 0.225  |
| 16  | 0.080 | 0.119  | 36  | 0.180 | 0.231  |
| 17  | 0.085 | 0.125  | 37  | 0.185 | 0.236  |
| 18  | 0.090 | 0.131  | 38  | 0.190 | 0.241  |
| 19  | 0.095 | 0.136  | 39  | 0.195 | 0.247  |

Table 5.3. The upper confidence limit  $p_0^u$  of the underlying probability of error of a recognition system, given that  $x$  identification errors were observed. The confidence level  $\alpha = 0.05$  and the number of trials is assumed to be  $N = 200$ .

and this is performed using the SAS function BETAINV (SAS, 1985a). Evaluation of  $\Pr(X \leq x|p = p_0) \leq \alpha$  requires that  $N$  be defined, so here  $N = 200$ , which corresponds to the number of identification trials performed in a single identification experiment. The confidence limits discussed in the remainder of this section are therefore applicable to the results of identification experiments reported elsewhere in this chapter. The values of  $p_0^u$  in Table 5.3 constitute the upper limit of the range of underlying probabilities of recognition error that could produce the observed value of  $x$  recognition errors. The best estimate of the underlying probability of error is

$$p' = x/N, \quad (5.5)$$

and  $p_0^u$  represents the upper confidence limit of this value.

It is important when comparing experimental outcomes to know the lower confidence level as well as the upper confidence level. The transforms  $p \rightarrow 1 - p$  and  $x \rightarrow N - x$  are invoked to calculate the lower limits and these are tabulated in Table 5.4. The relationship between  $p_0^l$  and  $p_0^u$  and the observed number of identification errors  $x$  is plotted in Fig. 5.6. It is apparent from Fig. 5.6 that when 10 identification errors are observed in an experiment, and the best estimate of the probability of error is  $p' = 0.050$ , the confidence interval for the underlying probability of error ranges between  $p_0^l = 0.027$  and  $p_0^u = 0.083$ .

The values of  $p_0^u$  and  $p_0^l$  differ most from those derived from the standard deviation  $s$  (as defined in (5.1)) when  $x$  is small and also when  $x$  is around 100 ( $p = 0.5$ ). Fig. 5.7 shows the upper confidence intervals ( $\alpha = 0.05$ )  $p_0^u - p'$  and  $2s$ . When the number of errors recorded is less than 18, the confidence interval is larger, and therefore more conservative, for the exact solution. However, above 18 recognition errors, the  $2s$  confidence interval is more conservative than the exact solution. Since many of the recognition experiments have small numbers of errors, the confidence intervals for the 0-20 range are of primary interest. Fig. 5.7 shows that, ignoring  $x = 0$ , using  $2s$  for estimating the confidence interval causes a maximum difference from the exact confidence interval of 0.005 (0.5%), for  $x = 1$  ( $p' = 0.005$ ). Although the difference of 0.005 in a confidence bound of 0.010 is significant, the  $2s$  confidence intervals are used for figures containing identification error rates because they are easier to compute. The  $2s$  confidence intervals will tend to be less conservative than the exact confidence interval.

| $x$ | $p'$      | $p_0^l$ such that<br>$\Pr(X \geq x p = p_0^l) \leq \alpha$ | $x$ | $p'$  | $p_0^l$ such that<br>$\Pr(X \geq x p = p_0^l) \leq \alpha$ |
|-----|-----------|--|-----|-------|--|
| 0   | undefined | undefined  | 20  | 0.100 | 0.067  |
| 1   | 0.005     | 0.000  | 21  | 0.105 | 0.071  |
| 2   | 0.010     | 0.002  | 22  | 0.110 | 0.076  |
| 3   | 0.015     | 0.004  | 23  | 0.115 | 0.080  |
| 4   | 0.020     | 0.007  | 24  | 0.120 | 0.084  |
| 5   | 0.025     | 0.010  | 25  | 0.125 | 0.088  |
| 6   | 0.030     | 0.013  | 26  | 0.130 | 0.093  |
| 7   | 0.035     | 0.017  | 27  | 0.135 | 0.097  |
| 8   | 0.040     | 0.020  | 28  | 0.140 | 0.101  |
| 9   | 0.045     | 0.024  | 29  | 0.145 | 0.106  |
| 10  | 0.050     | 0.027  | 30  | 0.150 | 0.110  |
| 11  | 0.055     | 0.031  | 31  | 0.155 | 0.114  |
| 12  | 0.060     | 0.035  | 32  | 0.160 | 0.119  |
| 13  | 0.065     | 0.039  | 33  | 0.165 | 0.123  |
| 14  | 0.070     | 0.043  | 34  | 0.170 | 0.128  |
| 15  | 0.075     | 0.047  | 35  | 0.175 | 0.132  |
| 16  | 0.080     | 0.051  | 36  | 0.180 | 0.137  |
| 17  | 0.085     | 0.055  | 37  | 0.185 | 0.141  |
| 18  | 0.090     | 0.059  | 38  | 0.190 | 0.146  |
| 19  | 0.095     | 0.063  | 39  | 0.195 | 0.150  |

Table 5.4. The lower confidence limit  $p_0^l$  of the underlying probability of error of a recognition system, given that  $x$  identification errors were observed. The confidence level  $\alpha = 0.05$  and the number of trials is assumed to be  $N = 200$ .

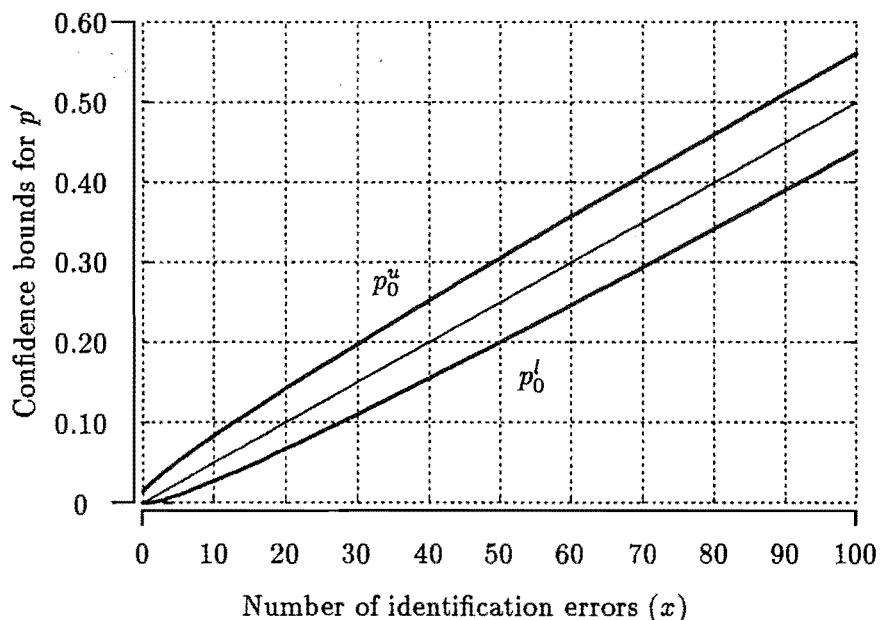
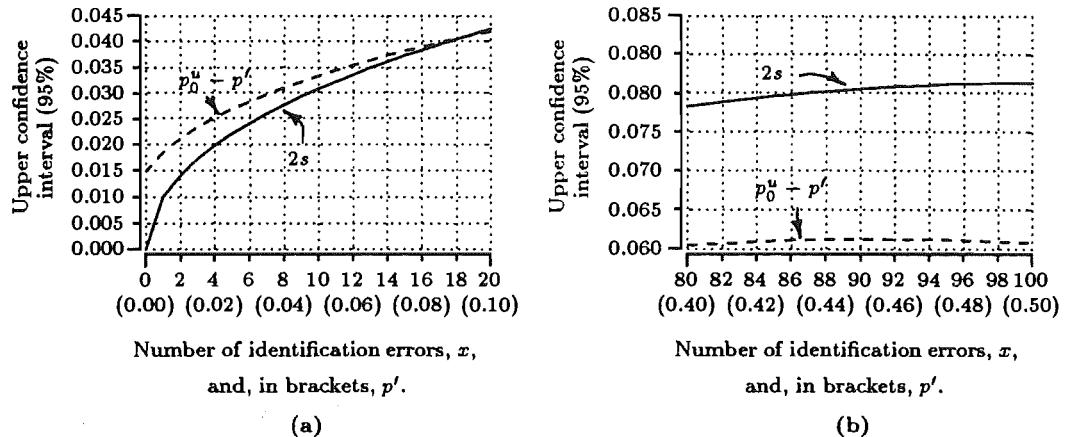


Figure 5.6. A graph of the 95% confidence interval when  $x$  errors are observed. The upper and lower bounds are from the  $e = Np_0$  column of Table 5.3 and Table 5.4 respectively.

### 5.3.3 Comparisons between two systems

Confidence intervals can be used to compute bounds of the underlying probability error of an identification system. However, such intervals are only applicable to independent trials of a single recognition system. In the work reported here the aim is to compare between two or more identification systems to determine which of them is the most accurate. Typically, the only difference between such systems is the features that each employs. Trials that use these different sets of features are not independent because the training utterances are the same for all sets of features and the same test utterances are used to create speaker templates. When two systems are compared under these conditions, the tests can be considered to be paired, because for each utterance there is



**Figure 5.7.** Deviation from  $x$  of the upper confidence ( $\alpha = 0.05$ ) limit for different numbers of observed identification errors: (a) 0 to 20 errors, (b) 80 to 100 errors.

a pair of results, one from each system. This section describes a method of calculating the significance of the differences in recognition performance between two systems (with paired results).

Suppose that test utterances are applied to two identification systems that are labelled here for convenience as system 0 ( $S_0$ ) and system 1 ( $S_1$ ). The probability of error for the two systems  $S_0$  and  $S_1$  is  $p_0$  and  $p_1$  respectively. The aim here is to decide whether experimental evidence indicates that  $p_0 = p_1$ ,  $p_0 \geq p_1$  or  $p_0 \leq p_1$ . Note that McNemar (1947) calls this a comparison of proportions, since  $p_0$  and  $p_1$  can be considered to represent the proportions observed from  $S_0$  and  $S_1$ . The joint performance of system 0 and system 1 can be summarized in a  $2 \times 2$  table as follows:

|       |           | $S_0$            |                  |
|-------|-----------|------------------|------------------|
|       |           | Incorrect        | Correct          |
| $S_1$ | Incorrect | $N_{00}(p_{00})$ | $N_{01}(p_{01})$ |
|       | Correct   | $N_{10}(p_{10})$ | $N_{11}(p_{11})$ |

where  $N_{ij}$  is the number of utterances that fit into the specified category.  $N_{01}$  is the number of utterances that  $S_0$  recognized correctly and  $S_1$  recognized incorrectly, and so on for the other  $N_{ij}$ . The numbers of interest in this table are  $N_{01}$  and  $N_{10}$  since they record the differences in recognition performance between the two systems.

The null hypothesis is that both of the systems have the same performance, i.e.,

$$H_0 : p_0 = p_1. \quad (5.6)$$

Following the notation of Gillick and Cox (1989), the total number of utterances that only one algorithm identified correctly is  $k = N_{01} + N_{10}$ . It is useful to introduce the notation  $E(N_{01})$  to describe the expected value of  $N_{01}$ . Under  $H_0$ , the expected value of  $N_{10}$  equals the expected value of  $N_{01}$  and therefore  $E(N_{01} - N_{10}) = 0$ . In other words  $(N_{01} - N_{10})$  has zero mean. The recorded difference between  $N_{01}$  and  $N_{10}$  must be normalized by the standard deviation of  $(N_{01} - N_{10})$  in order to be used as a test statistic. McNemar (1947) shows that

$$\sigma_{p_1 - p_0}^2 = \sigma_{p_1}^2 + \sigma_{p_0}^2 - 2\text{cov}(p_1, p_0) \quad (5.7)$$

$$= \frac{1}{N}(N_{01} + N_{10}) \quad (5.8)$$

and so defines a chi square distributed test statistic

$$\chi^2 = \frac{|N_{01} - N_{10}|^2}{N_{01} + N_{10}}. \quad (5.9)$$

Edwards (1948) points out that the chi square distribution is continuous, whereas the recorded error counts are discrete, so it is more correct to apply a continuity correction to  $|N_{01} - N_{10}|$ . This is achieved by subtracting 0.5 from the larger of  $N_{01}$  and  $N_{10}$  and by adding 0.5 to the smaller of  $N_{01}$  and  $N_{10}$ . In practice this can be written as

$$\chi^2 = \frac{(|N_{01} - N_{10}| - 1)^2}{N_{01} + N_{10}}. \quad (5.10)$$

$k$  is typically small in the recognition experiments performed here, and so the test proposed by McNemar (1947) provides an exact test of  $H_0 : p_0 = p_1$  for all sample sizes. Equation (5.10) is therefore used for assessing the differences in performance of various features for speaker identification. The  $\chi^2$  statistic in Equation (5.10) has 1 degree of freedom (McNemar, 1947), and once  $\chi^2$  has been computed for a particular pair of experiments, the chi-square distribution can be used to determine whether the statistic is significant. When attempting to disprove  $H_0$  using the  $\chi^2$  distribution, the value of  $\alpha$  is the probability that  $H_0$  is actually true. The smaller the value of  $\alpha$ , the more significant the differences are between  $S_0$  and  $S_1$ .

Gillick and Cox (1989) state that when  $k > 50$  and neither  $N_{01}$  nor  $N_{10}$  is close to 0, a normal approximation to the exact Binomial probability can be used and (5.10) can be rewritten as

$$W = \frac{|N_{01} - N_{10}| - 1}{\sqrt{N_{01} + N_{10}}}, \quad (5.11)$$

where the test statistic,  $W$ , has 0 mean and unity variance under  $H_0$ . However, when  $k$  is small it is not advisable to use  $W$  as a test statistic Lloyd (1990).

It is possible to extend the tests for differences in proportions so that a confidence interval for the estimation of  $|N_{01} - N_{10}|$  is developed. This is not pursued any further here and the reader is referred to Lloyd (1990) for more detail.

## 5.4 EVALUATION OF DIFFERENT SPEAKER IDENTIFICATION SYSTEMS

This section reports speaker identification accuracies for CEP, LPC, PARCOR, LTAGR and LTAS-VE features. Identification results are presented for each of the features individually, and then for certain combinations of features.

Features based on long-term averaging of the speech signal require utterances of reasonable duration so that sensitivity to short time variations in the utterance is removed. For this reason both the long-term features and the CEP and PARCOR features are evaluated throughout the entire utterance of the digits zero to nine. This is in contrast with work reported by other researchers (Soong *et al.*, 1987), who evaluated the variation in identification accuracy of CEP and LPC coefficients for different numbers of spoken digits. Recall that five utterances are used for training, ten for testing and that the population consists of twenty speakers.

### 5.4.1 Identification using the entire utterance

This section describes the recognition accuracies of PARCOR, LPC and CEP features for speaker identification when the entire utterance, including unvoiced and silence portions, is utilized for both training the VQ codebooks and for identification. The

| Codebook order | PARCOR vs CEP            | PARCOR vs LPC            | LPC vs CEP              |
|----------------|--------------------------|--------------------------|-------------------------|
| 2              | CEP ( $\alpha < 0.001$ ) | LPC ( $\alpha < 0.001$ ) | LPC ( $\alpha < 0.28$ ) |
| 4              | CEP ( $\alpha < 0.10$ )  | LPC ( $\alpha < 0.18$ )  | same                    |
| 8              | CEP ( $\alpha < 0.16$ )  | LPC ( $\alpha < 0.05$ )  | same                    |
| 16             | same                     | same                     | same                    |
| 32             | CEP ( $\alpha < 0.25$ )  | same                     | same                    |
| 64             | CEP ( $\alpha < 0.13$ )  |                          |                         |
| 128            | same                     |                          |                         |

**Table 5.5.** The best vocal tract features for performing speaker identification using the entire utterance. The confidence level  $\alpha$  is derived from the  $\chi^2$  value computed by applying McNemar's test (§5.3.3) to the speaker identification results of Fig. 5.8.

advantage of performing analysis on an entire utterance, including silence and unvoiced portions is that no preprocessing is required before features are abstracted from the speech signal.

The variation in the identification error rate for PARCOR coefficients as the number of codevectors is increased is illustrated in Fig. 5.8(a) and (b). As would be expected, the number of identification errors decreases as the number of vectors in the codebook is increased. The error rate falls at more than 3% per codebook size doubling, as the codebook size increases from 2 to 16, but as the size increases from 16 to 128 the error rate only reduces from 3 % to 1.5%. As the number of codevectors is increased past a certain threshold (in this instance 32), there is not necessarily a corresponding improvement in the identification accuracy.

The identification error rate for the CEP coefficients, shown in Fig. 5.8(c) and (d), follows a similar trend to the PARCOR coefficients, but with fewer errors. The identification error rate falls to 0.5% for codebooks of size 32, 64 and 128. The smaller identification error of CEP coefficients, compared with PARCOR coefficients, indicates that CEP coefficients are preferred for speaker identification.

In order to make the set of features examined more complete, error rates for LPCs with the likelihood-ratio distance measure ( $d_{LR}$ ) are computed. The error rate of the LPCs, depicted in Fig. 5.8(e), follows the same pattern as that of the other features. These results agree closely with those obtained by Soong *et al.* (1985) for vector quantized LPC vectors and using the likelihood ratio distance measure (Gray and Markel, 1976). The similar accuracies between the LPCs and the other features implies that it is acceptable to perform further experiments using just the PARCOR and CEP features and that although speaker identification using LPC<sup>1</sup> and  $d_{LR}$  is prominent in the literature, LPCs do not appear to offer any significant advantage over PARCOR or CEP features.

Table 5.5 tabulates the results of applying McNemars test to the identification results for entire utterances shown in Fig. 5.8. Features that give a value of  $\chi^2 = 0$ , or require  $\alpha > 0.30$  to be considered different, are recorded as having the 'same' identification performance.

To select the best feature, it is necessary to examine the performance of the features for different codebook sizes. From the PARCOR vs CEP column and the PARCOR vs

<sup>1</sup>Note that the codebooks in the case of LPC are limited to 32 vectors because larger numbers of vectors cause convergence difficulties for the VQ training algorithm. It was not considered necessary to investigate this any further since the identification error rate does not reduce significantly for codebooks larger than 32 vectors.

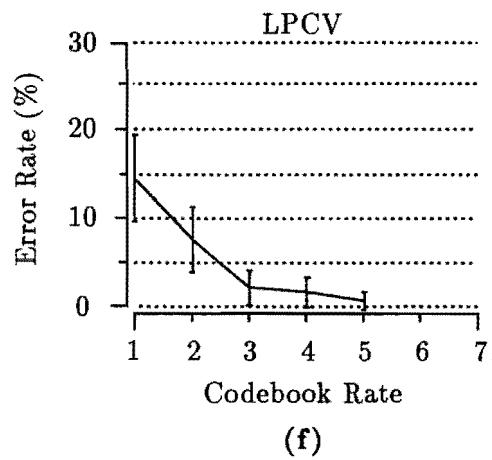
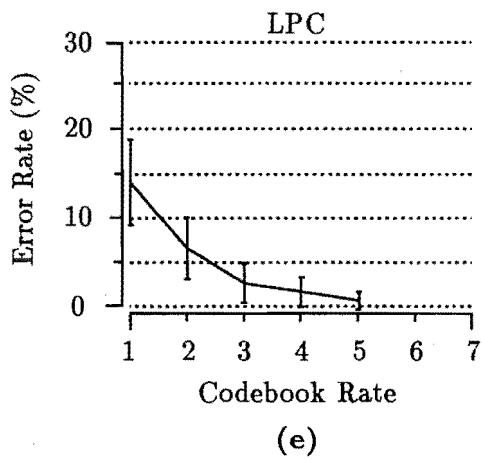
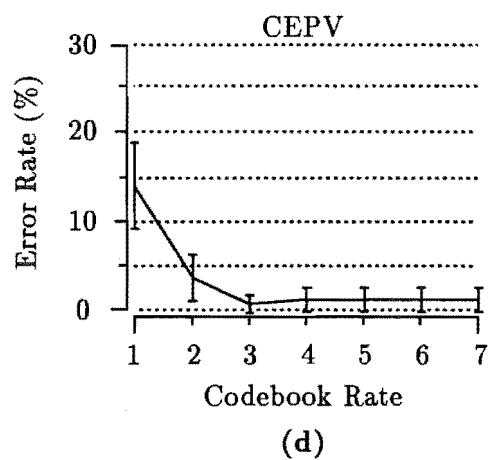
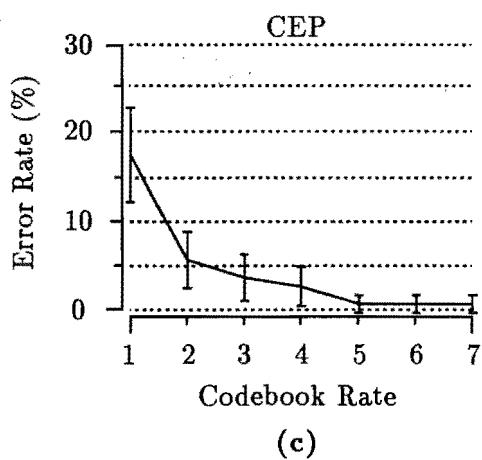
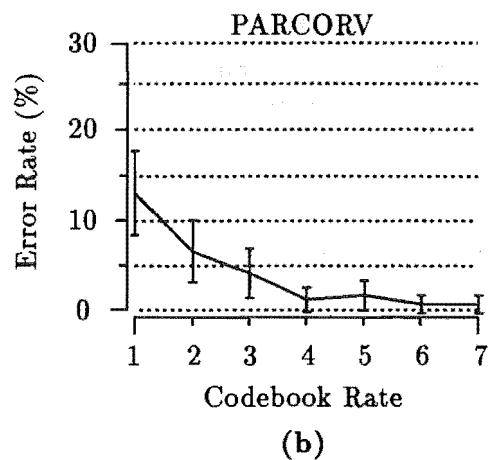
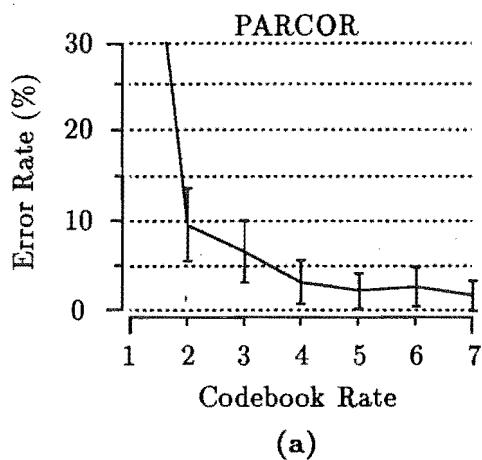


Figure 5.8. A comparison of percentage identification errors for different vocal tract features and utterances. (a), (c), (e) entire utterance; (b), (d), (f) voiced utterance.

