WAVELET-BASED TECHNIQUES FOR SPEECH RECOGNITION

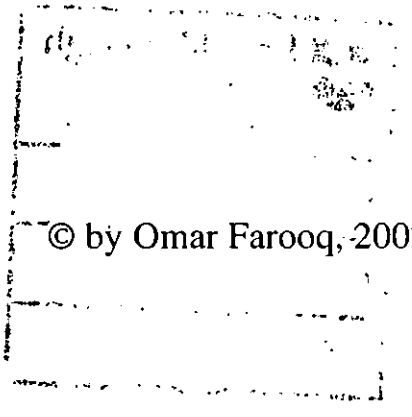
By

Omar Farooq

A doctoral thesis submitted in partial fulfilment of the requirements for the award of

Doctor of Philosophy of Loughborough University

June 2002



Dedicated to my parents

ABSTRACT

In this thesis, new wavelet-based techniques have been developed for the extraction of features from speech signals for the purpose of automatic speech recognition (ASR). One of the advantages of the wavelet transform over the short time Fourier transform (STFT) is its capability to process non-stationary signals. Since speech signals are not strictly stationary the wavelet transform is a better choice for time-frequency transformation of these signals. In addition it has compactly supported basis functions, thereby reducing the amount of computation as opposed to STFT where an overlapping window is needed.

New features based on the discrete wavelet transform (DWT) and the admissible wavelet packet (AWP) have been proposed. These features are not only speaker-independent but also overcome the problem of shift variance encountered in earlier approaches. To match the human auditory system, which has constant relative bandwidth channels, admissible wavelet packets have been proposed to obtain a similar 24-band structure. A simple classifier based on the Linear Discriminant Analysis (LDA) has been used for classification and the recognition performance has been compared with those of the standard Mel scale cepstral coefficients (MFCC) based features. An Artificial Neural Network based classifier has also been tested to assess the improvement that may be achieved by a non-linear classifier.

In order to establish the robustness of the proposed features, the noise performance is also studied at different levels of signal to noise ratios and a new technique of robust wavelet-based sub-band features has been proposed. Further more, a new pre-processing stage based on wavelet denoising is proposed to enhance the recognition performance under noisy (white Gaussian) conditions.

Finally, the 24-band admissible wavelet-based features are tested on a recogniser using the Hidden Markov Model (HMM) for the task of word recognition from continuous speech. The recognition performance was tested for clean as well as noisy speech and this was compared with the standard MFCC based features. The proposed denoising technique is found to increase the recognition performance significantly in the presence of noise for the MFCC as well as 24-band admissible wavelet-based features.

ACKNOWLEDGEMENTS

First of all, I would like to thank God for everything, including the ability to learn, the pleasure of sharing it and the wisdom of appreciating it.

I would like to thank my supervisor, Dr. Sekharjit Datta for his support, encouragement and enthusiastic discussions. I have been very fortunate to have the opportunity to work with him, and I can never thank him adequately for his help on both academic and non-academic matters. .

I would like to thank Prof. Bryan Woodward, my Director of Research for providing valuable suggestions during my research and proving valuable comments in the write-up of this thesis. I would also like to thank Dr. Chris Richards for invaluable discussions during the early stages of my research.

I want to thank my parents and family for the love and support they have provided during my three years of research, without which I would not have reached my current goals and aspirations.

I would like to offer special thanks to all of my friends, fellow research students and staff in the Department of Electronic and Electrical Engineering at Loughborough University. All of my experiences and interactions with these people have shaped my work and helped make my graduate experience worthwhile and unforgettable.

Finally, I would like to thank the UK Commonwealth Commission for sponsoring my research, and my home institution, Aligarh Muslim University, India for granting me leave to carry out this research.

GLOSSARY

ANN	Artificial Neural Network
ASR	Automatic Speech Recognition
AWP	Admissible Wavelet packet
CDCN	Codeword Dependent Cepstral Normalisation
CDHMM	Continuous Density Hidden Markov Model
CWT	Continuous Wavelet Transform
DB	Daubechies Wavelet
DHMM	Discrete Hidden Markov Model
DCT	Discrete Cosine Transform
DWT	Discrete Wavelet Transform
FCDCN	Fixed Codeword Dependent Cepstral Normalisation
FFT	Fast Fourier Transform
HMM	Hidden Markov Model
HTK	HMM Tool Kit
LDA	Linear Discriminant Analysis
MFCC	Mel Frequency Cepstral Coefficient
ML	Maximum Likelihood
MLP	Multi-Layer Perceptron
MRA	Multi-resolution Analysis
PLP	Perceptual Linear Prediction
POF	Probabilistic Optimum Filtering
RASTA	RelAtive SpecTrA
RPS	Root Power Spectrum
SCWT	Scalable Wavelet Transform
SDCN	SNR Dependent Cepstral Normalisation
SNR	Signal-To-Noise Ratio

STFT	Short Time Fourier Transform
TDNN	Time Delay Neural Network
TIMIT	Texas Instrument/ Massachusetts Institute of Technology
VQ	Vector Quantisation
WP	Wavelet Packet
WT	Wavelet Transform
ZCPA	Zero Crossing Peak Amplitude

NOTATIONS

$\mathbf{CWT(\tau,a)}$	Continuous wavelet transform
$\|\mathbf{x}-\mathbf{m_k}\|^2$	Euclidean distance
$\mathbf{V \oplus W}$	Direct sum of two vector spaces
$\langle \mathbf{x,y} \rangle$	Inner product $\sum_i \mathbf{x_i}.\mathbf{y_i^*}$
$\mathbf{A \cap B}$	The intersection of two sets A and B is the set of elements common to A and B.
$\mathbf{A \cup B}$	The union of two sets A and B is the set obtained by combining the members of each.
$\mathbf{DWTf[a^j,n]}$	Discrete wavelet transform
Σ	Covariance matrix
$\mathbb{R}$	The set of all real numbers
$\mathbf{L^2(\mathbb{R})}$	Space of finite energy function $\int_{-\infty}^{+\infty} \|\mathbf{x(t)}\|^2 \mathbf{dt} < \infty$
$\Delta \mathbf{f}$	Frequency width (bandwidth)
$\Delta \mathbf{t}$	Time width
ψ **(t)**	Mother Wavelet
E[x]	Expected value
P[x]	Probability
x(t)	Continuous signal
$\mathbf{x^*}$	Complex conjugate
x[n]	Discrete signal
Z	The set of all integers

LIST OF FIGURES

LIST OF TABLES

CONTENTS

CHAPTER 1

INTRODUCTION TO SPEECH RECOGNITION

Since the earliest days of computing, speech was considered to be the ultimate human/machine interface specifically for interacting with computers. This is because speech is the most effective means of communication between human beings. To make this interface possible, automatic speech recognition (ASR) capabilities are required by a machine. Although current ASR systems show great promise in providing this human/machine interface there is still a large gap to bridge. The major problems in the implementation of an ASR system are the complexity and variability of speech signals. The variations in speech signals may be due to the motion and size of the vocal tract articulators and their constraints. The variation of the acoustic media can also cause a difference in the speech signals. The main cause of variations can be classified into following groups:

1. **Acoustic media**: This includes noise in the background, interference created due to reverberation and changes in the environment, position and characteristics of the microphone. In the case of speech recognition over the telephone or mobile network, additional factors such as channel distortion and fading come into effect and can further distort and band-limit the signal.
2. **Inter-speaker variability**: Due to anatomical differences in vocal tracts and articulators, speech signals carry information about individual speakers along with the phonetic and linguistic information. The geographical origin of the

speaker may also add to the variations due to the presence of different dialects in various parts of the world.

3. **Intra-speaker variability**: Even for the same speaker the motion of the articulators is not the same for the same sound. Also there are problems of co-articulations in which the same phoneme is pronounced differently depending upon the context. Variations in speech can also be caused due to psychological state of the speaker, e.g. stress, joy, anger, hesitation, etc.

In spite of all these variations, speech signals are structured and are subject to phonetic and linguistic rules. The ASR tries to extract this structured information related to each phoneme for the purpose of recognition. Recently there have been efforts to integrate visual information with audio in order to enhance the performance of speech recognition [1], [2], [3]. A brief overview of an ASR is given in the next section.

1.1 Overview of automatic speech recognition systems

An ASR mainly consists of four processing units; the front end, acoustic modelling, language modelling and decoding. A general block diagram of an ASR system is shown in Figure 1.1. The acoustic processing front-end takes the input as analogue speech and in the simplest case may perform the analogue-to-digital (A/D) conversion only. Sometimes digital filters are used for pre-emphasis and noise removal. The microphone used for inputting speech may introduce noise at the line frequency, as well as low and high frequency loss and non-linear distortion. The A/D conversion itself introduces quantisation noise and a fluctuating DC bias. This process can yield sampled data with a signal to noise ratio in excess of 30dB. The function of the pre-emphasis filter is to boost the signal spectrum at a rate of approximately 20dB per decade. This is carried out firstly to boost the voiced section of the speech as it has a natural attenuation of about 20dB per decade due to the speech production mechanism itself [4]. Secondly, it emphasises the perceptually important band above 1kHz for which the ear is more sensitive.

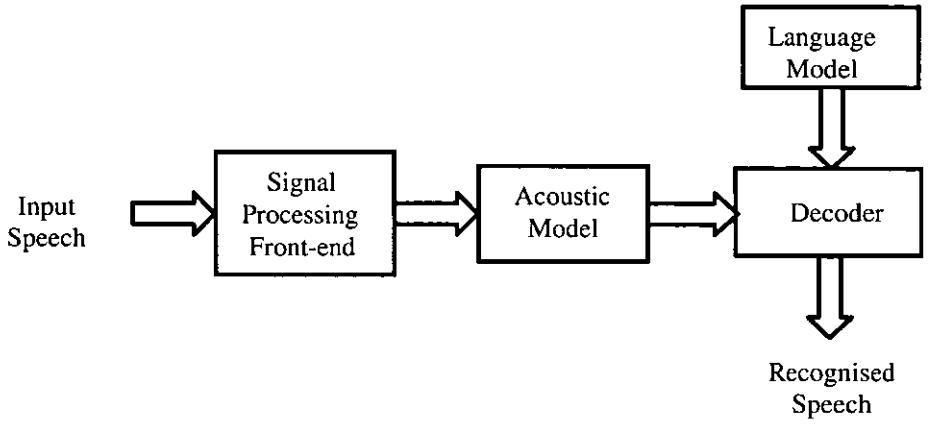


Figure 1.1: Block diagram of a simple ASR.

From the speech samples a duration $\mathbf{T_w}$ seconds is windowed and is used for processing at later stages. The window duration may be from 20ms to 30ms and the Hamming window is usually used for this purpose. A frame is formed from the windowed data with a typical frame duration ($\mathbf{T_f}$) of about 10 to 20ms. Since the frame duration is shorter than window duration there is an overlap of data and the percentage overlap is given as:

$$\textbf{\% Overlap} = ((T_w - T_f)*100)/T_w \quad (1.1)$$

As the number of samples in a frame is large, it will be very difficult for a classifier to work upon these samples directly for recognition. Furthermore, these samples carry information, much of which is redundant. An important unit in the acoustic front end is the feature extractor, which tries to extract sufficient and relevant information from the input samples thereby reducing the dimension. Usually time-frequency analysis is carried out during the frame period of the speech samples to extract features. Linear Predictive Coefficients (LPC), Perceptual Linear Prediction (PLP) coefficients and Mel Frequency Cepstral Coefficients (MFCC) are the most commonly used features for the task of speech recognition [4], [5], [6], [7], [8]. In a simple speech recognition system dealing with a small vocabulary the features extracted are passed to a classifier, which tries to recognise the input from these features. A classifier may be based on the Linear Discriminant Analysis (LDA) [9], [10], [11] or on the Artificial Neural Networks [12], [13], [14], [15], [16], [17] and classified without the use of a language model. In a more complex system the acoustic model may itself have a Hidden Markov Model (HMM) to encode the temporal evolution of features [8],

[18], [19].

The language model provides constraints in the occurrence of particular words and word sequences. This plays a very important role in determining the search space, which is crucial in making the speech recognition system respond in real time. A static language model has been used earlier in which the occurrence probability of a word sequence corresponding to the most recent N words is used to predict the probability of the current word. A language model that uses this approach is known as the N-gram model. Since the statistics of the model are developed during the training phase and remain fixed it is therefore a static model. In the case of a dynamic or adaptive language model the word probability depends on the input text observed until that time. This helps in improving the performance of the system in places where a large language source has sub-language structures within itself [6], [8], [20]. The last stage is the decoder (also known as the search stage). This predicts the most likely word sequence, given a language model, acoustic models and a sequence of utterances. The Viterbi algorithm is the technique most commonly used for decoding. It is a recursive transition network composed of the states of HMM in which each state can be reached from any other state. This search is time synchronous but the implementation is impractical because of the size of the search space. A Viterbi beam search is used in practice, which reduces the search space and a dynamic programming technique is used for this purpose [6], [21], [22].

An ASR is first trained before it can be used for a speech recognition application. The training speech data is first segmented into words and these words are further divided into smaller acoustic units called phonemes. These phonemes are essentially the building blocks of words irrespective of language. Since the content of the training speech is known, a lexicon is used to look into the phonetic transcription of each word. The problem of utterance recognition by an ASR boils downs to the recognition of these phonemes. The features are extracted from the phonemes for each frame duration and are then passed on for acoustic modelling. The HMM is first used to build a context-independent acoustic model for each phoneme. These acoustic models are refined using the entire data given in the training phase. This helps to build a speaker-independent phoneme model if many speakers are used during the training phase. To

overcome the problems of co-articulation, context-dependent acoustic models of these phonemes are also used. This takes into account the effect the previous and the next phoneme of the one currently under consideration. The technique increases the computational effort and the decoding time, but it results into considerable improvement in the performance of an ASR [8].

The training process is very important in improving the performance of a large vocabulary speaker-independent ASR system. The training data set should cover larger speaker distribution; this helps in improving the capabilities of an ASR for speaker-independent performance. Since a language model is derived during the training phase, the spoken utterance should be large enough in volume to generalise the calculation of word probability. During the testing phase the speech signal is segmented into phoneme duration, and for each phoneme features are extracted. Then a scoring is performed based on the earlier phoneme models developed during the training phase. These phonemes are then combined to give the formation of a word from a given lexicon with a score associated with it. The language model is then used to weight this word score by the probability of its occurrence in the context.

The complexity of an ASR varies in a wide range. It can be as simple as speaker-dependent isolated word recognition on a very complex, large vocabulary speaker-independent continuous speech recognition in a noisy environment with channel distortion conditions. Depending upon its application the training process is tailored to have optimal recognition performance.

1.2 Classification of ASR

Depending upon the complexity, speech recognition systems can be classified into the following categories:

Isolated words/digits recognition: These are the earliest reported systems used in speech recognition. Each word/digit is separated by a pause, which is used in order to mark the end of the spoken word/digit. These systems are essentially based on the template matching technique and the number of words recognised is therefore restricted to few tens only.

Continuous digit recognition: This is the next phase of development in speech recognition where continuous spoken digits are recognised without having the inter-digit pause.

Read speech: This has a vocabulary size of about 100 words, but the inter-word pause is still present in this case. The emergence of dynamic time warping is an important landmark, which improves the recognition performance considerably [23].

Continuous speech: This is the latest system on which a lot of research is continuing. The vocabulary size is around 100k words. These systems are mostly based on the HMM and use language information coupled with acoustic information to recognise the speech.

1.2.1 State of the Art

The state-of-the-art ASR systems are speaker-independent, have a large vocabulary and can recognise continuous speech. The word error rate of these complex systems is below 10% in a quiet environment [8]. However, it is found that in the case of background noise or in the presence of channel distortion the recognition performance degrades rapidly. Efforts are now focusing on improving the robustness of these ASR systems in the presence of noise and channel distortion. The first approach is based on the extraction of features that are inherently resistant to noise. The techniques used in this approach are RASTA (RelAtive SpecTrA) processing [24], one-sided auto-correlation linear predictive coefficient [25] and auditory model processing of speech [26]. The second approach is based on the compensation model, which tries to recover intelligible speech from corrupted speech in the feature parameter domain or at the pattern matching stage. Methods using the second approach are cepstral normalisation [27], probabilistic optimum filtering [28], [29] and parallel model combination [18].

1.3 Research Objectives

There has been considerable progress in speech recognition technology during the last two decades; however, the basic feature extraction process has

remained unchanged and is based on the short time Fourier transform. The Fourier transform of a signal assumes that the signal is stationary during the time of analysis. This assumption only partially holds for some of the rapidly varying phonemes such as unvoiced stops (e.g. /p/, /t/ and /k/). Hence, the recognition performance for some phonemes is found to be poor when Fourier methods are applied for feature extraction.

The work detailed in this thesis explores an alternative time-frequency transformation technique based on the wavelet transform, which is suitable for stationary as well as non-stationary signal analysis. Recently, there have been many attempts to use the wavelet transform for the feature extraction task. The discrete wavelet transform [10], [30], [31], [32] as well as wavelet packets [10], [11], [13], [14], [33] have been tried for this purpose. Earlier works [10], [11], [13] had problems using the wavelet transform because of the shift variance in the feature. The goal of this work is to extract new features by using the discrete wavelet transform and the wavelet packet that are both shift invariant and speaker independent. The thesis will also explore the possibility of using the admissible wavelet packets to design a Mel scaled filter similar to the one used for the MFCC. Furthermore, the performance of the extracted features will be evaluated in the presence of additive white Gaussian noise.

To reduce the effect of noise on the performance of speech recognition system various filtering methods have been proposed [28], [29]. In this thesis, a new pre-processing technique based on wavelet denoising [34] is also investigated to reduce the effect of white Gaussian noise on the recognition performance.

1.4 Organisation of the thesis

This thesis is divided into seven chapters. In Chapter 1 a basic overview of speech recognition systems is presented. The chapter also discusses some of the problems encountered in the speech recognition processes by using the conventional techniques.

In Chapter 2 a study of different types of features used for speech recognition is presented. LPC, PLP and MFCC-based feature extraction

processes and their relative advantages are examined in detail. Also the features for robust speech recognition are described in this chapter.

Chapter 3 reviews the recently proposed discrete wavelet transform-based features and their problems. In this chapter a new set of features for phoneme recognition based on the discrete wavelet transform is proposed. The recognition performance of these features is tested using a Linear Discriminant Analysis-based classifier and compared with the MFCC-based features.

Chapter 4 introduces some of the problems with the technique developed in Chapter 3 and proposes the use of the admissible wavelet packets for feature extraction. The recognition results obtained by using the admissible wavelet packets are discussed in this chapter. Further, a 24-band filter structure based on the admissible wavelet packet is also proposed, with a bandwidth closely following the Mel scale.

Chapter 5 begins by introducing the problems of noisy speech recognition. Modified admissible wavelet packet-based features are suggested for additive white Gaussian noise and their performance is evaluated for different signal-to-noise ratios. A new pre-processing stage based on wavelet denoising is finally proposed for the extraction of robust features.

Chapter 6 deals with implementation of a continuous speech recognition system for words from the TIMIT database, for features derived with a 24-band filter structure using the admissible wavelet packets developed in Chapter 4. Further, the recognition performance of the above features and MFCC features are studied in the presence of different levels of white Gaussian noise of zero mean. Finally the improvement achieved by using the proposed wavelet denoising before the feature extraction is evaluated.

Chapter 7 presents the general conclusions of this thesis and proposes possible improvements and directions of future research.

1.5 Original Contributions

The following novel techniques have been proposed in this thesis.

1. New log-energy based feature have been proposed using discrete wavelet transform and admissible wavelet packets for the task of speech recognition. These features are found to be shift invariant and have little variations with the change in speaker. These features are found superior to the wavelet coefficient based features.

2. Further, admissible wavelet packets have been used to design a 24-band filter structure that closely follows the Mel scale. First 13 discrete cosine transform coefficients obtained from the log-energy in each frequency band have been proposed as features. These features gave superior performance as compared to the MFCC features when the frame duration was taken as 32ms.

3. In order to reduce the effect of noise on the features extracted, a new wavelet based pre-processing technique has been proposed. This is based on the wavelet denoising of the input speech before the feature extraction phase. Both hard and soft thresholding have been applied to evaluate the improvement in the recognition performance for phoneme as well as word recognition task.

1.6 References

[1] K. W. Grant and P. F. Seitz, "The use of visible speech cues for improving auditory detection of spoken sentences", *Journal of Acoustical Society of America* vol. 108, no. 3, pp.1197-1208, Part 1, September 2000.

[2] C. V. Neti and A. Senior, "Audio-visual speaker recognition for video broadcast news", *DARPA HUB4 Workshop*, Washington D.C., March 1999.

[3] G. Potamianos and A. Potamianos, "Speaker adaptation for audio-visual speech recognition", *Proceedings of EUROSPEECH*, Budapest, 1999, vol. 3, pp. 1291-1294.

[4] J. W. Picone, "Signal modeling techniques in speech recognition", *Proceedings of the IEEE*, vol. 81, no. 9, pp. 1215-1247, September 1993.

[5] J. Gauvain and L. Lamel, "Large vocabulary continuous speech recognition: Advances and applications", *Proceedings of the IEEE*, vol. 88, no. 8, pp. 1181-1200, August 2000.

[6] J. Picone, W. J. Ebel and N. Deshmukh, "Automatic speech understanding: The next generation", *Digital Signal Processing Technology*, vol. CR57, pp.101-114, 1995.

[7] L. R. Rabiner and R. W. Schafer, *Digital processing of speech signals*, Prentice-Hall, Englewood Cliffs, New Jersey, USA, 1978.

[8] S. Young, "A review of large vocabulary continuous speech recognition", *IEEE Signal Processing Magazine*, September, pp. 45-57, 1996.

[9] S. Balakrishnama, A. Ganapathiraju and J. Picone, "Linear discriminant analysis for signal processing problems", *Proceedings of IEEE Southeastcon*, Lexington, Kentucky, USA, March 1999, pp. 36-39.

[10] C. J. Long, *Phoneme discrimination using non-linear wavelets methods*, Ph.D. thesis, Loughborough University, Department of Electronic and Electrical Engineering, February 1999.

[11] C. J. Long and S. Datta, "Discriminant wavelet basis construction for speech recognition", *Proceedings of 5th International Conference of Spoken Language Processing*, Sydney, Australia 30th November to 4th December 1998, vol. 3, pp. 1047-1049.

[12] T. Koizumi, M. Mori, S. Taniguchi and M. Maruya, "Recurrent neural network for phoneme recognition", *Proceedings of 4th International Conference on Spoken Language Processing*, Philadelphia, USA October 3-6 1996, vol. 1, pp. 326-328.

[13] C. J. Long and S. Datta, "Wavelet based feature extraction for phoneme recognition", *Proceedings of 4th International Conference of Spoken Language Processing*, Philadelphia, USA October 3-6 1996, vol. 1, pp. 264-267.

[14] E. Lukasiak, "Wavelet packets based features selection for voiceless plosives classification", *Proceedings of International Conference on Acoustic, Speech and Signal Processing*, 2000, vol. 2, pp. 689-692, 2000.

[15] A. Waibel, *Neural network approach for speech recognition*, Advances in speech processing, Edited by S. Furui and M. Sondhi, Marcel Dekker, 1991, pp. 555-595.

[16] A. Waibel, T Hanazawa, G. Hinton, K. Shikano and K. J. Lang, "Phoneme recognition using time delay neural network", *IEEE Transactions on Acoustic Speech and Signal Processing*, vol. 37, no. 3, pp. 328-339, March 1989.

[17] P. C. Woodland, *Spoken alphabet recognition using multilayer perceptron*, Neural Networks for speech Vision and Natural Language by Chapman and Hall, 1992, pp. 135-147.

[18] M. J. F. Gales and S. J. Young, "Robust continuous speech recognition using parallel model combination", *IEEE Transactions on Speech and Audio Processing*, vol. 4 no. 5, pp. 352-359, September 1996.

[19] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition", *Proceedings of IEEE*, vol. 77, no. 2, pp. 257-285, February 1989.

[20] R. Rosenfeld, "Two decades of statistical language modeling: where do we go from here?" *Proceedings of the IEEE*, vol. 88, no. 8, pp. 1270-1278, August 2000.

[21] N. Deshmukh, A. Ganapathiraju and J. Picone, "Hierarchical search for large vocabulary conversational speech recognition", *IEEE Signal Processing Magazine*, September 1999, pp. 84-107.

[22] H. Ney and S. Ortmanns, "Dynamic programming search for continuous speech recognition", *IEEE Signal Processing Magazine*, pp. 64-83, September 1999.

[23] H. Sakoe and S. Chiba, "A dynamic programming algorithm optimization for spoken word recognition", *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 26, no. 27, pp.43-49, 1978.

[24] H. Hermansky and N. Morgan, "RASTA processing of speech", *IEEE Transactions on Speech and Audio Processing*, vol. 2, no. 4, pp. 578-589, October 1994.

[25] K. H. Yuo and H. C. Wang, "Robust features for noisy speech recognition based on temporal trajectory filtering of short-time autocorrelation sequences", *Speech Communication,* vol. 28, pp. 13-24, 1999.

[26] D. S. Kim, S.Y. Lee R.M. Kil and X Zhu, "Auditory model for speech recognition in real world noisy environments", *Electronics Letters*, vol. 33, no. 1, pp. 12-13, 2nd January 1997.

[27] A. Acero and R. M. Stern, "Environmental robustness in automatic speech recognition", *Proceedings of International Conference on Acoustic, Speech and Signal Processing*, 1990, pp. 849-852, 1990.

[28] D. Y. Kim and C. K. Un, "Probabilistic vector mapping with trajectory information for noise-robust speech recognition", *Electronics Letters*, vol. 32, no. 17, pp. 1550-1551, 15th August 1996.

[29] L. Neumeyer and M. Weintraub, "Probabilistic optimum filtering for robust speech recognition", *Proceedings of International Conference on Acoustic, Speech and Signal Processing*, 1994, pp. I-417, I-420, 1994.

[30] B. T. Tan, M. Fu, A. Spray and P. Dermody, "The use of wavelet transform for phoneme recognition", *Proceedings of 4th International Conference of Spoken Language Processing*, Philadelphia, USA October 3-6 1996, vol. 4, pp. 2431-2434.

[31] Z. Tufekci and J. N. Gowdy, "Feature extraction using discrete wavelet transform for speech recognition", *Proceedings of IEEE Southeastcon*, Nashville, Tennessee, April 7- 9, 2000, pp. 116-123.

[32] H. Wassner and G. Chollet, "New cepstral representation using wavelet analysis and spectral transformations for robust speech recognition", *Proceedings of 4th International Conference on Spoken Language Processing*, Philadelphia, USA, vol. 1, pp. 260-263, October 1996.

[33] S. Chang, Y. Kwon and S. Yang, "Speech feature extracted from adaptive wavelet for speech recognition", *Electronics Letters*, vol. 34, no. 23, 12th November 1998, pp. 2211-2213.

[34] D. L. Donoho and I. M. Johnston, "De-noising by soft-thresholding," *IEEE Transactions Information Theory*, vol. 41, no. 3, pp.613-627, May 1995.

CHAPTER 2

FEATURE EXTRACTION

2.1 Introduction

In this chapter various existing techniques that have been used for feature extraction in speech recognition applications are discussed. The shortcomings of the newly proposed wavelet-based techniques are also elaborated at the end of this chapter. The acoustic front end as explained in Chapter 1 carries out the feature extraction process. To identify features suitable for speech recognition it is essential to understand the language phonetics. A language can be described by a set of abstract linguistic units called phonemes (basic sounds). The phonemes are the smallest meaningful unit in the phonology of a language. These phonemes combine together to form different words of the language. There are differences between 'phonemes' and phonetic elements. A 'phoneme' is a basic unit as linguistically defined. It is difficult to represent them in acoustic space as they hardly show one-to-one mapping, i.e. a phoneme may be represented by more than in phonetic elements. In this thesis they are all loosely termed as 'phonemes'. One of the possibilities for a speech recognition system is to identify the different phonemes for a given speech input and then later combine them to form valid words. Thus to implement a speech recognition system, the phoneme boundaries are to be detected. This process is known as segmentation. Recently a speech recognition system based on syllable identification has also been reported [1], however phoneme based recognisers are still most commonly used. The work carried out in this thesis is based on phoneme recognition, therefore to

identify phonemes it is necessary to understand about its type and production mechanism. The next section gives a brief introduction into the different types of phonemes.

2.2 Phoneme types

Sound is produced when air from the lungs is pumped out and passes though the trachea into the larynx where the vocal cords may vibrate due to the airflow. The air is then modulated by the pharynx and passed to the mouth and/or nasal cavity. The type of sound radiated depends upon the position of the various articulators (such as jaw, tongue, velum, lips etc) and the position of constriction in the vocal cord.

Vowels: The vowels are generally long in duration compared to consonant sounds and are especially well defined due to the open vocal tract. They can be easily and reliably recognised. Vowels can be classified according to the tongue position in three main categories. Front vowels are /iy/ (as in 'beet'), /ih/ (as in 'it'), /ae/ (as in 'at') and /eh/ (as in 'met'). Mid-position vowels are /aa/ (as in 'father'), /ax/ (as in 'all') and /ah/ (as in 'up'). Back vowels are /ux/ (as in 'foot') /uw/ (as in 'boot') and /o/ (as in 'obey').

Diphthongs: It is a gliding monosyllabic speech sound that starts at the near articulatory position of a vowel and moves to or towards the position of another vowel. The diphthongs in English are /ay/ (buy) /aw/ (down), /ey/ (bait), /oy/ (boy), /o/ (boat) and /ju/ (you) but, there is disagreement as to the total number of diphthongs.

Semi-vowels: These have vowel-like characteristics and the sound produced depends on the context in which they occur. The examples of the semi-vowel are /w/, /l/, /r/ and /y/.

Nasal Consonants: These are produced by the nasal excitation with the vocal tract completely constricted at some part of the oral passage. The velum is lowered so that the air flows through the nasal tract and the sound is radiated

through the nostrils. The nasal consonants are /m/ (lip constriction) /n/ (constriction behind the teeth) /ng/ (constriction just forward of velum).

Unvoiced Fricatives: Exciting the vocal tract by a steady airflow that becomes turbulent in the region of constriction of the vocal tract produces the unvoiced fricative. The constriction divides the vocal tract in two cavities and the sound is radiated from the lips. The location of the constriction decides the sound produced. The unvoiced fricatives are /f/ (constriction near the lips), /th/ (constriction near the teeth), /s/ (constriction in the middle of the oral tract) and /sh/ (constriction at the back of the oral tract).

Voiced Fricative: The voiced fricative are /v/, /dh/, /z/ and /zh/. These have the similar position of constriction with a difference that the vocal cord vibrates in this case.

Voiced and Unvoiced Stops: For the production of a stop or plosive sound, there is a complete constriction in the vocal tract allowing the pressure to build up, then it is suddenly released. The stop sounds are dynamic in nature and their properties are influenced by the vowel that follows. The voiced stops are /b/ (constriction at the lips), /d/ (constriction at the back of the teeth) and /g/ (constriction near the velum). The corresponding unvoiced stops with the similar constriction position are /p/, /t/ and /k/. The only difference is that in this case the vocal cords do not vibrate.

2.3 Overview of feature extraction techniques

To identify a phoneme some of its characteristics (features) in time/frequency or in some other domain must be known. The basic requirement of a feature extraction system is to extract a set of features for each of these phonemes. A feature can be defined as a minimal unit, which distinguishes maximally close phonemes. The feature vector extracted should possess the following properties:

1. Vary widely from class to class.
2. Stable over a long period of time.

3. Can be easily computed from the input speech samples.
4. Should be small in dimension.
5. Should be insensitive to the irrelevant variation in the speech background noise and channel distortion.
6. Should not have correlation with other features.

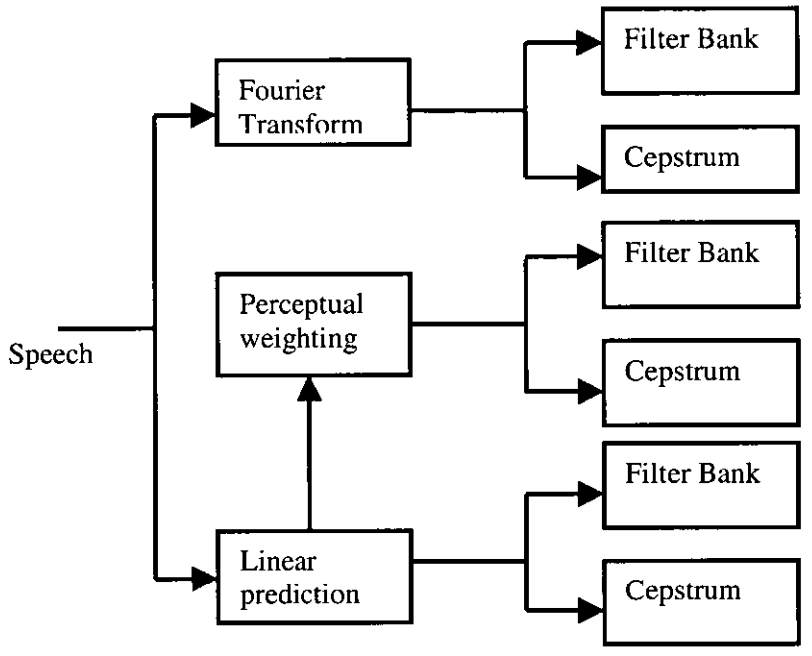


Figure 2.1: Block diagram of different feature extraction techniques.

Figure 2.1 shows the most commonly used techniques for feature extraction. The following techniques can be used directly/indirectly for extraction of features from speech signal:

Fourier analysis: - To have a frequency domain description of the speech signals, the Fast Fourier Transform (FFT) is used. The FFT coefficients can be used to extract the formant frequencies F_1, F_2, F_3 and F_4, which can be used as a set of features to identify the phonemes. The formant frequencies are the peak resonant frequencies of the vocal tract and F_1 is the lowest resonant frequency. The FFT analysis assumes that the signal is stationary; however, this is not strictly true for speech signals. Therefore, to force the above condition, the signal is analysed for smaller sub-intervals (called frames). This frame duration can be much shorter than the duration of the phoneme and is typically of 10-20ms duration. Due to the inertia of the articulators the speech signal is usually

stationary during this duration (with some exceptions). To evaluate the FFT coefficients in this frame duration a windowed version of the FFT is used, known as the Short Time Fourier Transform (STFT), which identifies the frequency components of the signal during the frame duration. If the windowed duration is small then the frequency resolution is poor (high time resolution), while a longer window gives better frequency resolution (poor time resolution). This is due to Heisenberg's Uncertainty Principle, which says that $\Delta t.\Delta f \leq 1/4\pi$, where Δt and Δf is time and the frequency resolution respectively. Thus by using the STFT, high resolution in either frequency or time domain can be achieved for a given window size. Figure 2.2 shows the tiling of the time-frequency axis by the STFT. The FFT has also been used to calculate power at the output of a filter bank, which is then used as a feature [2]. It has also been used for the calculation of Mel frequency cepstral coefficients to be discussed in this section later.

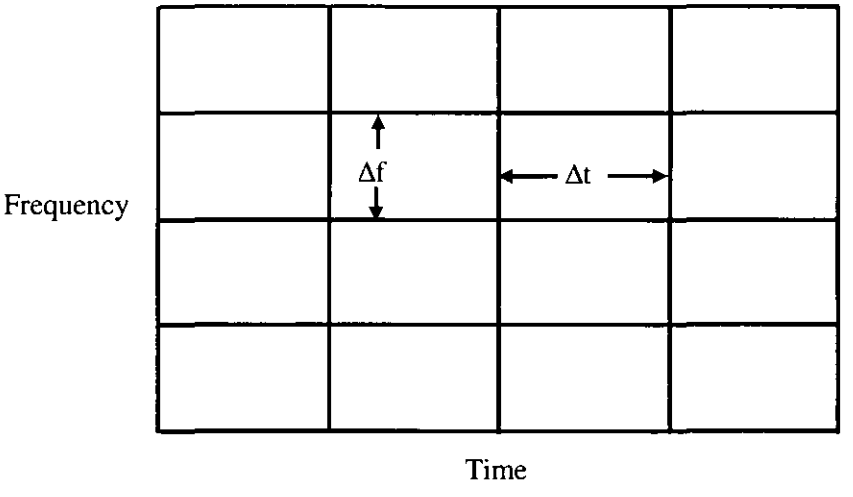


Figure 2.2: Uniform tiling of time-frequency plane by the STFT.

There are a number of window functions possible such as rectangular, triangular, Hanning, Hamming, Blackman etc. Hamming window is a good choice because it has the smallest side lobe magnitude for a given main lobe width [3]. This is because the STFT does not have compact support. The process of frame-based analysis of phonemes is shown in Figure 2.3. Although the introduction of the window reduces the effects of side-lobes, it increases the computation because the data in the overlapping region is processed twice.

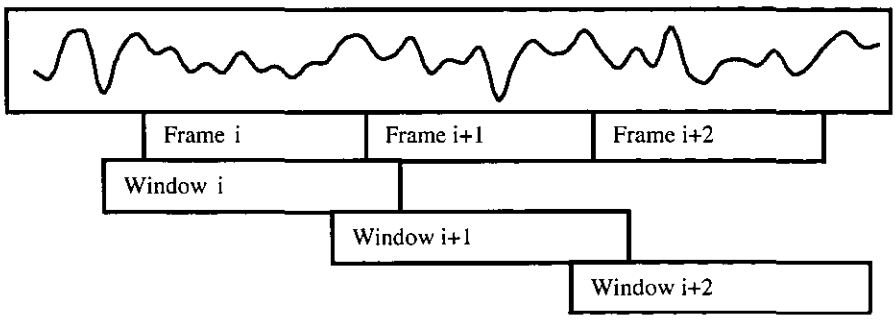


Figure 2.3: Window and frame structure used for analysing phonemes by using short time Fourier transform.

