SPEECH RECOGNITION IN NOISE USING WEIGHTED MATCHING ALGORITHMS

Nestor Becerra Yoma

A thesis submitted for the degree of
Doctor of Philosophy
to the Faculty of Science and Engineering of the University of Edinburgh
1998

Abstract

This thesis investigates the problem of automatic speech recognition in noise (additive and convolutional) by the development of Weighted Matching algorithms (WMA). The WMA approach relies on the fact that additive noise corrupts some segments of the speech signal more severely than others. As a result, WMA revises the classical concept of acoustic pattern matching in order to include the segmental signal to noise ratio (SNR) frame-by-frame. The problem of end-point detection is also addressed and a method based on autoregressive analysis of noise is also proposed for robust speech pulse detection. The technique is shown to be effective in increasing the discrimination between the speech signal and background noise.

Modified versions of the Dynamic Time Warping (DTW) and Hidden Markov Model (HMM) algorithms are proposed and tested in combination with reliability in noise cancelling weighting firstly using a novel noise cancelling neural net (LIN-Lateral Inhibition Neural Net) and then spectral subtraction (SS). The reliability in noise cancelling is a function of the local SNR and tries to measure the reliability of the information provided by the noise cancelling technique. A model for additive noise is proposed with the suggestion that the hidden clean signal information should be treated as a stochastic variable. This model is applied to estimate the uncertainty in noise cancelling using SS in a Mel filter bank, and this uncertainty (inverse of reliability) is employed to compute the weighting coefficient to be used in the modified DTW or Viterbi (HMM) algorithms. This uncertainty (in the form of a variance) is mainly caused by the lack of knowledge about the phase difference between noise and clean signals. The model for additive noise also suggests that SS could be defined by means of the expected value of the logarithm of the hidden clean signal energy given the noisy signal energy and the noise energy estimation.

The reliability in noise cancelling weighting is tested in an isolated word recognition task (digits) with several types of noise, and is shown to substantially reduce the error rate when SS is used to remove the additive noise using a poor estimation of the corrupting signal. The weighted Viterbi (HMM) algorithm is compared and combined with state duration modelling. It is shown that weighting the time varying signal information requires only a low computational load and leads to better results than the introduction of temporal constraints in the recognition algorithm. In combination with temporal constraints, the weighted Viterbi algorithm results in a high recognition accuracy at moderate SNR's without an accurate noise model.

When the signal is corrupted by both additive and convolutional noise, it is shown that the effect of the transmission channel function can be removed after the additive noise has been cancelled by means of SS. Cepstral Mean Normalization (CMN) and Maximum Likelihood estimation (MLE) of the convolutional noise are tested with SS and it is shown that the weighted Viterbi algorithm in combination with temporal constraints improves the performance of the additive and convolutional noise cancellation.

The techniques proposed in this thesis represent important theoretical contributions for speech recognition in noise and are interesting from the practical application point of view due to the simplicity of the assumptions made about the corrupting environment.

Acknowledgements

Firstly, I would like to extend my most sincere gratitude to Prof. Mervyn Jack, my first supervisor and Director of The Centre for Communication Interface Research (CCIR), for having guided and advised me about my work, my academic and professional career, for having always had time to discuss any idea or topic arising during my research, for having financially supported my attendance to several conferences, and for proof reading this thesis. I am grateful also for the support he has given me in whatever I have decided to do.

I would also like to specially thank Dr Fergus McInnes, my second supervisor, for all the corrections and improvements suggested over the last four years, for proof reading this thesis, and for having always at least a few minutes to discuss equations, algorithms or a paper that needed to be sent off on the following day. Dr McInnes also organized walks to the countryside near Edinburgh. Those walks gave me an opportunity to socialize with my colleagues and introduced me to the beautiful landscape of Scotland.

This work would not have been possible without the support of CNPq (Conselho Nacional de Pesquisa e Desenvolvimento) from Brazil, and Prof João Marcos Romano, my former supervisor at UNICAMP, who was essential in the application process for my PhD scholarship.