LPC column it is apparent that overall the PARCOR features do not perform as well as either the CEP or LPC features. The LPC vs CEP column indicates that the LPC features are slightly better than the CEP features, although only at the low level of significance  $\alpha < 0.28$ .

Table 5.5 also shows that as the order of the codebook is increased, the differences in speaker identification performance of the various features is reduced. To reduce the computation required to search codebooks, and incidentally, to reduce the storage requirements of the speaker templates, it is desirable to choose the smallest possible codebook that gives reasonable speaker identification performance. Since the reduction in the error rate is not significant for codebooks containing more than 16 codevectors, this would seem to be a reasonable size to select for performing speaker identification using entire utterances.

In addition, Table 5.5 shows that for a codebook of size 16 there is no significant difference in the performance of the LPC and CEP features. However, the simplicity of the CEP distance calculation compared with LPC  $d_{LR}$  distance computation (and VQ training), makes CEP16 the best feature.

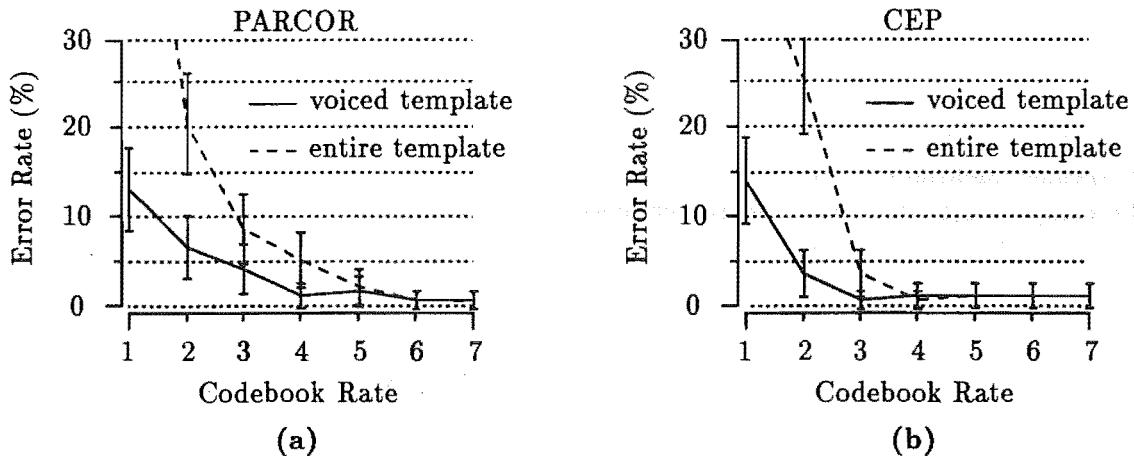
#### 5.4.2 Comparison of identification using voiced and entire utterances

This section describes speaker identification accuracies obtained when features of both the test and reference utterances are abstracted from only voiced portions of speech. The results obtained from voiced speech are compared with those obtained from the entire utterance.

The PARCORV and CEPV identification errors for voiced speech are depicted in Fig. 5.8(b) and (d). Comparing these two results with those obtained from PARCOR and CEP features it is apparent that the identification error rate for voiced frames falls more rapidly than that for the entire utterance. This is to be expected, since the PARCOR coefficients of voiced frames vary less than those of unvoiced frames, and the silences between words do not add to the speaker information that is recorded in the codebooks. This finding contrasts with that reported by Soong *et al.* (1985), who found that identification errors increased when only voiced frames were used for identification. However, their experimental procedure was different to that reported here in one important respect. Their codebooks were trained on the entire utterance, so codebooks that were designed to represent the voiced, unvoiced and silent portions of a person's utterances were matched against voiced frames from the test utterances. The results in Fig. 5.8 are for separate codebooks constructed for voiced and entire utterances.

The reduction in error rate that the voiced speech exhibits over unvoiced speech for CEP and PARCOR features is most significant for codebooks containing a small number of codevectors. As the number of codevectors is increased past 32 the difference in error rates is reduced. Smaller codebooks give such different error rates because codevectors in a small codebook describe the 'average' of a large number of training vectors. For voiced speech the training vectors contain less variation (due to fewer different sounds), so a codebook having few codevectors can represent the speech with higher accuracy than a similarly sized codebook can represent the entire utterance. As the number of codevectors in each codebook is increased the difference in the distribution of training vectors becomes less important so the voiced and entire codebooks tend to give the same error rates.

The LPC features (as depicted in Fig. 5.8(e) and (f)), in contrast to CEP and PARCOR features, do not appear to show any significant identification error rate variation between LPC and LPCV. It seems that LPC features, matched using the  $d_{LR}$  distance



**Figure 5.9.** A comparison of the percentage identification error rates for vocal tract features extracted from voiced utterances matched against codebooks derived from: (i) the entire utterance and (ii) voiced speech. (a) PARCOR features, (b) CEP features.

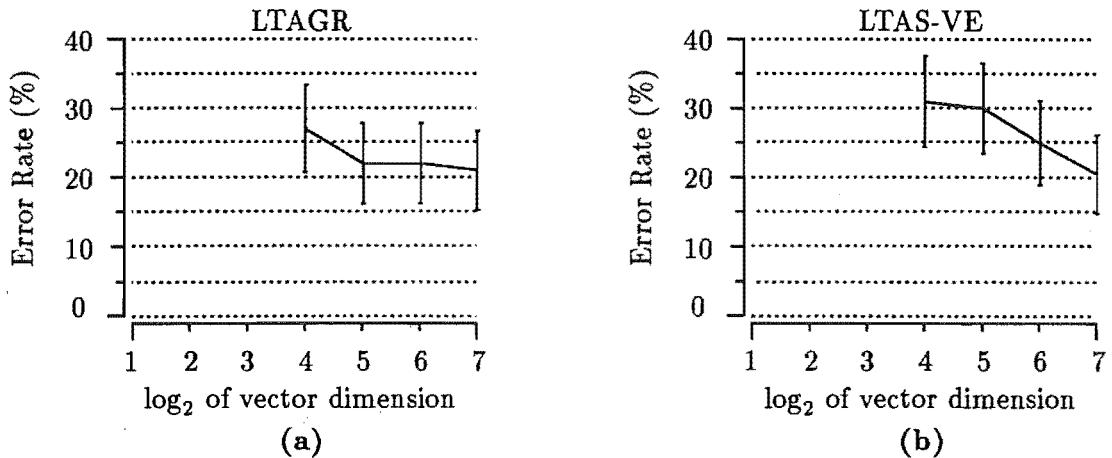
| Codebook order | PARCOR vs PARCORV            | LPC vs LPCV | CEP vs CEPV              |
|----------------|------------------------------|-------------|--------------------------|
| 2              | PARCORV ( $\alpha < 0.001$ ) | same        | CEPV ( $\alpha < 0.25$ ) |
| 4              | PARCORV ( $\alpha < 0.21$ )  | same        | same                     |
| 8              | same                         | same        | CEPV ( $\alpha < 0.05$ ) |
| 16             | PARCORV ( $\alpha < 0.13$ )  | same        | same                     |
| 32             | same                         | same        | same                     |
| 64             | PARCORV ( $\alpha < 0.13$ )  |             | same                     |
| 128            | same                         |             | same                     |

**Table 5.6.** Comparison between voiced only speech and the entire utterance for performing speaker identification. The confidence level is derived from the  $\chi^2$  value computed by applying McNemar's test (§5.3.3) to the speaker identification results of Fig. 5.8.

measure, are less affected by silence and unvoiced speech than the CEP and PARCOR features.

Table 5.6 contains the results of applying McNemar's test to speaker identification experiments performed using either the entire utterance, or voiced portions of the utterance. In all the paired experiments the voiced speech has a lower, or same, identification error rate as the entire utterance. In most of the paired tests the difference between identification error rates is not highly significant statistically. Notice that the LPC features give the same identification accuracy for both voiced and entire utterances, whereas the overall trend for CEP and PARCOR coefficients is that performing speaker identification using voiced portions of an utterance is much better than performing speaker identification using the entire utterance.

As mentioned previously, Soong *et al.* (1987) show that speaker identification using templates constructed from entire utterances is more accurate when entire utterances, rather than voiced only utterances, are used for testing. They deduce, therefore, that it is better to use the entire utterance for performing speaker identification and in an earlier paper Soong *et al.* (1985) explain, "we not only eliminate the need to separate



**Figure 5.10.** Variation in identification error rates for: (a) LTAGR and (b) LTAS-VE features as the dimension of the feature utilized for matching is varied from 16 ( $\log_2 16 = 4$ ) to 128 ( $\log_2 128 = 7$ ).

voiced frames from the input data, but also we improve the speaker recognition performance by using all the speech data". This statement is evaluated here by performing speaker identification experiments using voiced test utterances and matching against two different sets of templates. One set of templates is formed from the entire utterance and the other set of templates is formed from the voiced only portion of the utterances. Fig. 5.9 shows that the identification error rate for matching voiced test utterances against entire templates is higher than that obtained from matching voiced test utterances against voiced templates. Soong *et al.* (1987) were therefore not completely correct to say that "it is not wise to discard an unvoiced speech segment", since it is only unwise if the speaker templates are constructed from the entire utterance.

#### 5.4.2.1 Performance of the LTAGR and LTAS-VE

Fig. 5.10(a) shows the variation in the LTAGR identification error as the number of samples in the LTAGR feature is altered. The insignificant reduction in the percentage error as the vector dimension is increased above 32, indicates that adjacent sample points within the LTAGR are highly correlated, and that redundant speaker information is recorded.

The identification error rate for the LTAS-VE is shown in Fig. 5.10(b), and for a feature vector size of 128 the error rate is only 0.5% different from that obtained by performing identification using the LTAGR. That two features calculated by two distinctly different methods should give such similar error rates is surprising. It might be expected, therefore, that the identification errors derived from these two features would be highly correlated. However, examination of the distribution of the identification errors amongst the trial population, as illustrated in Fig. 5.11, shows that the number of errors associated with each individual varies between the LTAGR and LTAS-VE features, which implies that the features are uncorrelated.

From inspection of Fig. 5.10(b), it is apparent that the LTAS-VE is more sensitive to a reduction in feature dimension than the LTAGR. This indicates that increasing the bandwidth of the filters utilized to determine the long-term average spectrum, and thereby reducing the feature dimension, would significantly increase the identification error rate.

A similar effect to increasing the bandwidth of the LTAS-VE filters can be observed by altering the shape of the window used to calculate the LTAS-VE. The difference between using the Hamming and four-term Blackman windows (described in more detail

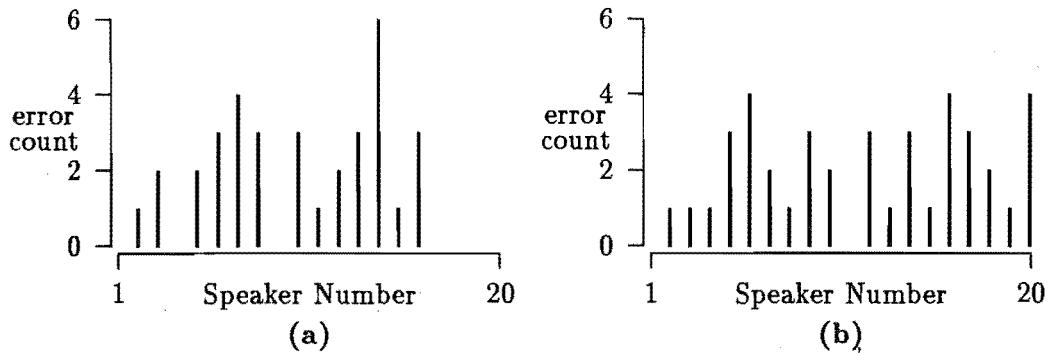


Figure 5.11. The identification error rate for each speaker using: (a) LTAGR and (b) the LTAS-VE as recognition features.

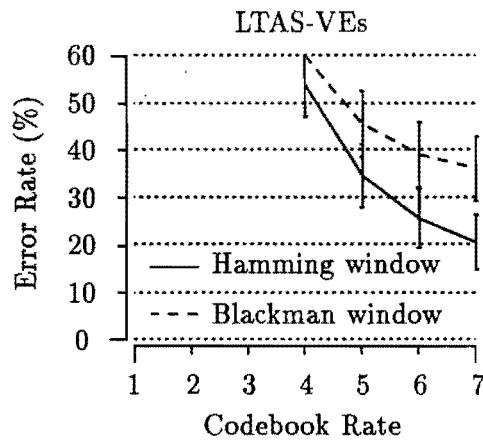


Figure 5.12. A comparison of the percentage identification error rates for LTAS-VE features calculated using different windows.

in Harris (1978)) when computing the LTAS-VE is evaluated. The main difference between these two windows is their spectral resolution, which is related to the width of the main lobe of the Fourier transform of the window. The Fourier transform of the Hamming window has a narrow main lobe and therefore allows good spectral frequency resolution, but has high sidelobe levels. The high sidelobes cause a considerable amount of leakage (§2.6.1). The Fourier transform of the four-term Blackman window has low sidelobe levels, but the wide main lobe causes considerable smoothing of the LTAS-VE. The identification error rates depicted in Fig. 5.12, show that the LTAS-VE with the highest frequency resolution, using the Hamming window, has the best identification error rate. The difference in speaker identification error-rate for the two windows is somewhat surprising. Clearly the spectral detail in the LTAS-VE contains information about a speaker's voice that is useful for distinguishing that person from other people, but it is difficult to draw any further conclusions. Section 6.2.1 discusses implications this observation has for clinical applications of the LTAS-VE.

| Step | Variable Entered | Wilk's Lambda |
|------|------------------|---------------|
| 1    | $P_2$            | 0.08562822    |
| 2    | $C_{c0}^l$       | 0.01723747    |
| 3    | $T_R$            | 0.00361524    |
| 4    | $A_s$            | 0.00088953    |
| 5    | $T_F$            | 0.00031008    |
| 6    | $P_l$            | 0.00012556    |
| 7    | $A_r$            | 0.00003162    |
| 8    | $C_{c3}^r$       | 0.00001021    |
| 9    | $C_{c0}^r$       | 0.00000461    |
| 10   | $C_{c2}^r$       | 0.00000228    |
| 11   | $C_{c1}^r$       | 0.00000084    |
| 12   | $N_p$            | 0.00000044    |
| 13   | $A_r$            | 0.00000023    |
| 14   | $S_1$            | 0.00000015    |
| 15   | $S_2$            | 0.00000009    |
| 16   | $C_{c3}^l$       | 0.00000006    |
| 17   | $C_{c2}^l$       | 0.00000003    |
| 18   | $C_{c1}^l$       | 0.00000002    |

Table 5.7. Output of stepwise discriminant analysis performed on the training descriptors.

#### 5.4.2.2 Various LTAGR based methods

It is of interest to ascertain whether speaker identification accuracy is maintained when the dimension of the LTAGR features is reduced by using the LTAGR descriptors defined in §4.3.2. A total of 21 descriptors are defined in §4.3.2, but it would be better if a smaller number of descriptors could be selected, and weighted, so that speaker identification is performed accurately.

The selection of the best subset of descriptors to describe the LTAGR is not a trivial task. The method adopted here is based on the stepwise inclusion of descriptors, as described in §3.4.2.1, and is performed using the STEPDISC procedure in SAS. Table 5.7 lists the order in which descriptors are included in the descriptor set. The effect of gradually including more variables in the LTAGR description is examined by increasing the subset of the variables used to perform speaker identification. Distances are weighted by the inverse of the pooled covariance matrix, as defined in (3.13). Fig. 5.13 shows that generally the identification error rate decreases as more descriptors are added to the subset of descriptors. However, the identification error rate portrayed in Fig. 5.13 does not always decrease as additional descriptors are added. This is because the training descriptors are different to the test descriptors, so the optimal set of descriptors and the pooled covariance matrix of the training descriptors (which provides descriptor weightings) will not necessarily provide the best performance for the test descriptors. The trade-off between having large training sets, and not describing the categories of recognition accurately enough is well-known. In the particular example reported here, increasing the number of samples in the training data would go some way towards providing more reliable estimates of the best descriptor weightings.

The best performance using LTAGR descriptors gave an error rate of 23.5% compared with 21% using all the samples in the LTAGR signal. Since the recognition

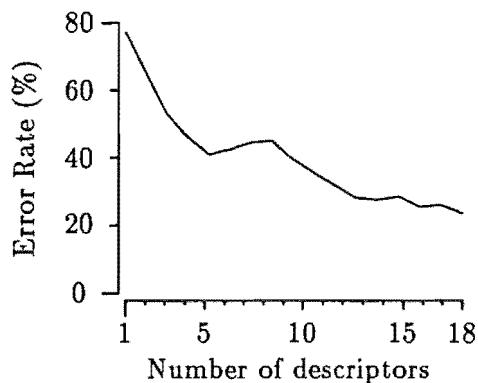


Figure 5.13. Variation in the speaker identification error rate as the number of descriptors used to represent the speaker's LTAGR is increased. See Table 5.7 for a listing of the descriptors.

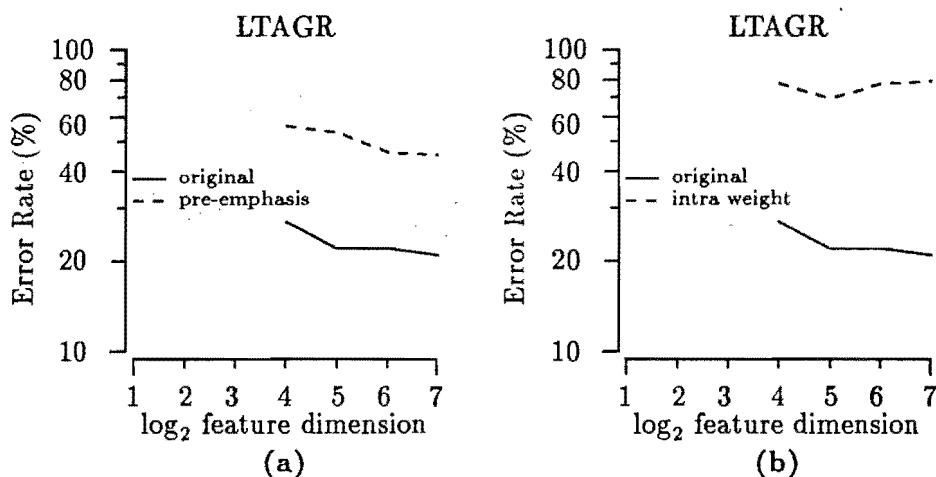


Figure 5.14. Two methods of performing speaker identification based on the LTAGR: (a) LTAGRs computed from pre-emphasized speech, (b) using a distance measure that weights contributions from samples in the LTAGR by the inverse of the pooled intra-speaker covariance matrix.

accuracy is lowered by using descriptors instead of the complete LTAGR, the complete LTAGR is used for performing the identification experiments presented in the remainder of this chapter.

In §4.5.1.1 it was demonstrated that the spectrum of the LTAGR matched closely to the long-term average spectrum of pre-emphasized voiced speech. The long-term average spectrum is computed from pre-emphasized speech, so the effect of pre-emphasis on the LTAGR is examined. Fig. 5.14(a) shows that the LTAGR of pre-emphasized speech has a higher identification error rate than the LTAGR of unemphasized speech. From this result it is concluded that the information characterizing speakers is reduced when high frequencies are emphasized with respect to the low frequencies.

The Euclidean distance, which is used to measure the difference between LTAGRs, weights the contribution from each sample in the LTAGR identically. A more sophisticated and commonly used approach is to weight each sample in the LTAGR by a weight that is related to its expected accuracy. Fig. 5.14(b) shows the effect of weighting the distance between LTAGR samples by the diagonal of the inverse of the pooled intraspeaker covariance matrix ( $\mathbf{W}$ , as defined by (3.10)). The rational is that those parts of the LTAGR that vary least within a speaker should perform best for distin-

| Codebook<br>size | CEPV               |                | PARCORV            |                |
|------------------|--------------------|----------------|--------------------|----------------|
|                  | $(1 - 0.95z^{-1})$ | $(1 - z^{-1})$ | $(1 - 0.95z^{-1})$ | $(1 - z^{-1})$ |
| 2                | 14.5 %             | 13 %           | 13.5 %             | 13 %           |
| 4                | 4 %                | 3.5 %          | 6 %                | 6.5 %          |
| 8                | 0.5 %              | 0.5 %          | 3.5 %              | 4 %            |
| 16               | 1.0 %              | 1.0 %          | 1.0 %              | 1.0 %          |
| 32               | 1.0 %              | 1.0 %          | 1.0 %              | 1.0 %          |
| 64               | 1.0 %              | 1.0 %          | 0.5 %              | 0.5 %          |
| 128              | 1.0 %              | 1.0 %          | 1.0 %              | 0.5 %          |

**Table 5.8.** The identification error rates for two different pre-emphasis filters. Voiced speech is used for the constructing the templates and for performing speaker identification.

guishing between speakers and should be weighted more strongly. However, the results in Fig. 5.14(b) show that this is not the case for the LTAGR. The increase in error rate is due to the normalization performed on the LTAGR before the pooled intraspeaker covariance is calculated. This causes samples around the LTAGR peak to have similar amplitudes, and to be weighted more in the distance calculation. However, these samples do not necessarily contain information relating to a person's identity. Therefore, weights derived from  $\mathbf{W}$  are not useful for computing distances between LTAGRs.

#### 5.4.3 The effect of varying pre-emphasis on vocal tract features

In the speaker identification experiments reported in the above sections the pre-emphasis filter is  $(1 - z^{-1})$ , which differs from the pre-emphasis filter of  $(1 - 0.95z^{-1})$  used by Soong *et al.* (1987) and Furui (1981). So as to ascertain whether the choice of pre-emphasis filter critically affects the identification error rate, experiments were performed using pre-emphasis filters of  $(1 - z^{-1})$  and  $(1 - 0.95z^{-1})$ .

Table 5.8 contains the speaker identification error rate for voiced speech that has been pre-emphasized by the two different pre-emphasis filters. Clearly the two different filters perform almost identically for CEPV and PARCORV coefficients, so the choice of pre-emphasis filter is immaterial.

Both Soong *et al.* (1987) and Furui (1981) use speech that has been sampled at 6.67kHz. The effect of raising the sampling rate to 10 kHz, while maintaining the same pre-emphasis filter, is to reduce the amount of emphasis applied at the higher frequencies. It follows that the pre-emphasis effect should be increased by choosing a filter constant nearer to unity. However, the experimental results reported in Table 5.8 do not support any definitive statement about which of these two pre-emphasis filters is best.