Another concept of time frequency analysis is Wigner-Ville Distribution (WVD) which has been borrowed from physics. It was originally invented by E. P. Wigner and later adopted in signal processing by J. Ville. The WVD of a signal **x(t)** is given by

$$\mathbf{W(t,f)} = \int_{-\infty}^{+\infty} \mathbf{x(t+\frac{\tau}{2}).x^{*}(t-\frac{\tau}{2})e^{-2\pi f\tau}d\tau} \qquad (2.1)$$

It is the Fourier transform of the signal's autocorrelation function with respect to the delay variable [4]. It can also be thought of as a short-time Fourier transform (STFT) where the windowing function is a time-scaled, time-reversed copy of the original signal. The WVD is a tool for time-frequency analysis because it generally has much better resolution than the short-time Fourier transform (STFT) method.

The WVD has two notable limitations:

- Cross-term calculations may give rise to negative energy.
- Aliasing effect may distort the spectrum such that a high frequency component may be mis-identified as a low frequency component.

These problems have been extensively studied; however, people are still on the lookout for better smoothing functions and alias-free distributions. Due to the above reasons WVD has not been tried since energy is an important feature for speech recognition task.

Linear predictive analysis: - The vocal tract can be modelled as a time varying all-pole filter with its excitation coming from the glottal pulses. The time

varying nature of the filter is essentially because different sounds have different constriction and articulatory positions. Thus, this model is used to extract features that could help in the classification of the speech. The linear prediction technique is used to derive the filter coefficients by minimising the mean square error between the input and the estimated samples of speech. 12 coefficients are usually sufficient to predict the speech. For a signal s_n with linear predictive coefficients a_i using p taps in the prediction filter the estimated signal $\tilde{s}_n$ is given by:

$$\tilde{s}_n = \sum_{i=1}^{p} s_{n-i} a_i \qquad (2.2)$$

These coefficients are extracted using the auto-correlation method or the covariance method [5]. The auto-correlation method is computationally efficient and gives a stable filter that is very important in the analysis-by-synthesis method of speech coding.

After linear predictive analysis the cepstral coefficients can be extracted or further filtering can be applied to calculate the power in each band to be used as features. Perceptual weighting has also been applied after the linear predictive analysis to shape the spectrum similar to that of human ear response.

Cepstral coefficients: - These coefficients are derived from the linear predictive coefficients and have an advantage of being independent (uncorrelated). Cepstral coefficients c_i can be calculated by the following equations:

$$c_1 = a_1$$

$$c_i = a_i + \sum_{k=1}^{i-1} (1 - k/i).a_k .c_{i-k} \quad 1 \le i \le p \qquad (2.3)$$

The cepstral coefficients extracted can be used as such (uniform weighting) or can be weighted by using some weighting function w_i [5]. A few examples of the weighting function are:

$$w_i = 1 \qquad \text{Uniform weighting} \qquad (2.4)$$

$$w_i = 1 + 0.5.p.\sin(\pi.i/p) \qquad 1 \le i \le p \qquad (2.5)$$

and the weighted cepstral coefficients are given as:

$$c_i' = c_i.w_i \qquad (2.6)$$

The cepstral coefficients can also be extracted by first taking the FFT of the speech samples and then taking the log magnitude of these samples. Psychophysical studies have shown that the human perception of the frequency contents of the sound does not follow a linear scale. For each tone with a frequency of f Hz a subjective pitch is measured on a scale called the "Mel" scale [6]. Mathematically the Mel scale is given by:

$$\mathbf{Mel(f) = 2595\log_{10}(1 + f/700)} \text{ where f is in Hz} \qquad (2.7)$$

It has been found that the subjective pitch is linear on the logarithmic frequency beyond 1000Hz. To have a similar response to that of human ear, filters have been designed to follow the Mel scale. For speech sampled at 16kHz a bank of 24 filters, each having a constant bandwidth of 100Hz below 1000Hz and then having a logarithmic increase up to 8000Hz, has been designed. The central frequency and the bandwidth of these filters are shown in Table 2.1.

The Mel filters have triangular profiles with overlapping bands, as shown in Figure 2.4. These filters must be normalised so that they do not increase energy in the higher frequency bands. The log of energy at the output of each filter is calculated and a Discrete Cosine Transform (DCT) is applied to give the Mel-frequency cepstral coefficients (MFCC). The MFCC is defined as the short-term spectral envelope of a speech signal after filtering. The lower order terms of the cepstral coefficients give the idea of smoothness of the spectrum and correspond mainly to the vocal tract response rather than to the fine spectral structures. These fine structures produce the artefacts that reduce the spectral matching. These artefacts can be minimised by truncating the infinite series to a finite value. Various techniques have been used to achieve this, such as linear predictive modelling, using raised sine liftering, Gaussian liftering and perceptually based linear predictive (PLP) analysis [7]. A PLP based root power sum (RPS) front end is reported to have better recognition rate [8].

Table 2.1: 24-band Mel scaled filters and their corresponding frequency bands

Filter number	Central frequency (Hz)	Bandwidth (Hz)
1	100	100
2	200	100
3	300	100
4	400	100
5	500	100
6	600	100
7	700	100
8	800	100
9	900	100
10	1000	124
11	1149	160
12	1320	184
13	1516	211
14	1741	242
15	2000	278
16	2297	320
17	2639	367
18	3031	422
19	3482	484
20	4000	556
21	4595	639
22	5278	734
23	6063	843
24	6954	969

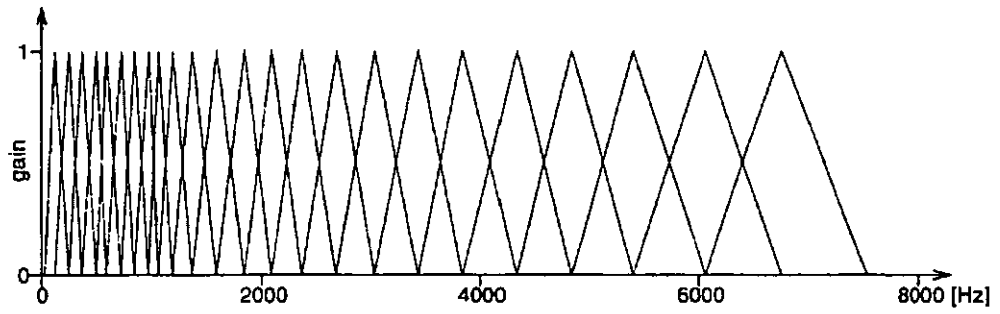


Figure 2.4: Frequency response of the Mel scale filter bank (not normalised).

2.4 Temporal Features

With the availability of FFT and linear predictive analysis, spectral measures of the overlapped windowed frames were used as features. Mostly the features extracted were based on the LPC or the cepstral coefficients. These analyses assume that the signal is stationary during the windowed duration. However, the temporal changes in the spectrum of the speech signal also play an important role in its perception. To have a robust speech recogniser these temporal changes must also be included in the set of feature vectors. Delta coefficients or the difference coefficients that measure the changes in the coefficients over time have been proposed for this purpose [9], [10].

Harte [11] proposed the calculation of the dynamic features from the cepstral coefficients by using the DCT. He first calculated the cepstral coefficients by passing the signal through a bank of 21 Mel scale band-pass filters, squaring, averaging, taking the log and then taking the DCT of the stacked values. The RASTA (relative spectral) method adds an extra pole to a finite impulse response band-pass filter used in delta processing based on knowledge of human hearing perception, in which further improvement in the recognition performance under noisy environment can be achieved [12]. Shen [10] has shown that by using a post-processing unit after the RASTA the recognition performance can be further improved in a noisy environment; however, the performance degrades in the absence of noise. The post-processing unit actually minimises the classification error by using a minimum classification error algorithm.

2.5 Features for noisy speech

Some of the speech recognition application areas may have to contend with a noisy environment, for example in factories, cars or aeroplanes. This calls for processing techniques that should be little affected by background noise and therefore on the performance of the recogniser. The human auditory system is robust to background noise, so to have a speech recogniser with robust performance, the human auditory system has been studied and many models have been proposed. Kim [13] proposed a simple zero crossing with peak amplitude (ZCPA) model as a robust front end for a speech recognition system in a noisy environment [13]. It consists of a bank of band-pass cochlear filters and a non-linear stage at the output of each cochlear filter. Kim [14] has studied the performance of the ZCPA model and suggested the cepstral measures based on this model. The results of linear predictive coefficient cepstrum, MFCC, ZCPA spectrum and ZCPA cepstrum (ZCPAC) under noisy conditions have been studied and higher recognition with ZCPAC is reported with a MLP as well as discrete HMM classifier.

RASTA [12] processing improves the performance of a speech recognition system in the presence of additive noise and different channel distortion conditions. However, RASTA processing may not always give an optimal solution for noisy speech recognition [15]. A cepstral mean subtraction has been used to remove a constant bias produced by the channel or background noise. This is achieved by subtracting the short-time average of the cepstral vector from the current cepstral vector [16]. A subtraction based on the power spectrum has also been applied for robust speech recognition [17].

The Cepstral normalisation has been carried out by various techniques in order to reduce the effect of noise and channel distortion on the extracted features [18]. The Probabilistic Optimum Filtering (POF) [19], [20] approach is based on the assumption that the clean speech cepstrum can be derived from the noisy cepstrum by using a linear transformation. In order to train the POF filter, stereo pairs of noisy and clean cepstral vectors are used at different levels of signal-to-noise ratio (SNR). One side of the auto-correlation [21] of the signal is selected and high pass filtering is also applied, which removes the slowly varying

component. This causes the noise to be removed, since it is assumed to be stationary, leaving the clean one-sided auto-correlation of the speech.

2.6 Wavelet transform for feature extraction

The STFT is at the heart of all the most commonly used feature extraction techniques. However, as explained earlier this transform assumes the signal to be stationary during the frame duration. In general this assumption is true except for the stop phonemes. Furthermore, it has a fixed time-frequency resolution, as shown in Figure 2.2. All these limitations arise because although the basis functions of the STFT are localised in frequency they are not localised in time. The wavelet transform is a time-frequency transform that can analyse non-stationary signals as well as stationary signals with multi-resolution capability. It uses basis functions called 'wavelets', which are localised both in time as well as frequency [22]. Due to these capabilities there has been growing interest in the use of wavelet for signal and image processing. The detail theory of the wavelet will be discussed in Chapter 3 and Chapter 4.

A discrete wavelet transform has been used earlier for feature extraction, [23], [24], [25]. In [23] the wavelet coefficients were used as features by first ranking them in terms of their energy. The top few wavelet coefficients having high energy were selected as features. However, as the discrete wavelet transform is shift variant, these features are not very reliable. With a small shift in the signal the wavelet coefficients would change, thereby resulting in a change in the feature vector. The problem becomes non-existent if the shift is an integral multiple of the sampling time; however, this cannot be guaranteed in a practical situation. In order to reduce the problem due to shift, the signal can be over-sampled or a different shifted version of the same signal can be given for feature extraction. Both of these solutions increase the computation load.

The discrete wavelet transform has also been used in the place of DCT for the MFCC features and has shown improved performance both in the case of clean and noisy speech recognition [26], [27]. Wavelet packets, which are more general form of the wavelet transform, have also been used for the feature extraction for speech recognition purpose [23], [28], [29], [30], [31]. The wavelet

packet approach is usually based on the best basis selection criterion, which suffers from the problem of shift variance [22]. In this work new sets of features based on the discrete wavelet transform and the wavelet packet have been proposed that are shift invariant and speaker independent. The next chapter gives a brief introduction to the wavelet transform and proposes new features for the phoneme recognition task.

2.7 References

[1] R. J. Jones, S. Downey and J. S. Mason, "Continuous speech recognition using syllables", *Proceedings of Eurospeech*, 1997, Greece, pp. 1171-1174, 1997.

[2] T. Koizumi, M. Mori, S. Taniguchi and M. Maruya, "Recurrent neural network for phoneme recognition", *Proceedings of 4th International Conference on Spoken Language Processing*, Philadelphia, USA, October 3-6 1996, vol. 1, pp. 326-328.

[3] B. P. Lathi, "*Signal processing and linear systems*", Berkeley Cambridge Press, California, USA, 1998.

[4] W. Mecklenbrauker and F. Hlawalsch, *The Wigner distribution: Theory and applications in signal processing*, Elsevier Science 1997.

[5] L. Rabiner and B H. Juang, *Fundamental of speech recognition*, Prentice Hall 1993.

[6] D. O'Shaughnessy, *Speech Communication: Human and Machine*, Addison-Wesley, 1987.

[7] H. Hermansky, "Perceptual linear predictive (PLP) analysis for speech", *Journal of Acoustical Society of America*, vol. 87, Issue 4, pp. 1397-1837, April 1990.

[8] J. C. Junqua, H. Wakita and H. Hermansky, "Evaluation and optimization of perceptually-based ASR front end", *IEEE Transactions on Speech and Audio Processing*, vol. 1, no. 1, pp. 39-48, January 1993.

[9] S. Furui, "Speaker independent isolated word recognition using dynamic features of the speech spectrum", *IEEE Transactions Acoustics Speech and Signal Processing*, vol. 34, no. 1, pp. 52-59, February 1986.

[10] J. Shen, "Discriminative temporal feature extraction for robust speech recognition", *Electronics Letters*, vol. 33, no. 19, pp. 1598-1600, 11th September 1997.

[11] N. Harte, S. Vaseghi and B. Milner, "Dynamic features for the segmental speech recognition", *Proceedings of 4th International Conference on Spoken Language Processing*, Philadelphia, USA October 3-6 1996, vol. 2, pp. 933-936.

[12] H. Hermansky and N. Morgan, "RASTA processing of speech", *IEEE Transactions On Speech and Audio Processing*, vol. 2, no. 4, pp. 578-589, October1994.

[13] D. S. Kim, J. H. Jeong and S. Y. Lee, "Feature extraction based on zero crossings and peak amplitudes for robust speech recognition in noisy environment", *Proceedings International Conference on Acoustic, Speech and Signal Processing*, Atlanta, USA, May 1996, pp. 61-64.

[14] D. S. Kim, S.Y. Lee R.M. Kil and X Zhu, "Auditory model for speech recognition in real world noisy environments", *Electronics Letters*, vol. 33, no. 1, pp. 12-13, 2nd January 1997.

[15] H. Y. Jung and S. Y. Lee, " On the temporal decorrelation of features parameters for noise robust speech recognition", *IEEE Transactions on Speech and Audio Processing*, vol. 8, no. 4, pp. 407-416, July 2000.

[16] M. G. Rahim, B. H. Juang, W. Chou and E. Buhrke, "Signal conditioning techniques for robust speech recognition", *IEEE Signal Processing Letters*, vol. 3, no. 4, pp. 107-109, April 1996.

[17] J. Xu and G. Wei, "Noise-robust speech recognition based on difference of power spectrum", *Electronics Letters*, vol. 36, no. 14, pp. 1247-1248, July 2000.

[18] A. Acero and R. M. Stern, "Environmental robustness in automatic speech recognition", *Proceedings of International Conference on Acoustic, Speech and Signal Processing*, 1990, pp. 849-852, 1990.

[19] D. Y. Kim and C. K. Un, "Probabilistic vector mapping with trajectory information for noise-robust speech recognition", *Electronics Letters*, vol. 32, no. 17, pp. 1550-1551, 15th August 1996.

[20] L. Neumeyer and M. Weintraub, "Probabilistic optimum filtering for robust speech recognition", *Proceedings of International Conference on Acoustic, Speech and Signal Processing*,1994, pp. I-417, I-420, 1994.

[21] K. H. Yuo and H. C. Wang, "Robust features for noisy speech recognition based on temporal trajectory filtering of short-time autocorrelation sequences", *Speech Communication* vol. 28, pp. 13-24, 1999.

[22] S. Mallat, *Wavelet tour signal of signal processing*, Academic press, 1999.

[23] C. J. Long, *Phoneme discrimination using non-linear wavelets methods*, Ph.D. thesis, Loughborough University, Dept. of Electronic and Electrical Engineering, February 1999.

[24] B. T. Tan, M. Fu, A. Spray and P. Dermody, "The use of wavelet transform in phoneme recognition", *Proceedings of 4th International Conference on Spoken Language Processing*, Philadelphia, USA October 3-6, 1996, vol. 4, pp. 2431-2434.

[25] H. Wassner and G. Chollet, "New cepstral representation using wavelet analysis and spectral transformations for robust speech recognition," *Proceedings of 4th International Conference on Spoken Language Processing*, Philadelphia, USA, October 3-6, 1996, vol. 1, pp. 260-263.

[26] N. Gowdy and Z. Tufekci, "Mel-scaled discrete wavelet coefficients for speech recognition", *Proceedings of International Conference on Acoustic, Speech and Signal Processing*, 2000, vol. 3, pp. 1351-1354.

[27] Z. Tufekci and J. N. Gowdy, "Feature extraction using discrete wavelet transform for speech recognition", *Proceedings of IEEE Southeastcon*, Nashville, Tennessee, April 7- 9, 2000, pp. 116-123.

[28] S. Chang, Y. Kwon and S. Yang, "Speech feature extracted from adaptive wavelet for speech recognition", *Electronics Letters* vol. 34, no. 23, pp. 2211-2213, 1998.

[29] C. J. Long and S. Datta, "Wavelet based feature extraction for phoneme recognition", *Proceedings of 4th International Conference on Spoken*

Language Processing, Philadelphia, USA, October 3-6 1996, vol. 1, pp. 264-267.

[30] C. J Long and S. Datta, "Discriminant wavelet basis construction for speech recognition", *Proceedings of 5th International Conference of Spoken Language Processing*, Sydney, Australia, 1998, vol. 3, pp. 1047-1049.

[31] E. Lukasiak, "Wavelet packet based features selection for voiceless plosives classification", *Proceedings of International Conference on Acoustic, Speech and Signal Processing*, 2000, vol. 2, pp. 689-692.

CHAPTER 3

DISCRETE WAVELET TRANSFORM FOR PHONEME RECOGNITION

In the previous chapter the limitations of using the STFT were discussed and the wavelet transform was proposed to overcome these limitations. In this chapter, first an introduction to the discrete wavelet transform (DWT) is given, then new features for speech recognition are proposed. Experimental results obtained by using these features for the phoneme recognition task are also presented in this chapter. Finally, the results of the proposed novel features using DWT are compared with the results of earlier studies.

3.1 Introduction to Wavelet Transform

The assumption in the STFT calculations, that the signal is stationary, is not strictly valid for the speech signal. Therefore, it is not an ideal choice for the time-frequency analysis of a signal. The obvious solution to this problem is to use an adaptive window size, which allocates more time to the lower frequencies and less time for the higher frequencies. Figure 3.1 shows the tiling of the time-frequency axis by the wavelet transform using variable window sizes and the STFT using a fixed window size. It shows that the time and frequency resolution is fixed for the case of the STFT while it is variable for the wavelet transform.

Thus, the filter must have a higher time resolution the higher the central frequency of the filter (i.e. $\Delta f/f$ = fixed or a constant 'Q' filter). By using this filter, two very short bursts can be separated in time by going up to the higher frequencies in the time-frequency plane. Therefore, this analysis can be used for signals that have short duration high frequency components and long duration, low frequency components (e.g. speech signal). All the filters are a scaled version of a prototype filter.

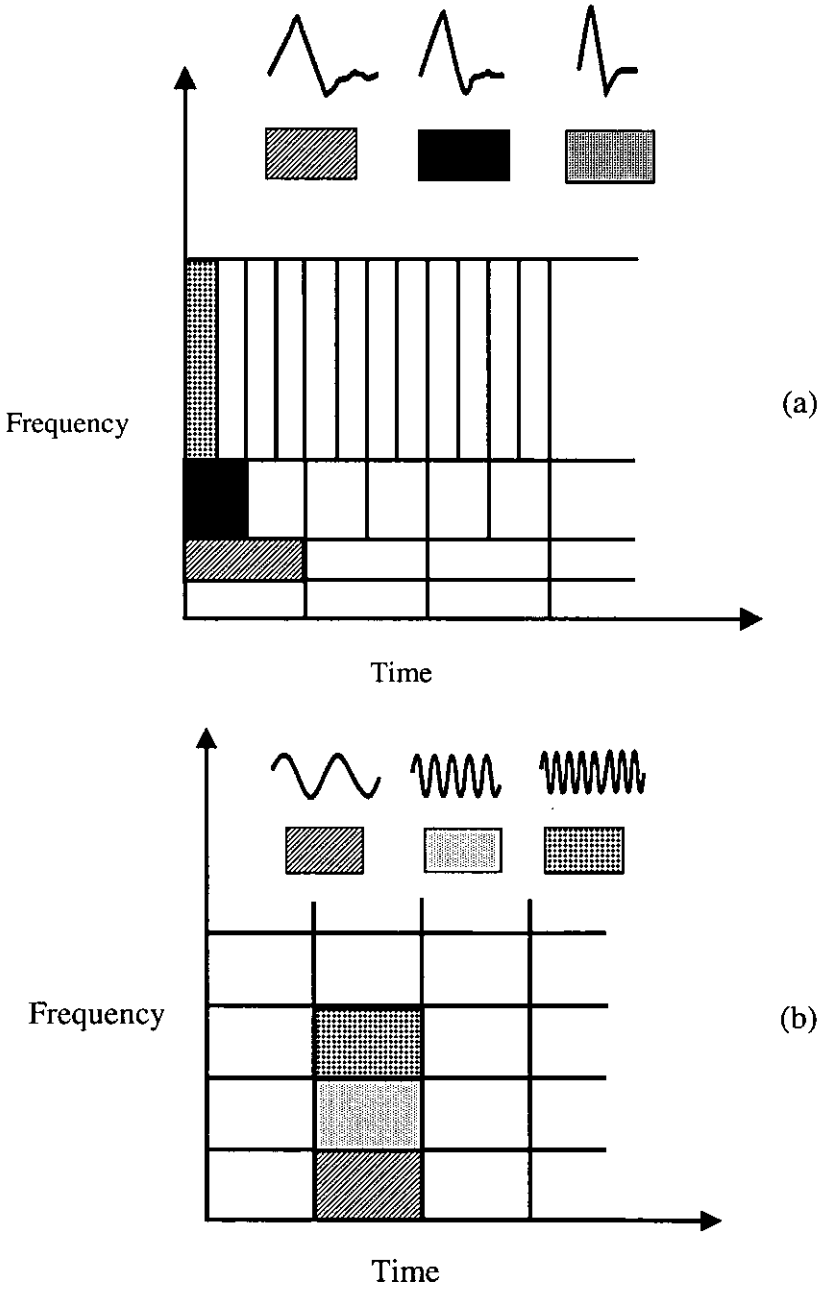


Figure 3.1: Tiling of the time frequency axis using (a) wavelet transform (b) short time Fourier transform.

The prototype filter or the wavelet function $\psi(t)$ is given by the following relation:

$$\psi_{a,\tau}(t) = a^{-1/2}\psi((t-\tau)/a) \tag{3.1}$$

where '**a**' is the scaling factor and 'τ' is the translation factor. The term $a^{-1/2}$ is used for energy normalisation. $\psi \in L^2(\mathbb{R})$ ($L^2(\mathbb{R})$ is finite energy space and $\mathbb{R}$ is a set of real numbers) [1], [2]. The continuous wavelet transform (CWT) of a signal $x(t)$ (where $x \in L^2(\mathbb{R})$) is given by:

$$CWT(a,\tau) = a^{-1/2}\int x(t).\psi^*((t-\tau)/a)dt \tag{3.2}$$

where '*' denotes a complex conjugate. The scaling parameter '**a**' gives the width of the wavelet and 'τ' gives the position. The function of the scaling parameter is similar to that of the scale used in maps. The higher values of '**a**' give a global picture of the signal. The scale changes in the continuous time signal do not alter the resolution because it can be reversed, which is not the case for a digital signal. In the discrete time signal, increasing the scale means down-sampling, thereby reducing the resolution while decreasing the scale will result in the up-sampling with no increase in resolution [2].

Another way of looking at the wavelet is as a basis function, and it can be seen as an inner product of the signal $x(t)$ and $\psi_{a,\tau}(t)$. Thus, it is a measure of similarity between the signal and a basis function called a wavelet. The difference between the Fourier transform and the wavelet transform (WT) is that the basis functions of the WT are localised in time while that of Fourier transform is not. Further, the WT does not have a single set of basis functions like the Fourier transform, which utilises just the Sine and Cosine functions. Instead, it has an infinite set of possible basis functions. The different wavelet families have different trade-offs between the compactness of the basis functions in space and their smoothness.

If ψ has a compact support of size **K**, there are **K** wavelets $\psi_{j,n}$ at each scale 2^j (which may have high amplitudes) whose support includes the isolated

singularity at t_0. To minimise the number of high amplitude coefficients the support size of ψ must be reduced. Higher vanishing moments produce large numbers of coefficients with smaller magnitudes. The support size of a function and the number of vanishing moments are *a priori* independent. However, a constraint is imposed on orthogonal wavelets, implying that if ψ has **p** vanishing moments then its support is at least of size **2p-1**. Daubechies wavelets are optimal in the sense that they have the minimum support size for a given vanishing moment [1].

These two considerations are very important when wavelets are applied for signal compression. If the signal has few isolated singularities and the signal is very regular between singularities, a higher vanishing moment wavelet is desirable. If the density of the singularities increases, it is better to decrease the size of support to achieve higher compression.

Figure 3.2 shows some examples of the wavelet basis functions and their corresponding order (vanishing moments). It can be seen that the smoothness of the 'Daubechies' wavelet basis increases with the increase in the order (number of vanishing moments), thereby making it for processing smooth signals with fewer singularities. However, the Daubechies wavelets are not symmetric, as seen in the Figure 3.2.

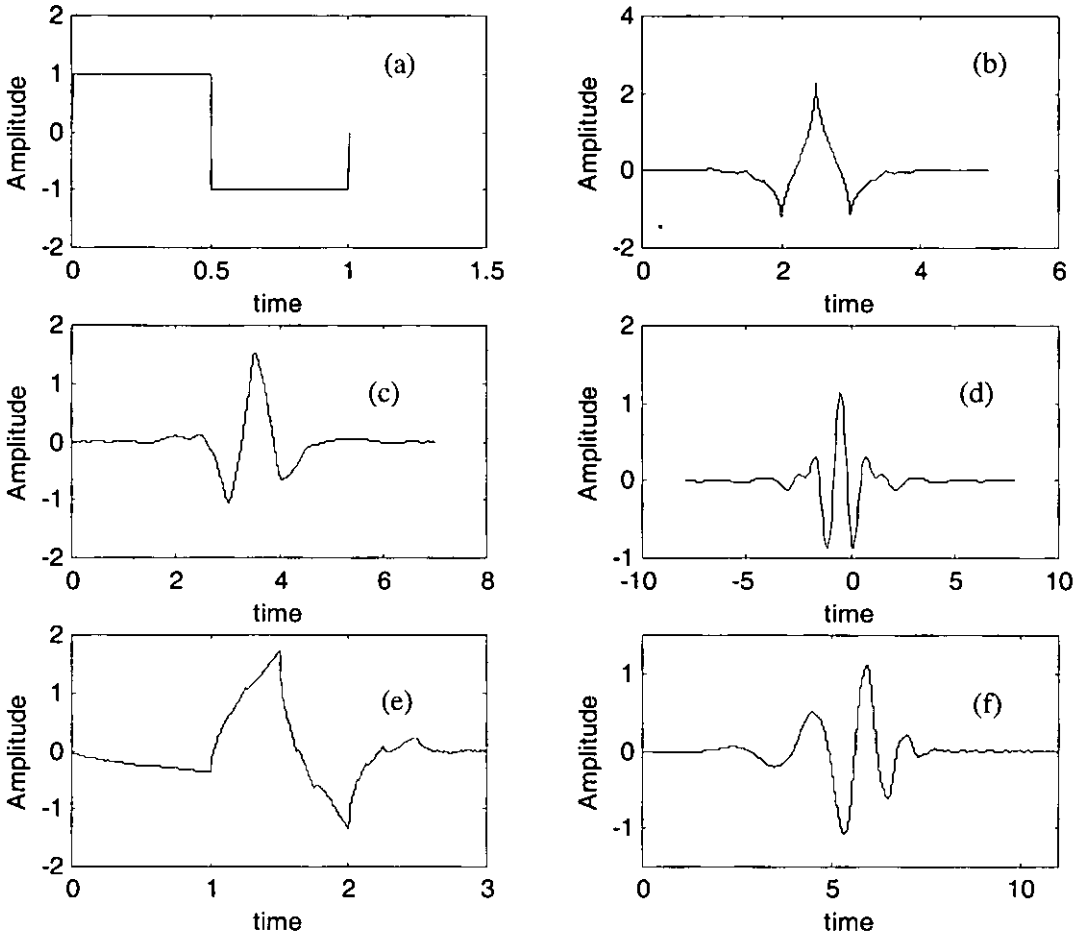


Figure 3.2: Some examples of basic wavelets (a) Haar wavelet (b) Coiflet of order 1 (c) Symmlet of order 4 (d) Meyer (e) Daubechies of order 2 (f) Daubechies of order 6.

Wavelet analysis provides immediate access to information that may not be available by Fourier analysis. The wavelet coefficients obtained give an idea of how close the signal is to a particular basis function. To recover the signal from the wavelet coefficients the function must satisfy the following condition:

$$\int_{-\infty}^{+\infty} \psi(t)dt = 0 \tag{3.3}$$

The recovered signal is the same as the original signal **x(t)** and is given by:

$$x(t) = c \iint_{a>0} a^{-2}.CWT(a,\tau).\psi_{a,\tau}(t)dad\tau \qquad (3.4)$$

where '**c**' is a constant and depends upon $\psi(t)$. The derivation of the equation 3.4 is discussed in APPENDIX A. The CWT has orthogonal decomposition as well as being isometric (i.e. preserves energy). Typically, CWT is over-complete and appropriate sampling is required to eliminate the redundancy. Discrete time scaling can be obtained by choosing the sampling grid as:

$$a = a_o^j, \quad \tau = ka_o^jT \quad a_o > 1 \qquad (3.5)$$

where **j** and **k** are integers. Thus the resulting filter is given by:

$$\psi_{j,k}(t) = a_o^{-j/2}\psi(t.a_o^{-j} - kT) \qquad (3.6)$$

and the wavelet coefficients can be obtained from the following relationship:

$$c_{j,k} = \int x(t).\psi^*_{j,k}(t)dt \qquad (3.7)$$

If the value of a_o is close to unity and **T** is small then the wavelet function is over-complete. The reconstructed signal is the same as the one obtained by the CWT and there is no restriction on the filter $\psi(t)$. However, if the samples are sparse (e.g. $a_o = 2$) then true orthonormal basis will be obtained for only very special values of $\psi(t)$ [1]. In other words, if the redundancy in the signal is large then there is not much restriction on the basis function $\psi(t)$, but if the sampling is critical then the basis function is highly constrained. If the energy of the wavelet coefficients relative to the signal energy lies within two positive frame bounds, **A** and **B**, then the family of wavelet coefficients constitutes a frame.

$$A\|x\|^2 \le \sum_{j,k} \left|\langle x, \psi_{j,k} \rangle\right|^2 \le B\|x\|^2 \qquad 0 < A \le B < \infty \qquad (3.8)$$

where $\langle\ \rangle$ is an inner product operator. If **A=B** then the frames formed are tight and if their value is equal to unity then it results in an orthonormal basis. A frame

of this type guarantees the unique representation of a signal in the $L^2(\mathbb{R})$ space. The ratio of **B/A** is a measure of the transformation redundancy; its value closer to unity gives faster convergence. There are some well-behaved functions that can be used as prototype orthonormal wavelets, but in the case of the STFT it is impossible to have an orthonormal basis function well localised in time and frequency.

For the problem of pattern recognition, the features extracted should have the shift invariance property. This means that the features extracted for the signal **x(t)** and **x(t-u)** should be the same. The STFT and the CWT preserve the features in the case of signal translation, but convolution and down-sampling present in the DWT destroys it unless **u** is an integer multiple of the sampling interval **kT** (Equation 3.5). In order to overcome the problem of translation variance, the dyadic wavelet transform or adaptive sampling is carried out. In the case of the dyadic wavelet the translation parameter 'τ' is not sampled. This transform gives a highly redundant representation of the signal.

An important property of the wavelet transform is its multi-resolution capabilities. The DWT with $a_0 = 2$ of the signal **x[m]** is given by:

$$\mathbf{DWTf[2^j,n]} = \sum_{m=0}^{N-1} x[m].2^{-j/2}\psi^*\left(\frac{m-n}{2^j}\right) \qquad (3.9)$$

where 2^j is the dilation of the orthogonal wavelet with information about the signal at 2^{-j} resolution. As the signal is analysed by a constant Q filter it provides a multi-resolution analysis of the signal over $L^2(\mathbb{R})$. In other words the space $L^2(\mathbb{R})$ is decomposed into a chain of nested subspaces, V_j.

Resolution increases → $\ldots \subset V_2 \subset V_1 \subset V_0 \subset V_{-1} \subset \ldots$ (Containment)

$$\lim_{j \longrightarrow +\infty} V_j = \bigcap_{j \in Z} V_j = [0] \qquad \text{(Uniqueness)}$$

$$\lim_{j \longrightarrow -\infty} V_j = \bigcup_{j \in Z} V_j = L^2(\mathbb{R}) \qquad \text{(Completeness)}$$

$$x(n) \in V_j \Leftrightarrow x(n/2) \in V_{j+1}, \quad j \in Z \qquad \text{(Scaling)}$$

where $\mathbf{Z}$ is a set of integers. The intersection of all the $\mathbf{V_j}$ is empty (Uniqueness), i.e. all the $\mathbf{V_j}$ are unique and at zero resolution ($2^{-j}\ j \to \infty$) all the information of the signal is lost. On the other hand at infinite resolution ($2^{-j}\ j \to -\infty$) the signal approximation converges to the original signal or the union of all the subspaces yields the $\mathbf{L^2(\mathbb{R})}$. Dilating functions in $\mathbf{V_j}$ by 2 enlarges the details by 2 and guarantees that it defines an approximation at a coarser resolution 2^{-j-1}. The space $\mathbf{V_j}$ can be decomposed into a sub-space $\mathbf{V_{j+1}}$ and another detailed space $\mathbf{W_{j+1}}$ that is orthogonal to $\mathbf{V_{j+1}}$. Thus

$$\mathbf{V_j = V_{j+1} \oplus W_{j+1}} \tag{3.10}$$

Hence $\mathbf{W_{j+1}}$ contains all the details necessary to go from one resolution to the next or

$$\mathbf{L^2(\mathbb{R}) = \bigoplus_{j \in Z} W_{j+1}} \tag{3.11}$$

The principle of the multi-resolution analysis (MRA) states that if there exists a scaling function which satisfies certain requirements, i.e. smoothness, continuity and orthonormality, such that:

$$\mathbf{\phi_{j,k}(t) = 2^{-j/2}\phi(t.2^{-j} - k)} \quad \mathbf{j = 0, 1, 2.....\ k = 1, 2, 3....} \tag{3.12}$$

forms an orthonormal basis for $\mathbf{V_j}$, then $\mathbf{W_j}$, its orthonormal complement, is similarly spanned by the orthonormal basis.

$$\mathbf{\psi_{j,k}(t) = 2^{-j/2}\psi(t.2^{-j} - k)} \quad \mathbf{j = 0, 1, 2.....\ k = 1, 2, 3....} \tag{3.13}$$

The approximate sub-space $\mathbf{V_0 \subset V_{-1}}$ can be created by integer translation of the scaling function. Using the scaling property there exists a sequence $\mathbf{g[n]}$ such that:

$$\mathbf{\phi(t) = \sqrt{2} \sum_{n=-\infty}^{+\infty} g[n]\phi(2.t - n)} \tag{3.14}$$

The detailed sub-space $\mathbf{W_0 \subset V_{-1}}$ also satisfies the similar equation:

$$\psi(t) = \sqrt{2} \sum_{n=-\infty}^{+\infty} h[n]\phi(2.t - n) \tag{3.15}$$

where **g[n]** is known as the smoothing or scaling filter and is low-pass in nature, while **h[n]** is known as the detail or wavelet filter and has a high-pass filter characteristics/response. Both of these filters satisfy the perfect reconstruction property. Thus in MRA, the signal in the space $\mathbf{V_j}$ is decomposed into an approximation sub-space $\mathbf{V_{j+1}}$ and a detailed sub-space $\mathbf{W_{j+1}}$. To reconstruct the signal in space $\mathbf{V_j}$, the approximate sub-space $\mathbf{V_{j+1}}$ and detailed sub-space $\mathbf{W_{j+1}}$ is added. A two-level of decomposition is shown in Figure 3.3. For the first level of decomposition the signal in the space $\mathbf{V_j}$ is first passed through a low pass (smoothing) filter and a high pass (wavelet) filter and then down sampled by a factor of 2. The second level of decomposition is applied on the signal in the approximation sub-space ($\mathbf{V_{j+1}}$) obtained by the previous decomposition. The corresponding binary tree structure is also shown in Figure 3.4 where the left child represents the lower frequency and the right child represents the higher frequency band. It is important to note here that this decomposing results into splitting of the approximate sub-space only, while the detailed sub-space obtained after decomposition is left untouched during the subsequent decompositions.

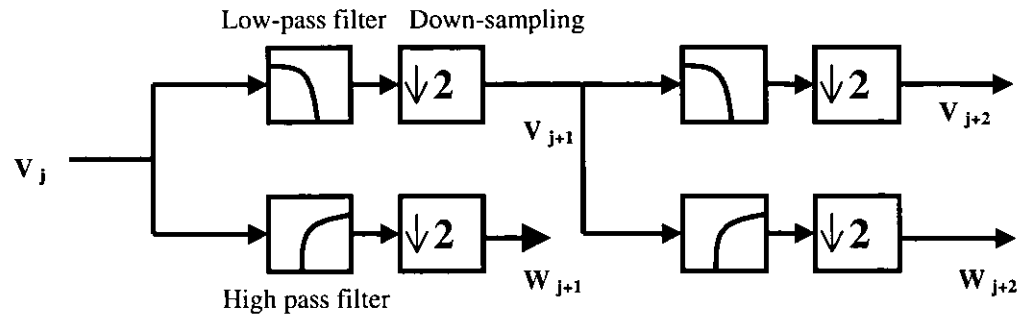


Figure 3.3: Two-level wavelet decomposition achieved by filtering and decimating the original signal

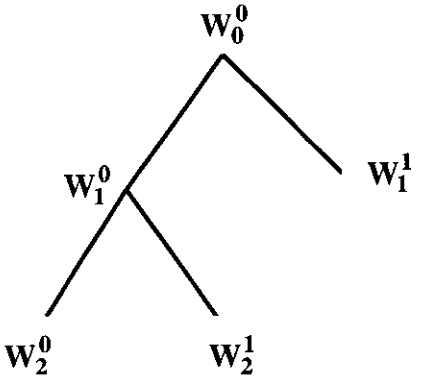


Figure 3.4: Left recursive binary tree structure obtained by the above scheme for decomposition by DWT.