My life in Edinburgh was one of the most exciting periods I have ever had and I would like to thank the people who contributed to make my PhD such an interesting experience. Dr Fabrizio Carraro, for the fruitful discussions on speech processing and for having helped me with computing and programming problems. He is a person keen to help anyone and, as friends, we shared many of our experiences here. Dr Ian Nairn, one of the first friends I made in Edinburgh, has been responsible for the computing facilities that made this work possible. He and his wife Jeanette also invited me several times for dinner parties and Burn's suppers, which allowed me to know a bit more about the traditions of Scotland. Dr Mark Schmidt was also one of my first friends here and introduced me to the facilities in CCIR. Mr Keith Edwards provided me with the telephone database used in one of the chapters of this thesis and the hardware to reproduce it. There were other PhD students at CCIR and I would like to mention Dr Hussain Salleh for the support we gave to each other, especially during the last year.

The last two chapters were completed with HTK (Hidden Markov Models Toolkit) available in CSTR. I would like to thank Steve Isard (Director) for having allowed me to use HTK, Dr Alan Black for having given me the instructions to compile the software, and the PhD student Simon King, who is also finishing his thesis, for having helped me in my first steps with the program and for the suggestions and discussions concerning the experiments with HMM's.

Outside the university I have had many friends that have helped me and with whom I enjoyed my free time. Among them I would like to mention Karina Rau and Mark Winskel.

Finally, I would like to thank my parents, Nestor and Isabel, who provided the right environment for study and always encouraged us in our achievements. I am especially grateful to my mother who was strong enough to give me emotional support after my father's death.

I dedicate this thesis to my late father, Nestor Becerra Acevedo, and my mother, Isabel Yoma Yoma.

Glossary

The following list defines all the important terms in this document. When pertinent, they are discussed more carefully in the text.

Additive noise External process (e.g. background and line transmission noises) that is added to the clean speech.

AR Denotes autoregressive analysis (see section 4.2).

ASRS Automatic Speech Recognition Systems.

Back-propagation Algorithm to train multilayer perceptrons (see also *multilayer perceptrons*).

Baum-Welch Algorithm based on the EM technique to train HMM's (see also *EM* and section 2.3.3).

CDCN Code-book Dependent Cepstral Normalization. Technique based on ML estimation of noise proposed to address the noise robustness of ASRS (see also *ASRS* and section 2.4).

CDHMM Continuous density HMM's (see also *HMM* and section 2.3.2).

CMN Cepstral mean normalization. Technique employed to address the problem of convolutional noise (see also *convolutional noise* and section 2.4).

Convolutional noise Effect introduced by the transmission channel response or the replacement of microphones.

DCT Discrete Cosine Transform.

Delta First order differential of static parameters generally computed over a window of 3 or 5 frames (see also *frame* and *static coefficients*).

Delta2 First order differential of delta parameters generally computed over a window of 3 or 5 frames (see also *frame* and *delta*).

DFT Discrete Fourier Transform.

DHMM Discrete HMM's (see also *HMM* and section 2.3.2).

DP Dynamic Programming (see section 2.2).

DTW Dynamic Time Warping. A technique for speech recognition where a test utterance is compared to a reference one (see section 2.2).

EM Expectation-Maximisation algorithm for ML estimation (see also *ML*, and sections 2.3.3 and B.2).

End-point detection determination of the start and end of an utterance (see section 4.5).

FIR Finite Impulse Response digital filter.

Frame Short period of time, usually 20 or 30ms, in which the speech signal is supposed stationary. In speech recognition, the signal is divided in overlapped frames with the same length. In every frame, parameters (e.g. MFCC) are estimated and the speech signal is then represented by a sequence of parameter vectors (see also *MFCC*).

Global SNR SNR in a long interval (e.g. several utterances).

HMM Hidden Markov Model. A technique for speech recognition based on the stochastic model of phonetic units or words (see section 2.3).

IIR Infinite Impulse Response digital filter.

IMELDA Integrated Mel-scale with LDA (see also *LDA*). Technique proposed to address the noise robustness of ASRS (see also *ASRS* and section 2.4).

LDA Linear Discriminant Analysis (see section 2.4).

LIN Lateral Inhibition Net. Noise cancelling neural net based on multilayer perceptrons (see also *multilayer perceptrons* and section 3.2).