#### 5.4.4 Combining features

Combinations of independent features are expected to be more accurate for speaker identification than individual features. Since different features produce error measures of differing magnitude and accuracy, the error measures must be scaled appropriately before being combined to produce an overall distance measure. A combined distance measure  $D_C(\mathbf{x}, \mathbf{y}_i)$  is used to specify different normalizing and weighting methods,

$$D_C(\mathbf{x}, \mathbf{y}_i) = k^1 \frac{D^1(\mathbf{x}, \mathbf{y}_i)}{D_N^1} + k^2 \frac{D^2(\mathbf{x}, \mathbf{y}_i)}{D_N^2} + \dots + k^F \frac{D^F(\mathbf{x}, \mathbf{y}_i)}{D_N^F}, \quad (5.12)$$

|                      | R1    | R2    | M1    | M2    | M3    | M4    | M5a   | M5b   |
|----------------------|-------|-------|-------|-------|-------|-------|-------|-------|
| CEP                  | •     | •     | •     | •     | •     | •     | •     | •     |
| PARCOR               |       |       | •     |       | •     | •     |       |       |
| LTAGR                |       |       | •     | •     | •     | •     | •     | •     |
| Voice only           |       | •     |       |       |       |       |       |       |
| $k$                  |       |       | 1     | 1     | $k_1$ | $k_2$ |       |       |
| normalization        |       |       | •     | •     | •     | •     |       |       |
| presorting           |       |       |       |       |       |       | •     | •     |
| presorting threshold |       |       |       |       |       |       | $p_1$ | $p_2$ |
| Codebook sizes       | R1    | R2    | M1    | M2    | M3    | M4    | M5a   | M5b   |
| 8                    | 3.5 % | 0.5 % | 4.0 % | 8.0 % | 4.0 % | 3.0 % | 3.0 % | 2.0 % |
| 16                   | 2.5 % | 1.0 % | 2.5 % | 5.0 % | 2.0 % | 1.5 % | 2.5 % | 2.0 % |
| 32                   | 0.5 % | 1.0 % | 2.5 % | 4.5 % | 1.0 % | 0.5 % | 2.0 % | 1.0 % |
| 64                   | 0.5 % | 1.0 % | 1.5 % | 4.5 % | 0.5 % | 0.5 % | 2.0 % | 1.0 % |

Table 5.9. The top half of the table contains a summary of different methods for incorporating more than one feature into the speaker recognition scheme. Capital 'R' is an abbreviation for reference and 'M' is an abbreviation for method. Definitions for feature weightings and the presorting threshold are as follows:  $k_1^f = \frac{\text{intraspeaker}^f}{\sigma_{\text{inter}}^f + \sigma_{\text{intra}}^f}$ ,  $k_2^f = \frac{\text{interspeaker}^f - \text{intraspeaker}^f}{\sigma_{\text{inter}}^f + \sigma_{\text{intra}}^f}$ ,  $p_1 = \frac{\text{intraspeaker}^f}{\sigma_{\text{inter}}^f + \sigma_{\text{intra}}^f} + \sigma_{\text{intra}}^f$  and  $p_2 = \frac{\text{intraspeaker}^f}{\sigma_{\text{inter}}^f + \sigma_{\text{intra}}^f} + 2\sigma_{\text{intra}}^f$ . The bottom half of the table gives the identification error rates for several different identification schemes which make use of more than one feature for identification.

where  $F$  is the total number of features,  $D_N^f$  normalizes the distance  $D^f(\mathbf{x}, \mathbf{y}_i)$  for feature  $i$ , and  $k^f$  weights the distance according to its expected accuracy. For convenience the normalized distance for feature  $f$  is expressed as

$$D_N^f(\mathbf{x}, \mathbf{y}) = \frac{D^f(\mathbf{x}, \mathbf{y}_i)}{D_N^f}. \quad (5.13)$$

The various combinations of features and weightings examined here are specified in the upper half of Table 5.9. Different methods of combining features are denoted 'M' and the two reference methods are denoted 'R'. The reference methods both use single features (CEP) that give good recognition performance. Note that methods M5a and M5b do not utilize (5.12), but are instead based on using a single feature (LTAGR) to presort the population before using another feature (CEP). The following sub-sections discuss the accuracies of various methods of combining features and compare the results with those obtained from the two reference methods. The LTAS-VE is not investigated here. It was rejected as a useful feature on the grounds that it is too computationally intensive to calculate compared with the other features (see §5.6).

#### 5.4.4.1 Method 1 (distance normalization, all features)

The first method for combining features normalizes the distances associated with each feature so that they can be added to those calculated from other features. Normalization is performed using the normalization factor,

$$D_N^f = \sum_{i=1}^N D(\mathbf{x}, \mathbf{y}_i), \quad (5.14)$$

where  $N$  is the total number of speakers.  $D_N^f$  is the sum of the distances between the test vector  $\mathbf{x}$  and each of the reference templates. The identification error when (5.12), (5.14) and  $k^f = 1$  are applied to the PARCOR and CEP codebooks and the LTAGR, for entire utterances is given in column M1 of Table 5.9. For a codebook of size 16 the recognition accuracy shows a slight improvement over that obtained from PARCOR16 coefficients (see Fig. 5.8), but no improvement over CEP coefficients by themselves. The error rate of the combined features does not decrease as rapidly with increase in codebook size as that of the CEP coefficients alone.

#### 5.4.4.2 Method 2 (distance normalization, CEP and LTAGR)

Recalling the high correlation between CEP and PARCOR coefficients reported in Table 5.2, it is expected that the PARCOR and CEP coefficients make similar contributions to the error calculated between test features vectors and reference templates. For this reason the PARCOR features are omitted from the feature set, leaving only the CEP and LTAGR features. Comparing M1 and M2 in Table 5.9, it is apparent that M2 has many more identification errors. This can be attributed to the removal of the PARCOR coefficients allowing distances from the LTAGR features to contribute more to the overall error measure, thereby increasing the number of identification errors towards the number of errors obtained using the LTAGR alone. It is therefore advantageous to combine contributions from different features in a way that incorporates prior knowledge about their expected accuracy.

#### 5.4.4.3 Method 3 (feature weighting with intraspeaker distance)

This is the first of two methods which normalize the error from a particular feature and then weight it in some way. The normalization step is that defined by (5.14) and the normalized distance  $D_N^f(\mathbf{x}, \mathbf{y}_i)$  is weighted by multiplying it by the average intraspeaker distance within the training set for that particular feature, i.e.  $k_1^f = \overline{\text{intraspeaker}}^f$ . For larger codebooks this method is slightly better than method 1, although it does not show any improvement over method 1 for a codebook of size 8.

#### 5.4.4.4 Method 4 (modified feature weighting)

The expected accuracy of a particular feature is governed by more than just the average intraspeaker distance. Fig. 5.15 shows an example of the distribution of intraspeaker and interspeaker distances. It is desirable that the overlap between the two distributions be as small as possible, implying that the difference between the interspeaker and intraspeaker distances should be as large as possible and the standard deviation of both of these distances should be small. Using these concepts, a scale factor,  $k_2^f$ , can be defined as

$$k_2^f = \frac{\overline{\text{interspeaker}}^f - \overline{\text{intraspeaker}}^f}{\sigma_{\text{inter}}^f + \sigma_{\text{intra}}^f}, \quad (5.15)$$

where  $\sigma_{\text{inter}}$  is the standard deviation of the interspeaker distance and  $\sigma_{\text{intra}}$  is the standard deviation of the intraspeaker distance. As Table 5.9 shows, the identification error rate using this weighting factor is smaller than that obtained using other weighting factors.

#### 5.4.4.5 Method 5 (presort using LTAGR)

This method utilizes the long-term average glottal response as a presorting selector to choose a sub-population for further identification by CEP features. To choose the

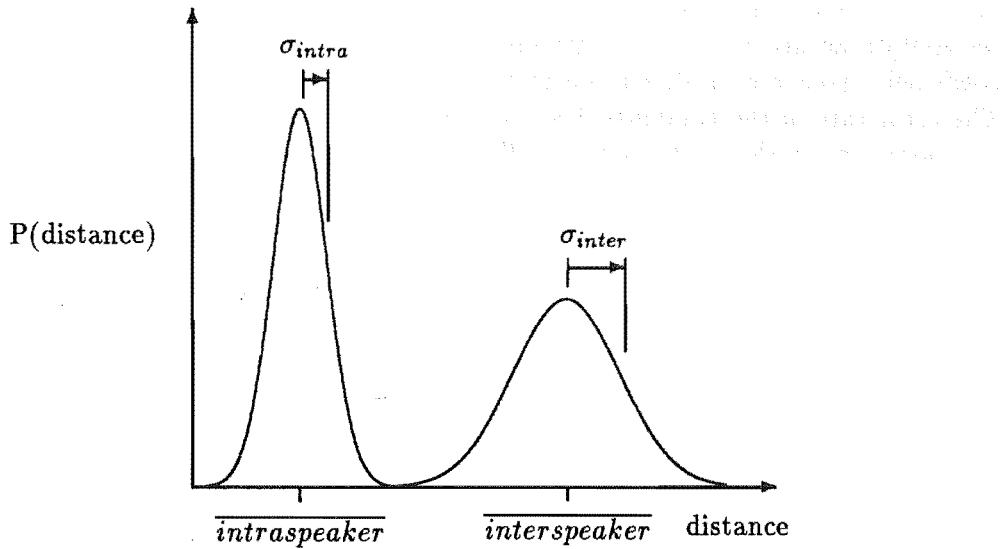


Figure 5.15. A representative probability density function of intraspeaker and interspeaker distances.

people out of the total population who are 'likely' candidates, LTAGR distances are compared with a predetermined threshold and those whose distances are less than the threshold are passed on to a recognition scheme that uses CEP features. Choosing a threshold level that is too small will result in fewer 'likely' candidates, which makes the final selection faster, but increases the probability of the correct person being excluded from the 'likely' candidates. A larger threshold has the opposite effect. Here the LTAGR threshold for selecting people is determined from the training data so as to ensure that the correct person has a low probability of being excluded from the selected sub-population. Information about the distances between the correct person and their templates is recorded in the distribution of intraspeaker distances. Two different selection thresholds are compared: one equal to the overall average intraspeaker distance plus one standard deviation and the other equal to the overall average intraspeaker distance plus two standard deviations. The results using the largest (average +  $2\sigma$ ) threshold compare favourably with the methods that use combined features. Using this method, codebooks of size 8 and 16 are at least as accurate as the other methods, and for codebooks of size 32 and 64 the error rate is increased by no more than 0.5% and in most cases is reduced. Notice that the error rate for M5a with a codebook of size 8 is smaller than that for R1 with a codebook size of 8. This occurs when the presort excludes a person who was identified incorrectly using R1.

The size of the selected sub-population relative to the original population is 0.52 and 0.59 for the thresholds specified by methods 5a and 5b respectively. A small threshold has the advantage of reducing the number of speakers in the sub-population, but in some instances the correct speaker will be excluded. Fig. 5.3 shows that when a test utterance is spoken by speaker 18 the threshold has to be set considerably higher than is required by the other speakers to ensure that speaker 18 remains in the sub-population. The large variation in one person's long-term average glottal response forces the threshold to be increased, causing a greater number of speakers to be included in the sub-population than would otherwise be required.

| Method | Advantages  | Disadvantages   |
|--------|---|---|
| R2     | Single feature.<br>Accurate for small codebook.<br>0.5% best IER. | Requires V/UV.  |
| M4     | Accurate for a large codebook.<br>0.5% best IER.                  | Not accurate for a small codebook<br>Requires the use of three separate features. |
| M5b    | Presorting lowers computational requirements.<br>1.0% best IER.   | Not so accurate for a small codebook.   |

**Table 5.10.** Advantages and disadvantages of the three best methods described in Table 5.9. IER is an abbreviation for identification error rate.

#### 5.4.4.6 Discussion of feature combination results

Table 5.9 shows that, with the exception of M2, the various combinations of features all give comparable identification performances, and do not offer significant improvements over the method R1. M4, the new method of combining contributions from several features, uses a weighting factor derived from both the intraspeaker and interspeaker distances in the training data to give lower error rates than the other less sophisticated methods of combining features.

Presorting the speakers by LTAGR distance before performing speaker identification using the CEP features allows the search population to be approximately halved. Further experiments on databases containing more speakers would be required to ascertain whether larger reductions are possible or whether the ratio of a half is independent of population size.

It is not possible to make a definitive statement about which of the identification methodologies is best. The advantages and disadvantages of methods R2, M4 and M5b are tabulated in Table 5.10. To understand some of the comments listed in Table 5.10, it is important to realize that a large codebook containing many codevectors takes longer to search and identification is much slower than for a small codebook. The actual system selected depends upon the relative importance assigned the points presented in Table 5.10.

## 5.5 FACTORS THAT REDUCE THE ACCURACY OF SPEAKER IDENTIFICATION

One of difficulties in performing speaker identification is that it is often not possible to guarantee that test utterances are recorded in the same conditions as the training utterance. This is especially true if recordings are taken over the public telephone system. Such recordings are dependent on the response of the telephone handset, any noise on the channel and the transmission characteristics of the telephone circuit. For the purposes of this discussion the system comprised of the microphone and transmission circuitry is called the recording channel.

A speaker identification system should be constructed to be as robust as possible to likely variations in the recording channel. This motivates evaluation of the effect of noise and several types of distortion on the identification accuracy of various features. Section 5.5.1 discusses the effect of noise and §5.5.2 examines the effects of distortion on

the speaker identification performance of CEPV, PARCERV, LTAS-VE and LTAGR features.

### 5.5.1 The effect of noise

In many situations the signal used for speaker identification may have a significant level of noise that ‘disguises’ any speaker specific information. Here the effect of noise is examined by adding noise to the original ‘clean’ digitized speech before performing speaker identification. Both random additive noise and speech correlated noise are examined.

#### 5.5.1.1 Gaussian noise

Various levels of Gaussian distributed random noise are added to the digitized test utterances to produce SNRs (as defined in §2.9.1) of 30, 20 and 10 dB. For CEPV, PARCERV and LTAS-VE features, voiced speech within these ‘noisy’ test utterances is extracted using the method outlined in §2.4.4 and the speech is pre-emphasized by  $1 - 0.95z^{-1}$  before speaker identification is performed by matching the utterances against codebooks (or templates) derived from ‘clean’ speech. The identification error rates for the CEPV, PARCERV, LTAS-VE and LTAGR features are portrayed in Fig. 5.16. The CEPV and PARCERV features are particularly strongly affected by Gaussian noise. The identification error rate rises from approximately 1% for high quality speech to more than 60 % for speech that has a 30 dB signal-to-noise ratio. Lower SNRs cause further degradation in the identification error rate.

The experiments performed here use features calculated directly from noisy speech without attempting to compensate for the noise at all. A logical extension of this study would be to incorporate methods to combat the effects of the noise by estimating its magnitude and adjusting the way the features are computed (Mansour and Juang, 1988; Junqua and Wakita, 1989). Un and Choi (1981) subtracted the estimated autocorrelation of the noise from the autocorrelation of the noisy speech before performing LPC analysis. An alternative method of countering the noise was reported by Noda (1988) who devised a spectral weighting scheme where high energy portions of the spectrum (formants) that were less affected by noise, were weighted more heavily than other lower energy portions of the spectrum.

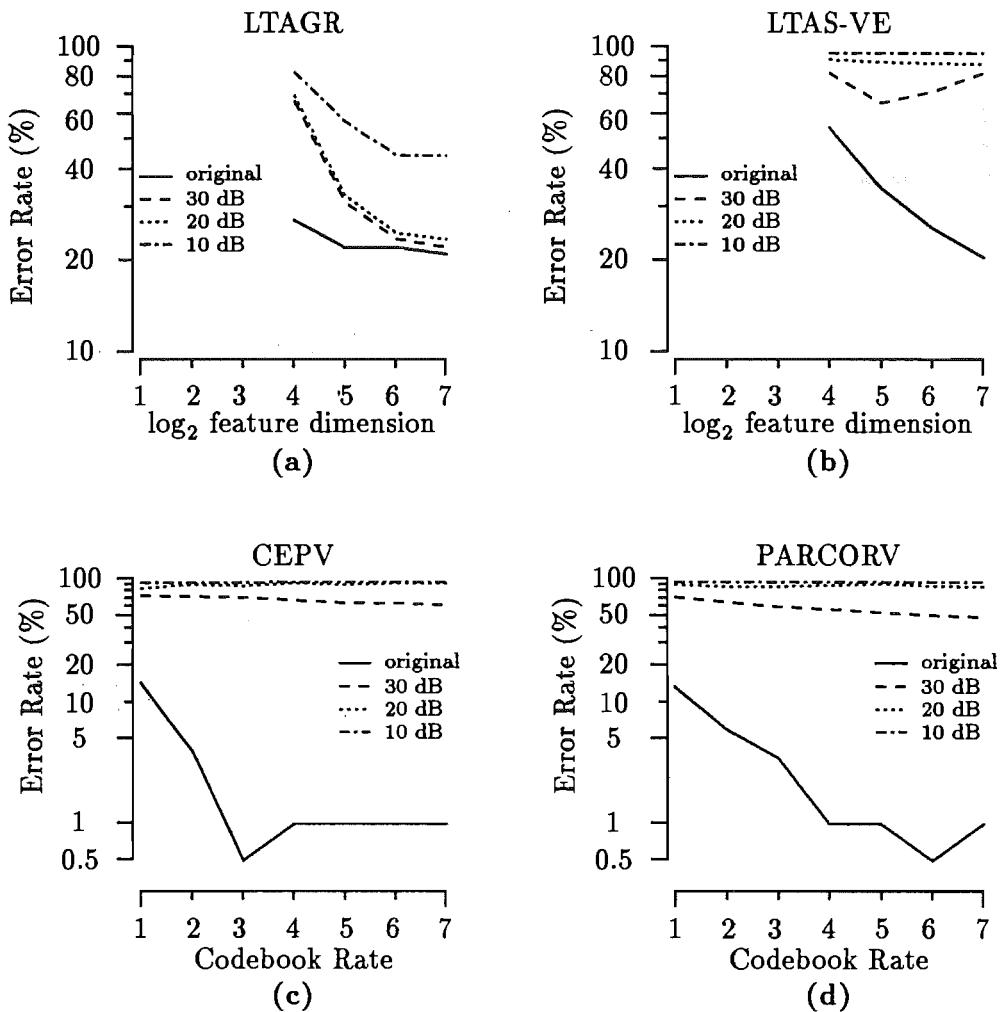
Due to the removal of random, zero mean, variations in the speech signal by the averaging process that occurs within the SAA algorithm, the LTAGR is relatively insensitive to Gaussian noise compared with the PARCERV, CEPV and LTAS-VE. Note that although the LTAS-VE incorporates averaging, the effect of the noise is not reduced because power spectra are averaged instead of time sequences.

#### 5.5.1.2 Speech correlated noise

Speech correlated noise is introduced here because it is a type of noise that corresponds to the noise levels in speech as perceived by listeners. Noise of low energy is generated when the speech signal contains little energy and noise of high energy is generated when the speech signal contains high energy. Equation (2.116) in §2.9.1 that defined speech correlated noise is restated here for convenience, i.e.

$$r(t) = s(t) + ks(t)n_0(t). \quad (5.16)$$

The signal-to-speech-correlated-noise ratios (or Q, which stands for quality) of 30, 20 and 10 dB are evaluated. Note that the noise source is uniformly distributed random noise.

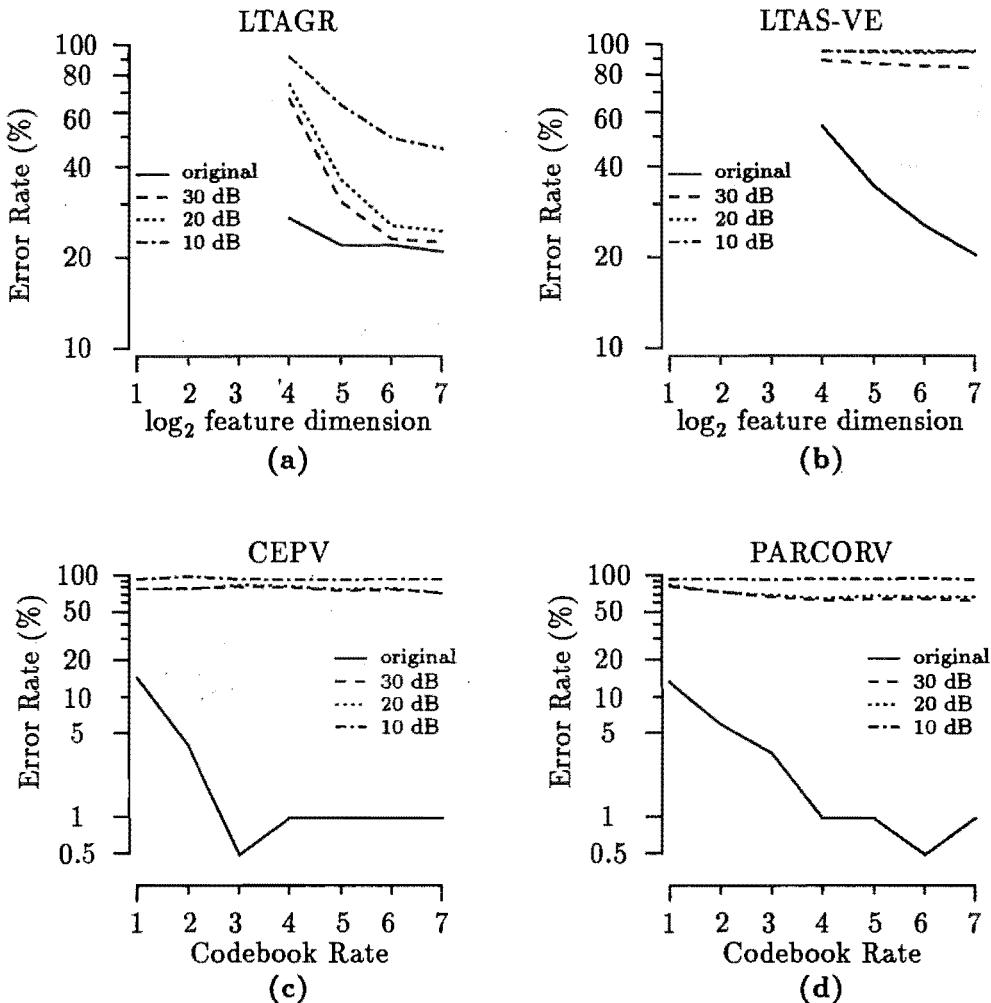


**Figure 5.16.** The identification error rates for different levels of contamination by Gaussian distributed random noise. Signal-to-noise ratios of 30, 20 and 10 dB are depicted for: (a) LTAGR, (b) LTAS-VE, (c) CEPV and (d) PARCORV features.