3.2 Review of feature extraction by the DWT

Owing to its multi-resolution capabilities and its ability to handle non-stationary signals, the DWT has been suggested for feature extraction in for phoneme recognition [3], [4], [5]. In [3] the discrete wavelet decomposition was applied on the phonemes extracted from the TIMIT database [6]. The wavelet coefficients obtained after decomposition were used as features by first ranking them in order of their energy. The top wavelet coefficients having high energy were selected as features. However, since the DWT is shift variant, these features were not very reliable. With small shift in the signal the wavelet coefficients would change, thereby resulting in a change in the feature vector. The problem is absent if the shift is an integral multiple of the sampling time; however, this

cannot be guaranteed in a practical situation. In order to reduce the problem due to shift, the signal can be over-sampled or the CWT can be applied. Tan [4] has reported that the Sampled Continuous Wavelet Transform (SCWT) normally improves the recognition of phonemes as compared to the Mel Frequency FFT cepstral coefficients and the discrete wavelet transform. Tan used a windowed wavelet (Morlet Wavelet) with 40 triangular band-pass filters. The MFCCs were extracted from a frame of 20ms with an overlapping of 10ms. The original speech was sampled at 16kHz and 12 cepstral coefficients were extracted and fed to the HMM system for recognition. In [5] the wavelet transform has been used to calculate the power spectrum of the signal which is used to extract the MFCC. Apart from feature extraction, the DWT has also been used for segmentation [7] of the speech signal as well as pitch detection [8], [9].

The features extracted are used for the recognition of phonemes by a classifier. The most commonly used classifiers are based on the Hidden Markov Model [10], [11], Linear Discriminant Analysis (LDA) [3], [12] or Artificial Neural Network [13], [14], [15]. Of all these, LDA is the simplest and easy to implement. In this thesis, it is the LDA classifier that is mostly used for the recognition of phonemes, while for the continuous speech recognition HMM-based classifier is used (discussed in Chapter 6). A brief introduction of the LDA classifier is given in the following section.

3.3 Linear Discriminant Analysis

For recognition of different classes based on features, template matching was the earliest technique used. Let $\mathbf{x}$ be the feature vector for the unknown input, and let $\mathbf{m}_1$, $\mathbf{m}_2$, ..., $\mathbf{m}_c$ be the templates (i.e., perfect, noise-free feature vectors) for the $\mathbf{c}$ classes. Then the error in matching $\mathbf{x}$ against $\mathbf{m}_k$ is given by:

$$\|\mathbf{x}-\mathbf{m}_k\| \tag{3.16}$$

Here $\|\mathbf{x}\|$ is called the **norm** of the vector $\mathbf{x}$. A minimum-error classifier computes $\|\mathbf{x}-\mathbf{m}_k\|$ for $\mathbf{k}$ = 1 to $\mathbf{c}$ and chooses the class for which this error is minimum. Since $\|\mathbf{x}-\mathbf{m}_k\|$ is also the distance from $\mathbf{x}$ to $\mathbf{m}_k$, it is also known as minimum-distance classifier. Clearly, a template matching system is a minimum-

distance classifier. Using the inner product to express the Euclidean distance from $\mathbf{x}$ to $\mathbf{m_k}$, it can be written as:

$$\|\mathbf{x}-\mathbf{m_k}\|^2 = (\mathbf{x}-\mathbf{m_k})'(\mathbf{x}-\mathbf{m_k})$$

$$= \mathbf{x}'\mathbf{x} + [\mathbf{m_k}'\mathbf{m_k} - \mathbf{m_k}'\mathbf{x} - \mathbf{x}'\mathbf{m_k}] \qquad (3.17)$$

The first term $\mathbf{x}'\mathbf{x}$ is the same for every class, i.e. for every $\mathbf{k}$. To find the template $\mathbf{m_k}$ that minimises $\|\mathbf{x}-\mathbf{m_k}\|$, it is sufficient to find the $\mathbf{m_k}$ that minimises the bracketed expression. Let a linear discriminant function $\mathbf{g(x)}$ be defined by:

$$\mathbf{g_k(x)} = \mathbf{m_k}'\mathbf{m_k} - \mathbf{m_k}'\mathbf{x} - \mathbf{x}'\mathbf{m_k} \qquad (3.18)$$

Then a minimum Euclidean distance classifier classifies an input feature vector $\mathbf{x}$ by computing $\mathbf{c}$ linear discriminant functions $\mathbf{g_1(x)}$, $\mathbf{g_2(x)}$, ... , $\mathbf{g_c(x)}$ and assigning $\mathbf{x}$ to the class corresponding to the minimum discriminant function. Linear discriminant functions can also be thought as correlation between $\mathbf{x}$ and $\mathbf{m_k}$, with the addition of a correction for the "template energy" represented by $\|\mathbf{m_k}\|^2$. With this correction included, a minimum Euclidean distance classifier is equivalent to a maximum correlation classifier.

The frequent problems encountered in the minimum distance classifier are:

- The features may be inadequate to distinguish the different classes
- The features may be highly correlated
- The decision boundary may have to be curved
- There may be distinct sub-classes in the data
- The feature space may be too complex

Some of the limitations of simple minimum Euclidean distance classifiers can be overcome by using a Mahalanobis distance measure. In particular, this can often solve problems caused by poorly scaled and/or highly correlated features. The Mahalanobis distance $\mathbf{r}$ for a feature vector $\mathbf{x}$ with mean vector $\mathbf{m_x}$ and $\mathbf{\Sigma}$ as covariance matrix is given as:

$$r^2 = (x - m_x)'\Sigma^{-1}(x - m_x) \qquad (3.19)$$

In the special case where the features are uncorrelated and the variances in all directions are the same, the Mahalanobis distance becomes equivalent to the Euclidean distance. The covariance of two features is a measure of their tendency to vary together. Where the variance is the average of the squared deviation of a feature from its mean, the covariance is the average of the products of the deviations of feature values from their means.

Let **{x[1,i], x[2,i], ... , x[n,i]}** be a set of **n** examples of feature **i**, and let **{x[1,j], x[2,j], ... , x[n,j]}** be a corresponding set of **n** examples of feature **j**. (That is, **x[k,i]** and **x[k,j]** are features of the same pattern, **k**) Similarly, let **m[i]** be the mean of feature **i**, and **m[j]** be the mean of feature **j**. Then the covariance of feature **i** and feature **j** is defined by:

$$\Sigma_{i,j} = \frac{\{[\,x[1,i] - m[i]\,][\,x[1,j] - m[j]\,] + \ldots + [\,x[n,i] - m[i]\,][\,x[n,j] - m[j]\,]\}}{n - 1} \qquad (3.20)$$

One can use the Mahalanobis distance in a minimum distance classifier as follows. Let $m_1, m_2, \ldots, m_c$ be the means (templates) for the **c** classes, and let $\Sigma_1, \Sigma_2, \ldots, \Sigma_c$ be the corresponding covariance matrices. To classify a feature vector **x**, the Mahalanobis distance from **x** to each of the means is measured, and **x** is assigned to the class for which the Mahalanobis distance is minimum.

The use of the Mahalanobis metric removes several of the limitations of the Euclidean metric such as:

- It automatically accounts for the scaling of the co-ordinate axes
- It corrects for correlation between the different features
- It can provide curved as well as linear decision boundaries

However, there is a price to be paid for these advantages. The covariance matrices can be hard to determine accurately, and the memory and time requirements grow quadratically rather than linearly with the number of features. These problems may be insignificant when only a few features are needed, but they can become quite serious when the number of features becomes large.

There is an important special case in which the Mahalanobis metric results in linear discriminant functions. This case occurs when the clusters in all **c** classes have the same covariance matrix Σ. In that case, we can expand the squared Mahalanobis distance from the feature vector **x** to the mean vector $\mathbf{m}_k$ as:

$$\begin{aligned} r^2 &= (\mathbf{x}-\mathbf{m}_k)^t \Sigma^{-1}(\mathbf{x}-\mathbf{m}_k) \\ &= \mathbf{x}^t\Sigma^{-1}\mathbf{x} - \mathbf{m}_k^t\Sigma^{-1}\mathbf{x} - \mathbf{x}^t\Sigma^{-1}\mathbf{m}_k + \mathbf{m}_k^t\Sigma^{-1}\mathbf{m}_k \end{aligned} \tag{3.21}$$

This is a similar expression as obtained for a minimum Euclidean distance classifier. Once again a linear discriminant function can be obtained by maximising the last three terms of the Equation 3.21. The linear discriminant function $g_k(\mathbf{x})$ is defined by:

$$g_k(\mathbf{x}) = \mathbf{m}_k^t\Sigma^{-1}\mathbf{m}_k - \mathbf{m}_k^t\Sigma^{-1}\mathbf{x} - \mathbf{x}^t\Sigma^{-1}\mathbf{m}_k \tag{3.22}$$

Although the above expression gives up the advantage of having curved decision boundaries, it retains the advantage of being invariant to linear transformations. In addition, it reduces the memory requirements from the **c** **d**-by-**d** covariance matrices to the **c** **d**-by **1** with a corresponding speed-up in the computation of the discriminant functions. Finally, when the covariance matrices are the same for all **c** classes, one can pool the data from all the classes and get a much better results from a limited amount of data.

This algorithm has been used for the classification of phonemes using the wavelet features and is referred to as the linear discriminant analysis (LDA) throughout this work (thesis).

3.4 Proposed feature extraction by using the DWT

It is obvious from the discussion in Section 3.2 that the use of wavelet coefficients directly as features is not a good choice. Taking the analogy from the human hearing, which is sensitive to the pressure (energy of the signal), the idea of energy-based feature has been used in speech recognition. This idea is here extended to the DWT-based features. Instead of using the energy of some of the

wavelet coefficients in a sub-space (sub-space is the same as a frequency band in the context of signal processing) as features, the energy in each frequency band is proposed. This scheme overcomes the problem of shift variance, as the energy in a band remains constant even if the signal is shifted. The feature extraction step based on the DWT proceeds as follows:

- A single frame of 32ms was selected, which gives 512 samples for speech sampled at 16kHz. This frame size was chosen, as it is large enough to accommodate all the phonetic variations required for identification. This also gives a dyadic length of the samples (number of samples which are integer multiple of 2^n), which is suitable for wavelet decomposition. If the phoneme length is less than 512 samples then it is padded with zeros to have all the frames of equal size. Let the phoneme sample be denoted by

$$\mathbf{x[nT]} \quad 1 \le \mathbf{n} \le 512 \qquad \mathbf{T = 1/fs} \qquad \mathbf{fs} = \text{sampling frequency} \tag{3.23}$$

- These frame samples are transformed using different levels of discrete wavelet decomposition. In general a 'p' level of discrete wavelet decomposition gives a 'p' detailed sub-space and an approximation sub-space. The transform was obtained by taking the inner product of the phoneme samples **x[nT]** and the wavelet function $\psi(\mathbf{t})$. The wavelet function used in this work was a 'Daubechies 6', where 6 indicates the vanishing moment or the order of wavelet used.

$$\mathbf{c_{j,k}} = \left\langle \mathbf{x}, \psi_{\mathbf{j,k}} \right\rangle \tag{3.24}$$

where

$$\psi_{\mathbf{j,k}}(\mathbf{t}) = \mathbf{a_0^{-j/2}}\psi\left(\frac{\mathbf{t - ka_0^j T}}{\mathbf{a_0^j}}\right) \tag{3.25}$$

- If $\mathbf{c_{j,k}}$ is the $\mathbf{j^{th}}$ wavelet coefficient in the $\mathbf{k^{th}}$ band then the total energy $(\mathbf{E_p})$ in the band **p** is given by:

$$\mathbf{E_p} = \sum_{\mathbf{j=1}}^{\mathbf{N_p}} (\mathbf{c_{j,p}})^2 \qquad \mathbf{p = 1,2,....L} \tag{3.26}$$

$$\mathbf{F_p} = \mathbf{E_p / N_p} \qquad \mathbf{p = 1,2,....L} \tag{3.27}$$

where N_p is the number of wavelet coefficients in the p^{th} band and **L** is the number of bands. The calculated energy is then divided by the number of the wavelet coefficients in the corresponding band, thereby giving the average energy per wavelet coefficients per band F_p. Since, by the DWT, partitioning of the frequency plane is such that the lower bandwidth occurs at the lower frequencies then progressively doubles towards the higher frequency end. This can be seen in Figure 3.5, which shows partitioning of the frequency band of a signal of bandwidth **B** Hz by three-level discrete wavelet decomposition. The first level of decomposition splits the band into two equal sub-bands at point **c**. The second level of decomposition splits the sub-band left of **c** into two at point **b** and the third level of decomposition further splits the sub-band at point **a**. This gives more number of samples at the higher frequency band as compared to the lower frequency band. Thus, the above division results in giving more weight at the lower frequency end and less at the higher frequency end. For a p-level of discrete wavelet decomposition there will be 'p+1' sub-bands, resulting in 'p+1' features to be used for classification.

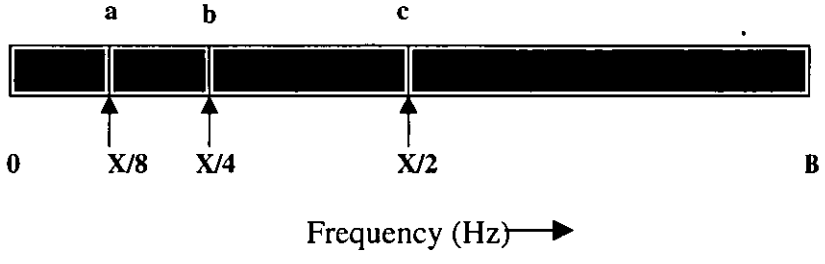


Figure 3.5: Partitioning of the frequency spectrum by a three-level discrete wavelet decomposition.

3.5 Results

The phonemes used for training the classifier and testing its performance were extracted from the TIMIT database [6]. A list of phonemes present in the TIMIT database is given in APPENDIX B. The two dialect regions DR1 (New England region) and DR2 (Northern part of USA) were selected for the extraction of the phonemes. A total of 151 speakers' data was used, of which 114 were used for training and the rest for testing the classifier. Out of the total of 152 speakers, 102 were male, of which 77 were used for training. Vowels, fricatives and stops from all possible contexts were extracted from the database (Table 3.1). The features were extracted by the method explained in Section 3.4 and passed to a classifier based on a simple LDA, where the classification was performed by using the Mahalanobis distance, as explained in Section 3.3.

Table 3.1: List of phonemes extracted from the TIMIT database

Vowels	/aa/, /ax/, /iy/
Unvoiced fricatives	/f/, /sh/, /s/
Voiced fricatives	/v/, /dh/, /z/
Unvoiced stops	/p/, /t/, /k/
Voiced stops	/b/, /d/, /g/

A range of experiments was carried out for the extraction of features by the DWT. In the first experiment, the level of decomposition by the DWT was varied from 4 to 7, thereby giving 5 to 8 features respectively for each 32ms frame duration of a phoneme. The mother wavelet used for this experiment was 'Daubechies 6'. The recognition performance achieved is shown in Figure 3.6. It can be seen from the results that there is not much increase in the recognition performance of the phonemes with the increase in the number of features. Instead there is a slight decrease in the recognition performance for the higher number of features. This reduction in recognition performance is attributed to the fact that the DWT recursively decomposes the lower frequency band only. This results into the energy features coming from the very low frequency bands. These

features are less significant and carry little discriminatory information from the phoneme classification point of view [12]. Thus, increasing the level of decomposition although increases the dimension of the feature but is not helpful in classification.

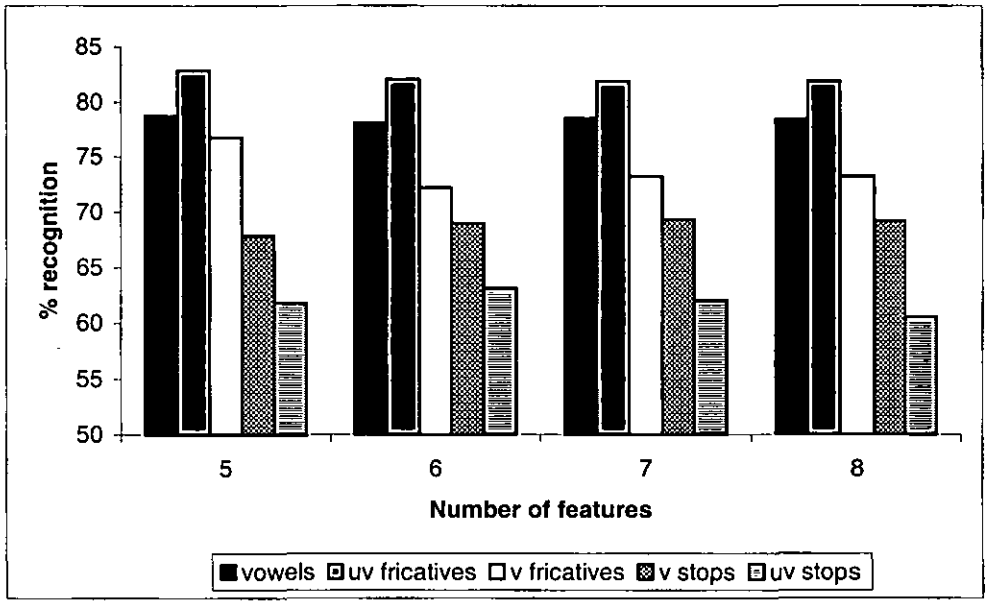


Figure 3.6: Recognition performance for phonemes with different numbers of features where uv indicates the unvoiced and v the voiced phonemes.

Table 3.2 shows the level of decomposition and the corresponding frequency bands obtained for a speech signal of 0-8kHz band. For example the lowest frequency band energy features form six or seven levels of decomposition by the DWT, but this is not very significant for classification as it carries little information about the phoneme. Thus, increasing the level of decomposition will not help to improve the recognition performance by the DWT.

Table 3.2: Partitioning of the 0-8kHz frequency band signal by different levels of decompositions by DWT

Level of decomposition	Bands obtained by DWT (kHz)
4	0-0.5, 0.5-1, 1-2, 2-4 & 4-8
5	0-0.25, 0.25-0.5, 0.5-1, 1-2, 2-4 & 4-8
6	0-0.125, 0.125, 0.25, 0.25-0.5, 0.5-1, 1-2, 2-4 & 4-8
7	0-.0625, .0625-0.125, 0.125, 0.25, 0.25-0.5, 0.5-1, 1-2, 2-4 & 4-8

A classifier based on an Artificial Neural Network (ANN) was also implemented to compare the improvement in recognition performance achieved by a non-linear classifier over a linear classifier. An ANN consists of a collection of neuron units communicating with each other. The signal travelling from one neuron to another is weighted and is passed through a non-linear activation function. The output of a neuron is the weighted sum of all the inputs passed through an activation function. The neurons are arranged in layers, with each neuron of the previous layer connected to all the neurons in the current layer. This gives a parallel architecture, thereby making it very fast and robust. An ANN can be trained to classify patterns either in a supervised or an unsupervised manner. In the case of supervised training the training pattern is applied at the input and the corresponding output is defined. The training progresses by adjusting the weights such that the error between the defined output and the actual output is minimised. The details of the different types of ANN and their training can be found in [16], [17], [18]. The Multi-layer Perceptron (MLP) [15], [19] as well as time delay neural network (TDNN) [13], [14] have been used as classifier for speech recognition applications.

A MLP is a class of ANN that can learn from the training set (under supervision) to classify the input patterns. For classification by the MLP, the data was divided into three sets: training, validation and testing. The validation set was used as a stopping criterion during the training process. This helps in avoiding the MLP to learn the local features of the training set. About 67% of the data are used for training, 10% for validation and the rest for testing the classifier. All three sets are mutually exclusive to each other. An MLP with one hidden layer of five neurones and three output neurones was simulated and trained for the classification of vowels (/aa/, /ax/ and /iy/) and stop phonemes (/p/, /t/ and /k/) [20], [21]. The results obtained are plotted in Figure 3.7.

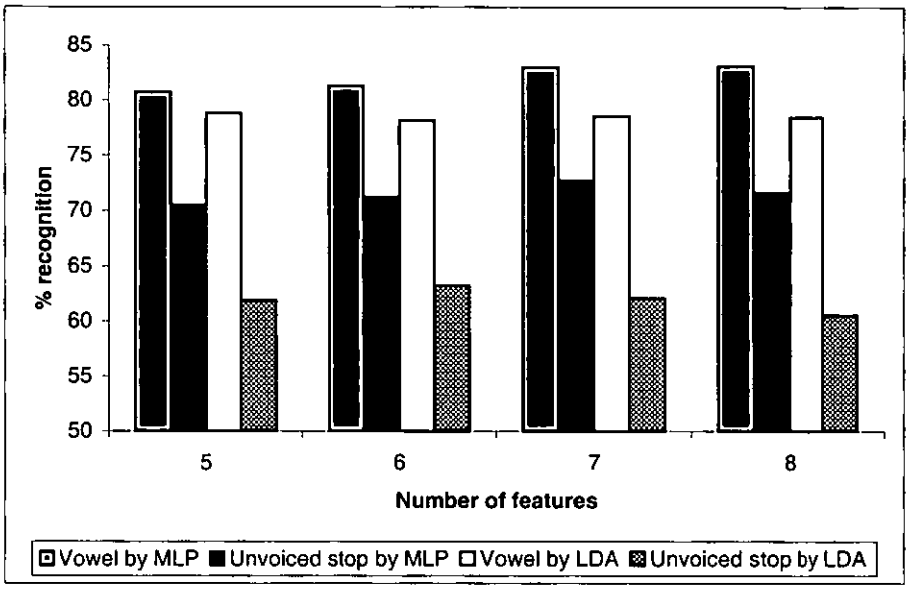


Figure 3.7: Recognition performance achieved by using linear discriminant analysis and a multi-layer perceptron network.

By comparing the performance of the MLP classifier for unvoiced stops and vowel classification it can be observed that by increasing the number of features further will not increase the recognition performance. The performance of the MLP classifier is found to be about 8-10% better than the LDA classifier for classification of the stop phonemes (Figure 3.7). For the case of vowel classification the improvement is not that large owing to the fact that vowels have themselves distinct formant frequencies causing the features to be linearly separable at lower levels of wavelet decomposition. It can also be seen that the recognition performance of the vowel improves consistently with an increase in the number of features while there is a distinct decrease in the recognition performance of the stops (/p/. /t/ & /t/) at the end [21]. The stop phonemes have most of the discriminatory information at the higher frequency band. By increasing the number of features, the lower frequency band is decomposed and the features corresponding to these lower bands are used for classification, which carries very little discriminatory information. Thus the recognition rate reduces not only for the LDA but also for the MLP classifiers. Although the MLP-based classifier gave better recognition performance than the LDA-based classifier, it was not used in the later tests because of the long training time requirement.

In the second experiment the frame was further sub-divided into sub-frames of 16ms and 8ms to account for the temporal evolution of the phonemes. As the speech signal is stationary for roughly 10ms duration (due to the physical limitations of the articulators) any further increase in the number of sub-frames was not tried. The feature extraction process, as explained earlier, is carried out for each sub-frame duration and the classification is carried out after 32ms (a frame duration). The subsequent features of a sub-frame give the dynamics of the features for a given phoneme and are important for classification of the stops. The results for unvoiced stops and vowels are shown in Figure 3.8 and Figure 3.9. It can be seen from the results in Figure 3.9 that the vowel recognition does not improve with the increase in the number of sub-frames. The reason for this is because the signal spectrum of a vowel is fairly constant from beginning to end of the phoneme; thus features obtained in each sub-frame have very little or no difference. Also, by increasing the number of features without having much discriminatory information the linear classifier's performance is expected to degrade. The recognition performance of the unvoiced stops shows considerable improvement when two sub-frames and four sub-frames are used (Figure 3.8). The increase in recognition of /p/, /t/ and /k/ is because of the marked difference in the shape of the signal waveform at the beginning and at the end. Thus, dividing into sub-frames helps in the recognition because the features are different in each sub-frame (due to the change in the signal, energy in each frequency band also changes, giving different features). Here the LDA-based classification was carried out because of the reason that if it shows better recognition then it implies these features will always give better performance for other non-linear classifiers as well. Therefore only the LDA-based classifier will be used for recognition from here onwards. The results of the phoneme recognition based on the DWT with different numbers of sub-frames features are given in Table 3.3.

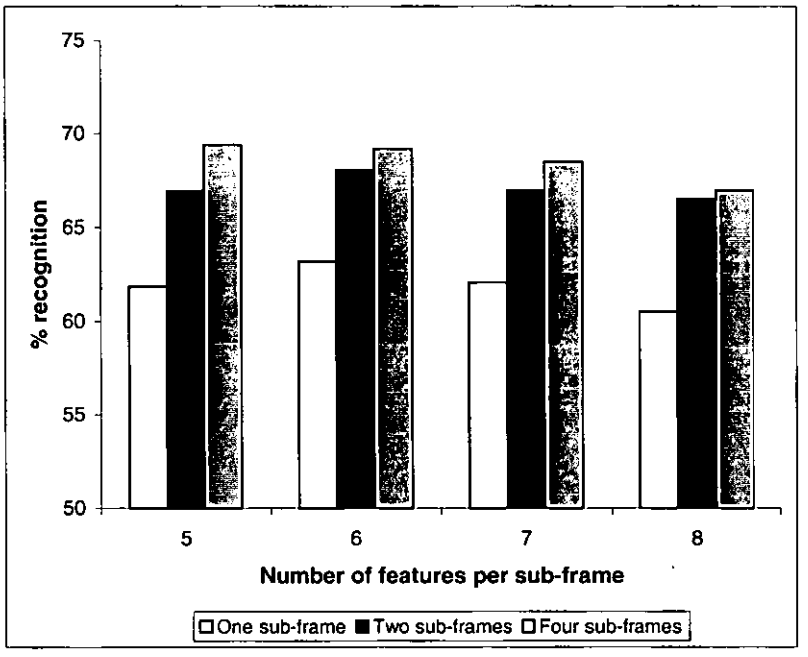


Figure 3.8: Recognition performance of the unvoiced stops vs. number of features per sub-frame for different number of sub-frames.

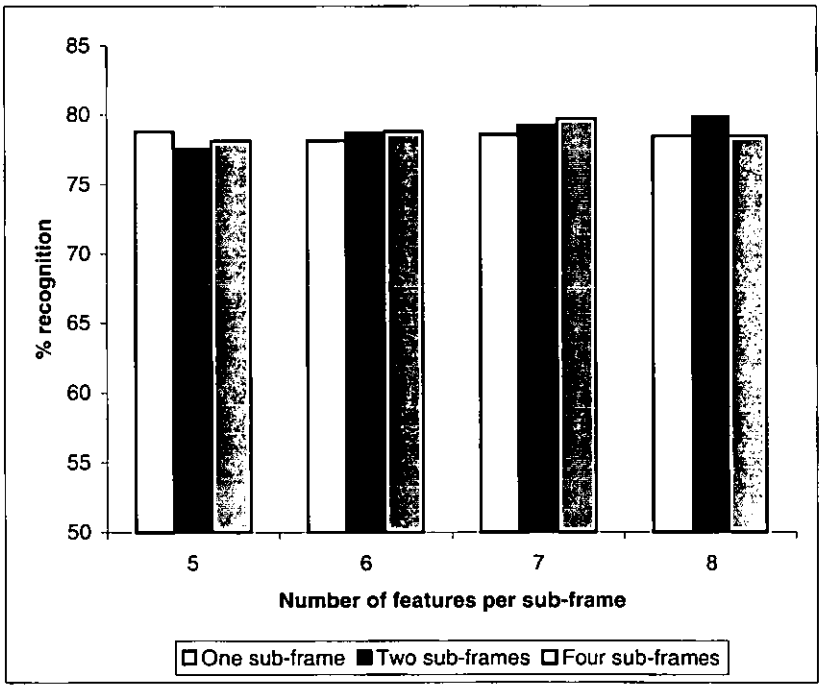


Figure 3.9: Recognition performance of the vowels vs. number of features per sub-frame for different number of sub-frames.

In the next experiment logarithmic compression was applied on the above features in order to reduce the dynamic range. The results obtained are for single frame of 32ms duration. The comparative recognition performance of the LDA classifier for recognition of vowels and unvoiced fricatives based on energy and log-energy features are shown in Figure 3.10 and Figure 3.11.

These plots also show the variation in recognition performance with the number of features. It is clear from the results that the log-energy features are far more superior as compared to simple energy-based features. However, the recognition performance does not improve with an increase in the number of features, as explained earlier. The variation of recognition performance of the unvoiced fricatives, unvoiced stops and vowels with the number of sub-frames and features is given in Table 3.4.

Table 3.3: Phoneme recognition performance achieved by the LDA classifier for different numbers of energy features extracted by the DWT.

No. of features	Phonemes	One frame	Two sub-frames	Four sub-frames
5	Vowels	78.80	77.55	78.12
	Unvoiced fricatives	82.88	83.07	78.21
	Voiced fricatives	76.77	77.53	71.97
	Voiced stops	67.89	66.05	70.40
	Unvoiced stops	61.86	66.30	69.40
6	Vowels	78.12	78.68	78.80
	Unvoiced fricatives	82.10	80.35	76.85
	Voiced fricatives	72.22	72.73	71.72
	Voiced stops	68.99	64.95	70.39
	Unvoiced stops	63.19	68.07	69.18
7	Vowels	78.57	79.25	79.70
	Unvoiced fricatives	81.91	80.16	75.25
	Voiced fricatives	73.23	73.74	70.96
	Voiced stops	69.36	66.79	70.76
	Unvoiced stops	62.08	66.96	68.51
8	Vowels	78.46	79.82	78.46
	Unvoiced fricatives	81.91	78.40	73.93
	Voiced fricatives	73.23	72.22	70.96
	Voiced stops	69.17	66.05	68.23
	Unvoiced stops	60.53	66.52	66.96

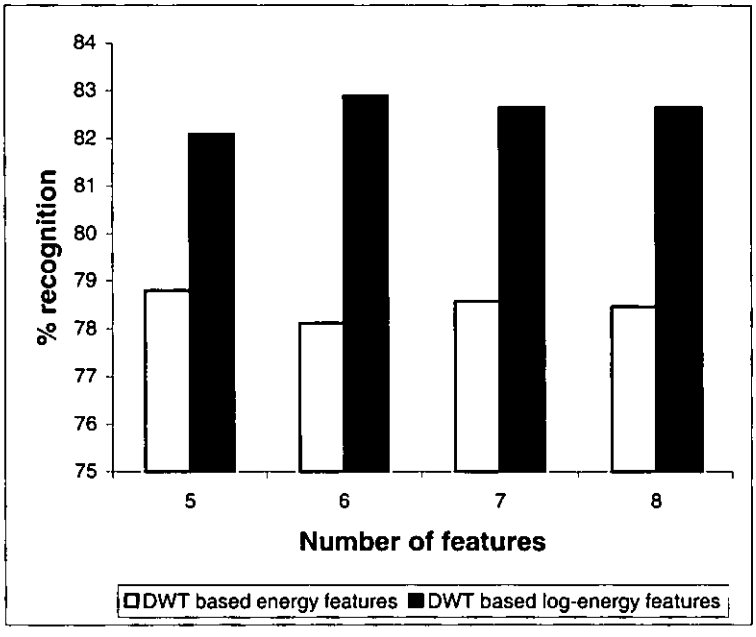


Figure 3.10: Improvement achieved by using log-energy features over simple energy features for vowel recognition.

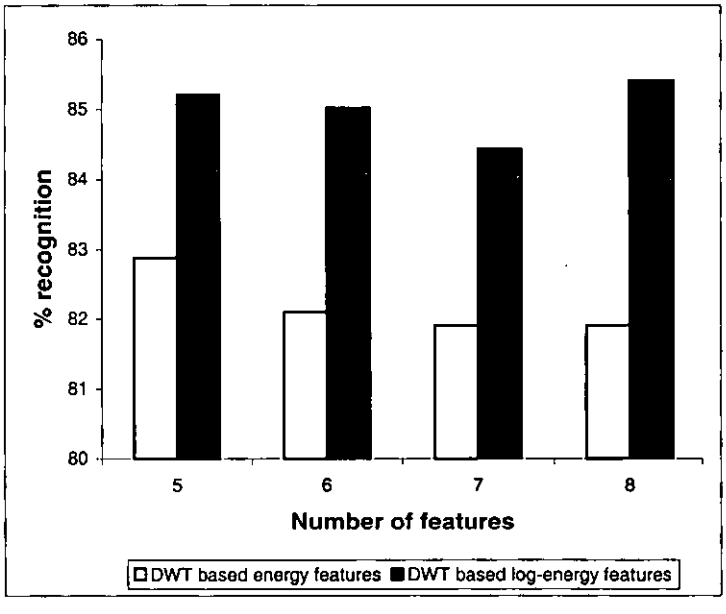


Figure 3.11: Improvement achieved by using log-energy features over simple energy features for unvoiced fricative recognition.

Table 3.4: Phoneme recognition performance with different numbers of log-energy features and sub-frames extracted by the DWT.

No. of features	Phonemes	One sub-frame	Two sub-frames	Four sub-frames
5	Vowels	82.09	80.54	80.73
	Unvoiced fricatives	85.21	85.41	84.82
	Unvoiced stops	66.08	71.40	74.94
6	Vowels	83.90	82.20	80.95
	Unvoiced fricatives	85.02	84.63	85.02
	Unvoiced stops	69.62	73.39	73.39
7	Vowels	82.65	81.41	81.63
	Unvoiced fricatives	84.44	84.82	84.82
	Unvoiced stops	68.96	74.94	74.06
8	Vowels	82.65	81.63	79.71
	Unvoiced fricatives	85.41	86.77	85.41
	Unvoiced stops	71.18	72.06	72.28

Lastly the performance of the log-energy features is then compared with the earlier proposed DWT coefficient-based features for the phoneme recognition task [3]. Both the tests are carried out on the same dialect regions of the TIMIT database using a 32ms frame duration and 'Daubechies 6' wavelet for the DWT. The DWT coefficient-based features are selected by ranking them by energy and taking the top 64 for classification. The comparative results are encouraging and are shown in Figure 3.12.

It shows improvement in recognition for both unvoiced fricatives and unvoiced stops; however, the performance is poor for vowel recognition. This is because the periodicity is easily represented with a large number of features. It should be noted that the proposed log-energy-based features are only 8 as compared to 64 wavelet coefficient features. Thus the task of the classifier is much simpler when the proposed features are used.

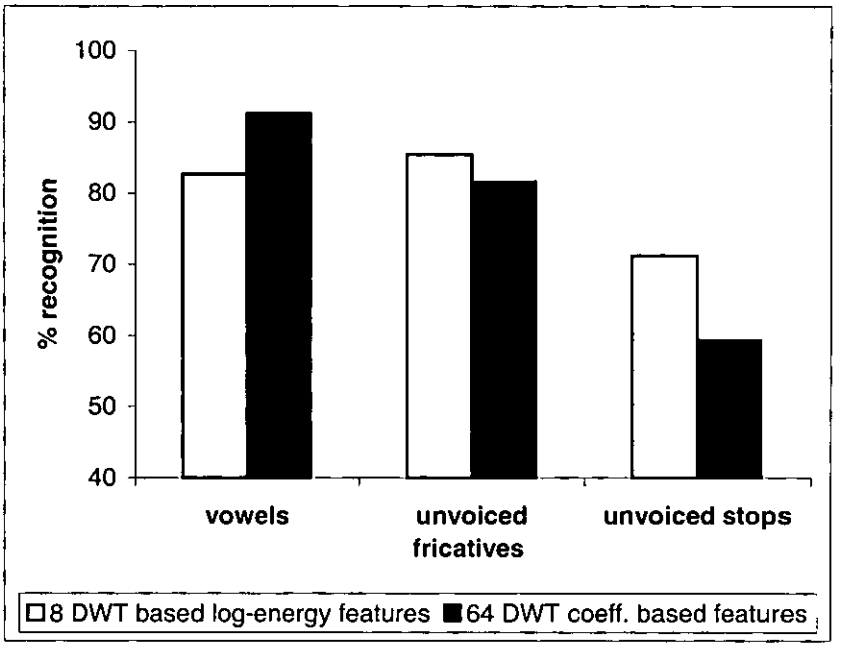


Figure 3.12: Phonemes recognition performance by log-energy and coefficients based features obtained by the DWT.

Some of these simulation results can be obtained by using the software provided at the back of this thesis. The usage of the software is given in brief in the APPENDIX E. There is also a 'readme.txt' file included with the software that gives the details of the functions developed in Matlab.

3.6 Summary

In this chapter, features based on the DWT have been explored and the recognition performance achieved by varying the number of sub-frames and features is studied. The results found show that the log-energy-based features perform the best and it shows better results for unvoiced stops and fricatives than the 64-wavelet coefficients. Further, the recognition performance is found to be better when a non-linear classifier based on the MLP is used. However, it is worth noting that there was no significant improvement in the recognition performance if the number of features were increased (by increasing the level of decomposition). This clearly indicates that in order to have further improvement in the recognition performance, the DWT cannot go beyond this limit. However, it is may be possible to increase the recognition performance if the new features

were derived from the band that had more perceptual information, e.g. 1-2kHz and 2-4kHz band. The DWT cannot decompose these bands (because of its limitation of decomposing the lower frequency band only), however; it is possible to do so by using a more general form of wavelet transform known as wavelet packets. In the next chapter this possibility is explored and new features based on the admissible wavelet packets and its application to the phoneme recognition are discussed.

3.7 References

[1] S. Mallat, *Wavelet tour signal of signal processing*, Academic press, 1999.

[2] O. Rioul and M Vetterli, "Wavelet and signal processing", *IEEE Signal Processing Magazine*, pp. 14-38, October 1991.

[3] C. J. Long, *Phoneme discrimination using non-linear wavelets methods*, Ph.D. thesis, Loughborough University, Department of Electronic and Electrical Engineering, February 1999.

[4] B. T. Tan, M. Fu, A. Spray and P. Dermody, "The use of wavelet transform in phoneme recognition", *Proceedings of 4th International Conference on Spoken Language Processing*, Philadelphia, USA, October 1996, vol. 4, pp. 2431-2434.

[5] H. Wassner and G. Chollet, "New cepstral representation using wavelet analysis and spectral transformations for robust speech recognition", *Proceedings of 4th International Conference on Spoken Language Processing*, Philadelphia, USA, October 1996, vol. 1, pp. 260-263.

[6] TIMIT Acoustic-Phonetic Continuous Speech Corpus, National Institute of Standards and Technology, Speech Disc 1-1.1, NTIS Order no. PB91-505065, October 1990.

[7] L. Janer, J. Marti C. Nadeu and E. L. Solano, "Wavelet transforms for non-uniform speech recognition systems", *Proceedings of 4th International Conference on Spoken Language Processing*, Philadelphia, USA, October 1996, vol. 4, pp. 2348-2351.

[8] L. Janer, J. J. Bonet and E. L. Solano, "Pitch detection and voiced/unvoiced decision algorithm based on wavelet transforms", *Proceedings of 4th International Conference on Spoken Language Processing*, Philadelphia, USA, October 1996, vol. 2, pp. 1209-1212.

[9] S. Kadambe and G. F. Boudreaux-Bartels, "Application of the wavelet transform for pitch detection of speech signals", *IEEE Transactions on Information Theory*, vol. 38, no. 2, pp. 917-924, Part 2 March 1992.

[10] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition", *Proceedings of IEEE*, vol. 77, no. 2, pp. 257-285, February 1989.

[11] S. Young, "A review of large vocabulary continuous speech recognition", *IEEE Signal Processing Magazine*, September, pp. 45-57, 1996.

[12] O. Farooq and S. Datta, "Wavelet transform for dynamic feature extraction of phonemes", *Acoustics Letters*, vol. 23, no. 4, pp. 79-82, 1999.

[13] A. Waibel, *Neural Network approach for speech recognition*, Advances in speech processing, Edited by S. Furui and M. Sondhi, Marcel Dekker, 1991, pp. 555-595.

[14] A. Waibel, T Hanazawa, G. Hinton, K. Shikano and K. J. Lang, "Phoneme recognition using time delay neural network", *IEEE Transactions on Acoustic, Speech and Signal Processing*, vol. 37, no. 3, pp. 328-339, March 1989.

[15] P. C. Woodland, *Spoken alphabet recognition using multilayer perceptron*, Neural Networks for speech Vision and Natural Language by Chapman & Hall, 1992, pp. 135-147.

[16] S. Y. Kung, *Digital neural networks*, Prentice Hall Inc., New Jersey, 1993.

[17] R. Linggard, *Neural network for speech processing: An introduction*, neural networks for speech vision and natural language by Chapman & Hall, 1992, pp. 129-134.

[18] C. G. Looney, *Pattern recognition using neural networks: Theory and algorithms for Engineers and Scientists*, Oxford University Press, New York, 1997.

[19] C. J. Long and S. Datta, "Wavelet based feature extraction for phoneme recognition", *Proceedings of 4th International Conference on Spoken*

Language Processing, Philadelphia, USA, October 1996, vol. 1, pp. 264-267.

[20] O. Farooq and S. Datta, "A neural network phoneme classification based on Wavelet features", *Proceedings of International Conference on Recent Advances Soft Computing*, June 29-30, 2000, DeMontfort University, Leicester, UK, pp. 75-77.

[21] O. Farooq and S. Datta, "Speaker independent phoneme recognition by MLP using wavelet features", *Proceedings of 6th International Conference on Spoken Language Processing*, Beijing, China, 16-20 October 2000, vol. 1, pp. 393-396.

CHAPTER 4

ADMISSIBLE WAVELET PACKETS FOR PHONEME RECOGNITION

In the previous chapter it had been established that the recognition performance could not be improved even if the number of features were increased. The increase in the feature dimension was achieved by increasing the level of decomposition by the DWT. This resulted in splitting the lower frequency bands only, producing new features from the very low frequency end of the spectrum. These very low frequency bands (less than 250Hz) are not as important as the mid-frequency band (about 500Hz-4kHz range) because the mid-band has more discriminatory information related to phonemes. Therefore it is desirable to have more features coming from the mid-frequency band region. This in turn implies further splitting of these bands into smaller sub-bands. However, as can been seen in Section 3.1, the DWT can successively decompose only the low frequency band obtained from the previous decomposition. This limitation can be overcome by using a more general form of wavelet transform proposed by Coifman [1], which decomposes the frequency axis arbitrarily. This new family of dyadic orthonormal wavelets is called wavelet packets and is uniformly translated in time to ensure that the entire time-frequency place is covered.

Orthonormal bases of $L^2(\mathbb{R})$ have also been constructed by dividing the time axis instead of the frequency axis. The time axis is segmented into different intervals and each interval is multiplied by cosine functions of different frequencies. This is known as local cosine bases due to the multiplication by the cosine function. Wavelet packets and local cosine bases form a dual family of bases. The partitioning of the time-frequency plane by the wavelet packet and the local cosine transform is shown in Figure 4.1 (a) and (b).

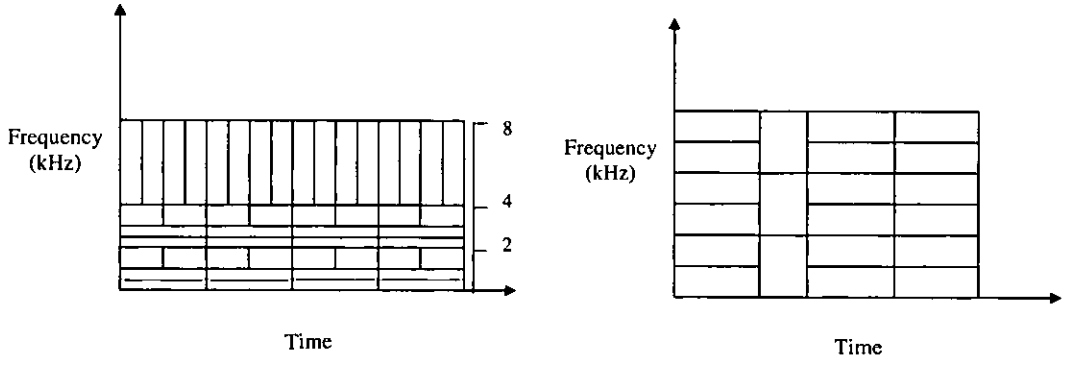


Figure 4.1: An example of tiling of time-frequency plane by (a) wavelet packet (b) local cosine basis

In this chapter the possibility of using wavelet packets for feature extraction is explored and the admissible wavelet packets are proposed for this purpose. Further, a filter structure based on the Mel frequency scale using the admissible wavelet packets has been proposed for the extraction of features.

4.1 Wavelet Packets

The DWT performs recursive decomposition of the lower frequency band obtained from the previous decomposition in a dyadic fashion. A speech signal sampled at 16kHz when decomposed once by the DWT will give two bands (0-4kHz & 4-8kHz). The second level of decomposition will partition the lower frequency band of 0-4kHz further into a band of 0-2kHz and 2-4kHz. In wavelet packet decomposition, the lower as well as the higher frequency band is decomposed into two sub-bands, thereby giving a balanced binary tree structure

as shown in Figure 4.2. Each node W_j^p, in the tree represents the depth (level) **j** and the number of node **p** to the left of it. The two wavelet packet orthogonal bases generated from a parent node (W_j^p) are defined as:

$$\psi_{j+1}^{2p}(k) = \sum_{n=-\infty}^{\infty} g[n]\psi_j^p(k - 2^j n) \tag{4.1}$$

$$\psi_{j+1}^{2p+1}(k) = \sum_{n=-\infty}^{\infty} h[n]\psi_j^p(k - 2^j n) \tag{4.2}$$

where **g[n]** is the scaling filter and **h[n]** is the wavelet filter.

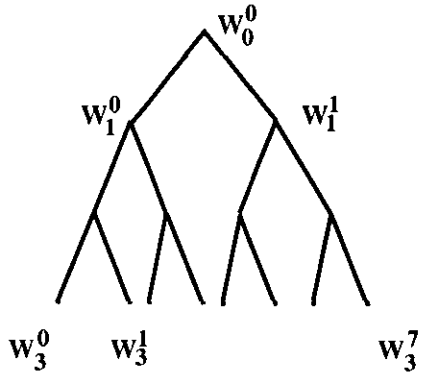


Figure 4.2: Balanced binary tree achieved by wavelet packet decomposition.

Full wavelet packet decomposition results in an over-complete set of bases. For a full **j** level wavelet packet decomposition there will be over $2^{2^{j-1}}$ orthogonal bases. From the above library of bases (also called packet table) the best basis is to be selected. A wavelet packet basis divides the frequency axis into intervals of varying sizes and covers each interval with a uniform translation in time. A selection of the best basis tries to have a best frequency partitioning for a given signal by reducing a cost function. The most commonly used cost function is the Shannon entropy [7].

There has been some recent research in the area of speech recognition where the wavelet packet transform has been used for feature extraction [2], [3], [4], [5], [6]. In [3], [4], [5] the wavelet packet decomposition has been used and different criteria such as best basis selection and local discriminant basis have been applied for feature selection. The wavelet packet parameters have been used for the recognition of stressed speech [6] and its performance is found to be

superior to the MFCC-based features. In this, the features are based on the energy of the wavelet coefficients in each band. A local cosine transform has also been used for feature extraction for the recognition of Korean digits [2].

4.2 Selection of the 'Best Basis'

In order to have a compact representation of a signal with a minimum set of orthogonal bases, the time-frequency tiling should be signal specific. This would correspond to an irregular sampling grid, locally adapted to the signal variations. This can be achieved by pruning the binary tree in a manner that minimises/maximises a given cost function. The cost function is highly dependent on the type of application, e.g. in compression, Shannon's entropy (this minimises the distortion) is usually used. However, for the recognition application, a cost function that reflects the distance between the classes is desirable [3]. Examples of these types of functions are the relative or cross entropy [7]. $\mathbf{L}^2(\mathbb{R})$ norm and logarithm of energy have also been used for this purpose. For a signal of dyadic length **N**, there exists more than $\mathbf{2}^{\mathbf{N/2}}$ bases, i.e. the number of admissible choices of tree structure. To compute the best tree just by comparison of all possible bases would cost more than $\mathbf{N2}^{\mathbf{N/2}}$ operations that are too large [7]. A faster search for the best basis was introduced by Coifman and Wickerhauser [1] using a dynamic programming technique that employed $O(N*\log_2(N))$ operations. In this algorithm the entropy of the parent node is compared with the sum of the entropies of the two children and the minimum is selected to contribute towards the best basis set. Although the best basis algorithm is good for the speech compression, it has problems when applied to the pattern recognition task. This is because of translation variance [7] as explained in the next section. For the phoneme recognition task, due to the presence of co-articulation, stress and emotion, the best basis will result in different bases for a given phoneme. In other words, this will cause the partitioning of the frequency axis in different ways for the same phoneme under different conditions. Thus, the features based on the best basis algorithm will not be a very good choice for phoneme classification. Also, the presence of noise and channel distortion will result into different basis selection.

4.2.1 Translation Variance

Both the wavelet packets and local cosine transform suffer from the problem of translation variance similar to the DWT. This poses a serious limitation on the use of these transforms for pattern recognition applications. Various methods have been proposed to overcome this problem [8], [9]. A 'spin-cyclic' procedure tried to remove the sensitivity of the wavelet coefficients due to translation in the final stage of classification. In this approach a circular shift of the training and the test signal between the interval $[-\tau,+\tau]$ (where τ is an integer and less than the signal dimension) is carried out. This generates 2τ extra samples for training and testing. For the shifted signals, the wavelet packet decomposition will give modified coefficients, thereby yielding a different basis when the cost function is minimised. This directly increases the load on the classifier if the best bases are to be used as features. Even if the energy in each band is used as features (similar to that of features proposed in Chapter 3), this may result into different number of features, which may further create problems in recognition.

For speech recognition if the full wavelet packet decomposition is applied then the energy in each frequency band will not be a good feature for recognition. This is because of the fact that wavelet packet will divide the frequency spectrum into sub-bands of equal bandwidth. Since the higher frequency bands have little discriminatory information, it will be better to have fewer features from these bands, otherwise they will reduce the recognition performance. Due to the above reasons neither the best basis criterion nor the full wavelet packet decomposition can be used effectively for the extraction of features from a phoneme.

For the speech recognition application it is desirable to have more features coming from the 500Hz-4kHz band and fewer from the 4-8kHz and 0-500Hz band. This specifically requires the tiling of the time-frequency plane which is neither similar to the DWT nor to the full wavelet packet decomposition but a special case of wavelet packet known as the admissible wavelet packets. The next section deals with the admissible wavelet packet and its uses for feature extraction.

4.3 Admissible wavelet packets

Instead of applying the full wavelet packet decomposition it is possible to selectively decompose the lower or the higher frequency band. This results in an admissible wavelet packet (AWP) structure and gives flexibility to have any tree structure ranging from the one achieved by the DWT to the one by the full wavelet packet. Thus it is capable of splitting the time-frequency plane in any desired manner. This property can be exploited such that the features carrying more discriminatory information can be extracted from the signal. A predefined partitioning of the frequency band can be performed taking into consideration the speech spectrum and human perception. This will result in an admissible binary tree structure, as shown in Figure 4.3. This structure is different from the tree structure obtained by full wavelet packet decomposition, as shown in Figure 4.2. It can also avoid the undesirable partitioning of the higher frequency bands, which are not very useful for recognition. Thus the problem of shift (that is present when applying best basis algorithm) in the phoneme will not affect the decomposition process as it is fixed. Out of the various possible admissible tree structures from a full wavelet packet tree, only a few are selected by considering the approximate energy distribution of the speech signal in entire frequency band. However, as all the phonemes will not have the same energy distribution, a fixed frequency axis partitioning will not be optimal for the recognition of a complete set of phonemes.

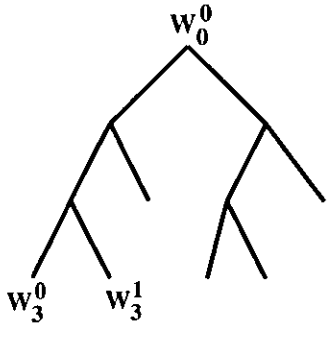


Figure 4.3: Admissible binary tree resulting from wavelet packet decomposition.

4.3.1 Feature Extraction by Admissible wavelet packets

To extract features from the phonemes, a frame duration of 32ms was chosen. The AWP decomposition was applied and the features were calculated by summing the energies of all the wavelet coefficients in a given frequency band. This step is similar to that explained in Section 3.3 for the DWT. Different AWP tree structures were selected, and the number of sub-frames was varied to see their effect on the recognition performance.

In the first experiment a single sub-frame of 32ms (equal to a frame duration) was selected and different admissible tree structures were used to give different numbers of features for the recognition of vowels. The recognition performance achieved (in Figure 4.4 and Figure 4.5) shows considerable improvement over the DWT-based features. Thus the use of AWPs overcomes the problem that was encountered by the DWT [10]. It can also be seen in Figure 4.4 that the recognition performance of the vowels improves with an increase in the number of features initially, then its effect is not so pronounced. For 5 features the recognition performance of the DWT-based features is better than the AWP-based features because it gives better frequency partitioning. However, it should be noted that the DWT a special case of the AWP where the lower frequency band is only decomposed.

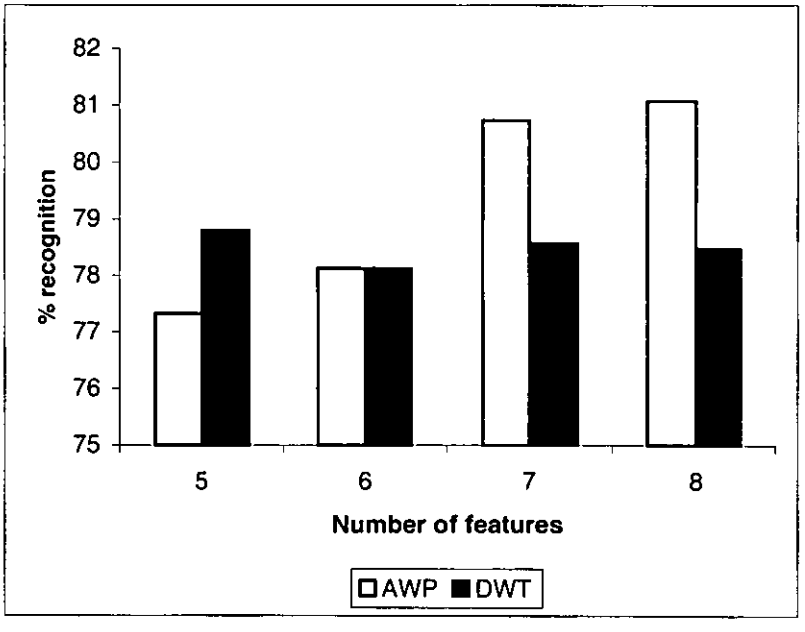


Figure 4.4: Recognition performance of the vowels by using a single sub-frame of 32ms duration.

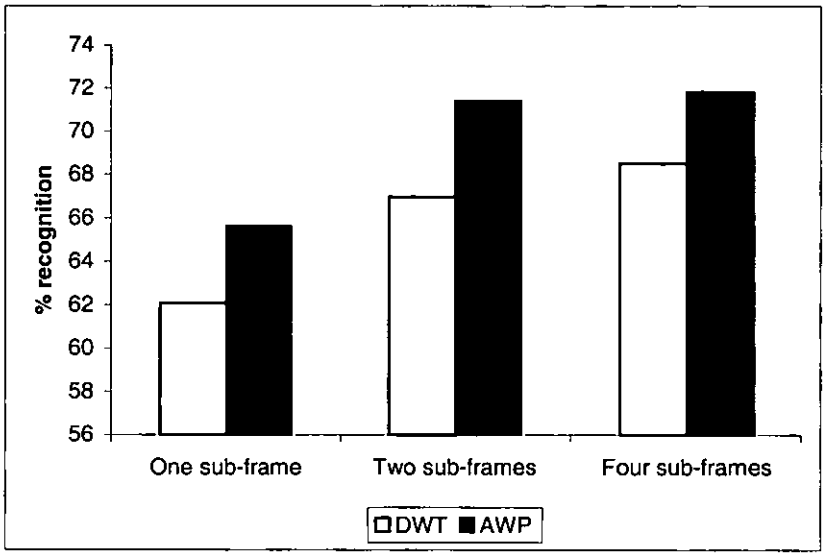


Figure 4.5: Classification of unvoiced stops (/p/, /t/ & /k/) by LDA using 7 features per sub-frame.

To include the temporal information about the evolution of features, the process of feature extraction should be carried out more than once in the 32ms duration. To incorporate this, a frame of 32ms was divided into smaller sub-frames. Experiments were carried out by varying the number of sub-frames and keeping the number of features per sub-frame fixed [10]. Figure 4.5 shows the classification performance achieved by the LDA for unvoiced stops with different numbers of sub-frames for the DWT and the AWP decomposition. The number of features used in both cases is 7 for each sub-frame duration. It is possible to have various admissible tree structures giving the same number of features, but the one giving the best recognition is shown here in Figure 4.5. It is clear that the recognition performance of the AWPs is superior to that of the DWT for the same number of features. This is because that the band structure achieved by the AWP (as shown in Table 4.1) is better in extracting the discriminatory features as compared to the DWT band structure.

Table 4.1: Frequency bands structure for 7 features obtained by the DWT and the AWP used in the experiment

	Bands obtained after decomposition (in Hz)
DWT	0-125, 125-250, 250-500, 500-1000, 1000-2000, 2000-4000, 4000-8000
AWP	0-500, 500-1000, 1000-1500, 1500-2000, 2000-3000, 3000-4000, 4000-8000

Detailed results for the unvoiced stops, unvoiced fricatives, vowels, voiced stops and voiced fricatives are given in Table 4.2. It shows the variation in the recognition performance of the phonemes with different numbers of sub-frames and different numbers of features. Although there are a large number of possible admissible tree structures for a given number of features, the structures chosen for these results are such that they give an overall best performance for the phoneme classes under consideration.

Similar to the Chapter 3, instead of using the energy in each band, the logarithm of energy was used as a feature [11]. The results obtained are shown in Figure 4.6, giving further improvement in the recognition performance over the simple energy features. The detailed results of the recogniser performance using the log-energy features are given Table 4.3, where the number of features as well as the number of sub-frames is also varied. Table 4.4 shows the different band structures used for the extraction of different numbers of features by the AWP. The same band structure was used to evaluate the recognition performance of phonemes in Table 4.2 and Table 4.3.

Table 4.2: Phoneme percentage recognition performance by LDA classifier with different number of energy features extracted by AWP.

No. of features	Phonemes	One frame	Two sub-frames	Four sub-frames
5	Vowels	77.32	76.98	76.08
	Unvoiced fricatives	83.07	82.68	79.57
	Voiced fricatives	77.78	78.28	75.76
	Voiced stops	69.36	64.04	65.52
	Unvoiced stops	63.19	66.96	70.29
6	Vowels	78.12	78.46	78.80
	Unvoiced fricatives	82.10	82.88	77.43
	Voiced fricatives	77.53	78.20	71.72
	Voiced stops	68.81	64.77	69.31
	Unvoiced stops	64.97	70.29	72.51
7	Vowels	80.73	80.73	80.16
	Unvoiced fricatives	81.71	81.91	76.46
	Voiced fricatives	77.27	78.03	71.97
	Voiced stops	67.34	65.32	68.05
	Unvoiced stops	65.63	71.40	71.84
8	Vowels	80.73	80.95	79.37
	Unvoiced fricatives	83.07	81.13	77.43
	Voiced fricatives	78.03	78.03	71.21
	Voiced stops	68.44	65.87	67.69
	Unvoiced stops	66.96	69.84	70.95

Table 4.3: Phoneme percentage recognition performance by LDA classifier for different number of log-energy features extracted by AWP.

No. of features	Phonemes	One frame	Two sub-frames	Four sub-frames
5	Vowels	79.14	79.37	80.39
	Unvoiced fricatives	87.35	87.16	85.60
	Unvoiced stops	66.96	72.73	72.06
6	Vowels	82.52	84.24	80.95
	Unvoiced fricatives	86.58	86.97	85.41
	Unvoiced stops	68.51	73.17	74.06
7	Vowels	82.65	85.49	82.65
	Unvoiced fricatives	86.77	86.97	86.77
	Unvoiced stops	72.28	75.83	74.72
8	Vowels	83.33	84.69	82.09
	Unvoiced fricatives	87.55	85.99	84.44
	Unvoiced stops	71.40	75.83	74.50

Table 4.4: The number of features and the corresponding frequency band selected by AWP decomposition for the extraction of features.

No. of features	Frequency bands obtained by AWP corresponding to the number of features (kHz)
5	0-1, 1-2, 2-3, 3-4, 4-8.
6	0-0.5, 0.5-1, 1-2 2-3, 3-4, 4-8.
7	0-0.5, 0.5-1, 1-1.5, 1.5-2, 2-3, 3-4, 4-8.
8	0-0.5, 0.5-1, 1-1.5, 1.5-2, 2-2.5, 2.5-3, 3-4, 4-8.

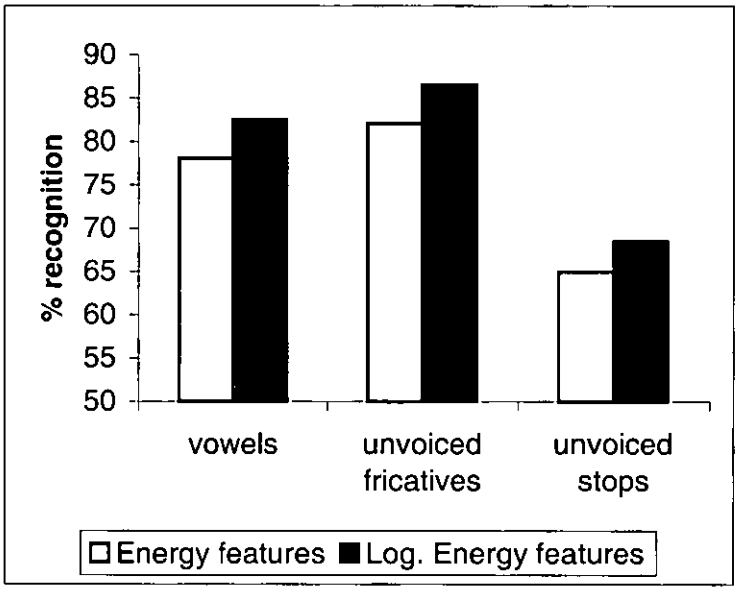


Figure 4.6: Comparative performance of energy and log energy features using 6 features in 32ms frame duration.

Since different phonemes have different spectra, it is obvious that to have the best recognition for a phoneme a specific admissible tree structure will be more suitable. However, this structure will not perform well for other classes of phonemes. In order to have an admissible tree structure that gives good recognition for all classes of phonemes, an analogy from the Mel scale that is used in MFCC can be used to give a similar structure. The next section discusses about the realisation of this filter structure using the AWPs.

4.3.2 Mel scaled wavelet filter bank

The Mel scale has been used since the 1980's for the design of filter bank structures to extract the MFCC-based features for speech recognition. These features have shown better performance as compared to all other features. Since AWP decomposition has the ability to partition the frequency axis in any desired manner, it was logical to think of designing a Mel scaled wavelet filter bank using the AWPs. Since a dyadic decomposition is used, the frequency band formed would not exactly follow the Mel scale. The signal bandwidth of 8kHz was split into 24 bands that closely follow the Mel scale [12]. The tree structure

obtained is shown in Figure 4.7 where the upper branch gives the higher frequency band and the lower branch gives the lower frequency band. The bandwidth of the filters obtained by using AWP decomposition is given in Table 4.5. This table also shows the corresponding Mel scaled filter central frequency and bandwidth used for the calculation of the MFCC. A similar approach has also been tried in [6] with 4kHz speech, the difference being that instead of using the DCT, the wavelet transform has been used on the 24 band log energies. This paper compare the performance achieved for stressed phoneme recognition by 24 wavelet coefficients and 20 MFCC features. However, it has been found that the best recognition performance is achieved with about 13 MFCC features and the performance may degrade if the number of features is increased.

For the calculation of phoneme features a window of 32ms duration was selected and decomposition by the AWP was performed to split the signal into 24 bands (as shown in Figure 4.7). The energy in each of these frequency bands was calculated and logarithmic compression applied to it. The Discrete Cosine Transform (DCT) was then applied to these 24 coefficients and the first 13 DCT coefficients were taken as features. The 13 MFCCs were also derived from 32ms duration of the speech signal. The LDA was used to classify the features extracted from the vowels, unvoiced fricatives, voiced fricatives, unvoiced stops and voiced stops. The results obtained are shown in Figure 4.8. It can be observed that the recognition performance for the voiced phoneme except for stops is better with the MFCC. This is due to the reason that the MFCC uses the STFT, which uses the sine and cosine bases. These bases are more efficient to extract the periodic structure from a signal. For the case of an unvoiced phoneme the features derived by AWP analysis are superior.

Table 4.5: Frequency bands of a wavelet-based Mel filter

Wavelet Filter				Mel Filter	
Filter number	Lower cut off frequency (Hz)	Higher cut off frequency (Hz)	Bandwidth (Hz)	Central frequency (Hz)	Bandwidth (Hz)
1	0	125	125	100	100
2	125	250	125	200	100
3	250	375	125	300	100
4	375	500	125	400	100
5	500	625	125	500	100
6	625	750	125	600	100
7	750	875	125	700	100
8	875	1000	125	800	100
9	1000	1125	125	900	100
10	1125	1250	125	1000	124
11	1250	1375	125	1149	160
12	1375	1500	125	1320	184
13	1500	1750	250	1516	211
14	1750	2000	250	1741	242
15	2000	2250	250	2000	278
16	2250	2500	250	2297	320
17	2500	2750	250	2639	367
18	2750	3000	250	3031	422
19	3000	3500	500	3482	484
20	3500	4000	500	4000	556
21	4000	5000	1000	4595	639
22	5000	6000	1000	5278	734
23	6000	7000	1000	6063	843
24	7000	8000	1000	6954	969

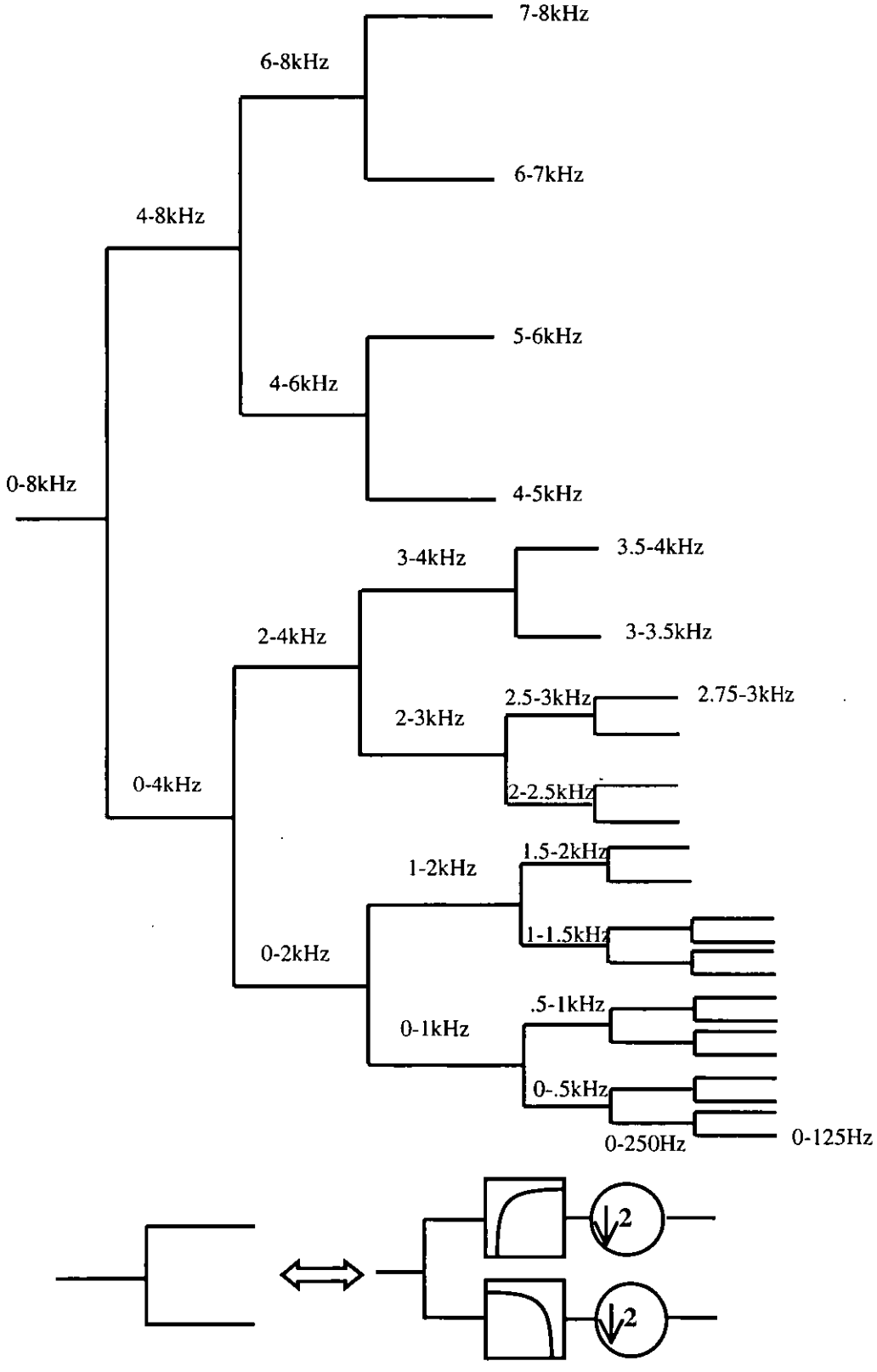


Figure 4.7: Admissible wavelet tree structure for 24-band filter.

The reason of getting better recognition in voiced stops by AWP analysis is because the stops have a sudden burst of high frequency and the signal is not stationary during the 32ms duration. This results in spilling of energy to the adjacent frequency bands when STFT processing is used, causing variations in the feature calculation. These features are correctly picked out by wavelet analysis because of its compact support, which thus improves their classification performance.

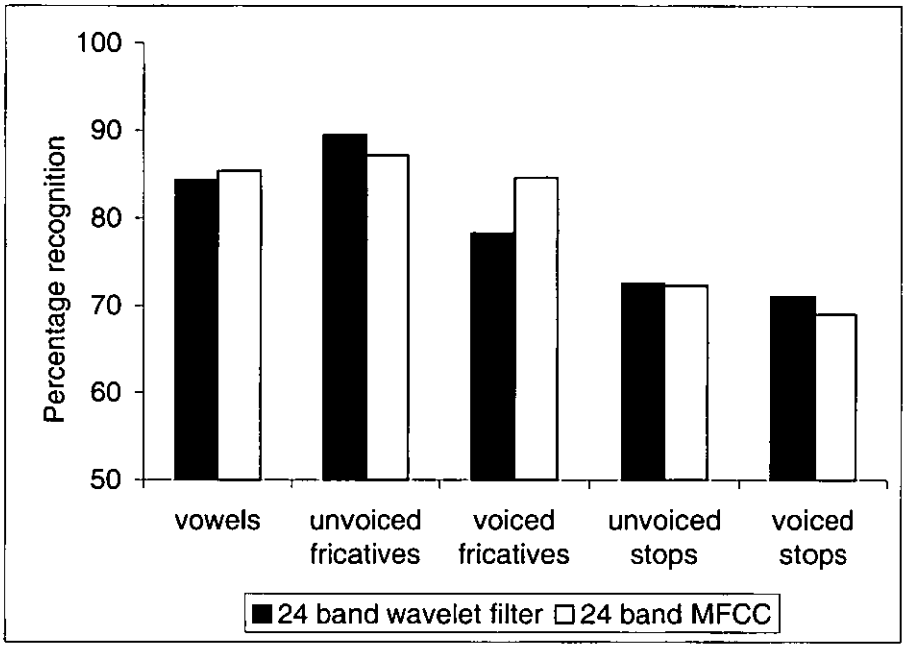


Figure 4.8: Classification of phonemes (32ms duration) using LDA classifier.

In the next experiment the frame duration was further split into sub-frames of 8ms duration each. This gave a total of about 52 features in 32ms duration. It is evident, by comparing the recognition score in Figure 4.8 and Figure 4.9, that the overall improvement in the recognition performance by AWP-based features found in the former is not reflected in the later. This is because of the fact that the signal is almost stationary during the 8ms duration and hence it is suitable for decomposition by the STFT. Since the wavelet can handle the non-stationary signal as well so it is more effective for large frame duration as compared to the STFT.

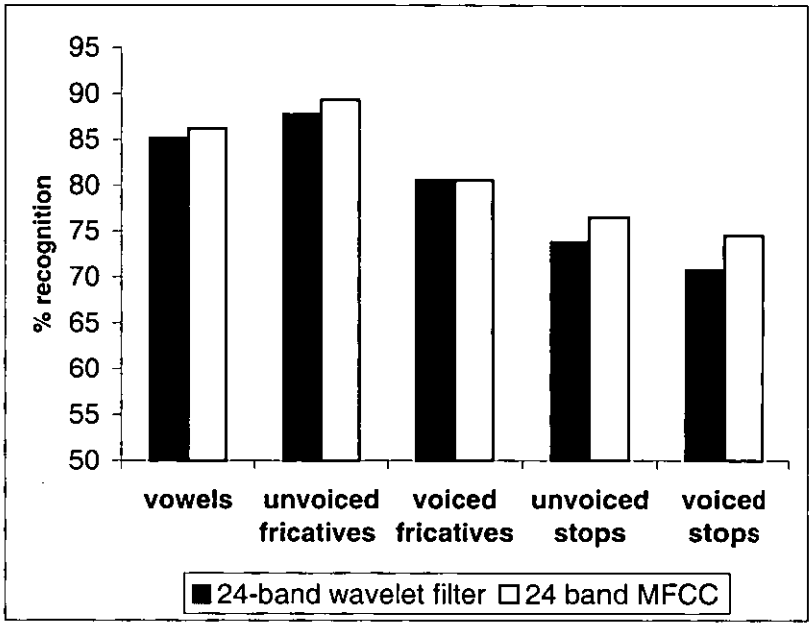


Figure 4.9: Recognition performance of the phonemes with features being extracted every sub-frame of 8ms duration.

Some of the techniques that can give an idea of the effectiveness of features for classification are Euclidean distance, Mahalanobis distance, Fisher's discriminant etc. Of these, Fisher's discriminant is superior because it accounts for the within class spread as well as the between class separation [13]. The next section gives a brief introduction to Fisher's discrimination criterion and uses it to evaluate Fisher's class separability for the MFCC and the AWP-based features.