Fig. 5.17 shows that the CEPV, PARCORV and LTAS-VE features all exhibit severe degradation, even at the smallest distortion level of 30 dB. The LTAGR performs significantly better than the other features, with an error rate of less than 30% for Q values of 20 dB and 30 dB and vector dimensions of 64 or 128. The advantage of LTAGR over the other features is best explained by substituting (2.106), which is the  $m^{th}$  pitch period of a speech signal  $r(t)$ , into (5.16) which describes a speech signal that is degraded by speech correlated noise. The  $m^{th}$  contaminated speech record is then written

$$r_m(t) = g_m(t - T_{em}) \odot v_m(t)(1 + kn_0(t)) + c_m(t)(1 + kn_0(t)). \quad (5.17)$$

Recall that  $v_m(t)$  is the  $m^{th}$  vocal tract response,  $g_m(t)$  is the  $m^{th}$  glottal excitation and  $c_m(t)$  is a contamination term. Since  $kn_0(t)$  is random, its effect will tend to be smoothed out of the LTAGR as the  $r_m$  records are averaged by SAA (as described in §2.8.2).



**Figure 5.17.** The identification error rates for different levels of contamination by speech correlated random noise as defined by (5.16). Q ratios of 30, 20 and 10 dB are depicted for: (a) LTAGR, (b) LTAS-VE, (c) CEPV and (d) PARCORV features.

### 5.5.2 The effect of non-ideal frequency response

Examination of the effect of a non-ideal frequency response is divided into two separate parts. The effects of a non-flat magnitude response and a non-linear phase response are treated separately since features that represent the power spectrum of the speech are expected to be less sensitive to a non-linear phase response and more sensitive to a non-flat magnitude response.

#### 5.5.2.1 The telephone channel

This section examines the speaker identification performance of CEPV, PARCORV, LTAGR and LTAS-VE features extracted from speech that has been bandlimited to contain only those frequencies that are typically present in speech transmitted over telephone lines. Fig. 5.18 shows the response for the linear phase filter that is applied to the speech before features are extracted.

The speaker identification error rates for the CEP, PARCOR, LTAGR and LTAS-VE features are depicted in Fig. 5.19. It is apparent that the reduction in information

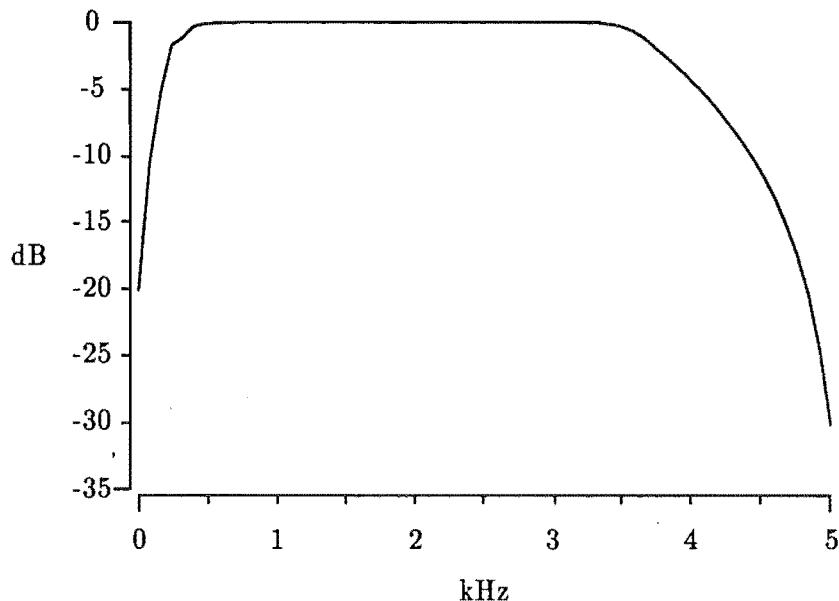


Figure 5.18. The frequency response of the filter used to model the attenuation of a telephone channel.

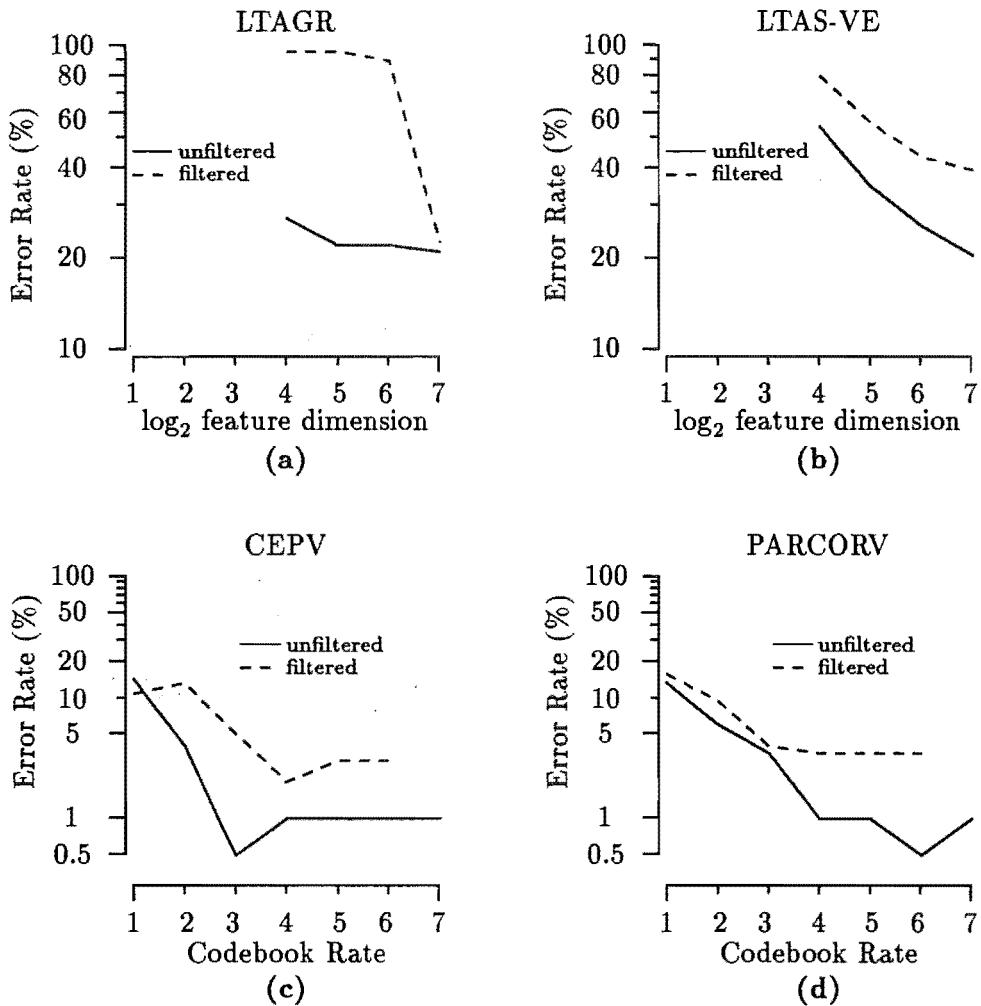
caused by bandpass filtering the speech signal increases the identification error by one or two percent for CEPV and PARCORM features. The LTAS-VE identification error rate is increased so significantly that it can not be considered to be useful for performing speaker identification using bandlimited speech (unless of course new templates were constructed). The LTAGR performs poorly when the feature size is reduced to less than 128 ( $2^7$ ), but for a feature size of 128 the identification error rate is increased only 1.5% above that obtained for high quality speech.

The identification error rates depicted in Fig. 5.19 indicate that it is feasible to accurately identify speakers from speech that has been bandlimited. However, channels exhibit a wide range of different characteristics, prompting further investigations into the effects of the types of distortion that might reasonably be expected to occur over a telephone channel.

### 5.5.2.2 Magnitude

Bogner (1981) identified 2000 and 2700 Hz as frequencies where the most significant differences between the magnitude responses of various telephone headsets occur. Based on Bogner's figures, the effect of the different headsets was simulated by passing the test utterances through either bandpass or bandstop linear phase filters with centre frequencies at either 2000 and 2700 Hz. The digital filters utilized were designed in MATLAB (The Math Works, 1990) using the specifications in Table 5.11. Note that the filters specified in Table 5.11 have only a few terms and so do not have harsh cutoffs. The frequency responses of these filters are plotted in Fig. 5.20 and the bandstop filters clearly provide the least distortion, since their magnitude response is essentially flat across the band, and the maximum attenuation is only 10 dB in the centre of the stop band.

For the PARCORM and CEPV features the effect of the magnitude distortion on the speaker identification error-rate is examined for voiced speech that is pre-emphasized with a  $(1 - 0.95z^{-1})$  filter. The training utterances consist of five undistorted utterances and the ten test utterances are filtered by the distortion filters before being



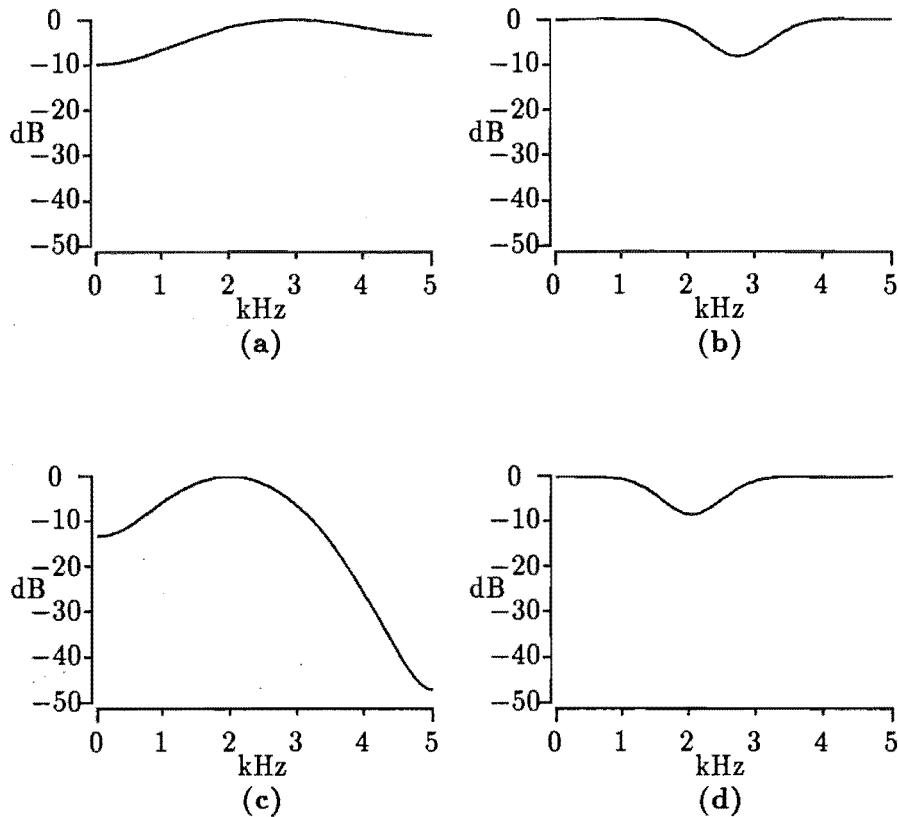
**Figure 5.19.** The identification error rates for speech passed through the telephone filter depicted in Fig. 5.18. The identification error rates plotted for: (a) LTAGR, (b) LTAS-VE, (c) CEPV and (d) PARCORY features.

| filter | type     | frequency | bandwidth | terms |
|--------|----------|-----------|-----------|-------|
| mag1   | bandpass | 2700      | 200       | 6     |
| mag2   | bandstop | 2700      | 200       | 16    |
| mag3   | bandpass | 2000      | 200       | 6     |
| mag4   | bandstop | 2000      | 200       | 16    |

**Table 5.11.** Filter specifications for modelling non-flat magnitude response in the transmission channel.

pre-emphasized and having features extracted. The effect of the various types of frequency response distortion on the speaker identification error-rate is shown in Fig. 5.21.

To begin with, the effect of attenuation in the 2000 Hz and 2700 Hz region is discussed. From the identification error rate results in Fig. 5.21, it would appear that attenuation of approximately 10 dB at 2000 Hz (Fig. 5.20(d)) and at 2700 Hz (Fig. 5.20(b)) has little affect on the error rate for LTAS-VE, CEPV and PARCORY features. For large vector sizes the LTAGR is insensitive to these types of distortion,

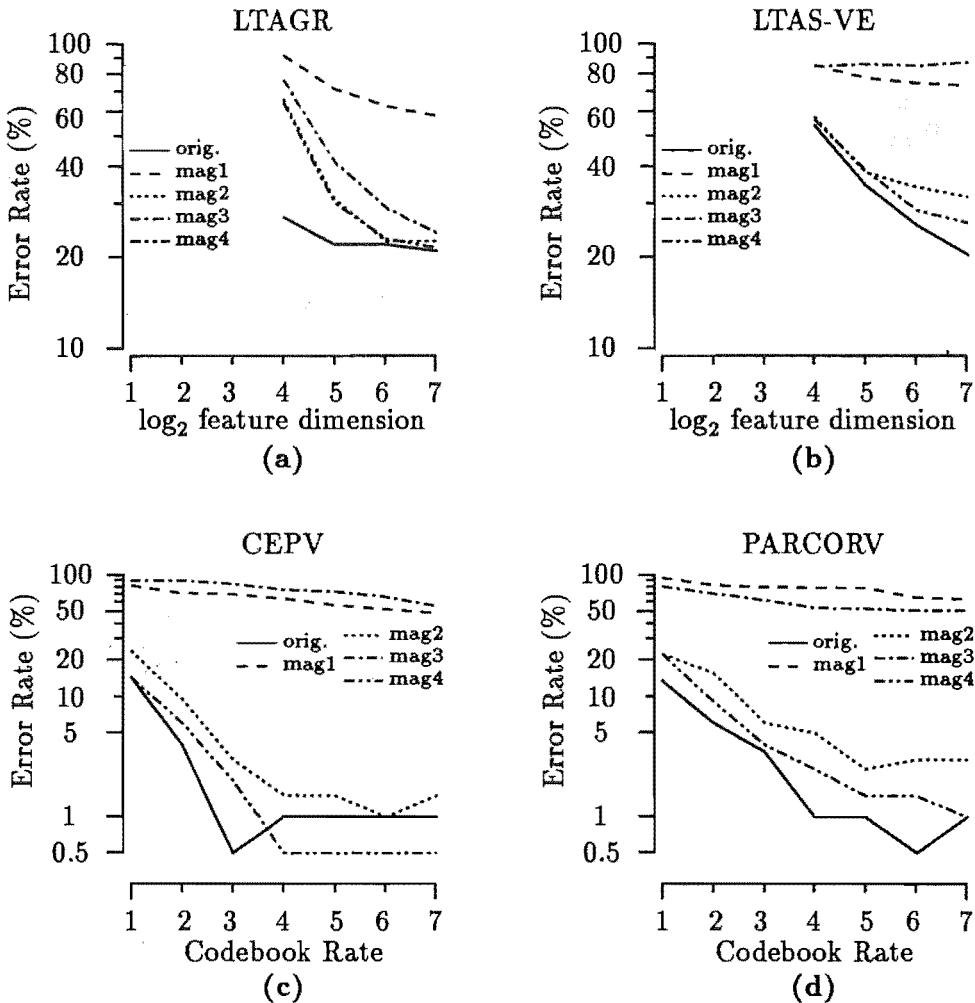


**Figure 5.20.** The magnitude of the filter frequency responses for the distortion filters with a non-flat frequency response. The filters are labelled: (a) mag1, (b) mag2, (c) mag3,(d) mag4.

but for smaller vector sizes the accuracy is significantly reduced. It is apparent that the identification error rate is essentially unaffected by the type of narrow attenuation depicted in Fig. 5.20(b) and (d).

Distortion filters with low frequency attenuation have a significant effect on the identification error rate. None of the features evaluated performed well with speech distorted by the filter depicted in Fig. 5.20(a). It is not surprising that features that describe the speech spectrum do not perform well for this distortion, since the frequencies that contain most of the speech energy (0-3kHz) are attenuated, and the spectral content of the speech is significantly altered.

The identification error rate obtained from using the LTAGR on speech distorted by the mag3 filter (Fig. 5.20(c)) is unusual because all the other features perform badly for this particular distortion. Comparison between mag1 and mag3 filter responses (see Fig. 5.20), shows mag3 to have more attenuation at both low and high frequencies, so the low identification error rate from speech distorted by the mag3 filter is unexpected. Furthermore, recall from §5.4.2.2 that the LTAGR performs poorly on speech that has the low frequencies attenuated and the high frequencies emphasized. However, Fig. 5.22 shows that the LTAGR of speech that has been distorted by mag1 has a very narrow central peak compared with the LTAGR of speech filtered by mag3. Although the mag1 distortion appears to be less severe than the mag3 distortion, its affect on the LTAGR is more pronounced. This can be attributed to the high frequencies tending to dominate the speech signal, causing ‘spikiness’ in the LTAGR. The ‘spikiness’ alters the shape of the main peak of the LTAGR and makes identification between ‘spikey’



**Figure 5.21.** The identification error rates for speech passed through the different magnitude distortion filters depicted in Fig. 5.20. The original speech (orig. in above figure) is filtered by mag1, mag2, mag3 and mag4 and the identification error rates plotted for: (a) LTAGR, (b) LTAS-VE, (c) CEPV and (d) PARCORM features.

LTAGRs and undistorted LTAGRs unreliable.

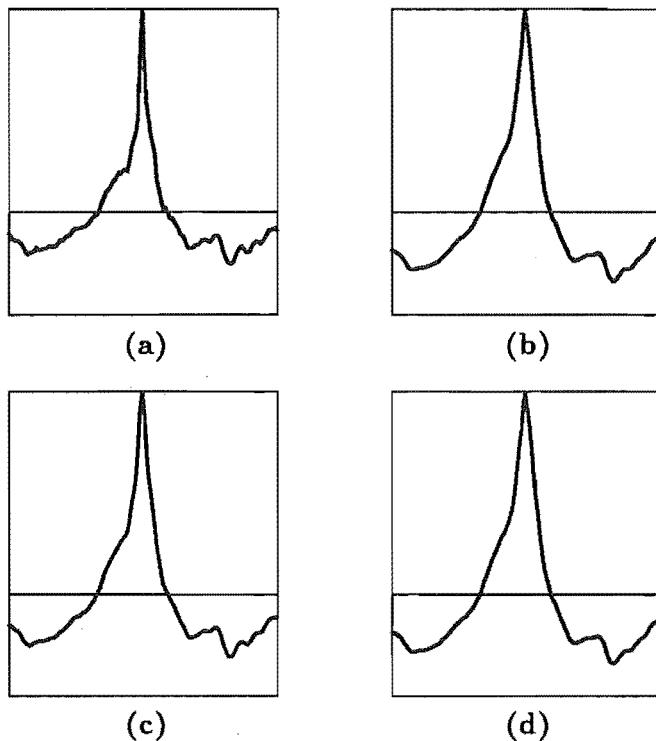
### 5.5.2.3 Phase

The effect of phase distortion on the identification error rate is examined by passing test utterances through a filter whose frequency response is specified by

$$H(f) = e^{i\psi(f)}, \quad (5.18)$$

where  $\psi(f)$  is the phase response of the filter.

The non-linear phase responses used here are based on the phase responses of tenth order Butterworth filters and a phase shifting filter. The filter specifications are summarized in Table 5.12. Filters with 300 Hz and 3500 Hz cutoffs were selected because these types of filter are commonly used to filter speech before transmission over the telephone network. The phases of the various phase distortion filters are plotted in



**Figure 5.22.** The LTAGRs obtained for the utterance AE1 after the speech has been filtered by filters having the magnitude responses depicted in Fig. 5.20. The LTAGR after filtering by: (a) mag1, (b) mag2, (c) mag3 and (d) mag4.

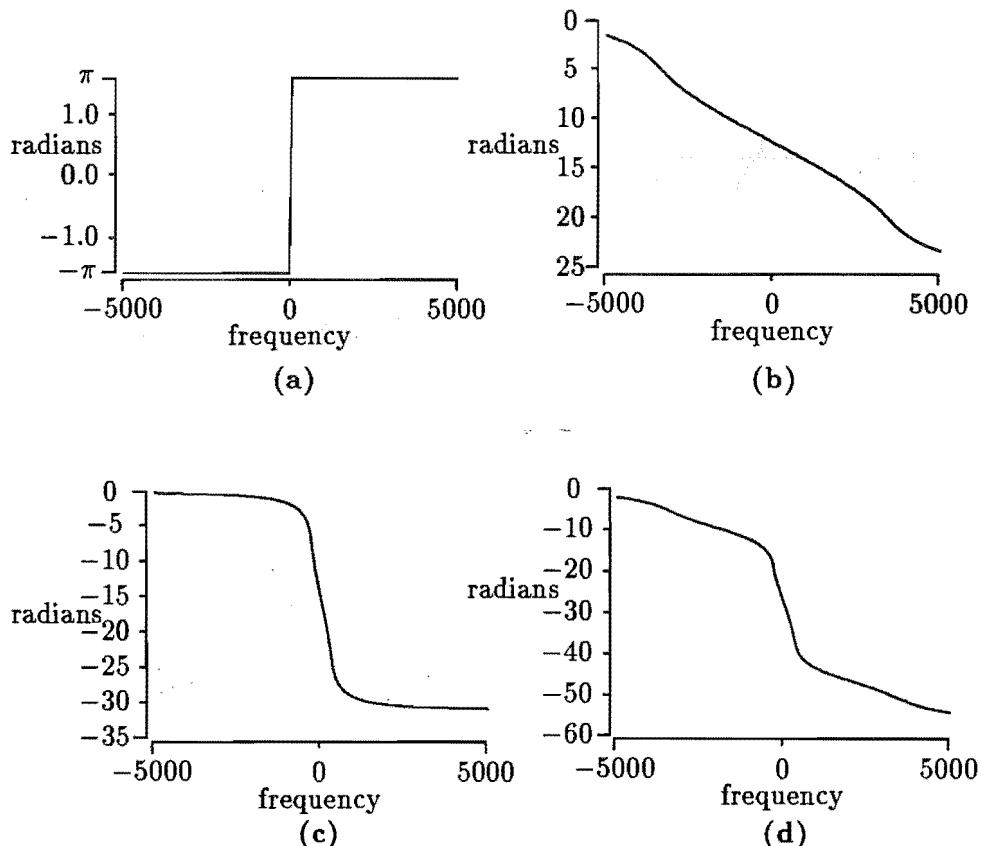
| filter | $\psi(f)$   |
|--------|---|
| ph1    | $\pi/2\text{sgn}(f)$  |
| ph2    | phase of 10th order lowpass Butterworth with 3500 Hz cutoff |
| ph3    | phase of 10th order highpass Butterworth with 300 Hz cutoff |
| ph4    | ph2 + ph3   |

**Table 5.12.** Definition of the phase responses of the different phase distortion filters.

Fig. 5.23. The phase responses are linear over much of the frequency range, with the non-linearities appearing most prominently in the region of the cut-off frequency. In order to examine the sensitivity of the features to extreme phase distortion, a Hilbert transform is used as one of the non-linearities. The impulse responses of the phase distortion filters are computed using the inverse FFT of a unity magnitude filter that has a phase corresponding to one of the phases depicted in Fig. 5.23. This impulse response is then convolved with the original speech to produce phase distorted speech.