4.4 Fisher's Discriminant

The ability of a feature to separate two classes not only depends on the distance between the classes but also the scatter within the classes. The Fisher's discriminant is based on these two measures and is given by:

$$f = \frac{(\mu_1 - \mu_2)^2}{\sigma_1^2 + \sigma_2^2} \tag{4.3}$$

where μ_1 and μ_2 are the mean and σ_1 and σ_2 are the variance of the two classes. Figure 4.10 shows the distribution of features V_1 and V_2 for the two-class identification problem. The between-class separation using feature V_1 is higher

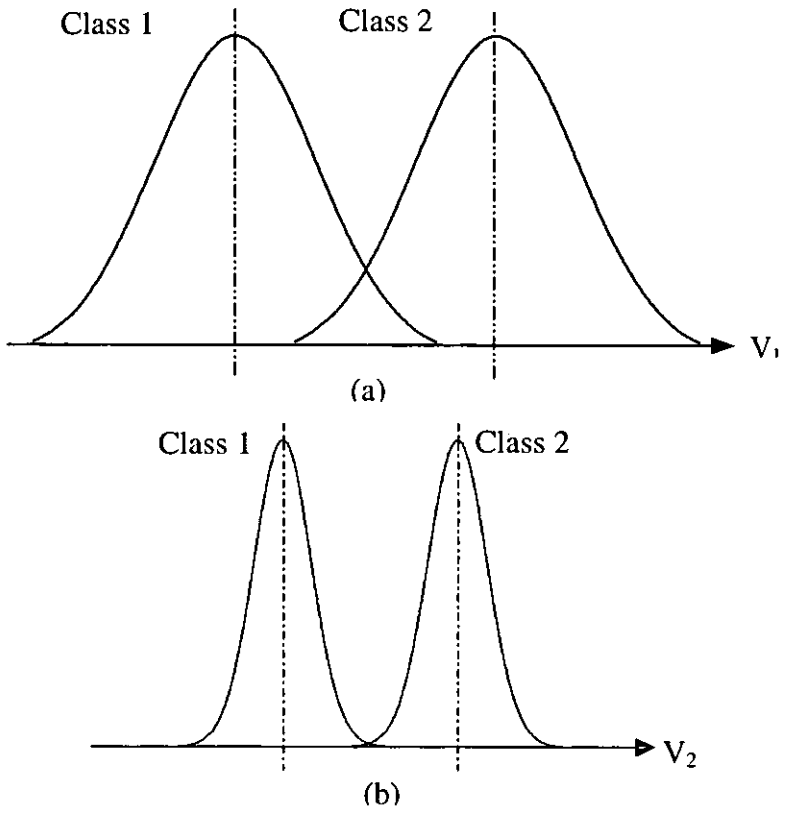


Figure 4.10: (a) Higher class separation and higher within class scatter (b) lower class separation and lower with-in class scatter.

but due to the lower variance of the feature V_2 it is superior for classification purposes. Thus within-class scatter is also an important parameter for classification purposes. With more than two classes, the class-to-class separation of features over all classes is evaluated. This is done by evaluating the variance of the class means. This variance is then compared to the average width of the distribution for each class, i.e. the mean of the individual variances. This measure is commonly called as the **F** ratio:

$$\mathbf{F} = \frac{\textbf{variance of the means (over all classes)}}{\textbf{mean of the variances (within classes)}} \qquad (4.4a)$$

If there are **n** measurements for each of the **m** different classes, then

$$\mathbf{F} = \frac{\frac{1}{(m-1)}\sum_{j=1}^{m}\left(\mu_j - \bar{\mu}\right)^2}{\frac{1}{m(n-1)}\sum_{j=1}^{m}\sum_{i=1}^{n}\left(x_{ij} - \mu_j\right)^2} \qquad (4.4b)$$

where x_{ij} is the i^{th} measurement of class **j**, μ_j is the mean of all the measurements of class **j** and $\bar{\mu}$ is the mean of all the measurements of all classes.

When **m**=2, Equation 4.4b reduces to Fisher's discriminant; therefore the **F** ratio is also referred as the generalised Fisher's discriminant. High **F** ratios means that the scatter among the classes is more than the spread of each class. However, it does not guarantee that none of them will overlap. Although the above measure is simple it is limited to a single feature and also assumes that same number of measurements are available for all the classes. To have a more general measure for the class separability, within-class and between-class scatter matrix are used.

A within-class scatter matrix gives the scatter of features around their respective class expected vector and is expressed by:

$$S_w = \sum_{i=1}^{L} P_i E\left[(x_i - \mu_i)(x_i - \mu_i)^T \mid \omega_i\right] \tag{4.5}$$

where P_i is the probability of occurrence of class ω_i, x_i is the feature vector of class ω_i and **L** is the number of classes. **E[.]** is the expectation operator. The between-class scatter matrix is the scatter of expected vectors around the mixture mean and is given as:

$$S_b = \sum_{i=1}^{L} P_i (\mu_i - \mu_0)(\mu_i - \mu_0)^T \tag{4.6}$$

where μ_0 represents the expected vector of the mixture distribution and is given by:

$$\mu_0 = \sum_{i=1}^{L} P_i \mu_i \tag{4.7}$$

In order to have criteria for class separability, a number is obtained by calculating the trace of the matrix **J** in Equation 4.8. For a set of features to be good for classification the value of **J** should be higher [13].

$$J = S_w^{-1} S_b \tag{4.8}$$

The results obtained by using Equation 4.8 for the 24-band AWP-based features and the MFCC features are shown in Table 4.6 where the features are extracted every 32ms and 8ms duration. It was found that the value of J is higher for the 24-band wavelet features as compared to the MFCC features when the feature extraction process is done after every 32ms while lower in the other case. This is due to the fact that the long duration signal of 32ms cannot be strictly taken as stationary, due to this reason the STFT is not suitable for time-frequency decomposition of the signal. Hence, some of the features are not properly picked up by the MFCC while the wavelet transform can do this effectively. Since the speech is almost stationary for a short duration of 8ms so this problem is not encountered here.

Table 4.6: Calculation of class separability by 24-band wavelet and MFCC features.

	24-band wavelet features	MFCC features
Features extracted every 32ms	0.0355	0.0346
Features extracted every 8ms	0.0167	0.0178

4.5 Summary

From the results obtained in this chapter it becomes clear that the log energy-based AWP features give better recognition performance as compared to the DWT-based features. This is because of the flexibility of splitting the lower or higher frequency bands by the AWPs. This capability has further been exploited to design a 24-band filter that closely follows the Mel scale. The 13 features extracted using this filter structure gave better classification of the unvoiced phonemes as compared to the MFCC-based features when the features were extracted after every 32ms. Calculating the Fisher discriminant measure further supports these results. However, if the features are extracted more frequently (every 8ms), the recognition results of the MFCC-based features become better. The features are shift invariant and are not very dependent on the speakers as seen in the results. The features extracted should not just satisfy the above criterion but also be robust to noise. To establish robustness of these

features, their performance in the presence of different levels of noise is to be tested. This issue of robustness of features is discussed in the next chapter and experimental results are reported for noisy phoneme recognition.

4.6 References

[1] R. R. Coifman and M. V. Wickerhauser, "Entropy base algorithm for best basis selection", *IEEE Transactions on Information Theory*, vol. 32, pp. 712-718, 1992.

[2] S. Chang, Y. Kwon and S. Yang, "Speech feature extracted from adaptive wavelet for speech recognition", *Electronics Letters*, vol. 34, no. 23, pp. 2211-2213, 12th November 1998.

[3] C. J. Long, *Phoneme discrimination using non-linear wavelets methods*, Ph.D. thesis, Loughborough University, Dept. of Electronic and Electrical Engineering, February 1999.

[4] C. J. Long and S. Datta, "Wavelet based feature extraction for phoneme recognition", *Proceedings of 4th International Conference on Spoken Language Processing*, Philadelphia, USA, vol. 1, pp. 264-267.

[5] E. Lukasik, "Wavelet packet based features selection for voiceless plosives classification", *Proceedings of International Conference on Acoustic Speech and Signal Processing*, Istanbul, Turkey, vol.2, pp, 689-692, 2000.

[6] R. Sarikaya and J. H. Hansen, "High resolution speech feature parametrization for monophone-based stressed speech recognition", *IEEE Signal Processing Letters*, vol. 7, no. 7, pp. 182-185, July 2000.

[7] S. Mallat, *Wavelet tour signal of signal processing*, Academic press, 1999.

[8] I. Cohen, S. Raz and D. Malah, "Orthonormal shift-invariant adaptive local trigonometric decomposition", *Signal Processing*, vol. 57, no. 1, pp. 43-64, February 1997.

[9] I. Cohen S. Raz and D. Malah, "Orthonormal shift-invariant wavelet packet decomposition and representation", *Signal Processing*, vol. 57, no.3, pp. 251-270, March 1997.

[10] O. Farooq and S. Datta, "Dynamic feature extraction by wavelet analysis", *Proceedings of 6th International Conference on Spoken Language Processing*, Beijing, China, 16-20 October 2000, vol. 4, pp. 696-699.

[11] O. Farooq and S. Datta, "Modified discrete wavelet features for phoneme recognition", *Proceedings of Workshop on Innovations in Speech Processing*, vol. 23, part 3, Stratford-upon-Avon, UK, pp. 93-99, 2001.

[12] O. Farooq and S. Datta, "Mel filter-like admissible wavelet packet structure for speech recognition", *IEEE Signal Processing Letters*, vol. 8, no. 7, pp. 196-198, July 2001.

[13] K. Fukunaga, *Introduction to statistical pattern recognition*, Academic Press, San Diego, California, USA, 1990.

CHAPTER 5

SPEECH RECOGNITION UNDER NOISY CONDITIONS

Due to recent advancement in ASR technology new applications are growing in areas such as recognition over the telephone network, the mobile network and the Internet. Thus an ASR system has to perform recognition under all these unknown environmental conditions. It has been well established that the performance of the ASR degrades substantially in the presence of a mismatch between the training and the test environment. This mismatch could be caused by various factors, such as noise, microphone distortion, channel characteristics, etc.

In this chapter the recognition performance of the wavelet-based features are evaluated in the presence of additive white Gaussian noise. The reason or choosing the Gaussian distribution is because in the presence of different noise sources, the overall distribution tends to be Gaussian. The performance of these features is also compared with MFCC-based features under the similar noisy conditions for various levels of signal-to-noise ratios (SNR). Further, these features are modified to make them robust to the noise. In addition, a new pre-processing based on wavelet denoising is also proposed in this chapter. This acts as a front end to reduce the effect of noise on speech signals before the feature extraction phase.

5.1 Introduction

Figure 5.1 shows a block diagram of a speech signal corrupted by noise at different stages while passing through a communication channel. First the signal is passed through a microphone, which picks up the background or ambient noise $d_1[n]$ along with the speech signal. This noise can be due to a noisy automobile, helicopter, aircraft cockpit or factory, etc. If the frequency response of the microphone is not flat over the speech signal bandwidth, it will also introduce

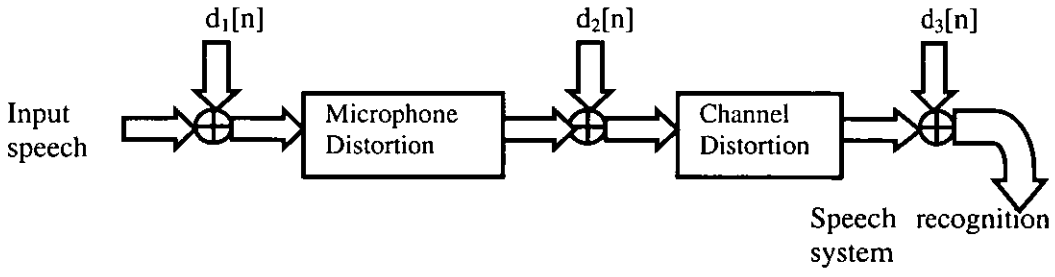


Figure 5.1: Addition of noise and distortion at different level when speech is transmitted through a channel.

some amount of distortion. This causes further degradation in the quality of speech. It is also possible that the microphone may reduce the bandwidth of the speech signal due to its poor and band-limited frequency response. At this stage, the speech signal can be affected by the emotions of the speaker (i.e. anger, fear, etc).

If the signal is passed through a channel, which may be a telephone or a mobile channel, degradation in the quality of speech signal occurs. These channels due to their limited bandwidth, cause reduction in the bandwidth of the signal, e.g. the telephone channel will restrict the bandwidth of the signal to 3.4kHz approximately. In addition to the bandwidth reduction a mobile channel will also introduce severe distortion in the signal due to the effect of fading (caused by multi-path propagation). The component $d_2[n]$ in Figure 5.1 shows the noise added to the signal while passing through the channel. The noise at the receiving end is modelled by $d_3[n]$, which is added to the signal after coming out of the channel. In order to have a robust ASR the above effects should be minimised. For this purpose various techniques have been proposed, some of which are discussed in the next section.

5.2 Robust Speech Recognition System

For robust ASR systems the basic strategies currently used are based on the following two approaches:

- Robust feature extraction
- Compensation technique

The first approach is based on the extraction of the features that are inherently resistant to noise. The techniques used under this category are RASTA (RelAtive SpecTrA) processing [1], one-sided auto-correlation LPC [2] and auditory model processing of speech [3]. The second approach is based on the compensation model, which tries to recover the original speech from the corrupted speech in the feature parameter domain or at the pattern-matching stage. Methods using the second approach are cepstral normalisation [4], probabilistic optimum filtering [5], [6] and parallel model combination [7].

RASTA processing performs filtering of the spectral components and tries to remove the changes that occur faster or slower than a typical range of change of speech signal [1]. This results in the removal of some of the noise components, thereby giving an improvement in the performance of an ASR in the presence of additive noise and different channel conditions. However, RASTA processing may not always give an optimal solution for noisy speech recognition [8]. A similar strategy based on the high-pass filtering approach is used in delta features [9]. This is calculated by taking the time derivative of the features, which is insensitive to a constant bias. This has the disadvantage of having only the transition information and hence it has to be used along with the other features for recognition. Cepstral mean subtraction is also an alternative way of high-pass filtering the cepstral coefficients. This is achieved by subtracting the short time average of the cepstral vector from the current cepstral vector [10].

If the noise is uncorrelated with the speech signal and is stationary, then short time auto-correlation (unbiased estimation [2]) of the signal will result in the auto-correlation of the signal plus the auto-correlation of noise. If one side of the auto-correlation is selected and high-pass filtering is applied, it will remove the slowly varying component. This causes the noise to be removed, since it is assumed to be stationary, leaving a clean one-sided auto-correlation of the

speech. The feature extraction process, which is similar to that of the MFCC features except that one-sided auto-correlation is used instead of the speech signal. A similar analogy has been taken in [11] where instead of using the auto-correlation (time domain approach); the power spectrum (frequency domain approach) is used for feature extraction. Here, differences in the power at the output of the filter are used as features. Both of these techniques have shown improvement in the recognition performance in the presence of different levels of SNR over the MFCC features. Although these techniques give some improvement in the recognition process, they have a basic assumption that the noise is stationary. These techniques will not be effective if the noise property changes with time as in the case of mobile environment.

The human auditory system is known to be robust to background noise, so in order to have speech recogniser with robust performance under noisy conditions, the human auditory system has been studied and a model has been proposed. Kim [3] proposed a simple zero crossing with peak amplitude (ZCPA) model as a robust front end for a speech recognition system in noisy environments. It consists of a bank of band-pass cochlear filter and a non-linear stage at the output of each filter. Kim [12] has also studied the performance of the ZCPA model and suggested that the cepstral measures based on this model show better performance as compared to MFCC under different noisy conditions.

Cepstral normalisation has been carried out by various techniques in order to reduce the effect of noise and channel distortion. SNR-dependent cepstral compensation (SDCN) [4] is based on the stereo database with simultaneous recording of the speech in the training and the testing environments. The individual frames are partitioned into subsets according to SNR in the testing environment. Compensation vectors corresponding to a given range of SNR are calculated by evaluating the average difference between the training and testing environments. When a new utterance is presented to the classifier, the SNR is evaluated first for each frame and then an appropriate compensation vector is added to the computed feature vector. Fixed Codeword-Dependent Cepstral Normalisation (FCDCN) [4] is a similar technique to SDCN with the difference that for each SNR, vector quantisation (VQ) is performed to compute the cepstral vector. This shows a small improvement in the recognition

performance as compared to SDCN [4]. If a new environment is presented in the test speech then the compensation corresponding to the environment closest to the new environment is applied.

The above cepstral compensation techniques are based on empirical procedures and perform compensation frame by frame. The Codeword-Dependent Cepstral Normalisation (CDCN) [4] is a model based on the compensation approach that tries to remove the effect of additive noise with linear filtering. This is achieved by using a maximum likelihood parameter estimate of the noise followed by a linear filtering effect. Although this requires a lot of computation the environment changes are slow, hence the computation of these parameters is not done frame by frame.

The Probabilistic Optimum Filtering (POF) [5], [6] approach is based on the assumption that the clean speech cepstrum can be derived from the noisy cepstrum by using a linear transformation. In order to train the POF filter, stereo pairs of noisy and clean cepstral vectors are used at different levels of SNR. During the testing phase the SNR for a sentence is computed and the transformation with the nearest SNR value is used to get the clean cepstral vector.

The parallel model combination is used to alter the parameters of a set of HMM based acoustic models, so that they reflect speech in a new acoustic environment [7]. This method has been found effective for the speech under noisy conditions as well as channel distortion.

All the above techniques proposed for the robust performance of an ASR are for the STFT-based features. Although there has been some research relating to the use of the wavelets for feature extraction, there has been no study carried out to evaluate its performance in the presence of noise. In the next section, the performance of wavelet-based features under noisy conditions is studied. Further modifications in these features are proposed to make them more robust to white Gaussian noise.

5.3 Experimentation on noisy speech recognition

Recently there has been some research in the use of sub-band based features [13], [14]. These features give an additional advantage of being robust to band-limited noise because this type of noise will affect the corresponding band features only. In this section the possibility of using sub-band features based on AWP is explored. The sub-band feature extraction process is similar to the one explained in Section 4.3.1 except for the fact that some additional processing steps are carried out. This was done to improve the recognition performance for the case of speaker-independent applications and also to make it robust to white noise.

The number of sub-bands chosen for feature extraction was 6 and 8, and AWP was used to give a band structure similar to that shown in Table 4.4. Total energy of the wavelet coefficients in each frequency band was calculated and normalised by dividing it by the number of wavelet coefficients in the corresponding band. A minimum band energy during a sub-frame (8ms duration) was identified and 50% of its value was subtracted from all the band energies. This process has an advantage that negative energy will never be encountered, which may result when noise is estimated or non-linear spectral subtraction is applied [15]. This technique also does not require speech/non-speech detection that is needed for the estimation of noise and is simple to implement. The logarithm of these energies in each band was used as a feature vector [13].

An additional feature, i.e. the variance of the sub-band energies, was also calculated and included in the feature vector. This is useful, as this feature is not affected by a constant offset that may be produced by a different speaker or noise. In the first experiment a 6-band structure derived by the AWP was used and different mother wavelets were tested to evaluate the phoneme recognition performance.

In order to have noisy speech, a white Gaussian noise of zero mean and different variance was generated and injected into the speech signal to obtain different levels of signal-to-noise ratios. Figure 5.2 shows a sample noise power spectral density was used to degrade the speech quality. In the first experiment the AWP that was used to decompose the signal into sub-bands using the

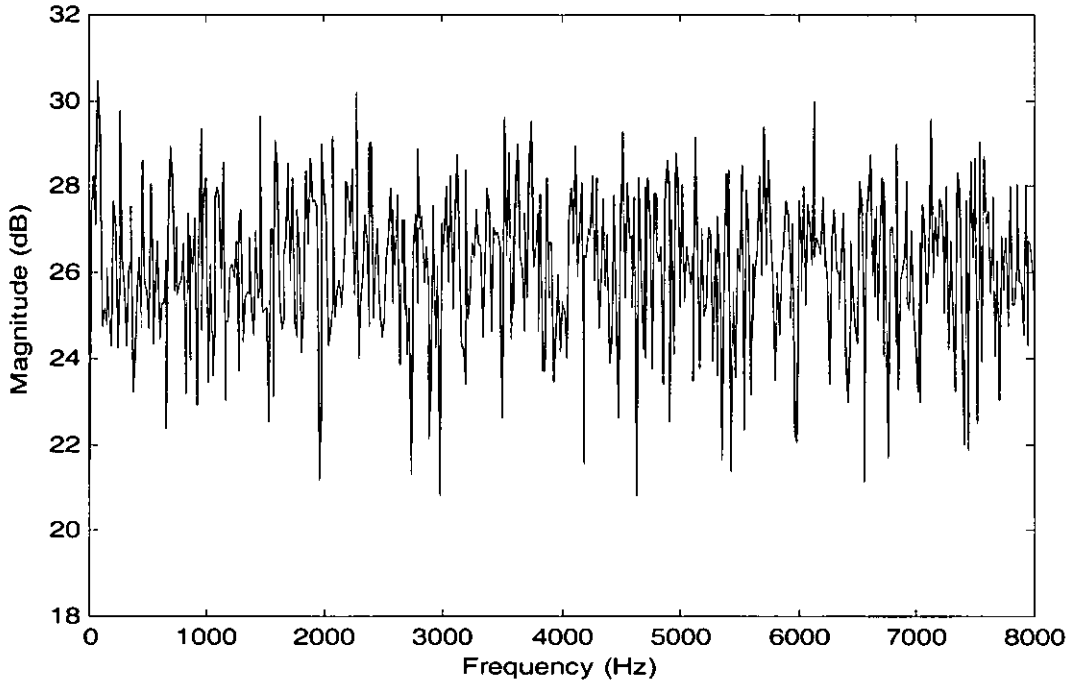


Figure 5.2: The power spectral density of the additive white noise added to the speech.

Daubechies 6-order wavelet (DB6). The frequency band of 0-8kHz was divided into 0-500Hz, 500-1kHz, 1-2kHz, 2-3kHz, 3-4kHz and 4-8kHz sub-bands and the features were calculated as explained earlier. This resulted in 7 features in each sub-frame, giving a total of 28 features for 32ms duration. 13 MFCC-based features were also extracted from each sub-frame duration. The comparative results obtained are shown in Figure 5.3.

It can be seen from the results above that the MFCC-based features perform especially well for vowel recognition, while similar recognition performance is achieved for fricative and stop recognition even though there is a reduction of about 46% in the feature dimension.

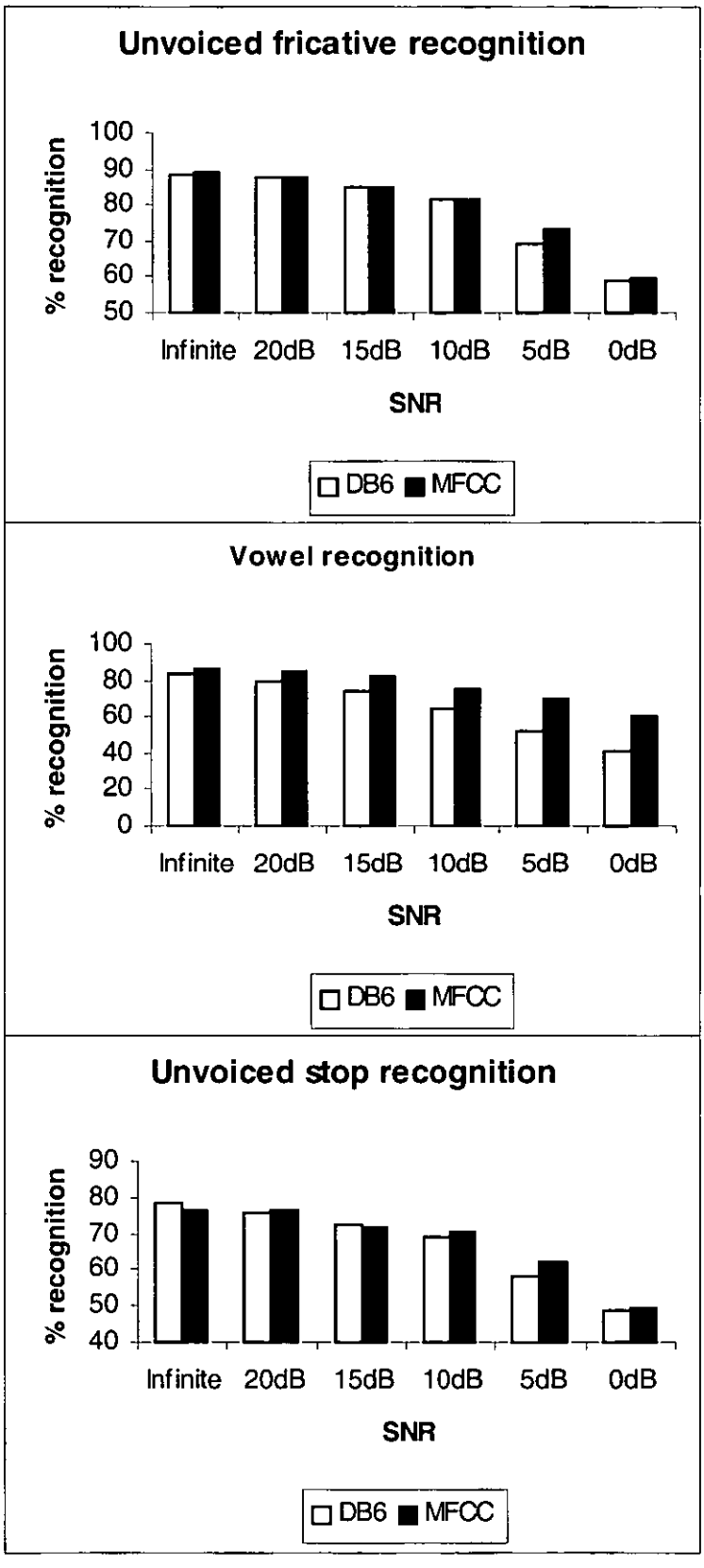


Figure 5.3: Recognition performance of the fricatives, vowels and stops by using 52 MFCC features and 28 AWP (DB6) based log-energy features.

In order to see the effect of mother wavelet on recognition performance, three different orders of Daubechies wavelets (DB2, DB6 and DB20) were tried. By using the higher order wavelet the support size increases and the function becomes smoother. The variation in recognition performance due to the change in the mother wavelet is because of the normalisation of the energy by the number of wavelet coefficients in the sub-band. The number of coefficients after a single decomposition by a mother wavelet 'DB **N**' of a signal **x[n]** is given by:

$$\mathbf{floor}\left(\frac{\mathbf{n}-\mathbf{1}}{\mathbf{2}}\right)+\mathbf{N} \tag{5.1}$$

where **n** is the length of decomposed signal and **floor(x)** is the greatest integer less than or equal to **x**. From Equation 5.1 it is clear that the number of coefficients obtained will depend on the mother wavelet as well as on the length of the input signal (the sub-frame duration). Since the sub-frame duration is fixed, the normalisation factor depends on the factor **N**. By analysing this it becomes clear that by using the lower order mother wavelet puts less relative emphasis on higher bandwidth features compared to the lower bandwidth features. Since the higher bandwidth occurs at higher frequencies, the classifier becomes less sensitive to the higher frequency features when lower order mother wavelets are used. If the multiplying factor (i.e. the reciprocal of the number of coefficients) corresponding to the band with minimum bandwidth is scaled to unity then the multiplying factors for different mother wavelets corresponding to different bandwidths is shown in Figure 5.4. Thus by choosing '**N**' higher the multiplying factor of 4-8kHz band will be larger, thereby giving more emphasis to this band. This may result in an improved recognition performance of phonemes with higher frequencies such as unvoiced fricatives, but it may also reduce the recognition of other phonemes because the higher frequencies carry speaker-dependent information as well. Figure 5.5 shows an improvement in the recognition of vowels and unvoiced fricatives with the increase in the wavelet order, but it is not true for the unvoiced stops. It can be seen that an increase in the wavelet order does not improve the overall phoneme recognition; although, it does increase the computational cost. For this reason, the best performing wavelet i.e. DB20 was chosen for the next phase of the experimentation.

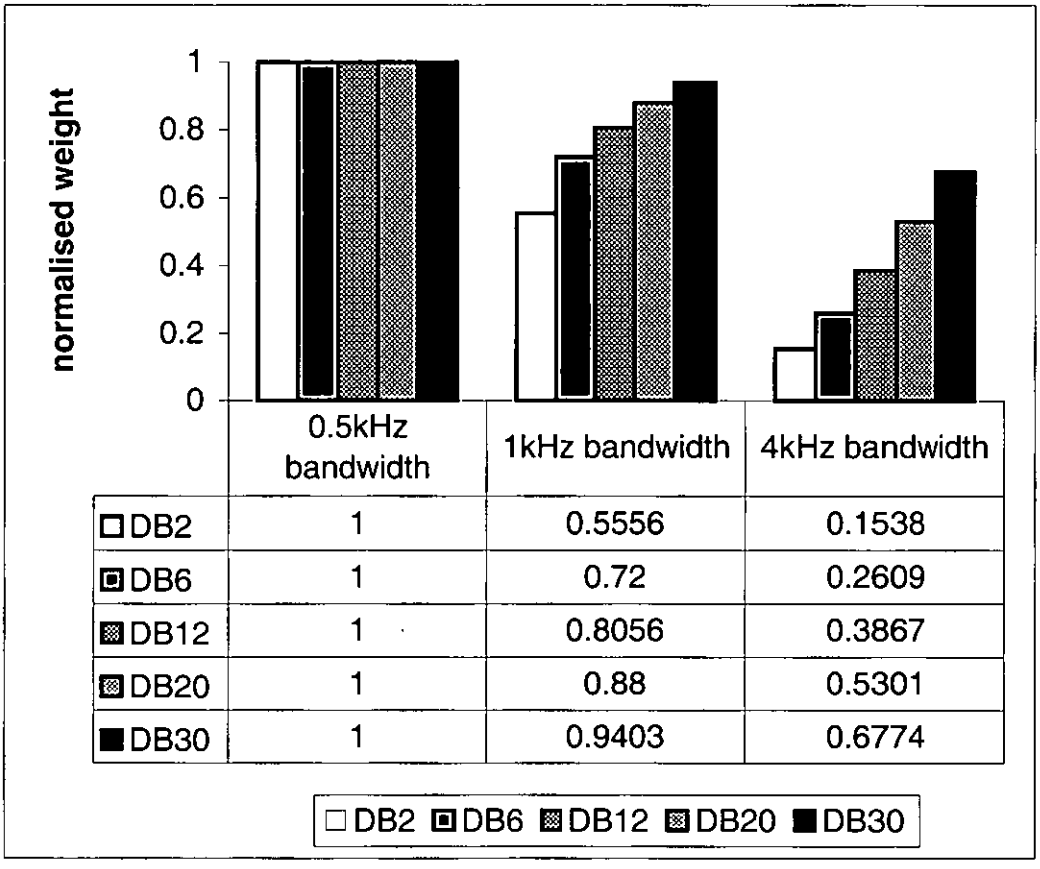


Figure 5.4: The weighting factor for different mother wavelets for a frame of 8ms.

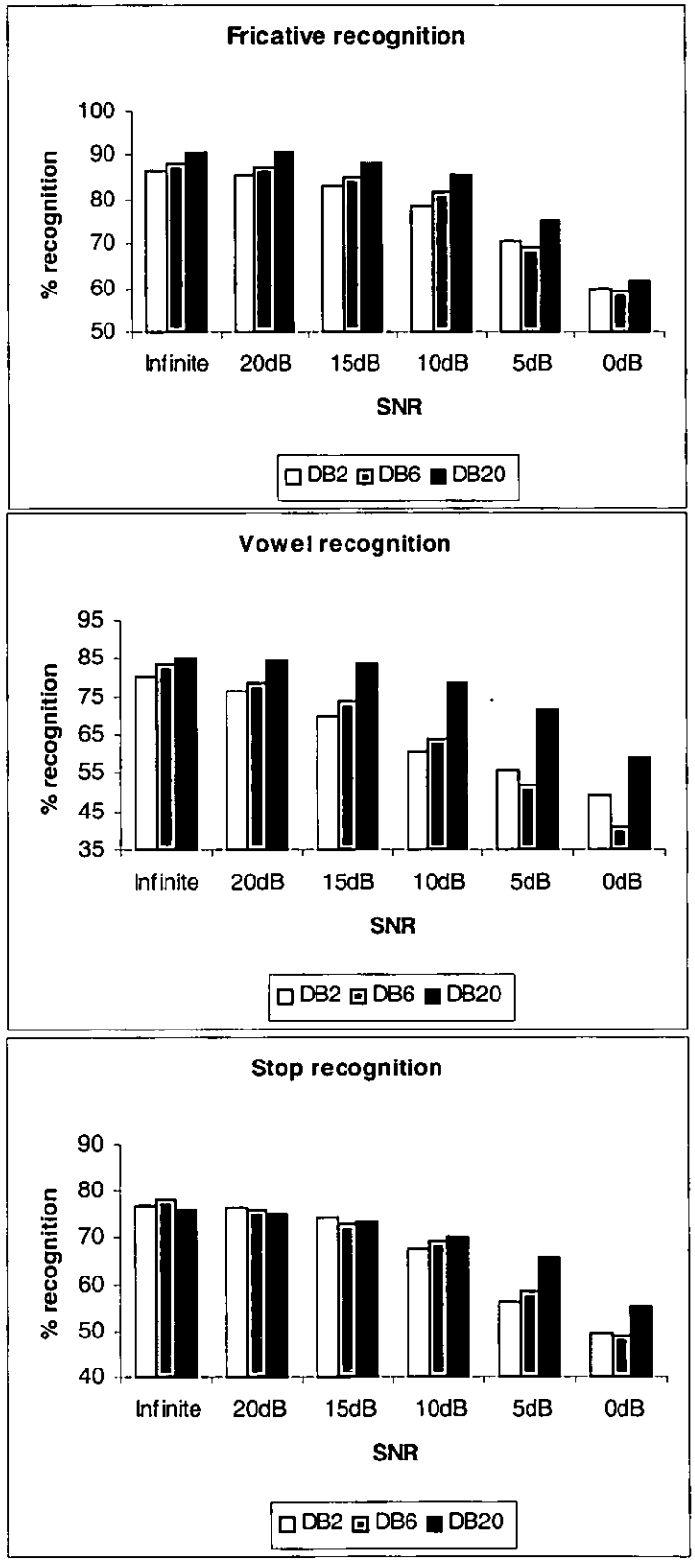


Figure 5.5: Recognition performance by using different order of Daubechies wavelet.

In the next set of experiment the number of sub-bands were increased from 6 to 8 and DB20 was chosen for decomposition. The AWP decomposition was applied such that the following 8 sub-bands; 0-500Hz, 500-1000Hz, 1-1.5kHz, 1.5-2kHz, 2-2.5kHz, 2.5-3kHz, 3-4kHz and 4-8kHz were obtained. Two sets of feature vectors were extracted, the first one was similar to the one extracted in the previous experiment (having 8 normalised log-energy features and one variance feature), while in the second set of feature vector the variance was omitted. The recognition performance achieved is shown in Figure 5.6. It can be seen that there is no significant improvement in the recognition performance when the variance feature is added. The variance feature is dependent on the distribution of sub-band energies. If the number of sub-bands is less (i.e. the bandwidth is larger) then the effect of speaker variation will not cause much variation in the sub-band energies for a given phoneme. This will give the same variance feature for a phoneme for different speakers' i.e. it will be a speaker-independent feature. However, if the number of sub-bands are more (i.e. the bandwidth is small), the energies in the sub-band may be different for a given phoneme for different speakers due to difference in their formant frequencies. This will cause the variance feature to change with the change in speaker making it not suitable for speaker-independent phoneme recognition task. Further, the noise energy (for the case of white noise) in each sub-band will be more uniform if the sub-bands are less in number (i.e. higher bandwidth) and hence the subtraction method proposed will be more effective to remove its effect.

The recognition performance of the features extracted by the 24-band filter derived by the AWP was also evaluated for speech under noisy conditions. The features were extracted every 8ms for a 32ms frame size. This gave a total of 52 features, which has the same dimension as the MFCC. The results obtained are shown in Table 5.1. It is evident from the results that the recognition by using the MFCC features is better than the AWP-based features. This is because 8ms sub-frame was used during which the signal is almost stationary, hence the STFT can extract all features effectively. This table also shows the recognition performance of the voiced stops/fricative using different features under various SNR levels.

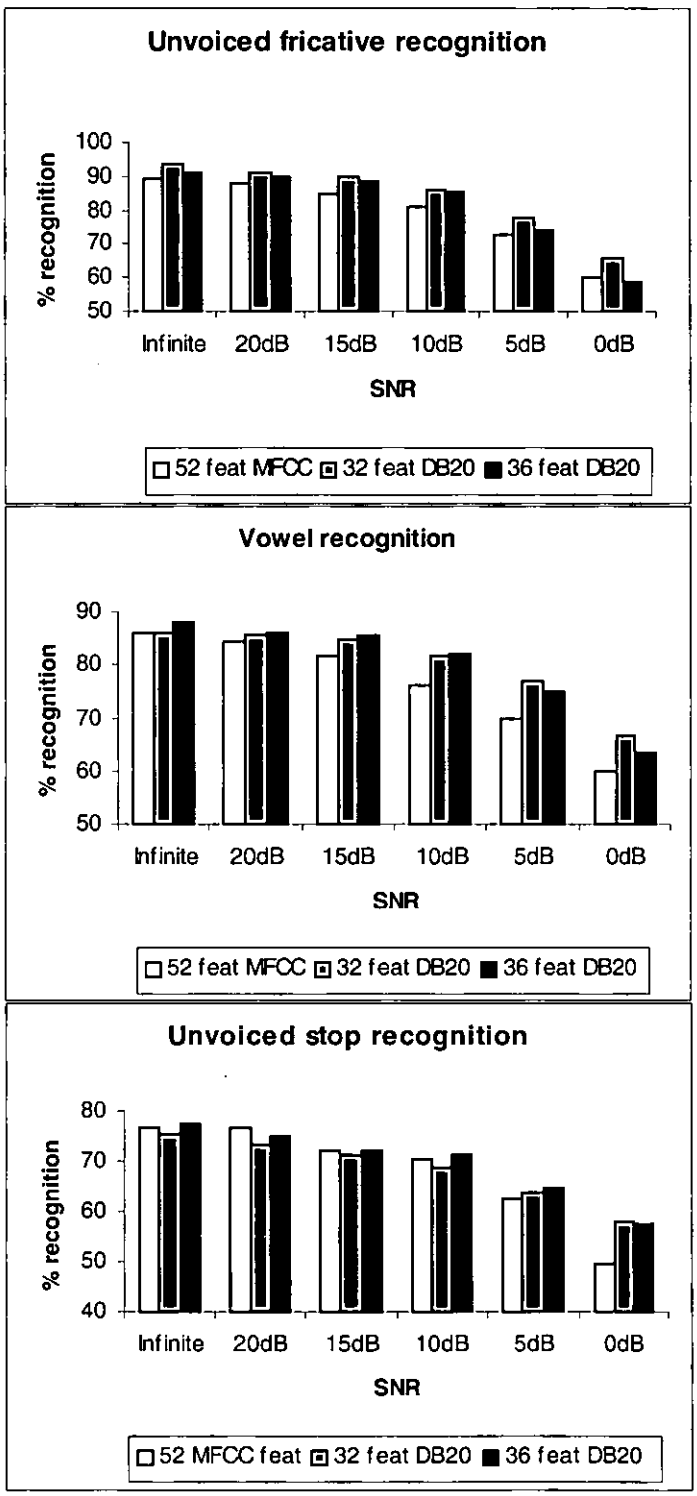


Figure 5.6: Comparative recognition performance achieved by using 52 MFCC and 32 and 36 AWP-based features with DB20.

Table 5.1: Percentage recognition performance of phonemes under noisy conditions with different features

A, B and C: 6 sub-band energies and a variance feature per sub-frame with wavelet DB2, DB6 and DB20 respectively

D: 13 features per sub-frame using MFCC

E: 13 features per sub-frame using AWP with 24-filter bank

F: 8 sub-band energy features per sub-frame using DB20 wavelet

G: 8 sub-band energies and a variance feature per sub-frame using DB20 wavelet.

		A	B	C	D	E	F	G
Unvoiced fricatives	Clean	86.58	88.33	90.66	89.30	87.74	93.58	91.05
	20dB	85.41	87.35	90.47	87.74	86.97	91.05	90.08
	15dB	83.07	85.21	88.52	84.63	84.44	89.69	88.33
	10dB	78.40	81.71	85.41	81.32	81.32	85.99	85.41
	5dB	70.43	69.26	75.29	72.96	71.40	78.02	73.74
	0dB	59.92	59.14	61.67	59.92	61.28	65.95	59.14
Vowels	Clean	80.05	83.67	85.26	86.17	85.15	86.05	87.98
	20dB	76.42	78.80	84.58	84.35	82.42	85.71	85.94
	15dB	69.84	73.47	83.33	81.86	80.95	84.69	85.49
	10dB	60.88	63.72	78.80	75.96	78.46	81.63	82.07
	5dB	55.56	51.93	71.43	69,73	72.79	76.76	75.06
	0dB	48.98	40.82	59.07	59.86	68.71	66.44	63.61
Unvoiced stops	Clean	76.94	78.27	75.83	76.50	73.84	75.38	77.38
	20dB	76.50	76.05	74.94	76.50	69.18	73.17	74.94
	15dB	74.28	72.73	73.39	72.06	68.07	71.40	72.06
	10dB	67.41	69.18	70.29	70.29	64.75	68.74	71.18
	5dB	56.10	58.31	65.63	62.53	60.09	63.86	64.53
	0dB	49.45	49.00	55.21	49.67	50.33	58.09	57.65
Voiced fricatives	Clean	73.48	76.52	81.31	80.56	80.56	78.03	79.56
	20dB	72.22	75.00	77.27	77.02	78.79	79.04	79.29
	15dB	70.20	72.72	74.75	73.74	75.51	75.25	76.26
	10dB	69.70	70.20	73.23	69.95	70.20	72.73	72.47
	5dB	64.39	64.90	70.45	66.41	64.90	65.91	68.18
	0dB	59.34	58.33	63.89	56.57	51.77	57.07	62.37
Voiced stops	Clean	66.61	74.91	77.62	74.55	70.76	71.12	72.56
	20dB	62.64	67.87	70.94	68.77	64.08	65.88	67.69
	15dB	63.90	61.55	67.33	63.54	60.45	61.73	62.45
	10dB	58.12	58.66	61.55	61.19	58.30	59.93	60.11
	5dB	48.92	53.97	58.66	58.12	53.61	57.40	56.68
	0dB	45.85	49.29	51.99	51.99	48.19	50.72	50.90

5.4 Wavelet-based denoising

In the traditional Fourier-based signal processing, out of band noise can be removed by applying the linear time-invariant filtering approach. However, the noise cannot be removed from the portions where it overlaps the signal spectrum. The denoising technique used in the DWT analysis is based on an entirely different idea and assumes the amplitude rather than the location of the spectrum of the signal to be different from the noise. The localising property of the wavelet is helpful in thresholding and shrinking the wavelet coefficients that helps in separating the signal from noise [16]. Denoising by wavelet is quite different from traditional filtering approaches because it is non-linear, due to a thresholding step. Let a signal **x[n]** be given as:

$$\mathbf{x[n] = f[n] + w[n] \quad 0 \le n \le N} \tag{5.2}$$

where **f[n]** is the original signal (which is assumed to be piecewise smooth) and **w[n]** is the white Gaussian noise with zero mean. Denoising of the signal **x[n]** by thresholding involves the following steps:

- Perform a suitable wavelet transform of the noisy data; (the wavelet basis may be chosen based on various factors including heavy computational burden, and ability to compress the energy of the signal into a very few, very large coefficients).
- Calculate the threshold δ depending upon the noise variance.
- Perform thresholding of the wavelet coefficients.
- The coefficients obtained from the step above are then padded with zeros to produce a legitimate wavelet transform and this is inverted to obtain the signal estimate.

The two types of thresholding commonly used are hard and soft thresholding. Usually thresholding is applied on the detailed coefficients (the coefficients at high-pass filter output after down sampling) obtained after wavelet decomposition of the signal and the approximate coefficients (the coefficients at low-pass filter output after down sampling) are left untouched. In the hard thresholding all the coefficients with absolute value below a threshold δ are

forced to zero. Mathematically, for the detailed coefficient $\mathbf{d_{ij}}$ the thresholding is carried out as follows:

$$\tilde{\mathbf{d}}_{ij}^{h} = \begin{cases} \mathbf{d_{ij}} & \text{if} \left|\mathbf{d_{ij}}\right| > \delta \\ \mathbf{0} & \text{if} \left|\mathbf{d_{ij}}\right| \leq \delta \end{cases} \tag{5.3}$$

For soft thresholding, the detailed coefficients are modified as follows:

$$\tilde{\mathbf{d}}_{ij}^{s} = \begin{cases} \mathbf{sign}(\mathbf{d_{ij}})(\left|\mathbf{d_{ij}}\right| - \delta) & \text{if} \left|\mathbf{d_{ij}}\right| > \delta \\ \mathbf{0} & \text{if} \left|\mathbf{d_{ij}}\right| \leq \delta \end{cases} \tag{5.4}$$

where **sign(x)** is +1 if **x** is positive and –1 if **x** is negative. Soft thresholding is similar to that of hard thresholding with the difference that the wavelet coefficients with magnitude above δ are reduced by a factor δ. Due to this reason soft thresholding is also called wavelet shrinkage. The difference between the soft and hard thresholding is shown in Figure 5.7 where the wavelet coefficients are shown before and after thresholding. Shrinkage of the wavelet coefficients is more helpful in reducing the noise from the signal as compared to the hard thresholding method. For this reason it is most commonly used for denoising the signals [17].

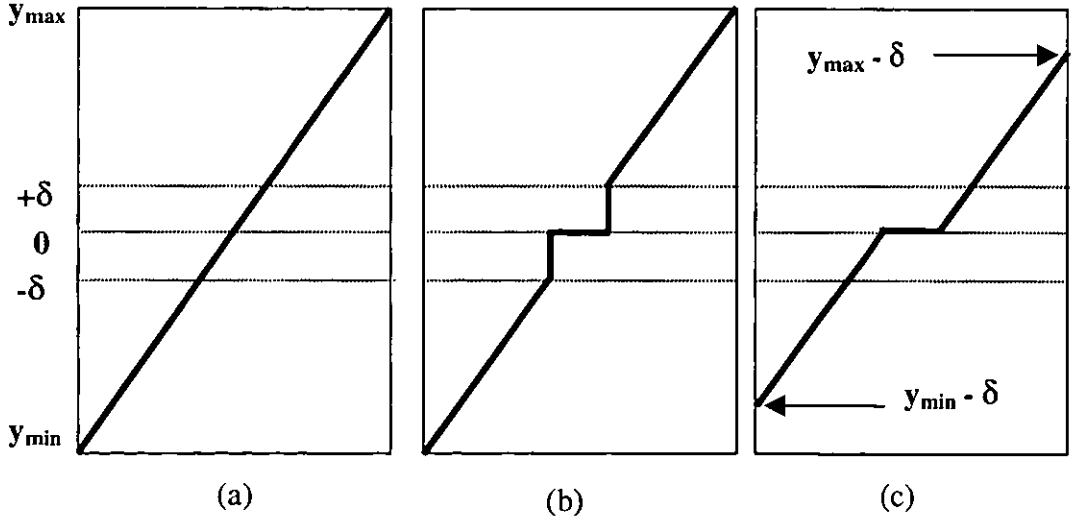


Figure 5.7: Plot of (a) the wavelet coefficients for original signal (b) the wavelet coefficients after hard thresholding (c) the wavelet coefficients after soft thresholding.

Donoho and Johnson [16] have shown that the optimal threshold for denoising is obtained by using the equation below:

$$\delta = \tilde{\sigma}.\sqrt{2.\log_e(N)} \tag{5.5}$$

where $\tilde{\sigma}$ is the noise standard deviation estimate. If the noise is Gaussian random variable then the wavelet coefficients after decomposition are also Gaussian random variable with the same variance σ^2 as noise [17]. A robust estimate of the noise variance is obtained with a median measurement, which is highly insensitive to isolated outliners of potentially high amplitudes. It is important to note that the median is different from the average (mean) value, which is more sensitive to large outliners. If $\{P_k\}_{0\le k<K}$ are **K** independent Gaussian random variables of zero mean and variance σ^2 then

$$E\{Med(|P_k|)_{0\le k<K}\} \approx 0.6745\sigma \tag{5.6}$$

where **E** is the expectation operator and **Med** is the median. The noise standard deviation estimate $\tilde{\sigma}$ is thus given as:

$$\tilde{\sigma} = \frac{1}{0.6745} Med\left(\left|\langle x, \psi_{1,m}\rangle\right|\right)_{0\le m<\frac{N}{2}} \tag{5.7}$$

The noise standard deviation estimate is carried out by performing one-level wavelet decomposition and calculating the median of the absolute values of detailed wavelet coefficients as shown in Equation 5.7. Once $\tilde{\sigma}$ is estimated, thresholding is usually applied to the detailed wavelet coefficients only. It is also possible to perform a '**j**' level wavelet decomposition and apply the thresholding to all the '**j**' detailed coefficients. Figure 5.8 and Figure 5.9 shows the hard and soft thresholding on a noisy speech signal for an unvoiced fricative and a vowel. The 'Daubechies 6' mother wavelet was used for denoising and in both the cases the thresholding was not applied to the approximate coefficients. With two-level discrete wavelet decomposition, where the thresholding is applied to the detailed coefficients only, the signal after denoising is smoother but it also loses some of the high frequency signal components as seen in Figure 5.8(d). This may cause

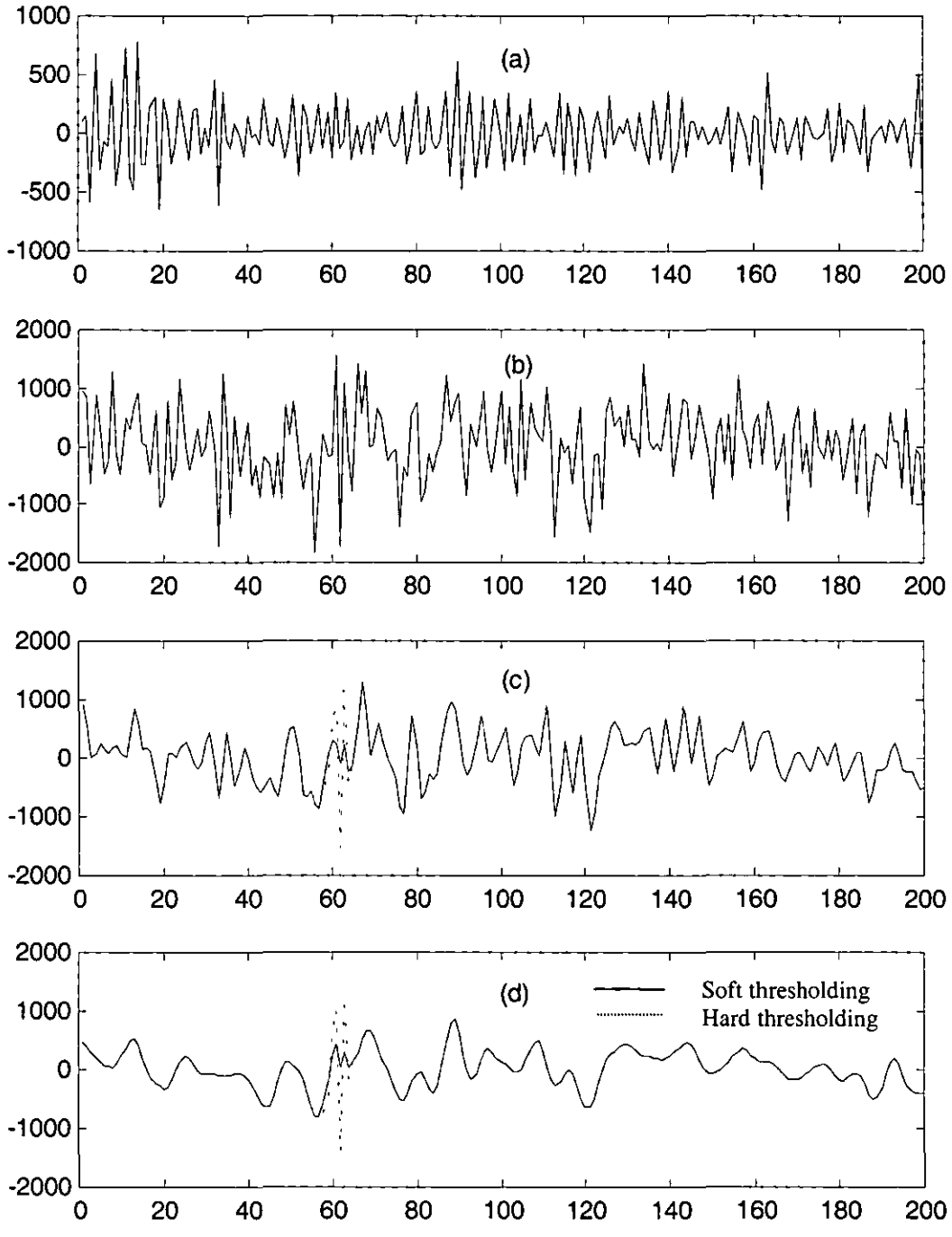


Figure 5.8: (a) Original signal of an unvoiced fricative (b) Signal with additive noise (c) Signal after denoising using one-level of decomposition (d) Signal after denoising using two-levels of decomposition.

reduction in the recognition performance at higher SNR for the phonemes with high frequency components. The unvoiced fricatives have very similar frequency characteristics to that of the white noise, thus application of denoising will cause

not only the removal of the noise but also some of the higher frequency components of the signal even when the single level decomposition is applied. This is clearly evident in Figure 5.8 (c) and the effect is more pronounced in Figure 5.8 (d). Also from Figure 5.8 (c) and (d) it is evident that a large-amplitude noise spike producing a high magnitude wavelet coefficient is retained as such after denoising when hard thresholding is applied but its effect is reduced by using soft thresholding (due to the shrinkage of the wavelet coefficients). This problem may be encountered at low SNR and will be absent at higher SNR.

However, for the vowels and the voiced phonemes the signal energy is mostly concentrated at the lower frequency end of the spectrum. Thus the denoising is expected to perform better in removing the noise for these phonemes as compared to the fricatives. This is evident if the Figure 5.8 and Figure 5.9 are compared.

Figure 5.10 shows the effect of the denoising processes on the power spectrum of the phoneme /aa/. The power spectrum of the phoneme changes considerably at the higher frequency end if the white Gaussian noise is added as seen in Figure 5.10 (a) and (b). The difference in the power spectrum of the phoneme is not significant for one-level denoising by soft or hard thresholding as in Figure 5.10 (b). However, a lower power spectral density is found with soft thresholding when two-level decomposition is applied (as seen in Figure 5.10 (c)). This results due to the shrinkage of wavelet coefficients, which reduces the overall amplitude of the signal.

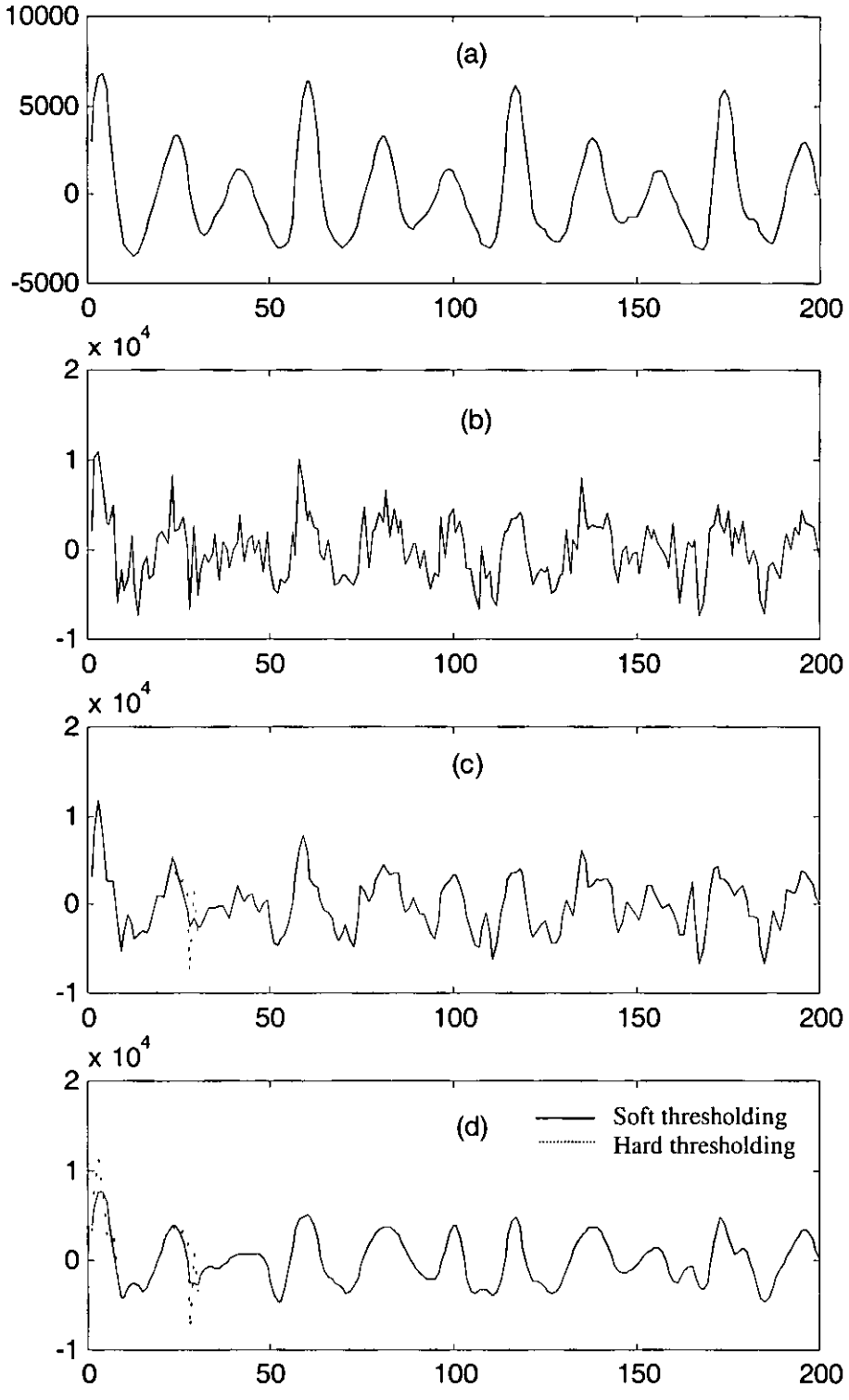


Figure 5.9: (a) Original signal of a vowel (b) Signal with additive noise (c) Signal after denoising using one-level of decomposition (d) Signal after denoising using two-level of decomposition.

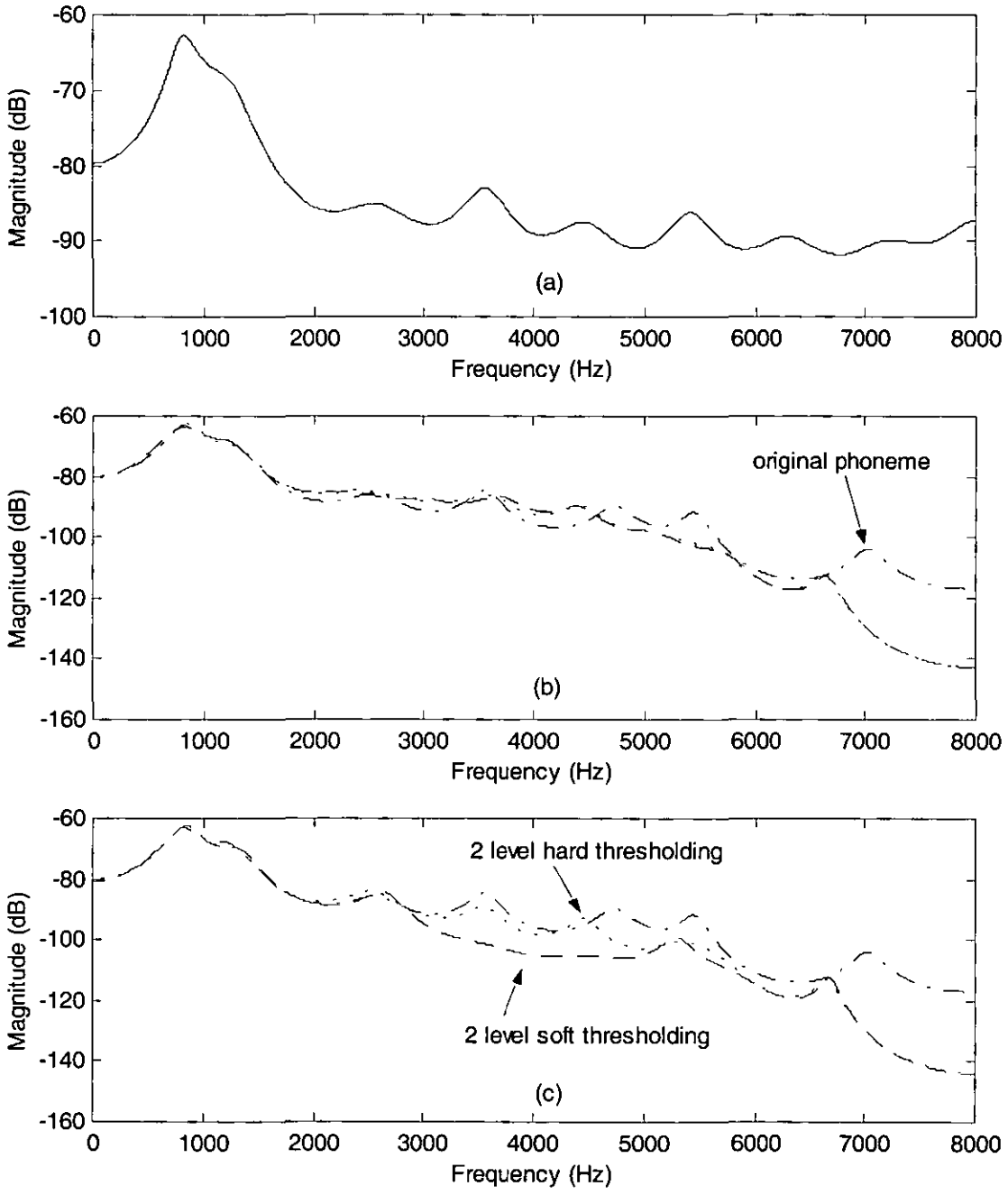


Figure 5.10: The power spectrum of (a) noisy phoneme (b) original phoneme and the phoneme after denoising with soft and hard thresholding with one-level decomposition (c) original phoneme and the phoneme after denoising with soft and hard thresholding with two-level decomposition.

5.5 Denoising for phoneme recognition

In order to reduce the noise from noisy phonemes, a wavelet-based denoising technique is proposed here. Denoising is carried out at a pre-processing stage before the feature extraction, such that the effect of noise on the features extracted is minimised. The block diagram of the proposed scheme is shown in Figure 5.11. The 'Daubechies 6' mother wavelet was used to perform one-level wavelet decomposition of the signal of 32ms frame duration. Since the noise added was white and Gaussian the threshold for denoising was calculated by using Equation 5.5. Soft thresholding of the noisy phoneme was carried out and the thresholding was applied to the detailed wavelet coefficients only. The signal was reconstructed after thresholding and then divided into sub-frames of 8ms duration for feature extraction. Finally the features were extracted by using the sub-band feature extraction technique (explained in section 5.3) in which 6 sub-bands were used. A total of 7 features per sub-frame were extracted using DB20 wavelet and the LDA was used for classification after every 32ms.

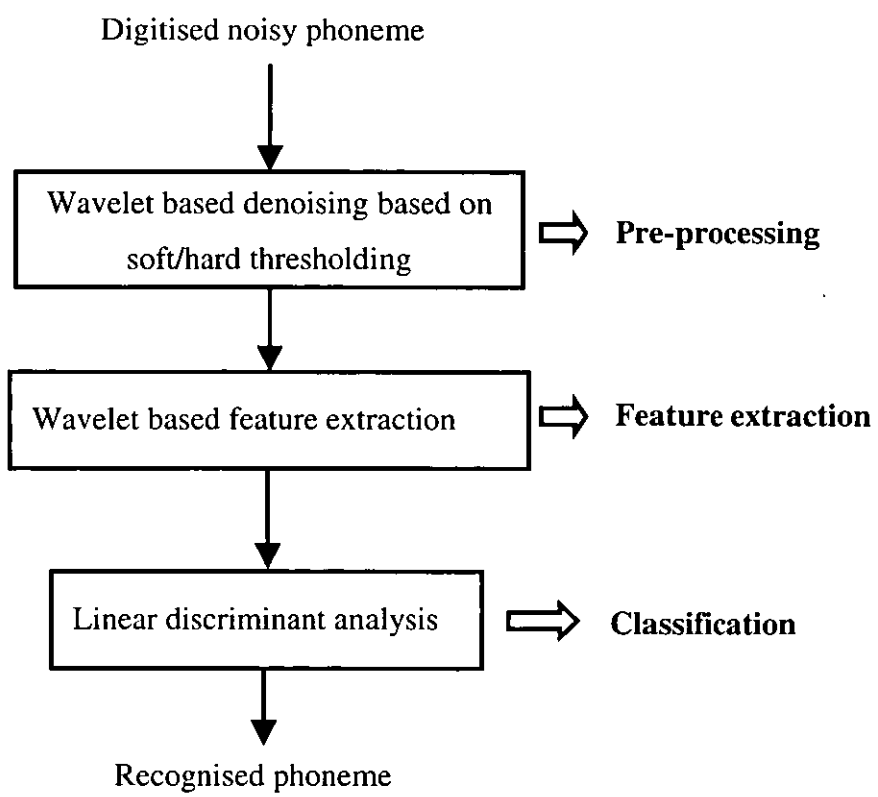


Figure 5.11: Block diagram of proposed robust feature extraction technique using wavelet based denoising.

The recognition performance with and without denoising for unvoiced fricatives is shown in Figure 5.12. Since the unvoiced fricatives have very high frequency components and the characteristics similar to white Gaussian noise, therefore when denoising is applied, it removes some of the signal part also, thereby, reducing the recognition performance for the clean and noisy phoneme. Only for the case of 0dB SNR, the process of denoising removes much of the white Gaussian noise causing the denoising-based features to perform better as compared to features without denoising.

For vowel recognition, the performance is found to be lower (< 1%) as seen in Figure 5.13 for clean and 20dB SNR. But at SNRs below 20dB the denoising helps in improving the recognition performance and the improvement increases as the noise power increases. A maximum improvement of about 24.4% is found at 0dB. The vowels show such improvement in recognition performance because of the periodic nature of the signal, which helps in isolating the noise from the signal part in the wavelet amplitude domain.

Hard thresholding was also tried for the denoising of noisy phonemes and the recognition results obtained are shown in Table 5.2 to Table 5.4. Each table shows the percentage recognition performance achieved when no denoising was used and when soft or hard thresholding was applied. In general it can be seen that with the presence of denoising, the improvement occurs only for very low SNR. Also, the best results are obtained with the vowels and the recognition performance degrades for the unvoiced fricatives. There is no specific advantage of soft or hard thresholding noticed from these results.

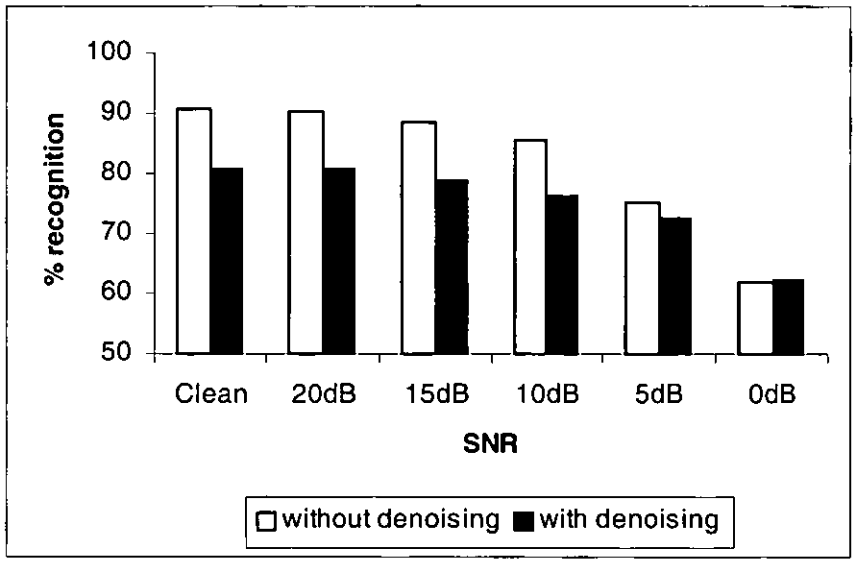


Figure 5.12: Recognition performance of unvoiced fricatives by using 6-band features with and without denoising.

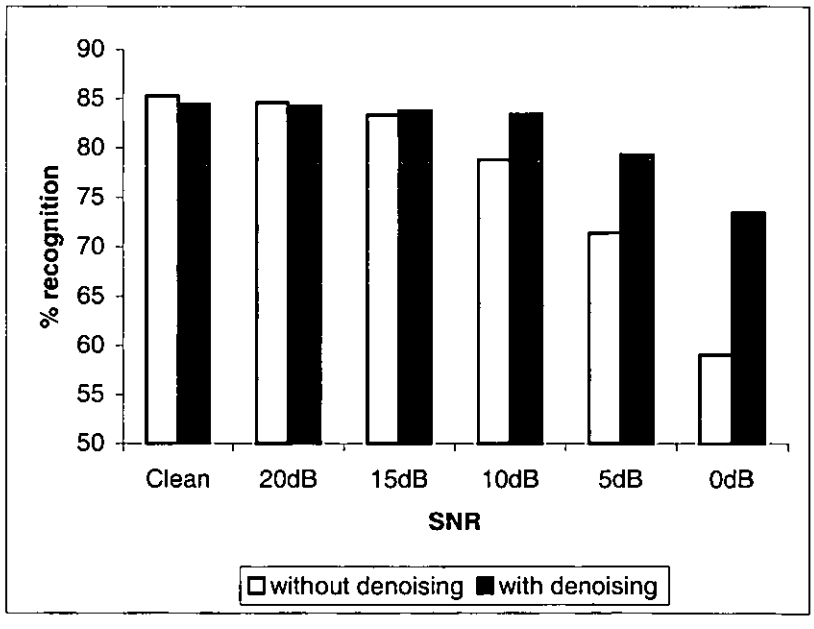


Figure 5.13: Recognition performance of the vowels by using 6-band features with and without denoising.

Table 5.2: The recognition performance of the 6 sub-band based features in the presence of different noise levels for

A: system not using denoising

B: system using denoising with soft thresholding

C: system using denoising with hard thresholding

		A	B	C
Unvoiced fricatives	Clean	90.66	80.74	84.24
	20dB	90.47	80.74	83.27
	15dB	88.52	78.79	82.30
	10dB	85.41	76.26	78.79
	5dB	75.29	72.76	69.84
	0dB	61.67	62.06	60.51
Unvoiced stops	Clean	75.83	76.50	74.72
	20dB	74.94	75.39	75.17
	15dB	73.39	74.26	75.61
	10dB	70.29	70.29	72.73
	5dB	65.63	66.08	66.74
	0dB	55.21	56.54	57.87
Voiced stops	Clean	74.73	75.81	73.47
	20dB	66.79	67.51	65.34
	15dB	63.90	65.16	63.72
	10dB	59.57	61.19	60.11
	5dB	58.21	59.57	58.30
	0dB	51.81	53.25	53.79
Voiced fricatives	Clean	79.29	79.80	76.77
	20dB	75.76	78.54	77.27
	15dB	73.74	78.54	78.28
	10dB	72.47	75.76	77.78
	5dB	68.18	67.93	68.43
	0dB	63.13	57.83	59.85
vowels	Clean	85.26	84.47	84.58
	20dB	84.58	84.24	84.47
	15dB	83.33	83.78	84.13
	10dB	78.80	83.45	83.56
	5dB	71.43	79.37	79.48
	0dB	59.07	73.47	72.00

Table 5.3: The recognition performance of the 24-band AWP-based features in the presence of different noise levels for

A: system not using denoising

B: system using denoising with soft thresholding

C: system using denoising with hard thresholding

		A	B	C
Unvoiced fricatives	Clean	87.74	78.40	77.04
	20dB	86.97	74.90	77.43
	15dB	84.44	73.15	75.68
	10dB	81.32	69.65	69.07
	5dB	71.40	57.00	62.84
	0dB	61.28	49.81	54.09
Unvoiced stops	Clean	73.84	69.40	71.40
	20dB	69.18	66.30	67.84
	15dB	68.07	68.29	69.18
	10dB	64.75	66.30	67.85
	5dB	60.09	61.20	62.97
	0dB	50.33	52.77	54.99
Voiced stops	Clean	70.76	69.86	68.59
	20dB	64.08	65.52	64.98
	15dB	60.45	61.19	61.19
	10dB	58.30	58.30	55.78
	5dB	53.61	58.30	54.51
	0dB	48.19	51.26	50.90
Voiced fricatives	Clean	80.56	77.02	79.29
	20dB	78.79	75.76	77.27
	15dB	75.51	71.72	74.49
	10dB	70.20	65.91	69.19
	5dB	64.90	63.88	63.13
	0dB	51.77	54.29	56.31
vowels	Clean	85.15	85.71	84.81
	20dB	82.42	84.24	83.11
	15dB	80.95	81.86	80.84
	10dB	78.46	79.93	78.91
	5dB	72.79	73.13	72.22
	0dB	68.71	70.18	68.82

Table 5.4: The recognition performance of the MFCC based features in the presence of different noise levels for

A: system not using denoising

B: system using denoising with soft thresholding

C: system using denoising with hard thresholding

		A	B	C
Unvoiced fricatives	Clean	89.30	78.21	78.79
	20dB	87.74	77.43	78.21
	15dB	84.63	74.12	75.88
	10dB	81.32	72.76	75.49
	5dB	72.96	63.04	68.48
	0dB	59.92	54.67	57.20
Unvoiced stops	Clean	76.50	76.05	77.83
	20dB	76.50	74.28	76.94
	15dB	72.06	72.51	73.39
	10dB	70.29	69.84	69.18
	5dB	62.53	63.19	63.41
	0dB	49.67	50.55	51.66
Voiced stops	Clean	74.55	73.29	73.10
	20dB	68.77	61.73	59.39
	15dB	63.54	57.76	54.51
	10dB	61.19	51.99	48.38
	5dB	58.12	50.18	47.11
	0dB	51.99	47.83	46.39
Voiced fricatives	Clean	80.56	78.54	77.02
	20dB	77.02	76.26	72.73
	15dB	73.74	70.45	69.19
	10dB	69.95	66.16	63.38
	5dB	66.41	58.33	56.82
	0dB	56.57	53.54	53.03
vowels	Clean	86.17	84.47	84.81
	20dB	84.35	80.84	81.75
	15dB	81.86	80.50	80.50
	10dB	75.96	74.04	74.38
	5dB	69.73	65.76	67.46
	0dB	59.86	58.50	57.82

5.6 Summary

New sub-band based features have been proposed in this chapter for adding robustness in the recognition performance. The recognition performance with different mother wavelets has also been explored and DB20 was found to be the best for 6 sub-band decomposition. Its performance was found superior to the MFCC-based features for unvoiced fricative and vowel under noisy conditions, although it used 28 features (as compared to 52 for MFCC). No significant improvement in the recognition was noticed if the sub-bands were increased from 6 to 8. Finally a novel pre-processing stage before the feature extraction has been proposed based on wavelet denoising. Both hard and soft thresholding were carried out for denoising the phonemes and the recognition results gave further improvement in the recognition performance under low SNR. However, a reduction in the recognition performance was found for the case of clean phonemes. A marked improvement in the performance of noisy vowels was noticed when the denoising was applied. The assumption that signal is smooth, for the threshold calculation is not totally correct for the fricatives and stops, hence this results in an incorrect calculation of the threshold. This causes the signal to be removed along with the noise thereby altering the features extracted and reducing the recognition performance.

All these experiments carried out takes a phoneme of 32ms duration from the TIMIT database; however, practically speaking this duration may vary over a wide range. Also, classifying on a small set of phonemes does not give an exact idea of the recognition performance that may be achieved for a larger set of phonemes. In order to overcome this difficulty a complete continuous speech recognition system was implemented by using the HTK (HMM Tool Kit) of Cambridge University. This system is described in detail in the next chapter.