Speaker templates for CEPV, PARCORV and LTAS-VE features are constructed from undistorted training utterances that have been pre-emphasized by a  $(1 - 0.95z^{-1})$  filter. Templates for the LTAGR are also formed from undistorted training utterances, but without any pre-emphasis. The test utterances are filtered by phase distorting filters before features are extracted.

The identification results for the various types of phase distortion are depicted in Fig. 5.24. It is apparent that phase distortion filter ph1 has essentially no effect on the identification error rate of CEPV, PARCORV and LTAS-VE features. This is



**Figure 5.23.** Phase responses applied to a flat magnitude response in the design of phase distortion filters. The filters are labelled: (a) ph1, (b) ph2, (c) ph3, (d) ph4.

to be expected since the CEPV, PARCORV and LTAS-VE features model the power spectrum of the speech.

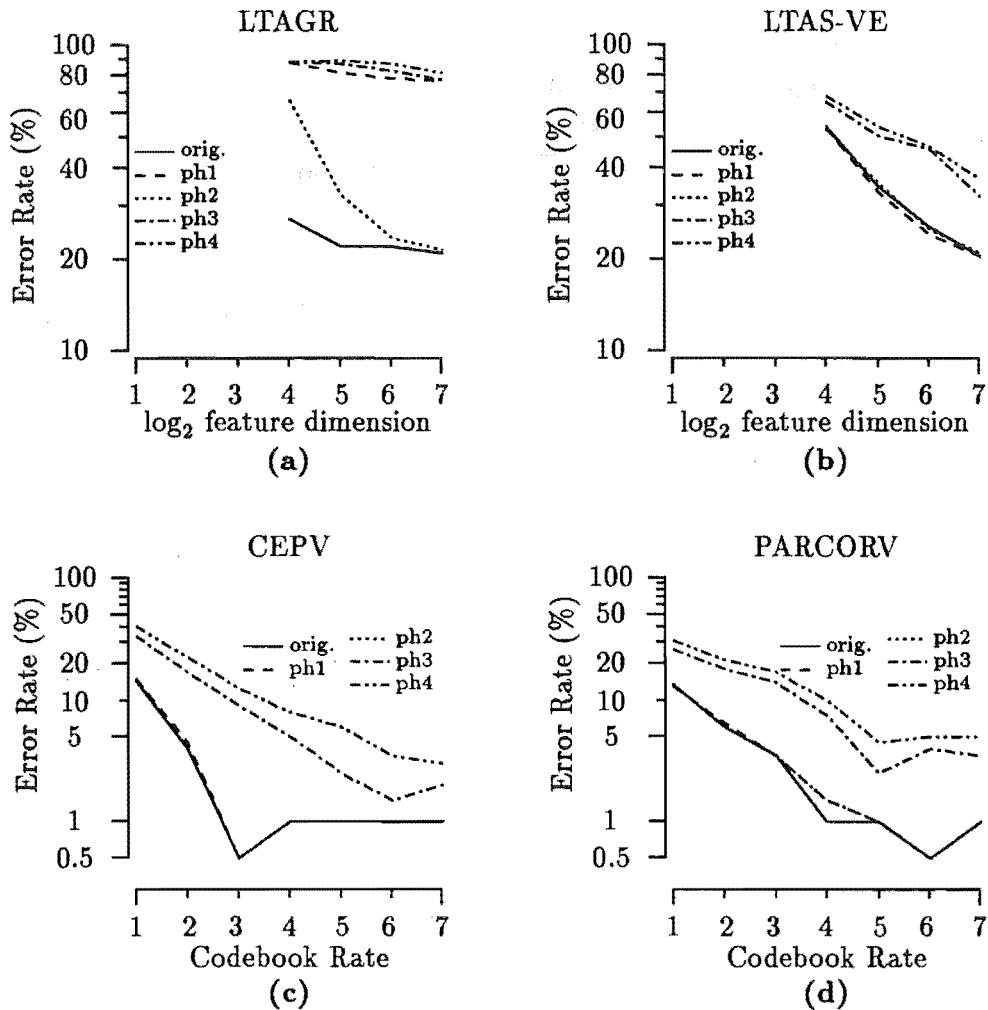
The CEPV, PARCORV and LTAS-VE features are insensitive to the ph2 high frequency non-linear phase distortion depicted in Fig. 5.23(b). Fig. 5.24(a) shows that the LTAGR is sensitive to this type of distortion, particularly when the LTAGR feature vector is of dimension 16 or 32. However, the identification error rate for vectors of 128 samples is not significantly different from that obtained for undistorted utterances. The difference of 44.5% between LTAGR identification error rates for vectors of dimension 16 and 128 indicates that the ph2 distortion causes the LTAGR to be sensitive to reductions in dimension.

The CEPV, PARCORV and LTAS-VE features all exhibit increased error rates of the order of 5% for ph3 and ph4 distortions. The LTAGR has error rates of greater than 80% for ph3 and ph4 distortions.

### 5.5.3 Summary of the effects of noise and frequency response distortion on speaker identification

The examination of the speaker identification error rates for LTAGR, LTAS-VE, PARCORV and CEPV features reveals that they are each affected differently by different types of distortion.

The LTAGR gives much lower error rates than the other features for both Gaussian noise and speech correlated noise. This is due to the averaging that occurs in the SAA



**Figure 5.24.** The identification error rates for original speech (orig. in the above figure) filtered with different phase nonlinearities, as specified by Fig. 5.23. The speech is passed through filters ph1, ph2, ph3 and ph4 and the identification error rates plotted for: (a) LTAGR, (b) LTAS-VE, (c) CEPV and (d) PARCERV features.

algorithm.

The effect of a non-flat frequency response in a channel varies, depending on the characteristics of the non-flat response. Small bands of attenuation (200 Hz wide) have a small effect on the identification error rate, whereas significant attenuation (10 dB or more) in the 0-500 Hz region causes a considerable increase in the speaker identification error rate for all the features except the LTAGR. The LTAGR gives approximately 20% identification error rate for speech that has both the high and low frequencies attenuated (mag3 in Fig. 5.20).

One limitation of the tests reported in §5.5.2.2 is that no attempt is made to correct features for the channel distortion. A reasonably straightforward method of doing this for CEP features is to take the average of the CEP coefficients across all the speech frames and then to subtract this average from all the frames (Birnbaum *et al.*, 1986). This is called channel normalization and is equivalent to deriving CEP coefficients for a speech frame that has been passed through a filter having a frequency response that

| Feature | Additions/s<br>Subtractions/s | Multiplications/s |
|---------|-------------------------------|-------------------|
| LTAGR   | 25 600                        | 0                 |
| PARCOR  | 140 000                       | 140 000           |
| CEP     | 150 000                       | 150 000           |
| LTAS-VE | 160 000                       | 560 000           |

**Table 5.13.** Computational requirements for calculating various features utilized in speaker identification experiments.

corresponds to the inverse of the long-term spectral characteristics of the channel. The long-term average CEP also contains information about the long-term average of the voice, and this is also removed from the CEP features. Experiments by Birnbaum *et al.* (1986) indicate that the accuracy of speaker verification is improved in most instances by channel normalization.

The experiment that examines the effect of different phase distortions highlights the sensitivity of the LTAGR to phase changes. Whereas features such as CEPV, PARCORV and LTAS-VE perform slightly worse (overall), the identification error rate of the LTAGR increases to 80-90% for all but the mildest of phase distortions. This means that the LTAGR can only be used in systems where the phase response of the speech ‘channel’ is reasonably constant.

## 5.6 COMPUTATIONAL REQUIREMENTS

A significant criterion for selecting between features, particularly when their performance is equivalent, is the computational effort required to use them in a speaker identification system.

Computational requirements for matching between vectors are described in terms of the dimensions of each of the vectors, so it is useful to specify  $P_{feature}$  to represent the number of elements in a feature vector. For example, if the CEP16V feature vector consists of 12 cepstral coefficients, it is denoted  $P_{CEP16V} = 12$ .

The computational effort can be divided into two separate parts: (1) calculation of the feature and (2) distance computation between features and reference templates.

### 5.6.1 Feature calculation

The estimated number of operations per second for each feature, as shown in Table 5.13, assumes a sampling rate of 10 000 samples per second. Figures for the number of operations apply to one second of speech, so ‘per second’ is assumed. The following paragraphs outline the assumptions and approximations in arriving at these figures.

In order to estimate the number of operations required to calculate the LTAGR, the pitch was assumed to be 100 Hz, implying a total of 100 glottal response frames per second. The comparisons required to determine the position of the glottal excitation (frame alignment) can be considered to be subtractions, which are similar computationally to additions. The total number of additions recorded in Table 5.13 is therefore composed of 12 800 additions for accumulating the 128 samples in the long-term average glottal response and 12 800 subtractions for locating the peak in the SAA frame. The final normalizing division has not been accounted for in the computation estimate, since it is considered insignificant compared with the total number of additions.

| Feature | Subtractions and additions/s                          | Multiplications/s                              | Typical number of operations/s |
|---------|---|--|--------------------------------|
| LTAGR   | $2 \times P_{LTAGR}$                                  | $P_{LTAGR}$                                    | 128                            |
| LTAS-VE | $2 \times P_{LTAS}$                                   | $P_{LTAS}$                                     | 128                            |
| PARCOR  | $2 \times S \times P_{PARCOR} \times \text{frames/s}$ | $S \times P_{PARCOR} \times \text{frames/sec}$ | 9 600                          |
| CEP     | $2 \times S \times P_{CEP} \times \text{frames/s}$    | $S \times P_{CEP} \times \text{frames/sec}$    | 9 600                          |

**Table 5.14.** Computational requirements for calculating the distance between features. Note that  $P$  is the feature order and  $S$  is the codebook size. The typical number of operations for PARCOR and CEP coefficients assumes  $S = 16$ ,  $P = 12$  and 50 frames per second of speech. The size of the LTAGR and LTAS-VE feature vectors is assumed to be 128.

The tabulated value for the computation required to determine PARCOR coefficients is based on 12 coefficients and frames that are 200 samples long, making a total of 50 frames per second. Most of the computation is in pre-emphasizing, windowing and calculating the autocorrelation since the Durbin-Levinson algorithm requires only order  $P_{PARCOR}^2$  operations to calculate PARCOR coefficients from autocorrelation coefficients. As well as the multiplication operations, the autocorrelation and pre-emphasis operations require a significant number of additions, estimated at 125 000 additions per second.

The CEP coefficients are determined iteratively from the LPC prediction coefficients and therefore only require an additional  $P_{CEP}^2$  operations per frame above the number specified for calculation of PARCOR coefficients.

The number of operations required to calculate the LTAS-VE is dependent upon the number of operations required to evaluate the discrete Fourier transform. Provided the length of the transform ( $N$ ) is a power of 2, as is the case in this situation, the FFT can be invoked. Each FFT requires  $2N \log_2 N$  real multiplications and  $3N \log_2 N$  real additions (Papoulis, 1980). The number of operations reported in Table 5.13 is a result of applying the aforementioned formulas to 128 point frames, accounting for windowing, and scaling by the number of frames per second.

### 5.6.2 Distance measures

The amount of computation required to evaluate the distance between a set of features and a template depends on the number of feature vectors that require matching and the dimension of each of the feature vectors.

Distances for the LTAGR, LTAS-VE, PARCOR and CEP features are computed using the Euclidean distance measure, which can be evaluated using a well defined number of operations consisting of  $N$  subtractions,  $N$  multiplications and  $N$  additions, where  $N$  is the number of elements in the feature vector.

In the situation where the reference template consists of a VQ codebook, multiple distance evaluations are required to determine the best match within the codebook. Furthermore, there will typically be a large number of vectors to match against the codebook. Table 5.14 tabulates the computations required to match frames of PARCOR or CEP coefficients against codebooks of size  $S$ , measured in computations per second of speech.

The computational advantage of a single characteristic vector, such as the LTAGR, is that the distance calculation only requires a single distance evaluation, whereas the distance calculation between a feature such as a CEP vector requires comparison

between many individual test vectors and a reference codebook which also contains many vectors.

## 5.7 SUMMARY

This section summarizes the main findings of this chapter. Some of the results cannot be considered original, since they confirm results obtained by other researchers, but in the context of the comparative experimental evaluation presented in this chapter it is important to confirm, or disagree, with the observations of other researchers.

- Analysis of training data can be performed to evaluate whether features are likely to be independent.
- Intraspeaker and interspeaker distances calculated from the training data are less correlated for voiced speech than for the entire utterance.
- The binomial distribution is used to give the confidence interval for the speaker identification error rate and McNemar's test is invoked to compare whether results from two experiments are significantly different.
- The identification error rate of all vocal tract features decreases as the codebook size is increased.
- The best vocal tract features for speaker identification using the entire utterance are LPC and CEP with identification error rates of 0.5%. CEP is preferred because the construction of template codebooks and the distance computations are more straightforward.
- Voiced speech gives smaller speaker identification error rates than entire speech utterances.
- The LTAGR has a speaker identification error rate of 21%. The use of LTAGR descriptors and discriminant analysis does not improve the identification error rate over using the entire LTAGR. Weighting the LTAGR by the inverse of the pooled intraspeaker covariance matrix increases the identification error rate significantly. LTAGRs determined from pre-emphasized speech also have increased identification error rates.
- Of all the methods of combining CEP, PARCOR and LTAGR features to improve the identification error rate, it is found that the method (M4) that uses weightings related to the interspeaker and intraspeaker distances in the training data gives the best results. However, in a statistical sense, M4 (modified weighting) was not significantly better than either CEPV or M5b (presort using LTAGR).
- The speaker identification performance of the LTAGR remains relatively unaffected by noise compared with LTAS-VE, CEPV and PARCORV features.
- The speaker identification error rate of LTAGR features is sensitive to variations in the phase response of the speech transmission channel. The increase in identification error rate is greater than that of LTAS-V, CEPV and PARCORV features.
- Computation of the LTAGR is an order of magnitude more efficient than either CEP, LTAS-VE or PARCOR features.

## CHAPTER 6

---

### CONCLUSIONS AND SUGGESTIONS FOR FURTHER RESEARCH

This thesis presents a thorough investigation of the usefulness of a new glottal based feature for performing speaker identification. Section 6.1 describes conclusions reached from the experimental investigations and §6.2 suggests possible directions of future research.

#### 6.1 CONCLUSIONS

The main aim of this thesis is to investigate the usefulness of the long-term average glottal response (LTAGR) for speaker identification and to compare it against vocal tract features. Several other aspects of the LTAGR are also investigated, and for this reason the concluding remarks are divided into two sections. Section 6.1.1 draws conclusions about the LTAGR as a feature and §6.1.2 is concerned with the significance of the speaker identification results reported in Chapter 5.

##### 6.1.1 The long-term average glottal response

The long-term average glottal response, computed using the shift-and-add (SAA) algorithm, represents the average of a person's glottal excitation convolved with the average vocal tract response throughout the utterance (§2.8.2).

Section 2.8.2 states that a requirement of using the SAA algorithm to calculate the long-term average glottal response is that only voiced sections of speech are used. In §2.8.5 a modification to the standard SAA algorithm (as defined by Brieseman *et al.* (1987)) for speech signals is proposed that simplifies the voiced/unvoiced decision. It is shown in §2.8.5 that the voiced/unvoiced decision can be approximated by applying a threshold to the peak of each frame to test whether it is of large enough amplitude to be added to the SAA accumulator. This has considerable computational advantages over more complicated voiced/unvoiced decision methods and, as shown in Fig. 2.25, does not significantly affect the shape of the LTAGR. Furthermore, as explained in §4.3.1, a peak threshold based voiced/unvoiced decision makes the SAA algorithm amenable to real-time computation.

An evaluation of the similarities and differences between the spectrum of the LTAGR and long-term average spectrum is reported in §4.5.1.1. Comparisons of these features using four different speakers (2 males and 2 females) shows that the spectrum of the LTAGR of pre-emphasized speech is similar (see Fig. 4.16) to the LTAS of voiced, pre-emphasized speech. Visual comparison of the LTAGR spectrum and the long-term average spectrum of unemphasized voiced speech (depicted in Fig. 4.16) shows that the differences are greater than those of the pre-emphasized speech, but the patterns of 'peaks' and 'valleys' in the spectra are the same.

### 6.1.1.1 Analysis of the long-term average glottal responses belonging to a large database

In §4.3.3 the variation of the LTAGR with accent and gender of the speaker is examined. LTAGRs are computed from the utterances of 352 American English speakers from 8 different accent regions. A total of 21 descriptors are defined to represent the characteristics of each LTAGR (§4.3.2).

Factor analysis and discriminant analysis are performed on the LTAGR descriptors to examine the correlations amongst descriptors to see if any ‘grouping’ or clustering occurs, and to evaluate the usefulness of the LTAGR for accent and sex discrimination.

Factor analysis of the LTAGR descriptors reveals that the first factor, which accounts for 67.7% of the total variance, is strongly correlated with descriptors that represent overall shape of the LTAGR. The next two factors are correlated with descriptors that measure the number of peaks in the LTAGR. Scatter plots of the LTAGR observations, plotted with axes comprising combinations of the three largest factors, show that the descriptors for speakers with different accents and genders are all grouped in a central ‘clump’, independent of either accent or gender. This means that the factor axes are independent of speaker gender and accent, so variations other than gender and accent dominate variations in the LTAGR descriptors.

Discriminant analysis is performed on the data to examine whether the LTAGR is useful for determining a speaker’s accent or gender. Section 4.3.3.2 shows that the LTAGR descriptors are not at all accurate at determining which regional accent a person has, but are 95.7% accurate for determining a person’s sex. The factor analysis and discriminant analysis results differ because the discriminant analysis specifically weights those descriptors that assist the specified discrimination. One limitation of this work is that the same speech data is used to obtain the distance weighting matrix (for the descriptors) as is used to evaluate the discrimination performance. Although this lowers the significance of the results, the small number of gender misclassifications (14 out of 352) indicates that the LTAGR is potentially useful for determining a speaker’s gender. The LTAGR gives an error rate comparable to those reported in the literature for cepstral coefficients (see §4.3.3.2), but requires less computation.

### 6.1.2 Speaker identification performance

This section of the conclusion is concerned with the speaker identification experiments reported in Chapter 5. Five different features are compared for speaker identification. Three of the features, the cepstral coefficients (CEP), the linear prediction coefficients (LPC) and the partial correlation coefficients (PARCOR) record vocal tract characteristics. The other two features, the long-term average spectrum (LTAS) and the long-term average glottal response (LTAGR) record characteristics that are related to the glottal pulse.

The speaker templates and matching methods differ for the vocal tract features and the long-term features. The CEP, LPC and PARCOR speaker templates consist of vector quantization codebooks that are formed using the Linde, Buzo and Gray (1980) training method described in §2.7.4.3. Distances between a test utterance and a speaker’s template are determined by computing the total error that occurs when the test utterance is quantized using the speaker’s codebook. The LTAS and the LTAGR templates are formed from the average of the LTAS and LTAGR features taken across five training utterances. For the LTAS and LTAGR the distances between a test utterance and speaker’s template are determined using a Euclidean distance measure between feature vectors.

Section 4.1 describes the speech database that was recorded to provide utterances

suitable for performing speaker identification experiments. Fifteen utterances of the digits zero to nine were recorded from twenty speakers (sixteen males and four females). The first five utterances are used for training and the last ten utterances are used for evaluating identification performances.

#### 6.1.2.1 Analysis of templates formed from the training data

The training data is analyzed to examine the expected usefulness of the CEP, PARCOR, LTAGR and LTAS features. The usefulness of each feature is assessed by examining the interspeaker and intraspeaker distributions for each individual feature. The trend, for both PARCOR and CEP coefficients, as the codebook size increases is for the variation in the intraspeaker distance to decrease. This implies that the variation of the distance measured between a person and their own template is less for larger codebooks. However, the interspeaker distance, which measures the separation between speakers, is also important. The intraspeaker distance and interspeaker distance are most widely separated when only voiced frames from the speech signal are used for recognition, and this leads to the improved identification accuracies reported in §5.4. In addition, the correlation between the intraspeaker and interspeaker distance is reduced when only voiced frames are used in training the codebooks, which means that a person's utterances can be close to their own template without being close to templates belonging to other speakers.

The correlations of the intraspeaker distances belonging to different features is used to assess whether features record the same information about speakers. The correlation (Pearson's coefficient) between the intraspeaker distances of the LTAGR and CEP (with codebooks containing 16 codevectors) is 0.00, which indicates that these two features represent different aspects of a person's speech. The LTAGR is uncorrelated with the vocal tract characteristics, as recorded by the CEP codebooks, and there are two possible reasons for this. First, the LTAGR contains phase information about the speech and, second, the LTAGR is dominated by information about a person's glottal characteristics. The long-term average spectrum, which can also be considered to record glottal characteristics (§3.5.5), has a low correlation of 0.01 with respect to CEP. The features related to the glottal characteristics of the speech therefore represent information that is independent of that described by the vocal tract features.

#### 6.1.2.2 Evaluation of different speaker identification features

A series of speaker identification experiments that evaluate the advantages and disadvantages of the different features for speaker identification are reported. The types of experiment fall into three broad categories. They are, experiments on individual features, experiments on combinations of features, and experiments on noisy or distorted speech. The following paragraphs present the conclusions reached from these experiments.

Comparison between LTAGR, LTAS, PARCOR, LPC and CEP coefficients for speaker identification reveals that the PARCOR, LPC and CEP features perform better than the LTAGR and LTAS features. The LTAGR and LTAS give approximately 20% identification error rate, while the identification error rate for PARCOR, LPC and CEP coefficients is approximately 0.5% for codebooks containing 32, 64 or 128 codevectors.

The difference in identification error rate between using the entire utterance and voiced only speech is examined in §5.4.2. There is a significant advantage in using voiced speech (see Table 5.6) when codebooks contain between 2 and 8 codevectors. However, the difference in identification error rates is insignificant when the size of

the codebooks is 16 or larger. This is because codebooks with more codevectors have enough resolution to adequately describe the wider range of unvoiced sounds.