5.7 References

[1] H. Hermansky and N. Morgan, " RASTA processing of speech", *IEEE* Transactions *on Speech and Audio Processing*, vol. 2, no. 4, pp. 578-589, October 1994.

[2] K. H. Yuo and H. C. Wang, "Robust features for noisy speech recognition based on temporal trajectory filtering of short-time autocorrelation sequences", *Speech Communication*, vol. 28, pp. 13-24, 1999.

[3] D. S. Kim, S. Y Lee and R. M. Kil, "Auditory processing of speech signals for robust speech recognition in real world noisy environment" *IEEE Transactions on Speech and Audio Processing*, vol.7, no. 1, pp. 55-69, January 1999.

[4] A. Acero and R. M. Stern, "Environmental robustness in automatic speech recognition", *Proceedings of International Conference on Acoustic, Speech and Signal Processing,* 1990, pp. 849-852, 1990.

[5] D. Y. Kim and C. K. Un, "Probabilistic vector mapping with trajectory information for noise-robust speech recognition", *Electronics Letters*, vol. 32, no. 17, pp. 1550-1551, 15th August 1996.

[6] L. Neumeyer and M. Weintraub, "Probabilistic optimum filtering for robust speech recognition", *Proceedings of International Conference on Acoustic, Speech and Signal Processing*, 1994, pp. I-417, I-420, 1994.

[7] M. J. F. Gales and S. J. Young, "Robust continuous speech recognition using parallel model combination", *IEEE Transactions on Speech and Audio Processing*, vol. 4, no. 5, pp. 352-359, September 1996.

[8] H. Y. Jung and S. Y. Lee, " On the temporal decorrelation of features parameters for noise robust speech recognition", *IEEE Transactions on Speech and Audio Processing*, vol. 8, no. 4, pp. 407-416, July 2000.

[9] S. Furui, "Speaker-independent isolated word recognition using dynamic features of speech spectrum", *IEEE Transactions on Acoustic, Speech and Signal Processing*, vol. 34, pp. 52-59, February 1986.

[10] M. G. Rahim, B. H. Juang, W. Chou and E. Buhrke, "Signal conditioning techniques for robust speech recognition", *IEEE Signal Processing Letters*, vol. 3, no. 4, pp. 107-109, April 1996.

[11] J. Xu and G. Wei, "Noise-robust speech recognition based on difference of power spectrum", *Electronics Letters*, vol. 36, no. 14, pp. 1247-1248, July 2000.

[12] D. S. Kim, S.Y. Lee R.M. Kil and X Zhu, "Auditory model for speech recognition in real world noisy environments", *Electronics Letters*, vol. 33, no. 1, pp. 12-13, 2nd January 1997.

[13] O. Farooq and S. Datta, "Robust features for speech recognition based on admissible wavelet packets", *Electronics Letters*, vol. 37, no. 5, pp. 1554-1556, 6th December 2001.

[14] P. M. McCourt, S. V, Vaseghi and B. Doherty, "Multi-resolution sub-band features and models for HMM-based phonetic modelling", *Computer Speech and Language*, vol. 14 no. 3, pp. 241-259, 2000.

[15] F. Korkmazskiy, F. K. Soong and O. Siohan, "Constrained spectrum normalisation for robust speech recognition in noise", *Workshop on Automatic Speech Recognition: Challenges for the new Millennium*, Paris, France, September 2000.

[16] D. L. Donoho and I. M. Johnston, "De-noising by soft-thresholding," *IEEE Transactions Information Theory*, vol. 41, no. 3, pp.613-627, May 1995.

[17] S. Mallat, *A wavelet tour of signal processing*, Academic Press, San Diego, 1998.

CHAPTER 6

CONTINUOUS SPEECH RECOGNITION USING WAVELET-BASED FEATURES

In order to recognise continuous speech, HMM based recognisers are commonly used. In this chapter, before discussing the results based on wavelet and MFCC features a brief review is presented of the HTK system [1], which uses HMM for acoustic modelling.

6.1 Introduction to HMM

The HMM is the most popular and successful stochastic approach to speech recognition generally used. This is due to the availability of efficient training and recognition algorithm. The HMM is used both for acoustic and language modelling. The basic theory of HMM is presented here in brief. The more detailed discussion of HMM theory and its application to speech recognition is given in [2], [3]. The use of HMM for speech recognition assumes that the speech can be divided into segments (typically of 10ms duration), during which the signal is stationary and the parameterisation of the waveform is carried out. However, this assumption is not strictly true specifically in the case of stops. Figure 6.1 shows a three-emitting state HMM model using the left-to-right topology. This topology is virtually the standard used in the case of speech

recognition and is used throughout in this work. The HMMs are characterised as follows:

- It consists of **N** distinct states, which are interconnected to each other. Each state is denoted by $s = \{s_1, s_2, \ldots, s_N\}$ and $q_i(\tau)$ indicates "being in" state s_i at time τ. For the multiple Gaussian mixture component the notation used is $q_{im}(\tau)$, which indicates being in mixture component M_m of the state s_i at time τ.

- The transition from one state to another is given by the state transition probability distribution $A=\{a_{ij}\}$ where

$$a_{ij} = P[q_j(\tau+1)|q_i(\tau)] \qquad 1 \le i,j \le N \tag{6.1}$$

For the special case of HMM where any state can be reached from any other state in a single step, $a_{ij}>0$ for all i,j. For other types of HMM $a_{ij}=0$ for one or more (i,j) pairs. The elements of the state transition matrix must obey the following constraint:

$$\sum_{j=1}^{N} a_{ij} = 1 \tag{6.2}$$

- The output probability distribution **B** of the observation symbol associated with each emitting state s_j is given as:

$$b_j(y(\tau)) = P[y(\tau)|q_j(\tau)] \tag{6.3}$$

where $y(\tau)$ is the feature vector at time τ. If the output distribution is based on discrete elements then the model is known as the Discrete HMM (DHMM). Alternatively if the output distribution is continuous then it is referred to as the Continuous Density HMM (CDHMM). In this case **B** becomes a probability density function. In this work CDHMMs are considered only.

- The initial state distribution is $\pi=\{\pi_i\}$ where

$$\pi = P[q_i(1)] \tag{6.4}$$

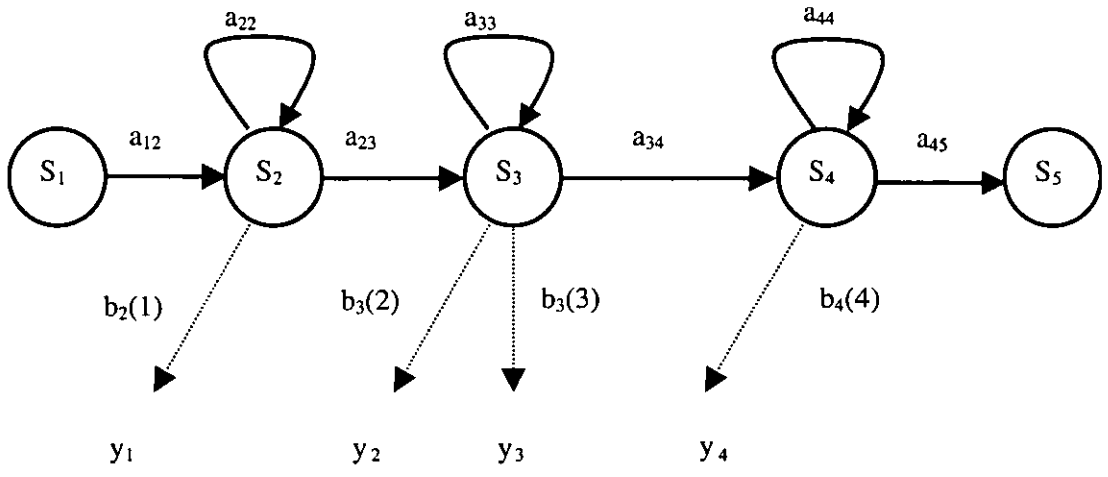


Figure 6.1: Left to right HMM model.

Usually in CDHMM the output probabilities are modelled by multivariate Gaussian distribution or as a mixture of these distributions. The formula for computing $\mathbf{b_j(y(\tau))}$ for the multiple Gaussian mixture components is given by

$$\mathbf{b_j(y(\tau)) = \sum_{m=1}^{M} c_{jm} N(y(\tau); \mu_{jm}, \Sigma_{jm})} \tag{6.5}$$

where **M** is the number of mixture components, $\mathbf{c_{jm}}$ is the weight of the $\mathbf{m^{th}}$ component and $\mathbf{N(y(\tau); \mu, \Sigma)}$ is a multivariate Gaussian with mean $\boldsymbol{\mu}$ and the covariance matrix $\boldsymbol{\Sigma}$, that is

$$\mathbf{N(y(\tau);\mu,\Sigma) = \frac{1}{\sqrt{(2\pi)^n |\Sigma|}} e^{-\frac{1}{2}(y(\tau)-\mu)' \Sigma^{-1} (y(\tau)-\mu)}} \tag{6.6}$$

where **n** is the dimensionality of $\mathbf{y(\tau)}$. The observation sequence $\mathbf{Y_T}$ is [y(1) y(2) .. y(T)]. The matrix **B** consists of a set of means, variances and mixture component weights. For convenience a compact representation of the HMM is used as $\mathcal{M} = (\mathbf{A,B})$. The details of using a HMM for the pattern recognition problem is given in APPENDIX C.

6.2 Speech recognition using HMM

The HMM parameter re-estimation can be carried out using the Baum-Welch algorithm [4], [5], [6]. This could also be used to find out the $\mathbf{P}[\mathbf{Y_T} \mid \mathcal{M}]$ and hence it could be used for recognition; however, in practice, the most likely state sequence associated with $\mathbf{Y_T}$ is commonly used. This likelihood is calculated using the same algorithm as the forward probability calculation (see APPENDIX C) except that a maximum operator replaces the summation. For a given model $\mathcal{M}$, let $\varphi_j(t)$ represent the maximum likelihood of observing speech vectors $\mathbf{y(1)}$ to $\mathbf{y(t)}$ and being in state $\mathbf{s_j}$ at time $\mathbf{t}$. The partial likelihood can be computed efficiently using the following recursion, also known as Viterbi algorithm

$$\varphi_j(t) = \max_{1 \le i \le N} [\varphi_i(t-1)a_{ij}]b_j(y(t)) \tag{6.7}$$

In continuous speech recognition there are a large number of possible word strings and it is not feasible to have a single concatenated model for each word string. Instead, a token passing framework is used [7]. In this scheme each state has a token associated with it. This token contains the history of the model sequence that was taken to get to that point and the current value of $\varphi_j(t)$. When a new vector is observed these tokens are updated and propagated to each state within the model. The most likely token calculated at the end state of each model is then propagated to all the connected models and the history of the token is updated.

In most of the speech recognition systems, it is not practical to propagate all the tokens. In order to reduce the computational load, a thresholding is applied to all the possible paths and those below this threshold are removed or pruned. The threshold is set at a fixed value below the current most likely path. This helps in reducing the computation but may introduce search errors.

The errors encountered during the recognition process of continuous speech can be classified into three categories. The error arising due to substitutions, where the wrong words are hypothesised, deletions, words left out of the hypothesis, and insertions where words are added. In order to minimise the

total number of errors in the recognition system, a fixed transition penalty score is added. This penalty score is used to set the level of insertion and deletions such that the performance is optimised. A grammar scale factor is also used to balance the information of the acoustic and language model. For the case of bigram language model the likelihood of a word $\mathbf{W_j}$ following a word $\mathbf{W_i}$ with associated observed data $\mathbf{Y_T}$, the log-likelihood is given by

$$\log(P[\mathbf{Y_T}, \mathbf{W_j} \mid \mathbf{W_i}]) = \log(P[\mathbf{Y_T} \mid \mathbf{W_j}]) + \mathbf{s}\log(P[\mathbf{W_j} \mid \mathbf{W_i}]) + \mathbf{p} \quad (6.8)$$

where **s** is the grammar scale factor and **p** is the fixed transition penalty.

6.2.1 HMM Parameter Tying

In small vocabulary systems, words can be used for modelling, but it is not possible to do the same for the case of medium and large vocabulary recognition systems. This is because insufficient examples of all the words may be present in the training set to develop a good model of whole words. For this reason it is necessary to tie sets of parameters together. Tying means that two Gaussian mixture components from two different states share the same distribution. Tying can be applied at model level, state level or Gaussian component level. In order to overcome the problem of insufficient data, the words are split into phonemes and phoneme models are developed. All observations of the same phoneme are tied together at the model level. This results in a context-independent model called the monophone set. Another advantage of the monophone model is that unseen words can be split into phonemes by looking up a dictionary, thus giving the capability of recognising new words, which was impossible at the word level model.

Although the monophone system performs well because of tying, it is well known that a phoneme model will not be the same due to the problem of co-articulation. In other words, the model of a phoneme will differ depending upon the context in which it appears. To model this effect, triphone models, also known as context-dependent models, are used. A triphone is a model of a phoneme, given particular preceding and following phonemes. For example the monophone and triphone sequence for the word "beat" is

beat = **sil b iy t sil** monophone sequence

beat = **sil-b+iy b-iy+t iy-t+sil** triphone sequence

The '**sil**' at the beginning and at the end indicates silence. The problem arising in the context-dependent model is because of insufficient data, which causes an inaccurate estimate of the of triphone models. Also there may be unseen triphones in the test set that do not appear in the training phase. To overcome this problem, schemes based on acoustically similar triphones clustering are used. Model-based [8], state-based decision tree clustering [9] and triphone model-based clustering [10] are the techniques used in which the latter has the limitation of handling the seen triphone only.

6.2.2 Recogniser specifications

A recogniser based on HMM was implemented for the task of word recognition. The speech recogniser used CDHMM to model the phonemes. Each context-independent phoneme was modelled by three emitting states, simple left-to-right CDHMM. The context-independent model was used to model context-dependent phoneme and the training was performed by using the Baum-Welch algorithm. Further, eight Gaussian density mixtures were used with state tying. No language model was used and all the words had equal occurrence probability. In order to minimise the total number of errors in the recognition system, a fixed transition penalty score was added. This penalty score was used to set the level of insertion and deletions such that the performance was optimised. This entire scheme was implemented using the HTK system [1].

6.3 Experiments for clean speech recognition

In order to test the recognition performance for continuous speech the TIMIT database was used. The two dialect regions (DR1 and DR2) were selected with two sentences (sa1 and sa2) spoken by all the speakers. The two sentences used are given below, with their corresponding phonetic transcript shown in Table 6.1.

"She had your dark suit in greasy wash water all year"

"Don't ask me to carry an oily rag like that"

Table 6.1: Words and their corresponding phonetic transcript

Word	Phonetic transcript
ALL	ao l
AN	ae n
ASK	ae s k
CARRY	k ae r iy
DARK	d aa r k
DON'T	d ow n t
GREASY	g r iy s iy
HAD	hh ae d
IN	ih n
LIKE	l ay k
ME	m iy
OILY	oy l iy
RAG	r ae g
SHE	sh iy
SUIT	s uw t
THAT	dh ae t
TO	t uw
WASH	w ao sh
WATER	w ao t axr
YEAR	y ih r
YOUR	y uh r

The speech signal **o(t)** was pre-emphasised by:

$$\hat{\mathbf{o}}(\mathbf{t}) = \mathbf{o}(\mathbf{t}) - \mathbf{0.97o}(\mathbf{t-1}) \tag{6.9}$$

A Hamming window of the form below was used for all the experiments

$$\hat{\mathbf{o}}(\mathbf{t}) = \left[\mathbf{0.54 - 0.46\cos}\left(\frac{\mathbf{2\pi t}}{\mathbf{T-1}}\right)\right]\mathbf{o(t)} \tag{6.10}$$

where **T** is the size of window. A window of 25ms duration was used with a frame length of 10ms for the feature extraction process.

6.3.1 Baseline system

The baseline system used in this work was based on MFCC as a feature vector ($\mathbf{O(t)}$) for recognition task. The MFCC features were derived from 24 Mel-spaced triangular filters. 13 MFCC, out of which 12 were the DCT coefficients (first 12 coefficients leaving the dc component), and a log of energy were selected as features. In order to include the dynamic features, delta coefficients were used. These delta coefficients were calculated by using the following regression formula

$$\Delta O(t) = \frac{\sum_{\tau=d_1}^{d_2} \tau\left(O(t+\tau) - O(t-\tau)\right)}{2\sum_{\tau=d_1}^{d_2} \tau^2} \tag{6.11}$$

where $\mathbf{d_2}$ is the width over which the dynamic coefficients are to be calculated. For regression, parameter $\mathbf{d_1 = 1}$ and for simple difference $\mathbf{d_1 = d_2}$. The use of differential coefficients has been extended to acceleration or the delta-delta coefficient, $\Delta O^2(t)$. This can be calculated by using the similar way as the delta coefficients. The delta and delta-delta coefficients were calculated and appended to give a complete 39-dimension feature.

6.3.2 Wavelet-based system

The 24-band filter structure obtained by using the AWP was used to extract 13 coefficients, as explained in Section 4.3.2. The filter structure was designed using the 'Daubechies 6' mother wavelet. In the case of the wavelet transform, there is no need to use an overlapping window since the wavelets have compact support. Thus, after a similar pre-emphasis as given in Equation 6.9, 13 features were extracted every 10ms. Delta and delta-delta coefficients were calculated and the 39 dimensional features vector was used for recognition. Further, 8 sub-band and 6 sub-band features (using 'Daubechies 20' wavelet) were also calculated along with the variance feature. These were then appended with the delta and delta-delta coefficients to give a 27- and 21-dimension feature vector respectively.

A dictionary was used to split the words of the two sentences into phonemes with a short pause (*sp*) at the end. This resulted in a total of 25

phonemes. A HMM was created for each phoneme and the training was done using the extracted features. Further, a three emitting state left to right HMM was used to model silence (***sil***) with an extra transition from state 2 to 4 and from state 4 to 2. Also, a single state ***sp*** model was created with a direct transition from the entry to the exit node. The ***sp*** model has its emitting state tied to the centre state of the ***sil*** model. The required topology of the two silence models is shown in Figure 6.2.

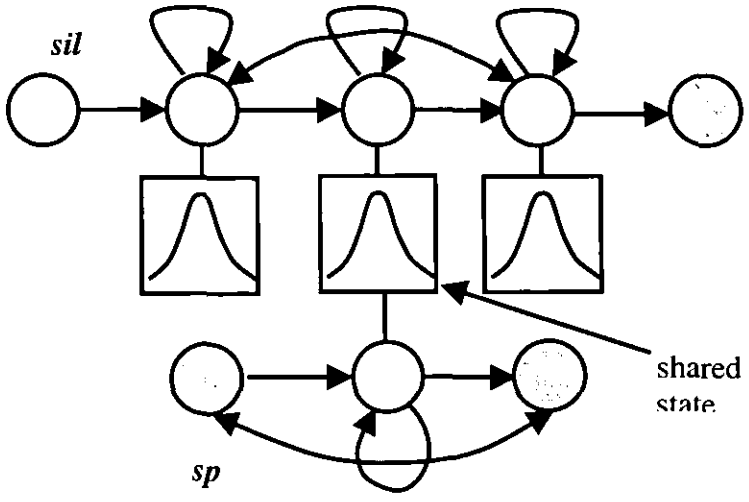


Figure 6.2: Silence model based on HMM.

In the final stage, monophone HMMs (context independent) were used to create triphone (context dependent) HMMs which resulted into a creation of 64 HMMs after tying states. There was no language model used for the recognition of continuous speech and all the words were assumed to have equal probability. This assumption is fairly true for the two sentences chosen for the task of recognition. A single Gaussian HMM, as well as multiple mixture components HMM, was used for training and testing.

The recognition results are evaluated in terms of the percentage correct and percentage accuracy. The percentage correct recognition is given as:

$$\textbf{Percentage correct} = \frac{\mathbf{N_w - D - S}}{\mathbf{N_w}} * \mathbf{100\%} \qquad (6.12)$$

where $\mathbf{N_w}$ is the total number of words in the reference transcript, and **D** and **S** are the total number of deletions and substitutions in the recognised transcript. The percentage accuracy is calculated as:

$$\text{Percentage accuracy} = \frac{N_w - D - S - I}{N_w} * 100\% \qquad (6.13)$$

where **I** is the total number of insertions in the recognised transcript. The number of insertions and deletions can be optimised by choosing the appropriate value of the transition penalty **p**. The best results were obtained by choosing **p=-10** and this value was used in all the later experiments. The results obtained by using the MFCC features and the 24-band wavelet-based features are shown in Table 6.2, where **H** is the number of words correctly recognised. The recognition results achieved by using the 6 sub-band and 8 sub-band features along with the energy variance features are also given at the end of Table 6.2.

It can be seen from the results that best recognition performance is achieved with 8 mixture and **p=-10**. The 24-band AWP-based features give the best percentage recognition, but due to large numbers of insertions, the percentage accuracy of MFCC-based features is the best. From the results of the previous chapters the recognition performance by 6 sub-band and 8 sub-band was found to be superior to MFCC and 24-band AWP features, but the results in Table 6.2 are just the opposite. This is because earlier only the three-class recognition problem was considered, but here all the 25 phonemes are classified by using these features, hence higher dimensions of features are helpful in giving better performance.

Table 6.2: Word recognition performance for continuous speech recognition

MFCC based features								
		% correct	**% accuracy**	**H**	**D**	**S**	**I**	**N_w**
Mixture=1	Training	99.67	98.58	2386	2	6	26	2394
p=0	Testing	99.10	97.30	770	1	6	14	777
Mixture=4	Training	99.92	99.33	2392	1	1	14	2394
p=0	Testing	99.36	98.58	772	0	5	6	777
Mixture=8	Training	99.92	99.37	2392	1	1	13	2394
p=0	Testing	99.23	98.46	771	1	5	6	777
Mixture=8	Training	99.87	99.58	2391	2	1	7	2394
p=-10	Testing	99.23	98.71	771	1	5	4	777

Table 6.2: cont.

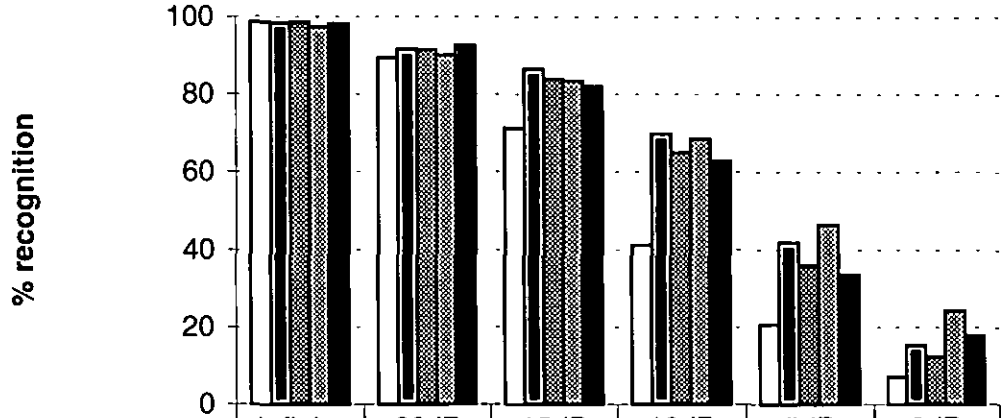
24-band AWP-based features								
		% correct	% accuracy	H	D	S	I	N_w
Mixture=1 p=0	Training	99.16	96.49	2374	6	14	64	2394
	Testing	98.84	93.31	768	2	7	43	777
Mixture=4 p=0	Training	99.50	97.49	2382	3	9	48	2394
	Testing	98.84	95.50	768	1	8	26	777
Mixture=8 p=0	Training	99.71	98.08	2387	2	5	39	2394
	Testing	98.97	95.75	769	1	7	25	777
Mixture=8 p=-10	Training	99.67	98.54	2386	3	5	27	2394
	Testing	98.97	96.53	769	1	7	19	777
8 sub-band AWP-based features								
		% correct	% accuracy	H	D	S	I	N_w
Mixture=1 p=0	Training	94.18	84.11	2217	20	117	237	2354
	Testing	91.51	76.83	711	10	56	114	777
Mixture=4 p=0	Training	96.98	90.70	2283	12	59	148	2354
	Testing	95.24	84.30	740	4	33	85	777
Mixture=8 p=0	Training	97.62	92.31	2298	9	47	128	2354
	Testing	95.88	86.36	745	4	28	74	777
Mixture=8 p=-10	Training	97.41	93.80	2293	13	48	85	2354
	Testing	96.01	88.16	746	4	27	61	777
6 sub-band AWP-based features								
		% correct	% accuracy	H	D	S	I	N_w
Mixture=1 p=0	Training	92.51	82.95	2187	38	139	226	2364
	Testing	91.38	79.28	710	11	56	94	777
Mixture=4 p=0	Training	95.60	89.89	2260	25	79	135	2364
	Testing	94.69	86.10	735	8	34	66	777
Mixture=8 p=0	Training	96.40	91.24	2279	23	62	122	2364
	Testing	95.62	87,64	743	7	27	62	777
Mixture=8 p=-10	Training	96.40	93.15	2279	34	51	77	2364
	Testing	95.37	89.45	741	9	27	46	777

6.4 Noisy speech recognition

In order to test the recognition performance under noisy conditions, noisy speech was generated by taking clean speech from the TIMIT database [11] and adding it with white Gaussian noise of zero mean and different variances. The baseline system and the 24-band filter using the AWP (explained in Section 6.6.1 and Section 6.6.2 respectively) were used for the recognition of noisy speech for different SNRs. The 6 sub-band and 8 sub-band based features were not used for this task as it gave inferior recognition performance for clean speech and was not expected to perform well for noisy speech.

The wavelet-based denoising stage proposed in Section 5.4 was also used before the feature extraction to assess the improvement in word recognition under noisy conditions. The process of denoising was carried out on a frame by frame basis. In the first experiment, one-level wavelet decomposition using the 'Daubechies 6' mother wavelet was applied over a frame duration and the threshold was calculated. The soft thresholding was carried out on detailed coefficients obtained by wavelet decomposition. The signal was then reconstructed using the same mother wavelet. This signal was then pre-emphasised and passed to the feature extractor, which was similar to the baseline system. In the second experiment, the threshold for denoising was calculated and two-level wavelet decomposition was carried out on the input speech signal. This resulted into splitting of 0-8kHz band into three sub-bands of 0-2kHz, 2-4kHz and 4-8kHz. The soft thresholding was applied on the detailed coefficients (two higher frequency bands) and the reconstructed signal was used for feature extraction. Similar tests were repeated using hard thresholding instead of soft thresholding.

It can be seen in Figure 6.3 that the percentage recognition accuracy by denoising with soft thresholding using one-level decomposition is 0.25% less and with two-level decomposition is 1.28% less as compared to the features extracted without denoising for the case of clean speech. This is due to the fact that some of the higher frequency components of phonemes are removed during denoising. This problem is encountered in the case of fricative recognition. However, for the case of voiced phoneme recognition the performance of the features with and without



	Infinite	20dB	15dB	10dB	5dB	0dB
MFCC	98.71	89.32	71.3	41.18	20.72	7.08
MFCC+S1	98.46	91.63	86.62	69.88	41.96	15.44
MFCC+H1	98.71	91.51	84.04	64.99	36.04	12.23
MFCC+S2	97.43	90.09	83.53	68.47	46.46	24.32
MFCC+H2	98.33	92.54	82.11	62.81	33.72	17.89

SNR

Figure 6.3: Recognition performance achieved by using

(a) MFCC features without denoising

(b) MFCC features with one-level denoising using soft thresholding

(c) MFCC features with one-level denoising using hard thresholding

(d) MFCC features with two-level denoising using soft thresholding

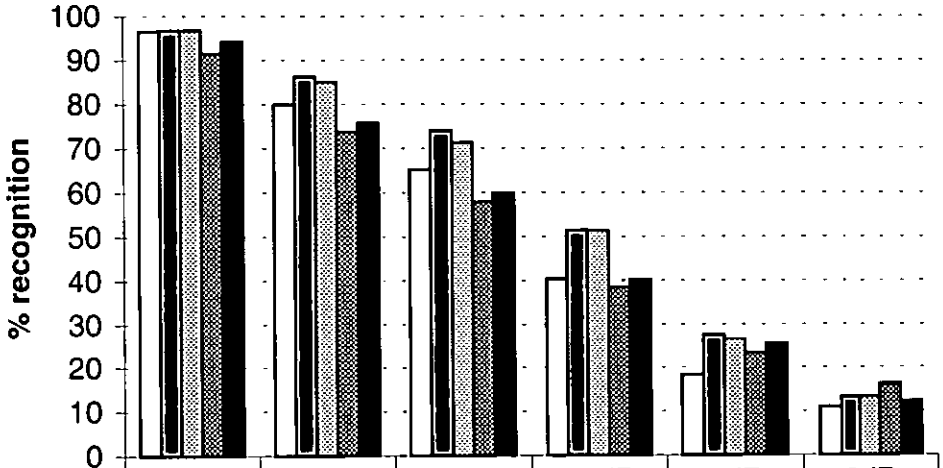
(e) MFCC features with two-level denoising using hard thresholding

denoising is approximately the same. Also, denoising with two-level decomposition results in slightly inferior recognition accuracy at higher SNR as compared to one-level decomposition. This is because some of the discriminatory information present in the 2-4kHz band is also removed along with the noise during the denoising process. The denoising is found to improve the recognition accuracy under the noisy conditions irrespective of hard or soft thresholding.

At SNR below 10dB the recognition accuracy with two-level denoising based features is found to be the best. However, the one-level denoising-based features perform better for SNR between 20dB to 10dB. The reason that two-level-based denoising performing better at lower SNR (higher level of noise) is understandable because it can remove the noise from the band 4-8kHz as well as

2-4kHz but one-level denoising cannot remove the noise from the 2-4kHz band. Hence, when the noise is more (i.e. lower SNR), two-level denoising is more effective in cleaning the speech signal.

The recognition performance achieved by using the hard thresholding is found to be lower as compared to soft thresholding (because of its shrinkage capabilities) for SNR below 15dB. However, it is always superior to the baseline system performance for noisy conditions. It is clear from Figure 6.3 that the recognition performance when the denoising is introduced is higher under noisy conditions and the soft thresholding usually gives the best result.



	Infinite	20dB	15dB	10dB	5dB	0dB
a	96.53	79.92	65.25	40.41	18.28	10.94
b	96.78	86.23	74	51.48	27.54	13.26
c	96.78	85.07	71.3	51.35	26.51	13.26
d	91.51	73.75	57.92	38.48	23.29	16.22
e	94.21	75.8	59.97	40.28	25.48	12.23

SNR

Figure 6.4: Recognition performance achieved by using
(a) wavelet features without denoising
(b) wavelet features with one-level denoising using soft thresholding
(c) wavelet features with one-level denoising using hard thresholding
(d) wavelet features with two-level denoising using soft thresholding
(e) wavelet features with two-level denoising using hard thresholding

The recognition accuracy achieved by using the 24-band AWP-based features is shown in Figure 6.4. One-level denoising with soft thresholding gives the best recognition performance except for 0dB SNR. For the case of two-level denoising, the results obtained by hard thresholding are found to be superior to the soft thresholding except at 0db SNR.

Comparing the performance of the MFCC and the 24-band AWP-based features; the former gives a better performance. Further, the denoising proposed here show considerable improvement in the recognition performance of both the feature extraction techniques. This establishes that denoising can be used as an effective tool at the pre-processing stage for the noisy speech recognition. The detailed results for these recognition tests are given in APPENDIX D.

6.5 Summary

The recognition results obtained by using the 24-band AWP-based features give 769 words correctly recognised words (**H** in Table 6.2 for test data) while 771 words are correctly recognised by MFCC-based features for the case of clean speech. It was found that there were more insertions when the wavelet-based features were used, causing the percentage accuracy of the word to be lower than the MFCC-based features. It is clear that the recognition performance by 6 sub-band and 8-sub-band based features are not comparable to the performance of the MFCC based features for the word recognition task since the number of phonemes to be identified is much higher than those considered in Chapter 5.

The denoising technique proposed was found to give substantial improvement in the word recognition accuracy for both MFCC and 24-band AWP-based features. Soft thresholding using one-level of decomposition was found to give better results under high and moderate SNR. When the signal power and noise power was about the same, two-level soft thresholding was found to be superior. This scheme has an advantage that there is no need to use complex algorithms to estimate the noise level, which requires the detection of speech and non-speech segment in an utterance.

6.5 References

[1] S. Young, D. Kershaw, J. Odell, D. Ollason, V. Valtchev and P. Woodland, "The HTK Book V3.0".

[2] L. Rabiner, *Introduction to speech recognition*, Prentice Hall International.

[3] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition", *Proceedings of IEEE*, vol. 77, no. 2, pp. 257-285, February 1989.

[4] L. E. Baum, T. Petrei, G. Soules and N. Weiss, "A maximisation technique occurring in the statistical analysis of probabilistic functions of Markov chains", *Annals of Mathematical Statistics*, vol. 41, pp. 164-171, 1970.

[5] L. A. Liporace, "Maximum likelihood estimation for multivariate observations of Markov source", *IEEE Transactions on Information Theory*, vol. 28, pp. 729-734, 1982.

[6] T. K. Moon, "The expectation-maximization algorithm", *IEEE Signal Processing Magazine*, vol. 13, no. 6, pp. 47-60, 1996.

[7] S. J. Young, N. H. Russel and J. H. S. Thornton, "Token passing; A conceptual model for connected speech recognition system", *Technical Report CUED/F-INFENG/TR38*, Cambridge university, 1989.

[8] L. R. Bahl, P. V. de Souza, P. S. Gopalkrishnan, D. Nahamoo and M. A. Picheny, "Context dependent modelling of phones in continuous speech using decision trees", *Proceedings of DARPA Speech and Natural Language Processing Workshop*, pp. 264-270, 1991.

[9] S. J. Young, N. J. J. Odell and P. C. Woodland, "Tree based state tying for high accuracy acoustic modelling", *Proceedings of ARPA Workshop on Human Language Technology*, pp. 307-312, 1994.

[10] S. J. Young and P. C. Woodland, "The use of state tying in continuous speech recognition", *Proceedings of Eurospeech*, pp. 2207-2210, 1993.

[11] TIMIT Acoustic-Phonetic Continuous Speech Corpus, National Institute of Standards and Technology, Speech Disc 1-1.1, NTIS Order no. PB91-505065, October 1990

CHAPTER 7

CONCLUSIONS

7.1 Overview

Although there has been considerable progress in the area of image processing using wavelets, there has not been much research in exploiting its capabilities for speech processing. This thesis has explored a number of different techniques in wavelet-based feature extraction for the purpose of speech recognition. The advantage of using the wavelet-based technique over the STFT is its ability to process stationary as well as non-stationary signals. The wavelets also have compact time-frequency support; therefore, any feature extraction technique does not require the use of an overlapping window, thereby reducing the computational cost as compared to the conventional STFT technique. The emphasis of this work lies in the extraction of features that are shift invariant and speaker independent. For this reason a simple LDA classifier was mostly used for the classification of phonemes. However, a more complex scheme has also been used in Chapter 6 for the word recognition problem. The results of the experimentation were reported in three chapters, each containing a different approach to utilise the properties of wavelets for phoneme recognition. Chapter 6 takes the best features extracted by wavelet techniques and applies then to the word recognition problem.

In Chapter 3, the DWT was used for feature extraction and instead of using the wavelet coefficients as features, energy in each of the frequency bands was calculated and used as features. This overcame the problem of shift variance

in features because the energy in a band remains constant for a phoneme with small shifts. Another advantage of using the energy-based features was the dimension reduction of the feature vector. Different levels of decomposition by the DWT were performed and the frame size was varied to observe its effect on the recognition performance. A logarithmic compression was also applied to the extracted energy features and the recognition performance was evaluated.

In Chapter 4 the limitations of the DWT, that it can only give left recursive binary tree structure was overcome by using the wavelet packets. Instead of using the Best-Basis algorithm, which requires a lot of processing for shift adjustments, admissible wavelet packets were used. Since the human hearing system is the best working speech recognition system to emulate, the Mel scale was adopted to design a 24-band filter structure using the AWP. The features derived from these 24-band filters were similar to those of the MFCC features. Different mother wavelets have also been explored to investigate their effect on the recognition performance.

An important quality of good features for speech recognition is their invariance in the presence of noise. Chapter 5 explores the phoneme recognition performance achieved by these features for noisy speech. Further modifications to these features were also proposed, which resulted in improving the phoneme recognition under noisy conditions. A new pre-processing based on wavelet denoising has also been proposed in this chapter. This technique is simple to implement and does not require the detection of speech and non-speech frames for estimating the noise. Due to the non-linear processing that is inherent because of thresholding, it can effectively remove additive Gaussian noise from the noisy speech. The effect of both hard and soft thresholding has been explored on the phoneme recognition task at various SNR levels.

Finally the performance of the wavelet-based features were tested using a state-of-the-art speech recogniser, the HTK. Continuous speech was taken from the TIMIT database and the word recognition was performed using the context-dependent phoneme model based on HMM. The pre-processing based on denoising was also performed for clean and noisy speech to evaluate the improvement in word recognition accuracy. These performances were also compared with the standard MFCC-based system under similar conditions.

The findings of this thesis can be summarised as follows.

7.1.1 Discrete Wavelet Transform

- Derivation of new energy-based features by the DWT retains the discrimination information as in the STFT. It also reduces the dimension of feature vectors as compared to the wavelet coefficient based features.

- These features overcome the problem of shift variance and speaker dependency.

- A comparative study of the recognition performance for different frame sizes and different levels of decomposition has also been carried out, which shows that an 8ms frame duration appears best for feature extraction.

- A non-linear classifier based on MLP gives better recognition performance as compared to the linear classifier based on LDA.

- A proposal of new features based on logarithmic compression of the energy features shows substantial improvement in the phoneme recognition over the simple energy features.

- The main drawback of the DWT-based features is the inability of the DWT to decompose the higher frequency band (which gives a left recursive binary tree structure). Thus, for higher levels of decomposition the features come from the very low frequency band having very little discriminatory information. This causes no improvement in the recognition performance even if the number of features is increased.