The correlations of the intraspeaker distances in §5.2.2 suggest that the LTAGR measures a voice characteristic that is not represented by the other features. Combinations of CEP, PARCOR and LTAGR features were therefore evaluated to see if identification performance could be improved. Five different methods of combining features were evaluated. Method 1 used the CEP, PARCOR and LTAGR features, with distance normalization being performed before the distance values from the three features were combined. Method 2 was the same except only the CEP and LTAGR features were used. Method 3 used CEP, PARCOR and LTAGR features, distance normalization and a weighting that corresponded to the intraspeaker distance. Method 4 used the same features as method 3, and a new method of weighting based on the interspeaker and intraspeaker distances in the training data. Method 5 used the LTAGR to presort the total population into a sub-population before identification was performed on the sub-population using the CEP. All of the above methods used the entire utterance for performing identification.

Method 4 gives a 1% improvement (to an identification error rate of 1.5%) over using CEPs by themselves for a codebook size of 16. This method also performs the best out of all the combined feature methods. Information recorded in the intraspeaker and interspeaker distributions within the training data is therefore useful for obtaining the appropriate weightings for combining distances calculated from separate features into a single distance measure. Computationally, method 4 has the drawback that it requires matching between both CEP and PARCOR codebooks. Method 4 is only better than CEP or PARCOR features alone when the codebook size is limited (see §5.4.4). In addition, if only voiced portions of the utterance are used, method 4 does not improve the error rate.

Method 5 uses the LTAGR to reduce the size of the speaker population before matching by the more accurate, but computationally intensive, CEP coefficients and codebooks. There is a slight decrease in accuracy, in the order of 1%, compared with CEP matching against the whole population and, on average, sorting with the LTAGR reduces the population to 0.56 of its original size.

In §5.6 the computation requirements of the features used for speaker identification are discussed under the headings of feature computation and identification requirements. The computational effort required to calculate the LTAGR is shown in §5.6 to be significantly less than that required for CEP, PARCOR or LTAS features. Matching between CEP and PARCOR test features and speaker templates is on a frame by frame basis which is more computationally intensive than either the LTAGR and LTAS (§5.6.2). However, this extra computation can be justified by the better speaker identification performance of the PARCOR and CEP features.

In summary, the various speaker identification results presented in §5.4 indicate that the LTAGR is not as accurate as CEP, LPC and PARCOR coefficients for performing speaker identification. Combining the LTAGR with CEP and PARCOR features does not produce a significant reduction in the speaker identification error rate.

### **6.1.2.3 Sensitivity of features to noise and distortion**

The effects of various types of noise and distortion on the speaker identification performance of the various features is reported in §5.5. The following conclusions are drawn from the identification experiments.

The LTAGR is insensitive to the effects of Gaussian noise and speech correlated noise compared with the CEP, PARCOR and LTAS features. This is because averaging occurs in the SAA algorithm that removes the effects of noise.

The effects of variations in the frequency responses of telephone headsets was evaluated by applying gain or attenuation in the 2.0 to 2.7 kHz region. The LTAGR, CEP and PARCOR features are insensitive to attenuation of approximately 10 dB in the frequency response of the speech transmission channel in the 2 to 2.7 kHz region (see §5.5.2.2). However, the speaker identification error rate of all the features increases to more than 50% when there is attenuation in the low frequency (0-2 kHz) region. This implies that any variations in the low frequency response of a channel will have a significant effect on the accuracy of speaker identification.

The LTAGR is shown to be sensitive to variations in the phase response of the speech transmission channel. The identification error rate increases to 80% or more for three of the four phase distortions tested. In contrast, the error rate of the CEPV and PARCERV features increases by approximately 5% for the various phase distortions applied and the error rate for the LTAS-V increases by less than 20%. This insensitivity of the CEPV, PARCERV and LTAS-V to phase distortion is expected since they do not record any phase information about the speech.

In summary, from the experimental findings reported in §5, it is apparent that the LTAGR cannot be justified as a useful feature for speaker identification unless the unusual situation of a noisy channel with reasonably consistent phase response occurs. In all other situations vocal tract features yield more accurate speaker identification.

## 6.2 SUGGESTIONS FOR FURTHER RESEARCH

The research reported here points to several areas of further investigation. These are grouped into those that use the LTAGR to assess a person's voice quality, as described in §6.2.1, and those relating to speaker identification, as discussed in §6.2.2. An alternative VQ structure for performing speaker identification is suggested in §6.2.2.5

### 6.2.1 The long-term average glottal response

In §4.4 it is suggested that the LTAGR can be used to extract similar information from the speech as might be expected from the LTAS. Furthermore, §4.4.3 explains that the LTAS is considered to be useful in clinical applications for monitoring changes in a client's voice. It would be interesting to compare the capabilities of the LTAS and the LTAGR in a clinical situation. Since the LTAGR is much simpler to compute than the LTAS (see §5.6), it could have considerable practical advantages provided that the same clinical information could be extracted.

It is also noted in §5.4.2.1 that the speaker information in the LTAS is affected by the shape of the window applied to the speech. This implies that some of the information characterizing the speaker is contained in the fine detail of the LTAS. Therefore, to characterize a person's voice for clinical assessment, or speaker identification, it is important to represent the fine structure of the LTAS. Much of the reported research into the clinical use of the LTAS makes use of coarse descriptors (Kitzing, 1986; Löfqvist, 1986) that discard the fine structure information. Further investigation into the information recorded in the fine structure of the LTAS is required to ascertain whether or not these coarse descriptors are actually best.

Section 5.4.4 introduced a speaker identification method that uses the LTAGR as a presort to lower the search population before a more accurate, computationally intensive, feature is used. In this role the LTAGR is being used to determine the general characteristics of a person's voice and to associate the voice with a group of people (voices) that have similar characteristics. This type of association could be useful in word recognition applications as well as speaker recognition applications. The idea

would be to have the speaker say a phonetically balanced phrase from which the LTAGR is extracted. The LTAGR could then be used to associate the speaker's voice with a group containing the same 'voice type', and word recognition could proceed using templates from the selected group. This type of system would be useful if people were to use a word recognition system for an extended period. An example of such a system would be a computerized drawing package, where a session of many hours duration could be started by users uttering a phonetically balanced phrase that 'tunes' the recognition system to their voice.

### 6.2.2 Speaker identification

There are many possible extensions to the speaker identification experiments reported in Chapter 5. The most obvious ones, such as the incorporation of time registration in the matching, and testing the effects of shorter phrases have been examined elsewhere in the literature (Soong and Rosenberg, 1988). However, a number of other research avenues remain.

#### 6.2.2.1 Methods of improving the accuracy of results

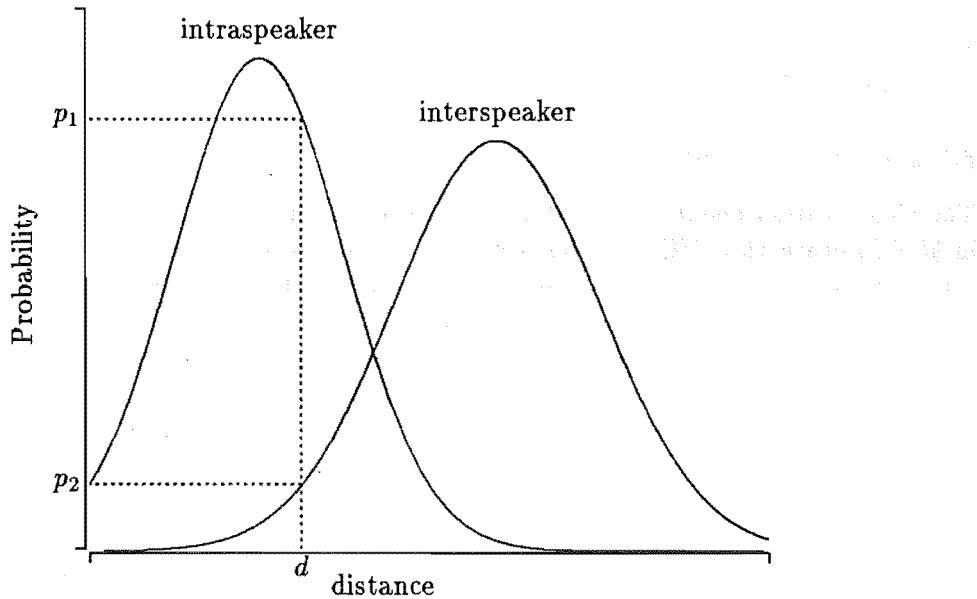
A drawback of the results reported in §5.4 is that it is not always possible to draw statistically significant conclusions between experiments that use vocal tract features. If further experiments were undertaken, the significance of the identification results could be improved in several ways. First, the number of speakers used in each identification experiment could be increased, thereby improving the accuracy of each experimental result. Second, the identification systems could be 'stressed' in some way, so as to increase differences in identification error rates. One method of stressing the systems is to reduce the length of the speech sample that is used for testing the identification system. In the work reported in this thesis it was not considered sensible to shorten the speech samples, because the aim was primarily to compare between the LTAGR and vocal tract features, and the LTAGR requires a long utterance to form a stable estimate. However, if further experiments were performed between vocal tract features alone, it would be quite feasible to use only portions of complete utterances.

#### 6.2.2.2 Weighting individual samples in the LTAGR

Another area requiring further research is the method of distance calculation between LTAGRs. Section 5.4.2.2 mentions that the application of intraspeaker weights to the LTAGR distance measure does not improve the accuracy of speaker identification. It would be useful to examine whether the weights applied to each sample for the purposes of distance calculation could be improved. The method of weighting using the inverse of the intraspeaker distance is not optimal for the reasons outlined in §5.4.2.2. A potentially better weighting would be one that takes into account the interspeaker as well as the intraspeaker distance. A possible form of such a weighting, specified for each sample  $i$  of the LTAGR, is

$$k_i = \frac{\overline{\text{interspeaker}}^i - \overline{\text{intraspeaker}}^i}{\sigma_{\text{inter}}^i + \sigma_{\text{intra}}^i}, \quad (6.1)$$

where  $\sigma_{\text{inter}}$  is the standard deviation of the interspeaker distance and  $\sigma_{\text{intra}}$  is the standard deviation of the intraspeaker distance as computed from all the training LTAGRs. Note that (6.1) has the same form as the weighting used for method 4 for combining contributions from various features into a single measure. Method 4 gave better identification results than the other methods of combining features, which implies that (6.1) might also give improved speaker identification results.



**Figure 6.1.** Intraspeaker and interspeaker probability distributions for speaker 1.

#### 6.2.2.3 Further evaluation of the LTAGR for presorting the speaker population

In §5.4.4 it was suggested that the LTAGR could be used to presort speakers so as to reduce the size of the population. However, the experiments reported in §5.6 require extension before conclusions can be drawn about whether presorting using the LTAGR is advantageous. In particular, it is important to ascertain whether the LTAGR can be used to select a sub-population that is significantly smaller (say a quarter) than the total population for large numbers of speakers. Such a reduction in the population would allow considerable computational savings when performing speaker identification.

#### 6.2.2.4 A method of assessing identification accuracy

Section 5.2.2 introduces the concept that the expected accuracy of a particular descriptor could be predicted from the intraspeaker and interspeaker distances of the training utterances. Noda (1989) describes a method for obtaining the probability that a speaker verification result is correct, given the distance  $d$ . In speaker identification, particularly for use in a court of law, it would be useful to determine a measure of confidence that a correct identification has been made. The confidence measure would be based on the distance that the test utterance is from the speaker's template and the spread of the speaker's training utterances. The confidence in a particular speaker identification result is related to how well the intraspeaker and interspeaker distributions are separated for that particular individual. The distribution of the intraspeaker and interspeaker distances can be modelled as a Gaussian distribution using the means and variances computed from the intraspeaker and interspeaker distances. If the intraspeaker and interspeaker distributions for speaker 1 are as depicted in Fig. 6.1, the distance,  $d$ , recorded between the test utterance and template 1 has a probability of occurrence  $p_1$  while the probability that this distance can occur when the utterance belongs to another speaker is given by  $p_2$ . From these two probabilities it should be possible to form an estimate of the confidence of the identification. One possible confidence measure would be  $p_1/p_2$ . Experiments could then be performed to examine how the confidence figure corresponds with the accuracy of actual speaker identification experiments.

Of course, it may also occasionally be important to determine the probability that

a particular test utterance does *not* belong to a specific individual. A modified version of the above scheme could be applied in that situation.

### 6.2.2.5 An alternative VQ structure

The identification results reported in §5.4 and the computation requirements presented in §5.6 indicate that VQ features are accurate for performing identification, but are computationally expensive. This section presents an alternative to the usual single VQ codebook per speaker that could considerably speed up speaker identification.

The essence of the new method is that information that is usually recorded in individual codebooks is instead recorded in a single large tree structured codebook. As explained in §2.7.4.4, a tree-structured codebook is efficient to search since every two distance computations halve the search space. For each centroid in the codebook it is necessary to record how well the centroid characterizes the speech of each of the speakers. This is explained further in the following paragraphs which are concerned with computing the codebook and the way in which such a codebook is used to perform speaker identification. The suggested algorithm for computing the codebook follows the LBG algorithm presented in §2.7.4.3.

It is useful for the purposes of this discussion to introduce a measure of how similar two frames of speech are to each other. This is called the *similarity* and an example of such a measure, expressed using terminology introduced in §2.7.3, is

$$S(|X(e^{j\theta})|^2, |\sigma/A(e^{j\theta})|^2) = \frac{\mathbf{a}^x \mathbf{R}^x \mathbf{a}^{xT}}{\mathbf{a} \mathbf{R}^x \mathbf{a}^T} \quad (6.2)$$

where  $\mathbf{a} \mathbf{R}^x \mathbf{a}^T$  is always greater than or equal to  $\mathbf{a}^x \mathbf{R}^x \mathbf{a}^{xT}$  (see Gray and Markel (1976)) and approaches  $\mathbf{a}^x \mathbf{R}^x \mathbf{a}^{xT}$  when the spectrum  $|\sigma/A(e^{j\theta})|^2$  is close to the spectrum  $|X(e^{j\theta})|^2$ . The reasons for using a similarity measure instead of a distance measure are explained in more detail later.

The iterative procedure for computing the codebook uses training vectors  $\mathbf{x}[pk]$  to model  $|X(e^{j\theta})|^2$ , where  $p$  is the speaker number and  $k$  is the number of the speech frame. Similarly,  $|\sigma/A(e^{j\theta})|^2$  is represented by the centroid vector  $\mathbf{y}[j]$ , where  $j$  is the centroid number. The similarity measure between a vector  $\mathbf{x}$  and a centroid  $\mathbf{y}$  can therefore be expressed concisely as  $S(\mathbf{x}, \mathbf{y})$ . The first seven steps of the following VQ training algorithm are the same as Algorithm 2.2, except that the training sequence is comprised of utterances from all the speakers. The notation used is introduced in §2.7.4.

#### Algorithm 6.1

Step 1: Initialisation: Fix the largest number of codevectors desired to be  $2^R$ , where  $R$  is an integer. Set  $M$  to the number of people and  $K_k$  to the vectors in the training sequence belong to speaker  $k$  and  $L$ , the number of codevectors, to 1. Define  $C_0 = \{\mathbf{x}[pk]; p = 0, \dots, M - 1; k = 0, \dots, K_k - 1\}$  and  $\mathbf{Y}(1) = \text{cent}C_0$ , the centroid of the entire training sequence.

Step 2: Splitting: Given  $\mathbf{Y}(L) = \{\mathbf{y}[j]; j = 0, \dots, L\}$ , split each codebook vector into  $\mathbf{y}_j + \epsilon$  and  $\mathbf{y}_j - \epsilon$ . Set  $\mathbf{Y}_m(2L) = \{\mathbf{y}_j + \epsilon, \mathbf{y}_j - \epsilon, j = 1, \dots, L\}$  and replace  $L$  by  $2L$ .

Step 3: Reset variables: Set  $m = 0$  and  $D_{-1} = \infty$ .

Step 4: Partitioning: Find the optimum partition for the codebook  $\mathbf{Y}_m(L), \mathcal{P}(\mathbf{Y}_m(L))$  using (2.93). Compute the resulting distortion

$$D_m = D(\{\mathbf{Y}_m(L), \mathcal{P}(\mathbf{Y}_m(L))\}). \quad (6.3)$$

Step 5: Termination Test: If  $(D_{(m-1)} - D_m)/D_m \leq \epsilon = 0.005$ , go to step 7. Otherwise continue.

Step 6: Update Codebook: find the next codebook  $\mathbf{Y}_{m+1}(L) = \text{cent}\mathcal{P}(\mathbf{Y}_m(L))$ , the centroids of the partitions for the codebook  $\mathbf{Y}_m(L)$ . Replace  $m$  by  $m + 1$  and go to step 4.

Step 7: Final Rate Test: Set  $\mathbf{Y}(L) = \mathbf{Y}_m(L)$ . If  $L < 2^R$  go to step 2 otherwise continue.

Step 8: Compute Similarities: Find the average similarity between each centroid and the speaker's vectors that fall in the cell. First, let  $j$  be the centroid number,  $p$  the speaker number and set  $S[pj] = 0$  for all  $p$  and  $j$ . The average similarity between speaker  $p$  and centroid  $j$  is given by

$$S[pj] = \frac{\sum_{k:\mathbf{x}[pk] \in C_j} S(\mathbf{x}[pk], \mathbf{y}[j])}{\|\mathbf{x}[pk] \in C_j; k = 0, \dots, K_k - 1\|} \text{ for all } p \text{ and } j \quad (6.4)$$

where  $\|\mathbf{x}[pk] \in C_j; k = 0, \dots, K_k - 1\|$  is the number of vectors belonging to speaker  $p$  in cell  $C_j$ .

Halt with the final quantizer  $\mathbf{Y}_m(L)$ , where codebook tree branches are specified by  $L = 2, \dots, 2^R$  and the similarity matrix  $S[pj]$ .

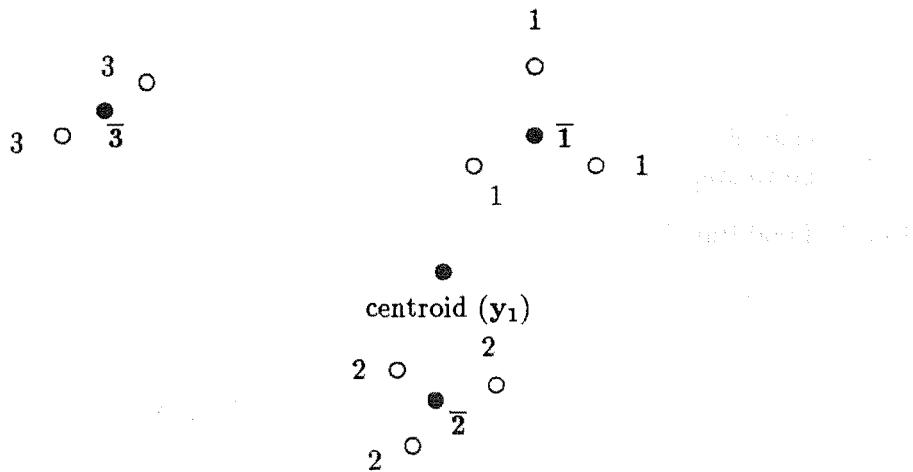
A particular cell,  $C_j$ , may not contain any vectors from a particular speaker,  $p$ , so  $\sum_{k:\mathbf{x}[pk] \in C_j} S(\mathbf{x}[pk], \mathbf{y}[j])$  will be zero and the similarity,  $S[pj]$ , is 0. This situation is the main reason for suggesting that a similarity score be used rather than a distance measure for measuring how far vectors are from the centroid. It would be difficult to determine a sensible distance to record when there are no vectors from a particular speaker in a cell. Obviously the distance should be large, but it is not clear how to select such a number. The similarity measure resolves this by allowing a similarity of 0 to be recorded.

An example of the vectors in a particular cell is depicted in Fig. 6.2, and for the purposes of this discussion the similarity between vectors can be considered to be inversely related to the distance between them. The cell depicted in Fig. 6.2 does not aid the discrimination between speakers 1 and 2 because the similarities between the speaker centroids and  $\mathbf{y}_1$  are approximately equal. However, speaker 4, who does not have any vectors within the cell, would have a similarity measure of 0 and such a variation in similarities would assist the discrimination of speaker 4 from the other three speakers. For a large single codebook to be useful, there must be enough resolution to ensure that for each speaker in the population there are several centroids contained within the codebook (vectors  $\mathbf{Y}_m(L)$ ) that are more similar to that particular speaker's speech than any other speaker's speech. There must therefore be at least as many codevectors as speakers, and probably round eight codevectors per speaker would be a good number to start experimenting with.

Identification uses a similarity score matrix to accumulate the similarity scores as test vectors are matched against the codebook. Given a test sequence of vectors  $\mathbf{w}[i]$  ( $0 < i \leq K - 1$ ) and a similarity score matrix  $SS[p]$ , initialize  $SS[p]$  by setting  $SS[p] = 0$  for all  $p$ . Compute  $SS[p]$  using

$$\left. \begin{aligned} m &= \arg \min_j d(\mathbf{w}[i], \mathbf{y}[j]) \\ SS[p] &= SS[p] + S[pm] \end{aligned} \right\} \text{ for } 1 < i \leq K - 1 \quad (6.5)$$

The person,  $p$ , with the highest similarity score is the identified speaker.



**Figure 6.2.** A stylized representation of a centroid,  $y_1$ , and the vectors from four speakers that fall in the cell associated with the centroid. The circles labelled 1 represent the positions of vectors obtained from speaker 1, and so on. Note that there are no circles labelled 4, because no vectors from speaker 4 occur in the example cell.

The result of evaluating  $\arg \min_j d(\mathbf{w}[i], \mathbf{y}[j])$  is a centroid number. However, the similarity used for identification has been computed as part of the training procedure and the position of the test vector  $\mathbf{w}[k]$  within the cell is lost. For the example depicted in Fig. 6.2, a test vector that is ‘close’ to the vectors belonging to speaker 3 is assigned to be more similar to speaker’s 1 and 2 than speaker 3. This would contribute to an incorrect identification. This situation can be avoided if enough centroids are used to ensure that the every speaker has several cells that are representative of their commonly occurring sounds, and in which they are most similar to the centroid.

Experimental verification is required to see whether the gain in computational efficiency using this method would be offset by a significant decrease in the identification accuracy. It would also be important to determine the optimum number of codevectors to use. One drawback of the above method is that the VQ codebook would most likely require recomputation every time a speaker was added to the population.