7.1.2 Admissible Wavelet Packets

- Proposing features based on the AWP shows an overall improvement in the classification of the phonemes over the DWT-based features. It also provides shift invariant features and is fast to compute contrary to the Best-Basis algorithm, which requires additional processing for shift adjustment.

- Different admissible wavelet tree structures have been tested for different frame sizes and the recognition performance evaluated.

- A novel filter using the AWP has been designed that splits the 0-8kHz speech signal into 24 bands closely following the Mel scale. The features derived by using these filters are similar to that of the MFCC features.

- Fisher class separability based on the 24-band AWP-based features appears to be higher than the MFCC features if the frame duration remains large. This gives a statistical measure that these features are better than MFCC features for large frame duration; however, for the lower frame duration the latter is found to be superior.

7.1.3 Robust Features

- The effect of white Gaussian noise on phoneme the recognition task based on the AWP features has been established. A new energy subtraction method is proposed for features to compensate the effect of noise and is found to show an improvement in phoneme recognition.

- The AWP-based features are found to perform better under very low SNR as compared to the MFCC.

- The effect of changing the mother wavelet on the phoneme recognition has also been investigated.

- A new pre-processing stage has been proposed based on denoising of the input signal using the wavelet transform and both soft and hard thresholding techniques have been tested. This scheme shows huge improvement in the recognition of vowels but the performance reduces for the unvoiced fricatives.

7.1.4 Continuous speech recognition using wavelet features

- The major advantage of the wavelet for feature extraction for continuous speech is its compact time and frequency support. This means that wavelet processing will not require an overlapping window function contrary to the STFT processing. This saves a lot of computation during the feature extraction phase.

- The pre-processing stage using wavelet denoising has also been evaluated for continuous speech recognition. Soft as well as hard thresholding techniques have been tested for both one-level and two-level denoising and considerable improvement in word recognition is achieved for both the MFCC as well as the AWP-based features.

The results of simulation without denoising the speech show that the recognition performance at lower SNR (in the range of 5dB to 0dB) is better when the wavelet based features are used. Thus these features will be useful for ASR in the application environment such as factory, cockpit communication and highway/motorway communication. The last two applications use a telephone network to transfer speech, therefore it will result in the reduction of speech bandwidth. This will require re-designing of the band splitting structure for feature extraction.

7.2 Future Work

Wavelet-based feature extraction makes theoretical sense to implement it for speech recognition systems. This thesis has explored the use of the wavelet transform for the extraction of energy based features, a concept similar to that in the STFT-based features. However, wavelets have even higher capabilities and the time information available by wavelet transform has not been used at all. This

information is of course not available when the conventional STFT technique is used. The timing information along with the band energy information can provide the temporal evolution of features. This may be helpful in avoiding the delta and delta-delta features and also enables the use of larger window duration.

Recognition performance in the presence of noise can be further improved by using more complex techniques of wavelet-based denoising. RASTA processing can also be used to enhance the performance under noisy conditions. Further, the resistance of wavelet-based features can be explored in the presence of real life noise other than white Gaussian.

A similar filter, such as the 24-band AWP, can be developed for the 0-4kHz bandwidth for the telephone channel and its performance can be studied in the presence of channel distortion as well as fading environment.

Finally, wavelet-based features should prove to be very useful for speaker recognition applications because of its multi-resolution capabilities.

APPENDIX A

A.1 Proof of Inverse Wavelet Transform

For a wavelet function $\psi(t)$ the continuous wavelet transform of a signal $x(t)$ is given as:

$$CWT(a,\tau) = a^{-1/2}\int x(t).\psi^*((t-\tau)/a)dt = x \otimes \bar{\psi}_a(\tau) \qquad (A.1)$$

where $\otimes$ is the convolution operator with

$$\bar{\psi}_s(u) = a^{-1/2}\psi^*(-u/a) \qquad (A.2)$$

Let $b(t)$ is given by:

$$b(t) = c\int_0^{+\infty}\int_{-\infty}^{+\infty} CWT(a,\tau).\frac{1}{\sqrt{a}}\psi(t-\tau/a)d\tau\frac{da}{a^2} \qquad (A.3)$$

The right hand side of the equation A.3 can be rewritten as a sum of convolutions

$$\begin{aligned} b(t) &= c\int_0^{+\infty} CWT(a,.) \otimes \psi_a(t)\frac{da}{a^2} \\ &= c\int_0^{+\infty} x \otimes \psi_a \otimes \bar{\psi}_s(t)\frac{da}{a^2} \end{aligned} \qquad (A.4)$$

where '.' Indicated the variable over which the convolution is performed. To prove that $b(t)$ is the same as $x(t)$, the Fourier transform of $b(t)$ is calculated.

$$\mathbf{b}(\omega) = \mathbf{c}\int_{0}^{+\infty} \mathbf{x}(\omega)\sqrt{\mathbf{a}}\psi^{*}(\mathbf{a}\omega)\sqrt{\mathbf{a}}\psi(\mathbf{a}\omega)\frac{\mathbf{da}}{\mathbf{a}^{2}}$$
$$= \mathbf{x}(\omega)\mathbf{c}\int_{0}^{+\infty} |\psi(\mathbf{a}\omega)|^{2}\frac{\mathbf{da}}{\mathbf{a}} \qquad \text{(A.5)}$$

since ψ is real $|\psi(-\omega)|^{2} = |\psi(\omega)|^{2}$. By changing the variable $\xi = \mathbf{a}\omega$ in the above equation the result obtained is

$$\mathbf{b}(\omega) = \mathbf{x}(\omega)\mathbf{c}\int_{0}^{+\infty} \frac{|\psi(\xi)|^{2}}{\xi}\mathbf{d}\xi \qquad \text{(A.6)}$$

From the equation A.6 it is clear that the recovered signal **b(t)** is the same as **x(t)** if

$$\mathbf{c}^{-1} = \int_{0}^{+\infty} \frac{|\psi(\xi)|^{2}}{\xi}\mathbf{d}\xi < +\infty \qquad \text{(A.7)}$$

Thus equation k.3 can be used to recover the signal back from the transformed domain (or in other words is the inverse transform) if equation A.7 is satisfied. This condition is known as the admissibility condition. To guarantee this condition the wavelets must have zero mean and should be continuously differentiable.

APPENDIX B

List of phonemes present in the TIMIT database

Phoneme Types	Phoneme symbols
Stops	/b/, /d/, /g/, /p/, /t/, /k/, /dx/, /q/
Affricates	/jh/, /ch/
Fricatives	/s/, /sh/, /z/, /zh/, /f/, /th/, /v/, /dh/
Nasals	/m/, /n/, /ng/, /em/, /en/, /eng/, /nx/
Semivowels and Glides	/l/, /r/, /w/, /y/, /hh/, /hv/, /el/
Vowels	/ax-h/, /aa/, /ae/, /ah/, /ao/, /aw/, /ax/, /axr/, /ay/, /eh/, /er/, /ey/, /ih/, /ix/, /iy/, /ow/, /oy/, /uh/, /uw/, /ux/

APPENDIX C

C.1 HMM for Pattern Matching

The HMM is used to determine the probability of the observation vector, $\mathbf{Y_T}$, given a hypothesised word string **W**. The word string is mapped to the appropriate set of models using the lexicon, the task is to compute $\mathbf{P[Y_T|\mathcal{M}]}$ where $\mathcal{M}$ is the set of HMM linked with word string **W.** Initially only the simplified case of a single model $\mathcal{M}$, will be examined. The observation sequence is defined as

$$\mathbf{Y_T} = \mathbf{y(1)y(2)...y(T)} \tag{C.1}$$

where each observation $\mathbf{y(\tau)}$ is an n-dimensional vector

$$\mathbf{y(\tau)} = \left[\mathbf{y_1(\tau)} \;\; \mathbf{y_2(\tau)} \;\; \ldots \;\; \mathbf{y_n(\tau)} \right]^{\mathbf{T}} \tag{C.2}$$

The total probability is given by summing over all possible paths through the model that ends at the appropriate final state. The probability of the observation sequence for a given model $\mathcal{M}$ is given by

$$\mathbf{P[Y_T|\mathcal{M}]} = \sum_{\theta \in \Theta} \mathbf{P[Y_T|\theta, \mathcal{M}]P[\theta|\mathcal{M}]} \tag{C.3}$$

$$= \sum_{\theta \in \Theta} \mathbf{a}_{\theta_{TN}} \prod_{\tau=1}^{\mathbf{T}} \mathbf{a}_{\theta_{\tau-1}\theta_\tau} \mathbf{b}_{\theta_\tau}(\mathbf{y(\tau)}) \tag{C.4}$$

where Θ is the set of all **K** possible state sequences of length **T** in the model $\mathcal{M}$

$$\Theta = \left\{\theta^{(1)}, \theta^{(2)}, ..., \theta^{(K)}\right\} \tag{C.5}$$

and θ_τ is the state occupied at time τ in the path θ. The model is initialised such that $\theta_0 = 1$.

The single model discussed till now is to be modified for it to work on the continuous speech. Many models are required for this purpose and are to be connected together to form the word string. This combination of model is achieved by linking two models together as shown in Figure C.1. The end non-emitting state of model A is combined with the start-state of model B and

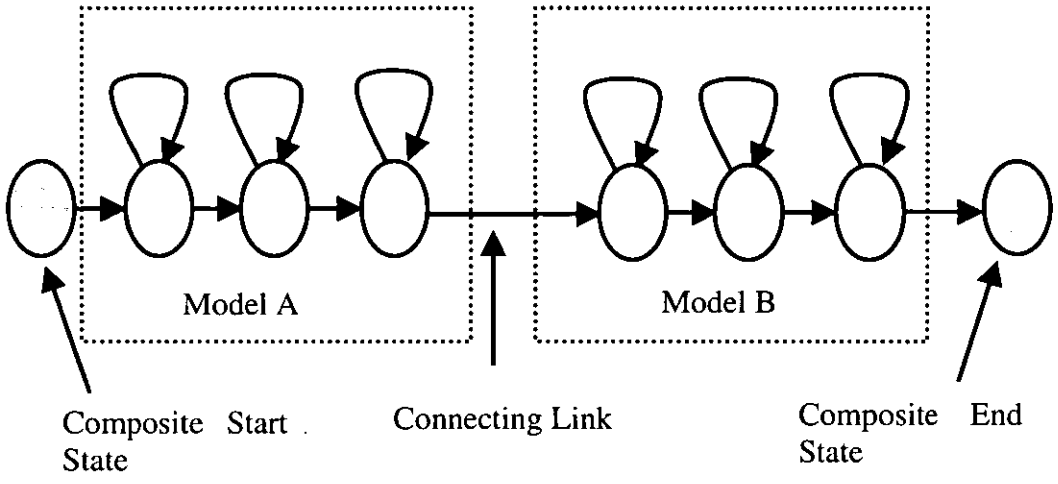


Figure C.1: Composite Hidden Markov Model AB

replaced by the connecting link. The start-state of the model A becomes the composite start state and the end state of model B becomes composite end state of the model. The set of possible states, Θ, is the set of all path of length T in the composite model AB. Equation C.4 is never implemented directly as it is computationally very expensive and forward-backward algorithm is used for its computation.

C.2 Forward-Backward Algorithm

The forward-backward algorithm is an efficient way of calculating the probability of an observation sequence being generated by a particular set of

models. The assumption made here is that the probability of a particular observation only depends upon the current state. Consider the forward probability $\alpha_j(t)$ and the backward probability $\beta_j(t)$. These are the joint probabilities of the partial observation sequence and being in state $q_j(t)$ of the given model $\mathcal{M}$. These are defined as:

$$\alpha_j(t) = P[y(1), y(2), \ldots . y(t), q_j(t) | \mathcal{M}] \tag{C.6}$$

$$\beta_j(t) = P[y(t+1), y(t+2), \ldots . y(T), q_j(t) | \mathcal{M}] \tag{C.7}$$

Using these definitions it is possible to compute by iterations the values of $\alpha_j(t)$ and $\beta_j(t)$. From the HMM definition, the initial conditions for $\alpha_j(0)$ are:

$$\alpha_1(0) = 1 \tag{C.8}$$

$$\alpha_j(0) = 0 \text{ if } j \neq 1 \tag{C.9}$$

Then for $1 \leq t \leq T$ and $2 \leq j \leq N\text{-}1$

$$\alpha_j(t) = \left[\sum_{i=1}^{N} \alpha_i(t-1) a_{ij} \right] b_j(y(t)) \tag{C.10}$$

and terminates with

$$\alpha_N(T) = \sum_{i=2}^{N-1} \alpha_i(T) a_{iN} \tag{C.11}$$

A similar set of recursion can be defined for the backward probability. The initial conditions in this case are:

$$\beta_j(T) = a_{jN} \text{ for } 1 \leq j \leq N \tag{C.12}$$

and the iterative scheme from $t=T-1$ to $t=0$ is:

$$\beta_i(t) = \sum_{j=1}^{N} a_{ij} b_j(y(t+1)) \beta_j(t+1) \text{ for } 1 \leq j \leq N \tag{C.13}$$

$$\beta_N(t) = 0 \tag{C.14}$$

The probability of a particular observation sequence is given by:

$$P[Y_T \mid \mathcal{M}] = \alpha_N(T) = \beta_1(0) = \sum_{j=1}^{N} \alpha_j(t) \beta_j(t) \tag{C.15}$$

The above equation gives the probability of a particular utterance and is also useful in the re-estimation formulae for the HMM.

C.3 Estimation of HMM Parameters

The estimation of the HMM parameter is essentially an optimisation problem. The task is to obtain a set of models that according to a criterion matches the available training data well. The criterion used may be Maximum Mutual Information (MMI) or the Maximum Likelihood (ML), however ML criterion is more commonly used. In this work the ML estimation will be used for the parameter estimation.

The aim of the ML estimation is to obtain a set of HMMs $\mathcal{M}$, such that they maximise $\mathcal{X}_{\text{mle}}(\mathcal{M})$ where:

$$\mathbf{X}_{\text{mle}}(\mathcal{M}) = \mathbf{P}[\mathbf{Y}_{\mathbf{T}} \mid \mathcal{M}] \tag{C.16}$$

Due to the large dimension of the problem (mean, variance and weight for large vocabulary system) the training process is very slow. To overcome this problem, Baum-Welch algorithm based on Expectation Maximisation (EM) technique is used. The Baum-Welch algorithm[1] is designed such that:

$$\mathcal{X}_{\text{mle}}(\tilde{\mathcal{M}}) \geq \mathcal{X}_{\text{mle}}(\mathcal{M}) \tag{C.17}$$

where $\tilde{\mathcal{M}}$ is the new estimate of the model set. This is done by introducing an auxiliary function, $\mathcal{A}(\mathcal{M}, \tilde{\mathcal{M}})$, defined as:

$$\mathcal{A}(\mathcal{M}, \tilde{\mathcal{M}}) = \sum_{\theta \in \Theta} \mathbf{P}[\mathbf{Y}_{\mathbf{T}}, \theta \mid \mathcal{M}] \log(\mathbf{P}[\mathbf{Y}_{\mathbf{T}}, \theta \mid \tilde{\mathcal{M}}]) \tag{C.18}$$

Maximising this auxiliary function ensures the $\mathcal{X}_{\text{mle}}(\mathcal{M})$ is non-decreasing satisfying Equation C.17. Equation C.18 can be used as many times,

---

[1] L. E. Baum, T. Petrei, G. Soules and N. Weiss, "A maximisation technique occurring in the statistical analysis of probabilistic functions of Markov chains", Annals of Mathematical Statistics, vol. 41, pp. 164-171, 1970.

each time replacing the original model set by the new model set estimate. Each iteration is guaranteed not to decrease the likelihood. This results into a local ML estimate of the parameters. Baum first proved the convergence of this algorithm and it was later extended to mixture distribution and vector observations. For CDHMM with Gaussian output distribution the model parameters can be estimated in an iterative manner. The details of the derivation of the formulae can be found in[2], only the results are quoted here. The probability of being in a particular mixture component is given by:

$$\mathbf{L_{jm}(\tau) = P[q_{jm}(\tau) \mid Y_T, \mathcal{M}]}$$

$$= \frac{1}{\mathbf{P[Y_T \mid \mathcal{M}]}} \mathbf{U_j(\tau) c_{jm} b_{jm}(y(\tau)) \beta_j(\tau)} \qquad \text{(C.19)}$$

where

$$\mathbf{U_j(\tau)} = \begin{cases} \mathbf{a_{1j}}, & \mathbf{if\ \tau = 1} \\ \sum_{i=2}^{N-1} \alpha_i(\tau-1) a_{ij}, & \mathbf{otherwise} \end{cases} \qquad \text{(C.20)}$$

The extended dataset is **{y(1), y(2), ...,y(T), L(1), L(2),..., L(T)}**, where $\mathbf{L(\tau)}$ is the matrix whose elements $\mathbf{L_{jm}(\tau)}$ are the probability of being in state $\mathbf{s_j}$ and component $\mathbf{M_m}$ at time τ. The terms frame/state alignment and frame/state component alignment is used to denote $\mathbf{L_j(\tau)}$, where $\mathbf{L_j(\tau)=P[q_j(\tau)|Y_T,\mathcal{M}]}$, and $\mathbf{L_{jm}(\tau)}$ respectively. Using the extended dataset the estimation of the mean variance and mixture weight is given by

$$\tilde{\mu}_{jm} = \frac{\sum_{\tau=1}^{T} \mathbf{L_{jm}(\tau) y(\tau)}}{\sum_{\tau=1}^{T} \mathbf{L_{jm}(\tau)}} \qquad \text{(C.21)}$$

$$\tilde{\Sigma}_{jm} = \frac{\sum_{\tau=1}^{T} \mathbf{L_{jm}(\tau)(y(\tau) - \tilde{\mu}_{jm})(y(\tau) - \tilde{\mu}_{jm})^T}}{\sum_{\tau=1}^{T} \mathbf{L_{jm}(\tau)}} \qquad \text{(C.22)}$$

---

[2] L. A. Liporace, "Maximum likelihood estimation for multivariate observations of Markov source", IEEE Trans. on Information Theory, vol. 28, pp. 729-734, 1982.

$$\tilde{c}_{jm} = \frac{\sum_{\tau=1}^{T} L_{jm}(\tau)}{\sum_{\tau=1}^{T} L_{j}(\tau)} \tag{C.23}$$

The transition probabilities are re-estimated by

$$\tilde{a}_{ij} = \frac{\sum_{\tau=1}^{T-1} \alpha_i(\tau) a_{ij} b_j(y(\tau+1)) \beta_j(\tau+1)}{\sum_{\tau=1}^{T-1} \alpha_i(\tau) \beta_i(\tau)} \tag{C.24}$$

where $1 < i, j < N$. The transition from the non-emitting states are re-estimated by

$$\tilde{a}_{1j} = \frac{1}{P[Y_T \mid M]} \alpha_j(1) \beta_j(1) \tag{C.25}$$

$$\tilde{a}_{iN} = \frac{\alpha_i(T) \beta_i(T)}{\sum_{\tau=1}^{T} \alpha_i(\tau) \beta_i(\tau)} \tag{C.26}$$

These re-estimation formulas are used to train all the HMMs used in this work. The details of how multiple component models are built are given in HTK manual.

APPENDIX D

Table D.1: Test results for wavelet-based word recognition without denoising for a total of 777 words.

	% Correct	% Accuracy	H	D	S	I
Clean	98.97	96.53	769	1	7	19
20dB	86.74	79.92	674	38	65	53
15dB	74.26	65.25	577	70	130	70
10dB	50.19	40.41	390	151	236	76
5dB	24.20	18.28	188	253	336	46
0dB	13.90	10.94	108	409	260	23

Table D.2: Test results for wavelet-based word recognition with one-level denoising with soft thresholding.

	% Correct	% Accuracy	H	D	S	I
Clean	99.23	96.78	771	1	5	19
20dB	91.63	86.23	712	18	47	42
15dB	82.63	74	642	32	103	67
10dB	64.09	51.48	498	54	225	98
5dB	36.68	27.54	285	161	331	71
0dB	15.19	13.26	118	462	197	15

Table D.3: Test results for wavelet-based word recognition with one-level denoising with hard thresholding.

	% Correct	% Accuracy	H	D	S	I
Clean	98.97	96.78	769	1	7	17
20dB	90.99	85.07	707	16	54	46
15dB	79.92	71.3	621	33	123	67
10dB	63.45	51.35	493	50	234	94
5dB	34.23	26.51	266	188	323	60
0dB	14.41	13.26	112	480	185	9

Table D.4: Test results for wavelet-based word recognition with two-level denoising with soft thresholding.

	% Correct	% Accuracy	H	D	S	I
Clean	98.33	91.51	764	1	12	53
20dB	84.94	73.75	660	17	100	87
15dB	75.03	57.92	583	20	174	133
10dB	63.32	38.48	492	24	261	193
5dB	42.60	23.29	331	74	372	150
0dB	19.05	16.22	148	278	351	22

Table D.5: Test results for wavelet-based word recognition with two-level denoising with hard thresholding

	% Correct	% Accuracy	H	D	S	I
Clean	98.33	94.21	764	1	12	32
20dB	85.71	75.8	666	12	99	77
15dB	76.96	59.97	598	18	161	132
10dB	62.42	40.28	485	36	256	172
5dB	43.50	25.48	338	139	300	140
0dB	14.54	12.23	113	428	236	18

Table D.6: Test results for MFCC-based word recognition without denoising

	% Correct	% Accuracy	H	D	S	I
Clean	99.23	98.71	771	1	5	4
20dB	93.05	89.32	723	19	35	29
15dB	78.12	71.3	607	56	114	53
10dB	46.59	41.18	362	138	227	42
5dB	21.75	20.72	169	401	207	8
0dB	7.08	7.08	55	668	54	0

Table D.7: Test results for MFCC-based word recognition with one-level denoising with soft thresholding

	% Correct	% Accuracy	H	D	S	I
Clean	99.10	98.46	770	0	7	5
20dB	94.21	91.63	732	7	38	20
15dB	89.83	86.62	698	18	61	25
10dB	74.77	69.88	581	45	151	38
5dB	45.30	41.96	352	168	257	26
0dB	16.22	15.44	126	468	183	6

Table D.8: Test results for MFCC-based word recognition with one-level denoising with hard thresholding

	% Correct	% Accuracy	H	D	S	I
Clean	99.10	98.71	770	1	6	3
20dB	93.95	91.51	730	8	39	19
15dB	88.42	84.04	687	12	78	34
10dB	70.01	64.99	544	40	193	39
5dB	38.48	36.04	299	179	299	19
0dB	12.48	12.23	97	504	176	2

Table D.9: Test results for MFCC-based word recognition with two-level denoising with soft thresholding

	% Correct	% Accuracy	H	D	S	I
Clean	98.71	97.43	767	1	9	10
20dB	95.62	90.09	743	3	31	43
15dB	90.60	83.53	704	8	65	55
10dB	79.79	68.47	620	20	137	88
5dB	59.33	46.46	461	70	246	100
0dB	29.09	24.32	226	242	309	37

Table D.10: Test results for MFCC-based word recognition with two-level denoising with hard thresholding

	% Correct	% Accuracy	H	D	S	I
Clean	99.23	98.33	771	1	5	7
20dB	97.04	92.54	754	3	20	35
15dB	90.99	82.11	707	8	62	69
10dB	77.22	62.81	600	23	154	112
5dB	49.03	33.72	381	93	303	119
0dB	22.65	17.89	176	304	297	37

APPENDIX E

E.1 Software Introduction

A phoneme recognition software with GUI facility is provided in the attached CD at the back of this thesis. It contains a demo version of the software developed using MATLAB version 5.3 in the 'Program' directory. It requires 'Wavelet toolbox' and 'Statistics toolbox' for the execution of some of the routines. In order to run the demo, execute the file 'asrdemo'. This will produce a new window (as shown in Figure E.1) giving the user three options.

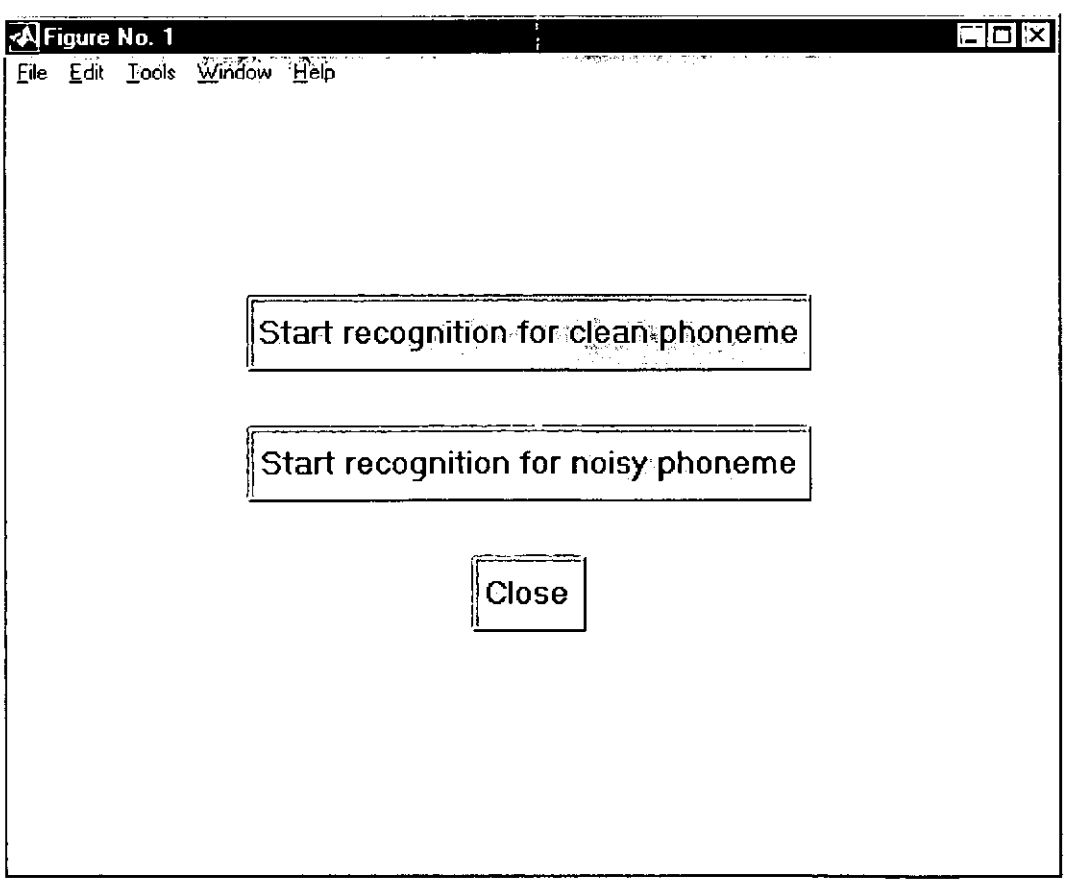


Figure E.1: The window displayed when running the 'asrdemo'

1. Recognition of the clean phoneme recognition.
2. Recognition of the noisy phoneme recognition.
3. Exit the demo.

Selection of first or second option will open another window and will ask user to select from further different options available. If the user clicks the first option he has to select from the following options (as shown in Figure E.2):

1. The phoneme class to be recognised (unvoiced fricatives, unvoiced stops or vowels).
2. The type of features to be extracted (DWT, AWP or MFCC base).
3. The number of features to be extracted during every 8ms duration. For MFCC it has a default value of 13.
4. The phoneme file to be used for the extraction of these features. These are the 'wave' files and are supplied with this software.

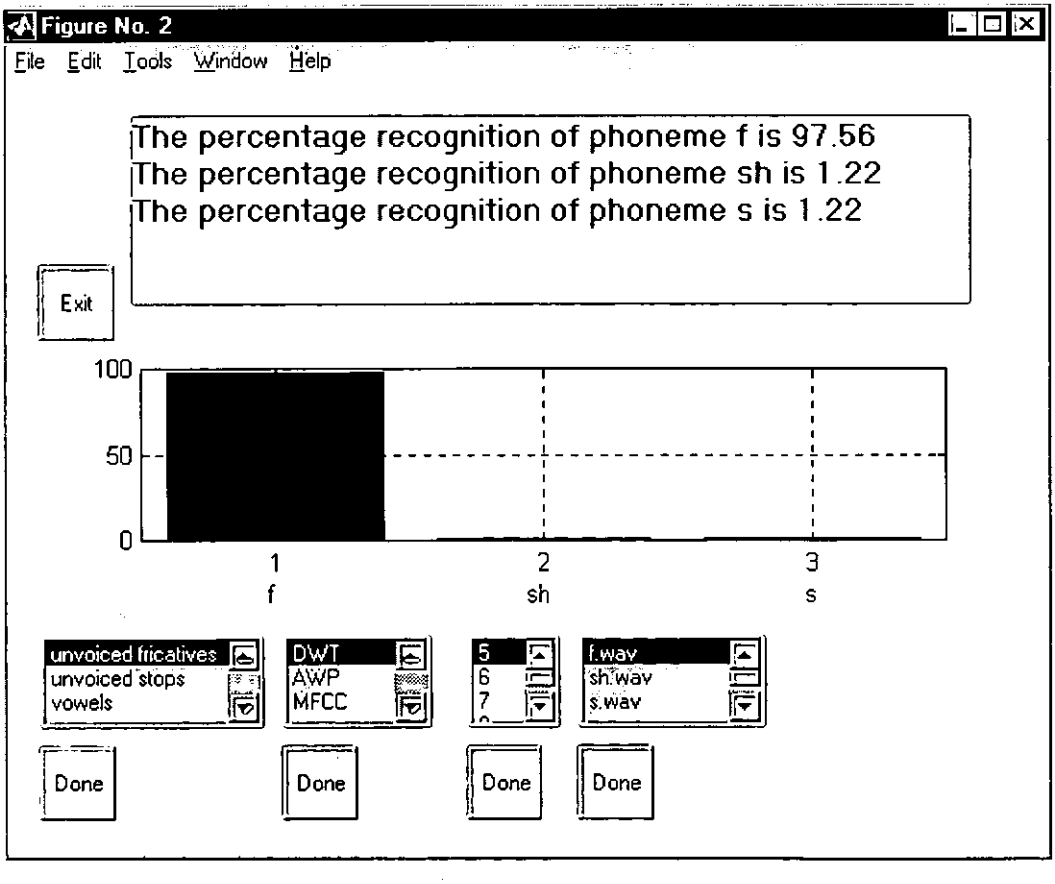


Figure E.2: Window for the recognition of clean speech

Once the selections are made correctly the classification results are displayed in percentage as well as a bar chart.

In the case of user selecting the noisy phoneme recognition, the user is provided with the following options in a new window (as shown in Figure E.3).

1. The phoneme class to be recognised (unvoiced fricatives, unvoiced stops or vowels).
2. The type of features to be extracted (AWP or MFCC base).
3. The number of features to be extracted during every 8ms duration.
4. The phoneme file to be used for the extraction of these features.
5. The level of noise present in the phoneme (20dB, 10dB or 0dB).
6. The option whether denoising is to be applied (select 2) or not (select 1).
7. If denoising is to be applied then the type of denoising (hard/soft) and the level of denoising (one level or two levels).

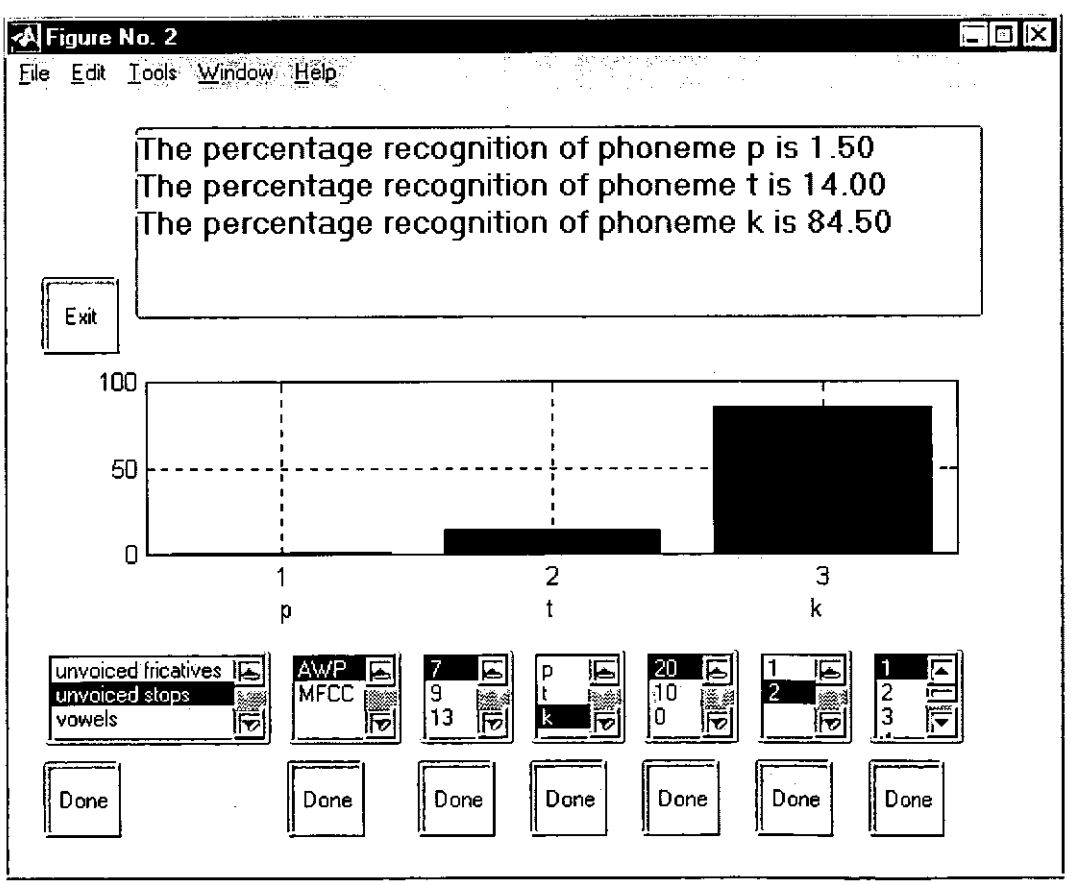


Figure E.3: Window for the recognition of noisy speech

After making these selections the feature extraction is carried out and the classification is performed by using the Linear Discriminant Analysis (LDA). The LDA uses the training feature file that is supplied with the software. Once the classification of the clean/noisy phonemes has been performed the second window can be closed by the exit options and the first window can be used for the recognition of other phonemes.

The asrdemo uses different Matlab 'functions', 'wave' files of the phonemes for feature extraction and 'txt' file to train the classifier. The details of these files are included in 'Readme.txt' file provided with the software.

PUBLICATIONS

Journal Papers

1. O. Farooq and S. Datta, "Modified discrete wavelet features for phoneme recognition", *Proceedings of Workshop on Innovations in Speech Processing*, vol. 23, part 3, Stratford-upon-Avon, UK, 2001, pp. 93-99.
2. O. Farooq and S. Datta, "Speech recognition with emphasis on wavelet based feature extraction", *IETE Journal of Research*, vol. 48, no. 1, pp. 3-13, Jan-Feb. 2002.
3. O. Farooq and S. Datta, "Phoneme recognition using wavelet based features", Accepted for publication in the *Journal of Information Sciences.*

Journal Letters

1. O. Farooq and S. Datta, "Wavelet Transform for dynamic feature extraction of phonemes", *Acoustics Letters*, vol. 23, no. 4, 1999, pp. 79-82.
2. O. Farooq and S. Datta, "Mel filter-like admissible wavelet packet structure for speech recognition", *IEEE Signal Processing Letters*, vol. 8, No. 7, July 2001, pp. 196-198.
3. O. Farooq and S. Datta, "Robust features for speech recognition based on admissible wavelet packets", *Electronics Letters*, vol. 37, no. 5, 6th December 2001, pp. 1554-1556.

Conference Papers

1. O. Farooq and S. Datta, "A Neural Network phoneme classification based on Wavelet features", *Proceedings of International Conference on Recent Advances Soft Computing*, June 29-30, 2000, De Montfort University, Leicester, UK pp. 73-78.

2. O. Farooq and S. Datta, "Dynamic feature extraction by wavelet analysis", *Proceedings of 6th International Conference on Spoken Language Processing*, Beijing, China from 16-20 Oct. 2000, vol. 4, pp.696-699.

3. O. Farooq and S. Datta, "Speaker independent phoneme recognition by MLP using wavelet features", *Proceedings of 6th International Conference on Spoken Language Processing*, Beijing, China from 16-20 Oct. 2000, vol. 1, pp.393-396.

4. O. Farooq and S. Datta, "Admissible wavelet packet transform based features for phoneme classification", *Proceedings of 3rd Conference on Research in Electronic Communication and Software*, University of Keele, UK, 09-11 April, 2001, pp.3-4.

5. O. Farooq and S. Datta, "Wavelet packet based features for noisy speech recognition", *17th International Congress on Acoustics*, Rome Italy, Sept. 2001.

6. S. Datta and O. Farooq, "Modified discrete wavelet features for phoneme recognition", *17th International Congress on Acoustics*, Rome Italy, Sept. 2001.

7. S. Datta and O. Farooq, "Wavelet based front-end for automatic speech recognition" presented in *6th International Workshop on recent trends in Speech, Music and Allied Signal Processing*, New Delhi India, 2001.

8. O. Farooq and S. Datta, "16-band filter derived by admissible wavelet packet for phoneme recognition", *Joint Conference on Information Sciences*, Durham, USA, March 2002, pp. 192-195.

9. O. Farooq and S. Datta, "Robust wavelet based features for phoneme recognition using wavelet denoising", *Proceedings of 4th Conference on Research in Electronic Communication and Software*, 17-19 April, Nottingham University, UK, 2002, pp. 15-16.

10. O. Farooq and S. Datta, "A novel wavelet based pre-processing for robust features in ASR", *Proceedings of International Symposium on Communication Systems, Networks and Digital Signal Processing*, Staffordshire Univ., UK (accepted).

11. O. Farooq and S. Datta, "Mel-scaled wavelet filter based features for noisy unvoiced phoneme recognition", *Proceedings of 7th International Conference on Spoken Language Processing*, Denver, Colorado, USA, 2002 (accepted).