#### 6.2.2.6 Future trends

Ideally speaker identification would be performed in conjunction with word recognition. If this were possible, systems could automatically reconfigure themselves for each user by performing speaker recognition while responding to voice commands. To date there has been little experimentation with this sort of combined system, but in theory there is no reason why it could not be implemented.

Until computers are able to listen and respond to speech in much the same way as humans, spoken interaction with computers is going to remain somewhat of a novelty. Such interaction will only be achieved after much research, investigation and experimentation, and there remains a considerable way to go.

---

## REFERENCES

- ABRAMOWITZ, M. and STEGUN, I. (1965), *Handbook of mathematical functions*, Dover Publications, Inc., New York.
- ANANTHAPADMANABHA, T.V. and FANT, G. (1982), 'Calculation of the true glottal flow and its components', *Speech Communication*, Vol. 1, December, pp. 167-184.
- ANANTHAPADMANABHA, T.V. and YEGNANARAYANA, B. (1979), 'Epoch extraction from linear prediction residual for identification of closed glottis interval', *IEEE Transactions in Acoustics, Speech and Signal Processing*, Vol. ASSP-27, No. 4, August, pp. 309-319.
- ANTEX ELECTRONICS CORPORATION (1990), *Series 2/Model SX10 Digital Audio Processor Datasheet*, Antex Electronics Corporation, 16100 South Figueroa St, Gardena, Calif. 90248.
- ATAL, B.S. (1972), 'Automatic speaker recognition based on pitch contours', *Journal of the Acoustical Society of America*, Vol. 52, No. 6 (Part 2), pp. 1687-1697.
- ATAL, B.S. (1974), 'Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification', *Journal of the Acoustical Society of America*, Vol. 55, No. 6, June, pp. 1304-1312.
- ATAL, B.S. (1985), 'Linear predictive coding of speech', In FALLSIDE, F. and WOODS, W.A. (Eds.), *Computer Speech Processing*, Prentice Hall, New Jersey, Chap. 4.
- ATAL, B.S. and HANAUER, S.L. (1971), 'Speech analysis and synthesis by linear prediction', *Journal of the Acoustical Society of America*, Vol. 50, No. 2 (Part 2), pp. 637-655.
- ATAL, B.S. and RABINER, L.R. (1976), 'A pattern recognition approach to voiced-unvoiced-silence classification with applications to speech recognition', *IEEE Transactions in Acoustics, Speech and Signal Processing*, Vol. ASSP-24, No. 3, June, pp. 201-212.
- ATAL, B.S. and RABINER, L.R. (1986), 'Speech research directions', *AT & T Technical Journal*, Vol. 65, No. 5, September, pp. 75-88.
- ATTILI, J.B., SAVIC, M. and CAMPBELL, JR., J.P. (1988), 'A TMS32020-based real time, text-independent, automatic speaker verification system', In *International Conference on Acoustics, Speech, and Signal Processing*, IEEE, pp. 599-602.
- BARNWELL, T.P. (1980), 'Windowless techniques for LPC analysis', *IEEE Transactions in Acoustics, Speech and Signal Processing*, Vol. ASSP-28, No. 4, August, pp. 421-427.

- BATES, R.H.T. (1982), 'Astronomical speckle imaging', *Physics Reports*, Vol. 90, No. 4, October, pp. 203-297.
- BATES, R.H.T. and CADY, F.M. (1980), 'Towards true imaging by wideband speckle interferometry', *Optics Communications*, Vol. 32, No. 5, March, pp. 365-369.
- BATES, R.H.T. and McDONNELL, M.J. (1986), *Image Restoration and Reconstruction*, The Oxford Engineering Science series, Oxford University Press, Oxford.
- BENDIKSEN, A. and STEIGLITZ, K. (1990), 'Neural networks for voiced/unvoiced speech classification', In *International Conference on Acoustics, Speech, and Signal Processing*, IEEE, pp. 521-524.
- BERGER, T. (1971), *Rate Distortion Theory*, Prentice-Hall, Inc., Englewood Cliffs, New Jersey.
- BERGLAND, G.D. (1969), 'A guided tour of the fast Fourier transform', *IEEE Spectrum*, July, pp. 41-52.
- BIRNBAUM, M., COHEN, L.A. and WELSH, F.X. (1986), 'A voice password system for access security', *AT & T Technical Journal*, Vol. 65, No. 5, September, pp. 68-74.
- BLYTH, C.B. (1986), 'Approximate Binomial confidence limits', *Journal of the American Statistical Association*, Vol. 81, No. 395, September, pp. 843-855.
- BOGNER, R.E. (1981), 'On talker verification via orthogonal parameters', *IEEE Transactions in Acoustics, Speech and Signal Processing*, Vol. ASSP-29, No. 1, February, pp. 1-12.
- BOLT, R.H., COOPER, F.S., DAVID, JR., E.E., DENES, P.B., PICKETT, J.M. and STEVENS, K.N. (1970), 'Identification of a speaker by speech spectrograms', *Journal of the Acoustical Society of America*, Vol. 47, No. 2 (Part 2), pp. 597-612.
- BONDER, L.J. (1983), 'The  $n$ -tube formula and some of its consequences', *ACUSTICA*, Vol. 52, pp. 216-226.
- BOVES, L. (1984), *The phonetic basis of perceptual ratings of running speech*, Foris publication, Dordrecht, Holland.
- BRACEWELL, R.N. (1986), *The Fourier Transform and its Applications*, Circuits and systems, McGraw-Hill Book Company, New York, 2nd ed.
- BRICKER, P.D. and PRUZANSKY, S. (1966), 'Effects of stimulus and duration on talker identification', *Journal of the Acoustical Society of America*, Vol. 40, No. 6, pp. 1441-1449.
- BRICKER, P.D., GNANADESIKAN, R., MATHEWS, M.V., PRUZANSKY, S., TUKEY, P.A., WACHTER, K.W. and WARNER, J.L. (1971), 'Statistical techniques for talker identification', *Bell Systems Technical Journal*, Vol. 50, No. 4, April, pp. 1427-1454.
- BRIESEMAN, N.P. (1984), *A New Algorithm For Musical Pitch Estimation*, Master's thesis, Electrical and Electronic Engineering Department, University of Canterbury, New Zealand.

- BRIESEMAN, N.P., THORPE, C.W. and BATES, R.H.T. (1987), 'Nontactile estimation of glottal excitation characteristics of voiced speech', *IEE Proceedings A*, Vol. 134, No. 10, December, pp. 807-813.
- BRIESEMAN, N.P., THORPE, C.W., ELDER, A.G. and ROLLS, A. (1989), *Sigproc Users Guide and Reference Manual*, Dept. of Elect. Eng., University of Canterbury, Christchurch, N.Z.
- BRIGHAM, E.O. (1974), *The fast Fourier transform*, Prentice-Hall Inc, Englewood Cliffs, New Jersey.
- BURTON, D.K. (1987), 'Text-dependent verification using vector quantization source coding', *IEEE Transactions in Acoustics, Speech and Signal Processing*, Vol. ASSP-35, No. 2, February, pp. 133-143.
- BURTON, D.K., SHORE, J. and BUCK, J. (1985), 'Isolated word speech recognition using multisection vector quantization codebooks', *IEEE Transactions in Acoustics, Speech and Signal Processing*, Vol. ASSP-33, No. 4, August, pp. 837-849.
- BUZO, A., GRAY, A.H., GRAY, R.M. and MARKEL, J. (1980), 'Speech coding based upon vector quantization', *IEEE Transactions in Acoustics, Speech and Signal Processing*, Vol. ASSP-28, No. 5, October, pp. 562-574.
- CATFORD, J.C. (1977), *Fundamental problems in phonetics*, Edinburgh University Press, Edinburgh.
- CHANDRA, S. and LIN, W.C. (1974), 'Experimental comparison between stationary and nonstationary formulations of linear prediction applied to voiced speech analysis', *IEEE Transactions in Acoustics, Speech and Signal Processing*, Vol. ASSP-22, No. 6, December, pp. 403-415.
- CHENG, Y.M. and O'SHAUGHNESSY, D. (1989), 'Automatic and reliable estimation of glottal closure instant and period', *IEEE Transactions in Acoustics, Speech and Signal Processing*, Vol. ASSP-37, No. 12, December, pp. 1805-1815.
- CHILDERS, D.G., WU, K., BAE, K.S. and HICKS, D.M. (1988), 'Automatic recognition of gender by voice', In *International Conference on Acoustics, Speech, and Signal Processing*, IEEE, pp. 603-606.
- CLARK, T.M., KENNEDY, W.K. and BATES, R.H.T. (1990), 'Towards a real time computer word recognition system using the TMS32030', In *Proc. NELCON, (New Zealand National Electronics Conference)*, pp. 295-303.
- COLEMAN, R.O. (1973), 'Speaker identification in the absence of inter-subject differences in glottal source characteristics', *Journal of the Acoustical Society of America*, Vol. 53, No. 6, pp. 1741-1743.
- COMREY, A.L. (1973), *A first course in Factor Analysis*, Academic Press Inc., New York.
- COOLEY, W.W. and LOHNES, P.R. (1971), *Multivariate data analysis*, John Wiley & Sons Inc., New York.
- CRUTTENDEN, A. (1986), *Intonation*, Cambridge Textbooks in Linguistics, Cambridge University Press, Cambridge.

- DARPA (1988), 'Getting started with the DARPA TIMIT CD-ROM', Distributed with the DARPA TIMIT Acoustic Phonetic Continous Speech Database. This documentation is a compilation of papers.
- DAVEY, B.L.K. (1989), *Advances in blind deconvolution*, PhD thesis, Electrical and Electronic Engineering Department, University of Canterbury, Christchurch, New Zealand, May.
- DAVEY, B.L.K. and THORPE, C.W. (1987), 'Image and signal reconstruction by shift-and-add', In *IPENZ conference proceedings*, Institution of Professional Engineers of New Zealand, Christchurch, May, pp. 147-157.
- DE SOUZA, P. (1983), 'A statistical approach to the design of an adaptive self-normalizing silence detector', *IEEE Transactions in Acoustics, Speech and Signal Processing*, Vol. 31, No. 3, June, pp. 678-684.
- DODDINGTON, G.R. (1985), 'Speaker recognition - identifying people by their voices', *Proceedings of the IEEE*, Vol. 73, No. 11, November, pp. 1651-1665.
- DUDLEY, H. (1939), 'Remaking speech', *Journal of the Acoustical Society of America*, Vol. 11, No. 2, October, pp. 169-177.
- DUNCAN, G. and JACK, M.A. (1988), 'Formant estimation algorithm based on pole frequency focussing offering improved noise tolerance and feature resolution', *IEE Proceedings F*, Vol. 135, No. 1, February, pp. 18-32.
- DUNTEMAN, G.H. (1984), *Introduction to multivariate analysis*, SAGE Publications, London.
- EDWARDS, A.L. (1948), 'Note on the "correction for continuity" in testing the significance of the difference between correlated proportions', *Psychometrika*, Vol. 13, No. 3, September, pp. 185-187.
- EDWARDS, M.L. and SHRIBERG, L.D. (1983), *Phonology: Applications in Communicative Disorders*, College-Hill Press, San Diego.
- ELDER, A.G., BATES, R.H.T., BRIESEMAN, N.P., CLARK, T.M., FRIGHT, W.R., GARDEN, K.L., KENNEDY, W.K., SQUIRES, P.L., TURNER, S.G. and THORPE, C.W. (1987), 'Real time speech therapy aid', *Proc. NELCON, (New Zealand National Electronics Conference)*, Vol. 24, pp. 115-118.
- ENDRES, W., BAMMACH, W. and FLÖSSER, G. (1971), 'Voice spectrograms as a function of age, voice disguise, and voice imitation', *Journal of the Acoustical Society of America*, Vol. 49, No. 6 (Part 2), pp. 1842-1848.
- ETON (1974), *Eton four-figure mathematical and statistical tables*, Eton Press Ltd., Box 8203, Christchurch, New Zealand.
- FALLSIDE, F. (1985), 'Frequency-domain analysis of speech', In FALLSIDE, F. and WOODS, W.A. (Eds.), *Computer Speech Processing*, Prentice Hall, New Jersey, Chap. 3.
- FALLSIDE, F. and WOODS, W.A. (Eds.) (1985), *Computer Speech Processing*, Prentice Hall, New Jersey.
- FANT, G. (1973), *Speech Sounds and Features*, Current Studies in Linguistics, MIT Press, Massachusetts.

- FLANAGAN, J.L. (1972), *Speech analysis, synthesis and perception*, Springer-Verlag, Berlin, 2nd ed.
- FLANAGAN, J.L. and CHERRY, L. (1969), 'Excitation of vocal-tract synthesizers', *Journal of the Acoustical Society of America*, Vol. 45, No. 3, March, pp. 764-769.
- FRY, D.B. (1979), *The physics of speech*, Cambridge textbooks in Linguistics, Cambridge University Press, Cambridge.
- FUJIMURA, O. (1968), 'An approximation to voice aperiodicity', *IEEE Transactions on Audio and Electroacoustics*, Vol. 16, No. 1, March, pp. 68-72.
- FURUI, S. (1974), 'An analysis of long-term variation of feature parameters of speech and its application to talker recognition', *Electronics and Communications in Japan*, Vol. 57-A, No. 12, pp. 34-42.
- FURUI, S. (1981), 'Comparison of speaker recognition methods using statistical features and dynamic features', *IEEE Transactions in Acoustics, Speech and Signal Processing*, Vol. ASSP-29, No. 3, June, pp. 342-350.
- FURUI, S. and ITAKURA, F. (1973), 'Talker recognition by statistical features of speech sounds', *Electronics and Communications in Japan*, Vol. 56-A, No. 11, pp. 62-71.
- FUSSELL, J.W. (1991), 'Automatic sex identification from short segments of speech', In *International Conference on Acoustics, Speech, and Signal Processing*, IEEE, pp. 409-412.
- GALLAGER, R.G. (1968), *Information theory and reliable communication*, John Wiley and Sons, Inc, New York.
- GILLICK, L. and COX, S.J. (1989), 'Some statistical issues in the comparison of speech recognition algorithms', In *International Conference on Acoustics, Speech, and Signal Processing*, IEEE, pp. 532-535.
- GLENN, J.W. and KLEINER, N. (1968), 'Speaker identification based on nasal phonation', *Journal of the Acoustical Society of America*, Vol. 43, No. 2, pp. 368-372.
- GNANADESIKAN, R. (1977), *Methods for statistical data analysis of multivariate observations*, John Wiley & Sons, New York.
- GOLD, B. and RABINER, L.R. (1969), 'Parallel processing techniques for estimating pitch periods of speech in the time domain', *Journal of the Acoustical Society of America*, Vol. 46, No. 2 (Part 2), pp. 442-448.
- GORSLUCH, R.L. (1983), *Factor Analysis*, Lawrence Erlbaum Associates, Hillsdale, New Jersey.
- GRAY, R.M. (1984), 'Vector quantisation', *IEEE Acoustics, Speech and Signal Processing Society Magazine*, April, pp. 4-29.
- GRAY, R.M., BUZO, A., GRAY, JR., A.H. and MATSUYAMA, Y. (1980), 'Distortion measures for speech processing', *IEEE Transactions in Acoustics, Speech and Signal Processing*, Vol. ASSP-28, No. 4, August, pp. 367-376.
- GRAY, JR., A.H. and MARKEL, J.D. (1976), 'Distance measures for speech processing', *IEEE Transactions in Acoustics, Speech and Signal Processing*, Vol. ASSP-20, No. 5, October, pp. 380-391.

- HARDCastle, W.J. (1976), *Physiology of speech production*, Academic Press, London.
- HARRIS, F.J. (1978), 'On the use of windows for harmonic analysis with the discrete Fourier transform', *Proceedings of the IEEE*, Vol. 66, No. 1, pp. 51-83.
- HOLLIEN, H. and MAJEWSKI, W. (1977), 'Speaker identification by long-term spectra under normal and distorted speech conditions', *Journal of the Acoustical Society of America*, Vol. 62, No. 4, October, pp. 975-980.
- HOLLIEN, H., MAJEWSKI, W. and DOHERTY, E.T. (1982), 'Perceptual identification of voices under normal, stress and disguise speaking conditions', *Journal of Phonetics*, Vol. 10, pp. 139-148.
- HUFFMAN, D.A. (1973), 'A method for the construction of minimum-redundancy codes', In SLEPIAN, D. (Ed.), *Key papers in the development of information theory*, IEEE Press, New York, pp. 47-50. Also in, Proc. IRE, vol. 40, pp. 1098-1101, Sept. 1952.
- IEEE (1969), 'Recommended practice for speech quality measurements', *IEEE Transactions on Audio and Electroacoustics*, Vol. 17, No. 3, September, pp. 225-246.
- ISHIZAKA, K. and FLANAGAN, J.L. (1972), 'Synthesis of voiced sounds from a two-mass model of the vocal cords.', *Bell Systems Technical Journal*, Vol. 51, No. 6, July-August, pp. 1233-1268.
- ITAKURA, F. (1975), 'Minimum production residual principle applied to speech recognition', *IEEE Transactions in Acoustics, Speech and Signal Processing*, Vol. 23, No. 1, February, pp. 67-72.
- ITAKURA, F. and SAITO, S. (1968), 'Analysis synthesis telephony based in the maximum likelihood method', In *The 6th International Conference on Acoustics*, Tokyo, Japan, August 21-28, pp. C17-C20.
- ITAKURA, F. and SAITO, S. (1973), 'On the optimum quantization of feature parameters in the PARCOR speech synthesiser', In FLANAGAN, J.L. and RABINER, L.R. (Eds.), *Speech Synthesis*, Dowden, Hutchinson & Ross, Inc., Stroudsburg, Pennsylvania, pp. 301-304. From Conf. Speech Commun. Process., 1972, p434-437.
- JAYANT, N.S. and NOLL, P. (1984), *Digital coding of waveforms, principles and applications to speech and video*, Prentice-Hall, Inc., Englewood Cliffs, New Jersey.
- JOHNSON, N.L. and KOTZ, S. (1969), *Discrete distributions*, Distributions in Statistics, Houghton Mifflin Co., Boston.
- JUANG, B.H. (1984), 'On the hidden markov model and dynamic time warping for speech recognition - a unified view', *AT & T Technical Journal*, Vol. 63, No. 7, September, pp. 1213-1243.
- JUANG, B.H., WONG, D.Y. and GRAY, JR., A.H. (1982), 'Distortion performance of vector quantization for LPC voice coding', *IEEE Transactions in Acoustics, Speech and Signal Processing*, Vol. ASSP-30, No. 2, April, pp. 294-303.
- JUNQUA, J.C. and WAKITA, H. (1989), 'A comparative study of cepstral lifters and distance measures for all pole models of speech in noise', In *International Conference on Acoustics, Speech, and Signal Processing*, pp. 476-479.

- KASHYAP, R.L. (1976), 'Speaker recognition from an unknown utterance and speaker-speech interaction', *IEEE Transactions in Acoustics, Speech and Signal Processing*, Vol. ASSP-24, No. 6, December, pp. 481-488.
- KAY, S.M. and MARPLE, JR., S.L. (1981), 'Spectrum analysis—A modern perspective', *Proceedings of the IEEE*, Vol. 69, No. 11, November, pp. 1380-1419.
- KERSTA, L.G. (1962), 'Voiceprint identification', *Nature*, December, p. 1253.
- KINSLER, L.E., FREY, A.R., COPPENS, A.B. and SANDERS, J.V. (1982), *Fundamentals of Acoustics*, John Wiley & Sons, New York, 3rd ed.
- KIOZUMI, T., TANIGUCHI, S. and HIROMITSU, S. (1985), 'Glottal source-vocal tract interaction', *Journal of the Acoustical Society of America*, Vol. 78, No. 5, November, pp. 1541-1547.
- KIRKLAND, J.R. and GARDEN, K.L. (1991), 'Neural magic and speech recognition', In *Proc. NELCON, (New Zealand National Electronics Conference)*, August, pp. 42-47.
- KITAWAKI, N. and NAGABUCHI, H. (1988), 'Quality assessment of speech coding and speech synthesis systems', *IEEE Communications Society Magazine*, October, pp. 36-44.
- KITZING, P. (1986), 'LTAS criteria pertinent to the measurement of voice quality', *Journal of Phonetics*, No. 14, pp. 477-482.
- KLECKA, W.R. (1980), *Discriminant Analysis, Quantitative Applications in the Social Sciences*, SAGE Publications, Beverly Hills.
- KNORR, S.G. (1979), 'Reliable voiced/unvoiced decision', *IEEE Transactions in Acoustics, Speech and Signal Processing*, Vol. 27, No. 3, June, pp. 263-267.
- KOHONEN, T. (1990), 'The self-organizing map', *Proceedings of the IEEE*, Vol. 78, No. 9, September, pp. 1464-1479.
- KRASNER, M., WOLF, J., KARNOFSKY, K., SCHWARTZ, R., ROUCOS, S. and GISH, H. (1984), 'Investigation of text-independent speaker identification techniques under conditions of variable data', In *International Conference on Acoustics, Speech, and Signal Processing*, IEEE, pp. 18B.5.1-18B.5.4.
- KRATZENSTEIN, C.R. (1782), 'Sur la naissance de la formation des voyelles', *Journal of Physiology*, Vol. 21, pp. 358-381.
- KREYSZIG, E. (1979), *Advanced Engineering Mathematics*, John Wiley & Sons, New York, 4th ed.
- KROON, P., DEPRETTERE, E.F. and SLUYTER, R.J. (1986), 'Regular-pulse excitation-A novel approach to effective and efficient multipulse coding of speech', *IEEE Transactions in Acoustics, Speech and Signal Processing*, Vol. ASSP-34, No. 5, October, pp. 1054-1063.
- KWON, S.Y. and GOLDBERG, A.J. (1984), 'An enhanced LPC vocoder with no voiced/unvoiced switch', *IEEE Transactions in Acoustics, Speech and Signal Processing*, Vol. ASSP-32, No. 4, August, pp. 851-858.

- LARAR, J.N., SCHROETER, J. and SONDHI, M.M. (1988), 'Vector quantisation of the articulatory space', *IEEE Transactions in Acoustics, Speech and Signal Processing*, Vol. ASSP-36, No. 12, December, pp. 1812-1818.
- LAVER, J. (1980), *The phonetic description of voice quality*, Cambridge studies in linguistics, Cambridge University Press, Cambridge.
- LEHISTE, I. (1970), *Suprasegmentals*, MIT Press, Massachusetts.
- LIEBERMAN, P. and BLUMSTEIN, S.E. (1988), *Speech physiology, speech perception, and acoustic phonetics*, Cambridge Studies in Speech Science and Communication, Cambridge University Press, Cambridge.
- LIM, C.A., ELDER, A.G., CLARK, T.M. and BATES, R.H.T. (1990), 'Software implementation of Hidden Markov model for recognition of isolated digits uttered by New Zealand speaker', In *Proc. NELCON, (New Zealand National Electronics Conference)*, pp. 287-294.
- LINDE, Y., BUZO, A. and GRAY, R.M. (1980), 'An algorithm for vector quantiser design', *IEEE Transactions in Communications*, Vol. COM-28, No. 1, January, pp. 84-95.
- LINDEMAN, R.H., MERENDA, P.F. and GOLD, R.Z. (1980), *Introduction to bivariate and multivariate analysis*, Scott, Foresman and Company, Glenview, Illinois.
- LINGGARD, R. (1985), *Electronic synthesis of speech*, Cambridge University Press, Cambridge.
- LLOYD, C.J. (1990), 'Confidence intervals from the difference between two correlated proportions', *Journal of the American Statistical Association*, Vol. 85, No. 412, December, pp. 1154-1158.
- LÖFQVIST, A. (1986), 'The long-time average spectrum as a tool in voice research', *Journal of Phonetics*, No. 14, pp. 471-475.
- LOOKABAUGH, T.D. and GRAY, R.M. (1989), 'High-resolution quantization theory and the vector quantization advantage', *IEEE Transactions on Information Theory*, Vol. 35, No. 5, September, pp. 1020-1033.
- LUBKER, J.F. and MOLL, K.L. (1965), 'Simultaneous oral-nasal air flow measurements and cinefluorographic observations during speech production', *Cleft Palate Journal*, Vol. 2, pp. 257-272.
- LUMMIS, R.C. (1971), 'Real-time techniques for speaker verification by computer.', *Journal of the Acoustical Society of America*, Vol. 50, p. 106(A).
- MACLAGAN, M.A. (1982), 'An acoustic study of New Zealand vowels', *The New Zealand speech therapist's journal*, Vol. 37, pp. 20-26.
- MAKHOUL, J. (1975), 'Linear prediction: A tutorial review', *Proceedings of the IEEE*, Vol. 63, No. 4, April, pp. 561-580.
- MAKHOUL, J. (1977), 'Stable and efficient lattice methods for linear prediciton', *IEEE Transactions in Acoustics, Speech and Signal Processing*, Vol. ASSP-25, No. 5, October, pp. 423-428.
- MAKHOUL, J., ROUCOS, S. and GISH, H. (1985), 'Vector quantisation in speech coding', *Proceedings of the IEEE*, Vol. 73, No. 11, November, pp. 1551-1558.

- MANSOUR, D. and JUANG, B.H. (1988), 'A family of distortion measures based upon projection operation for robust speech recognition', In *International Conference on Acoustics, Speech, and Signal Processing*, IEEE, pp. 36-39.
- MARKEL, J.D. and DAVIS, S.B. (1979), 'Text-independent speaker recognition from a large linguistically unconstrained time-spaced data base', *IEEE Transactions in Acoustics, Speech and Signal Processing*, Vol. ASSP-27, No. 1, February, pp. 74-82.
- MARKEL, J.D. and GRAY, JR., A.H. (1976), *Linear prediction of speech*, Springer-Verlag, Berlin.
- MARKEL, J.D., OSHIKA, B.T. and GRAY, JR., A.H. (1977), 'Long-term feature averaging for speaker recognition', *IEEE Transactions in Acoustics, Speech and Signal Processing*, Vol. ASSP-25, No. 4, August, pp. 330-337.
- MATSUMOTO, H. (1989), 'Text-independent speaker identification from short utterances based on piecewise discriminant analysis', *Computer Speech and Language*, No. 3, pp. 133-150.
- MAX, J. (1960), 'Quantizing for minimum distortion', *IRE Transactions on Information Theory*, Vol. IT-6, March, pp. 7-12.
- McGONEGAL, C.A., RABINER, L.R. and ROSENBERG, A.E. (1977), 'A subjective evaluation of pitch methods using LPC synthesised speech', *IEEE Transactions in Acoustics, Speech and Signal Processing*, Vol. ASSP-25, No. 3, June, pp. 221-229.
- McGONEGAL, C.A., ROSENBERG, A.E. and RABINER, L.R. (1979), 'The effects of several transmission systems on an automatic speaker verification system', *Bell Systems Technical Journal*, Vol. 58, No. 9, November, pp. 2071-2087.
- MCNEMAR, Q. (1947), 'Note on the sampling error of the difference between correlated proportions or percentages', *Psychometrika*, Vol. 12, No. 2, June, pp. 153-157.
- MOHN, JR., W.S. (1971), 'Two stastical feature evaluation techniques applied to speaker identification', *IEEE Transactions in Computers*, Vol. C-20, No. 9, September, pp. 979-987.
- MOORER, J.A. (1974), 'The optimum comb method of pitch period analysis of continuous digitized speech', *IEEE Transactions in Acoustics, Speech and Signal Processing*, Vol. ASSP-22, No. 5, October, pp. 330-338.
- MOULINES, E. and DI FRANCESCO, R. (1990), 'Detection of the glottal closure by jumps in the statistical properties of the speech signal', *Speech Communication*, Vol. 9, pp. 401-418.
- MURRY, T. and SINGH, S. (1980), 'Multidimensional analysis of male and female voices', *Journal of the Acoustical Society of America*, Vol. 68, No. 5, November, pp. 1294-1300.
- NAIK, J.M. (1990), 'Speaker verification: A tutorial', *IEEE Communications Society Magazine*, Vol. 28, No. 1, January, pp. 42-48.
- NAIK, J.M., NETSCH, L.P. and DODDINGTON, G.R. (1989), 'Speaker verification over long distance telephone lines', In *International Conference on Acoustics, Speech, and Signal Processing*, IEEE, pp. 524-527.

- NODA, H. (1988), 'Frequency-warped spectral distance measures for speaker verification in noise', In *International Conference on Acoustics, Speech, and Signal Processing*, IEEE, pp. 576-579.
- NODA, H. (1989), 'On the use of the information on individual's position in the parameter space for speaker recognition', In *International Conference on Acoustics, Speech, and Signal Processing*, IEEE, pp. 516-519.
- O'CONNOR, J.D. (1973), *Phonetics*, Penguin, Harmondsworth.
- OPPENHEIM, A.V. and SCHAFER, R.W. (1968), 'Homomorphic analysis of speech', *IEEE Transactions on Audio and Electroacoustics*, Vol. AU-16, No. 2, June, pp. 221-226.
- OPPENHEIM, A.V. and SCHAFER, R.W. (1975), *Digital signal processing*, Prentice-Hall Inc., New Jersey.
- OPPENHEIM, A.V. and WILLSKY, A.S. (1983), *Signals and Systems*, Prentice-Hall signal processing series, Prentice-Hall, Inc., New Jersey.
- O'SHAUGHNESSY, D. (1986), 'Speaker recognition', *IEEE Acoustics, Speech and Signal Processing Society Magazine*, October, pp. 4-17.
- PAPOULIS, A. (1980), *Circuits and Systems, Modern Approach*, HRW Series in Electrical and Computer Engineering, Holt, Rinehart and Winston, Inc., New York.
- POLLACK, I., PICKETT, J.M. and SUMBY, W.H. (1954), 'On the identification of speakers by voice', *Journal of the Acoustical Society of America*, Vol. 26, No. 3, May, pp. 403-406.
- PRUZANSKY, S. and MATHEWS, M.V. (1964), 'Talker-recognition procedure based on analysis of variance', *Journal of the Acoustical Society of America*, Vol. 36, No. 11, November, pp. 2041-2047.
- RABINER, L.R. (1989), 'A tutorial on Hidden Markov Models and selected applications in speech recognition', *Proceedings of the IEEE*, Vol. 77, No. 2, February, pp. 257-286.
- RABINER, L.R. and JUANG, B.H. (1986), 'An introduction to hidden Markov models', *IEEE Acoustics, Speech and Signal Processing Society Magazine*, January, pp. 4-16.
- RABINER, L.R. and LEVINSON, S.E. (1981), 'Isolated and connected word recognition - theory and selected applications', *IEEE Transactions in Communications*, Vol. COM-29, No. 5, May, pp. 621-659.
- RABINER, L.R. and SAMBUR, M.R. (1975), 'An algorithm for determining the end-points of isolated utterances', *Bell Systems Technical Journal*, Vol. 45, No. 2, February, pp. 297-315.
- RABINER, L.R. and SCHAFER, R.W. (1978), *Digital Processing of Speech Signals*, Prentice-Hall Signal Processing Series, Prentice-Hall, Inc., Englewood Cliffs, New Jersey.
- RABINER, L.R., CHENG, M.J., ROSENBERG, A.E. and McGONEGAL, L.A. (1976), 'A comparative performance study of several pitch detection algorithms', *IEEE Transactions in Acoustics, Speech and Signal Processing*, Vol. ASSP-24, No. 5, October, pp. 399-418.

- RABINER, L.R., ATAL, B.S. and SAMBUR, M.R. (1977), 'LPC prediction error—analysis of its variation with the position of the analysis frame', *IEEE Transactions in Acoustics, Speech and Signal Processing*, Vol. ASSP-25, No. 5, October, pp. 434–442.
- RABINER, L.R., ROSENBERG, A.E. and LEVINSON, S.E. (1978), 'Considerations in dynamic time warping algorithms for discrete word recognition', *IEEE Transactions in Acoustics, Speech and Signal Processing*, Vol. ASSP-26, No. 6, December, pp. 575–582.
- RABINER, L.R., LEVINSON, S.E. and SONDHI, M.M. (1983), 'On the application of vector quantization and hidden Markov models to speaker-independent, isolated word recognition', *Bell Systems Technical Journal*, Vol. 62, No. 4, April, pp. 1075–1105.
- RABINER, L.R., JUANG, B., LEVINSON, S.E. and SONDHI, M.M. (1985), 'Recognition of isolated digits using Hidden Markov Models with continuous mixture densities', *AT & T Technical Journal*, Vol. 64, No. 6, July-August, pp. 1211–1233.
- REICH, A.R. (1981), 'Detecting the presence of vocal disguise in the male voice', *Journal of the Acoustical Society of America*, Vol. 69, No. 5, May, pp. 1458–1461.
- REICH, A.R. and DUKE, J.E. (1979), 'Effects of selected vocal disguises upon speaker identification by listening', *Journal of the Acoustical Society of America*, Vol. 66, No. 4, October, pp. 1023–1028.
- REICH, A.R., MOLL, K.L. and CURTIS, J.F. (1976), 'Effects of selected vocal disguises upon spectrographic speaker identification', *Journal of the Acoustical Society of America*, Vol. 60, No. 4, October, pp. 919–925.
- ROBINSON, E.A. (1980), *Physical applications of stationary time-series*, Charles Griffin & Company Ltd., London.
- ROSENBERG, A.E. (1973), 'Listener performance in speaker verification tasks', *IEEE Transactions on Audio and Electroacoustics*, Vol. AU-21, No. 3, June, pp. 221–225.
- ROSENBERG, A.E. (1976), 'Evaluation of an automatic speaker-verification system over telephone lines', *Bell Systems Technical Journal*, Vol. 55, No. 6, July/August, pp. 723–744.
- ROSENBERG, A.E. and SAMBUR, M.R. (1975), 'New techniques for automatic speaker verification', *IEEE Transactions in Acoustics, Speech and Signal Processing*, Vol. ASSP-23, No. 2, April, pp. 169–177.
- ROSS, M.J., SHAFFER, H.L., COHEN, A., FREUDBERG, R. and MANLEY, H.J. (1974), 'Average magnitude difference function pitch extractor', *IEEE Transactions in Acoustics, Speech and Signal Processing*, Vol. ASSP-22, No. 5, October, pp. 353–362.
- ROTHENBURG, M. (1983), 'An interactive model for the voice source', In BLESS, D.M. and ABBS, J.H. (Eds.), *Vocal Fold Physiology: Contemporary research and clinical issues*, College-Hill Press, San Diego, Chap. 12.

- SAMOUELLIAN, A. and HOLMES, W.H. (1985), 'Real time pitch estimation', In *20th International Electronics Convention*, IREE, Melbourne, 30 Sept - 4 Oct, pp. 1019-1021.
- SAS (1985a), *SAS User's guide: Basics*, SAS Institute Inc., Cary, NC, USA, 5th ed.
- SAS (1985b), *SAS User's guide: Statistics*, SAS Institute Inc., Cary, NC, USA, 5th ed.
- SCHARF, B. (1970), 'Critical bands', In TOBAIS, J.V. (Ed.), *Foundations of modern auditory theory*, Academic Press, New York.
- SCHMIDT-NIELSEN, A. and STERN, K.R. (1985), 'Identification of known voices as a function of familiarity and narrow-band coding', *Journal of the Acoustical Society of America*, Vol. 77, No. 2, February, pp. 658-663.
- SCHROEDER, M.R. (1966), 'Vocoders: Analysis and synthesis of speech', *Proceedings of the IEEE*, Vol. 54, No. 5, May, pp. 720-734.
- SCHROEDER, M.R. (1968), 'Reference signal for signal quality studies', *Journal of the Acoustical Society of America*, Vol. 44, No. 6, pp. 1735-1736.
- SCHROEDER, M.R. (1975), 'Models of hearing', *Proceedings of the IEEE*, Vol. 63, No. 9, September, pp. 1332-1350.
- SCHROEDER, M.R. (1984), 'Linear prediction, entropy and signal analysis', *IEEE Acoustics, Speech and Signal Processing Society Magazine*, Vol. 1, No. 3, July, pp. 3-11.
- SCHROEDER, M.R. (1985), 'Linear predictive coding of speech: Review and current directions', *IEEE Communications Society Magazine*, Vol. 23, No. 8, August, pp. 54-61.
- SHANNON, C.E. (1948), 'A mathematical theory of communication', *Bell Systems Technical Journal*, Vol. 27, No. 3, July, pp. 379-423.
- SHANNON, C.E. (1949), 'Communication in the presence of noise', *Proc. of the IRE*, Vol. 37, No. 1, January, pp. 10-21.
- SHIRAI, K., MANO, K. and ISHIGE, S. (1988), 'Speaker identification based on frequency distribution of vector-quantized spectra', *Systems and Computers in Japan*, Vol. 18, No. 6, pp. 63-72.
- SIEGEL, L.J. (1979), 'A procedure for using pattern classification techniques to obtain a voiced/unvoiced classifier', *IEEE Transactions in Acoustics, Speech and Signal Processing*, Vol. ASSP-27, No. 1, February, pp. 83-89.
- SINTON, A.M. (1986), *Contributions to astronomical and medical information processing*, PhD thesis, Electrical and Electronic Engineering Department, University of Canterbury, Christchurch, New Zealand.
- SKINNER, P.H. and SHELTON, R.L. (Eds.) (1978), *Speech, language and hearing: Normal processes and disorders*, Addison-Wesley Publishing Co., Reading, Massachusetts, USA.
- SLEPIAN, D. (1976), 'On bandwidth', *Proceedings of the IEEE*, Vol. 64, No. 3, March, pp. 292-300.

- SONDHI, M.M. (1968), 'New methods of pitch extraction', *IEEE Transactions on Audio and Electroacoustics*, Vol. AU-16, No. 2, June, pp. 262-266.
- SONDHI, M.M. (1979), 'Estimation of vocal-tract areas: The need for acoustical measurements', *IEEE Transactions in Acoustics, Speech and Signal Processing*, Vol. ASSP-27, No. 3, June, pp. 268-273.
- SONDHI, M.M. and GOPINATH, B. (1971), 'Determination of vocal-tract shape from impulse response at the lips', *Journal of the Acoustical Society of America*, Vol. 49, No. 6 (Part 2), pp. 1867-1873.
- SONDI, M.M. and SCHROETER, J. (1987), 'A hybrid time-frequency domain articulatory speech synthesizer', *IEEE Transactions in Acoustics, Speech and Signal Processing*, Vol. ASSP-35, No. 7, July, pp. 955-967.
- SONG, K.H. and UN, C.K. (1983), 'Pole-zero modelling of speech based on high-order pole model fitting and decomposition methods', *IEEE Transactions in Acoustics, Speech and Signal Processing*, Vol. ASSP-31, No. 6, December, pp. 1556-1565.
- SOONG, F.K. and ROSENBERG, A.E. (1988), 'On the use of instantaneous and transitional spectral information in speaker recognition', *IEEE Transactions in Acoustics, Speech and Signal Processing*, Vol. 36, No. 6, June, pp. 871-879.
- SOONG, F.K. and SONDHI, M.M. (1988), 'A frequency-weighted Itakura spectral distortion measure and its application to speech recognition in noise', *IEEE Transactions in Acoustics, Speech and Signal Processing*, Vol. ASSP-36, No. 1, January, pp. 41-48.
- SOONG, F.K., ROSENBERG, A.E., RABINER, L.R. and JUANG, B.H. (1985), 'A vector quantization approach to speaker recognition', In *International Conference on Acoustics, Speech, and Signal Processing*, IEEE, pp. 11.4.1-11.4.4.
- SOONG, F.K., ROSENBERG, A.E., JUANG, B.H. and RABINER, L.R. (1987), 'A vector quantization approach to speaker recognition', *AT & T Technical Journal*, Vol. 66, No. 2, March/April, pp. 14-26.
- STEVENS, K.N., WILLIAMS, C.E., CARBONALL, J.R. and WOODS, B. (1968), 'Speaker authentication and identification: A comparison of spectrographic and auditory presentations of speech material', *Journal of the Acoustical Society of America*, Vol. 44, No. 6, pp. 1596-1607.
- STREMLER, F.G. (1982), *Introduction to communication systems*, Addison-Wesley Publishing Company, Reading, Massachusetts, 2nd ed.
- SUTHERLAND, A.M., JACK, M.A. and LAVER, J. (1988), 'Improved pitch detection algorithm employing temporal structure investigation of the speech waveform', *IEE Proceedings F*, Vol. 135, No. 2, April, pp. 169-174.
- TAKAGI, T. and KUWABARA, H. (1986), 'Contributions of pitch, formant frequency and bandwidth to the perception of voice personality', In *International Conference on Acoustics, Speech, and Signal Processing*, IEEE, pp. 889-892.
- THE MATH WORKS (1990), *PRO-MATLAB User's Guide*, The Math Works, Inc, 21 Eliot St, South Natick, MA 01760, USA.

- THORPE, C.W. (1990), *Processing of speech and other sounds.*, PhD thesis, Electrical and Electronic Engineering Department, University of Canterbury, Christchurch, New Zealand.
- THORPE, C.W. and BATES, R.H.T. (19XX), 'Speech analysis/companding/resynthesis by shift-and-add and Clean', *IEE Proceedings I*, Vol. , No. , p. . Submitted for publication.
- TUCKER, W.R. and BATES, R.H.T. (1978), 'A pitch estimation algorithm for speech and music', *IEEE Transactions in Acoustics, Speech and Signal Processing*, Vol. 26, No. 6, pp. 597-604.
- TURNER, S.G. (1986), *Real-time speech analysis for use with impaired speech aids*, Master's thesis, Electrical and Electronic Engineering Department, University of Canterbury, Christchurch, New Zealand.
- UN, C.K. and CHOI, K.Y. (1981), 'Improving LPC analysis of noisy speech by autocorrelation subtraction method', In *International Conference on Acoustics, Speech, and Signal Processing*, IEEE, pp. 1082-1085.
- VAN LANCKER, D., KREIMAN, J. and EMMOREY, K. (1985), 'Familiar voice recognition: patterns and parameters. Part 1: Recognition of backward voices', *Journal of Phonetics*, Vol. 13, pp. 19-38.
- VELUS, G. (1988), 'Variants of cepstrum based speaker identity verification', In *International Conference on Acoustics, Speech, and Signal Processing*, IEEE, pp. 583-586.
- VOIERS, W.D. (1964), 'Perceptual bases of speaker identity', *Journal of the Acoustical Society of America*, Vol. 36, No. 6, June, pp. 1065-1073.
- WAKITA, H. (1973), 'Direct estimation of the vocal tract shape by inverse filtering of acoustic speech waveforms', *IEEE Transactions on Audio and Electroacoustics*, Vol. AU-21, No. 5, October, pp. 417-427.
- WARD-SMITH, A.J. (1980), *Internal Fluid Flow*, Clarendon Press, Oxford.
- WATSON, C.I., KENNEDY, W.K. and BATES, R.H.T. (1990), 'Towards a computer based speech therapy aid', In *Proc. SST, (Australian International Conference on Speech Science and Technology)*, November, pp. 234-239.
- WATSON, C.I., KENNEDY, W.K. and BATES, R.H.T. (1991), 'A computer based speech training aid for the speech impaired: Development and evaluation', In *Proc. NELCON, (New Zealand National Electronics Conference)*, August, pp. 105-110.
- WENDLER, J., RAUHUT, A. and KRÜGER, H. (1986), 'Classification of voice qualities', *Journal of Phonetics*, No. 14, pp. 483-488.
- WILLIAMS, C.E. and STEVENS, K.N. (1972), 'Emotions and speech: Some acoustical correlates', *Journal of the Acoustical Society of America*, Vol. 52, No. 4 (Part 2), pp. 1238-1250.
- WISE, J.D., CAPRIO, J.R. and PARKS, T.W. (1976), 'Maximum likelihood pitch estimation', *IEEE Transactions in Acoustics, Speech and Signal Processing*, Vol. ASSP-24, No. 5, October, pp. 418-423.

- WITTEN, I.H. (1982), *Principles of computer speech*, Academic Press, London.
- WONG, D.Y., JUANG, B. and GRAY, JR., A.H. (1982), 'An 800 bit/s vector quantisation LPC vocoder', *IEEE Transactions in Acoustics, Speech and Signal Processing*, Vol. ASSP-30, No. 5, October, pp. 770-780.
- WOODWARD, P.M. (1953), *Probability and information theory, with applications to radar*, Electronics and waves, Pergamon Press Ltd, London.
- WU, L. and FALLSIDE, F. (1991), 'On the design of connectionist vector quantizers', *Computer Speech and Language*, Vol. 5, pp. 207-229.
- YAMADA, Y., TAZAKI, S. and GRAY, R.M. (1980), 'Asymptotic performance of block quantizers with difference distortion measures', *IEEE Transactions on Information Theory*, Vol. IT-26, No. 1, January, pp. 6-14.
- YOUNG, M.A. and CAMPBELL, R.A. (1967), 'Effects of context on talker identification', *Journal of the Acoustical Society of America*, Vol. 42, No. 6, pp. 1250-1254.
- ZHENG, Y.C. and YUANG, B.Z. (1988), 'Text-dependent speaker identification using circular hidden Markov models', In *International Conference on Acoustics, Speech, and Signal Processing*, IEEE, pp. 580-582.