LMS least-mean-square. Adaptive filtering algorithm based on the gradient technique (see section 4.2).

Local SNR *see segmental SNR.*

Lombard effect Distortion on the speech signal when the speaker is led to speak more loudly in the presence of additive background noise.

Mel Perceptual frequency scale. According to psychoacoustic experiments, the perception is more sensitive to variations in the formant central frequencies when the formants are in the low part of the spectrum than when they are in the high one.

Mel filter bank array of filters based on the Mel scale. Below 1kHz the filters have approximately the same band width, but above 1kHz the band width is roughly directly proportional to the central frequency.

MFCC Mel Frequency Cepstral Coefficients. Parametrisation based on the Mel filter bank analysis. In MFCC the logarithm of the output energy is computed for all the filters, and then the cepstral transform (DCT inverse) is applied in order to reduce the number of parameters and increase the discriminability (see also *Mel filter bank*, *DCT* and section A.1).

ML Maximum Likelihood (see section 2.3.3).

MLE Maximum Likelihood Estimation (see section 2.4 and Appendix B).

Multilayer perceptrons Neural net composed by an input layer of sensory or source units, one or more hidden layers of computation nodes with a non-linear function, and an output layer of computation nodes (see section 3.2).

PDF Probability Density Function.

PMC Parallel Model Combination. Technique proposed to address the noise robustness of ASRS (see also *ASRS* and section 2.4).

Rasta RelAtive SpecTrAl.Technique proposed to address the noise robustness of ASRS (see also *ASRS* and section 2.4).

Segmental SNR SNR in a short interval such as 20 or 30 ms, which is generally the duration of a frame (see also *frame*).

SMC Short-term Modified Coherence. Technique to address the problem of additive noise in ASRS (see also *additive noise* and *ASRS* and section 2.4).

SNR Signal-to-Noise Ratio. Generally SNR is defined as being the logarithm in dB (ten times the logarithm to base 10) of the ratio between the clean and noise signal energies.

SS Spectral Subtraction. An additive noise cancelling technique based on the subtraction of the noise from the noisy signal energy (see also *additive noise*, and sections 2.4 and 5.5).

Static coefficients Parameters extracted from a single frame (see also *frame*, *delta* and *delta2*).

Temporal constraints Restrictions imposed to state durations in HMM's to improve the recognition accuracy (see also *HMM* and section 6.6.1).

Viterbi Decoding algorithm used in speech recognition based on HMM's (see also *HMM* and section 2.3.3).

WMA Weighted Matching Algorithms. Speech recognition algorithms that take into consideration the degree of signal distortion frame-by-frame (see section 2.5).

List of symbols

$a_{i,j}$: Transition probability in HMM. It is defined as being the probability of going (or staying if $i = j$) from state i to state j (see HMM in Glossary).

$b_j(T_t)$: Output probability in HMM. It is defined as being the probability of observing the frame T_t at time t given that at time t the process is in state $s_t = j$ (see HMM in Glossary).

C : The covariance of a multivariate Gaussian distribution.

$d(t, r)$: Local distance between parameter vectors T_t and R_r. In this thesis, the local distance corresponds to the squared Euclidean distance between the vectors.

$D(T, R)$: Overall distance between a testing T and a reference R sequences (see also R and T). This overall distance corresponds to $G(L_R, L_T)$ normalized to the alignment path length and is estimated by means of DTW (see $G(t, r)$ and DTW in Glossary).

E_m : The logarithm of the energy at the output of the filter m.

$G(t, r)$: Minimum overall or global matching distance along the optimum alignment path from the start point until $d(t, r)$. This global distance $G(t, r)$ defines the DP equation for the DTW algorithm and is used to obtain $D(T, R)$ (see $D(T, R)$ and DTW in Glossary).

I_i : Probability of state i being the first one (see HMM in Glossary).

$H_A(z)$: Autoregressive filter estimated in non-speech intervals to model an additive noise process.

H^c : Convolutional noise in the cepstral domain represented by a vector of constants.

L_R : Length in number of frames of a reference parameter vector sequence.

L_T : Length in number of frames of a testing parameter vector sequence.

$LI()$: Output of LIN (Lateral Inhibition Net), a noise cancelling neural net proposed in Chapter 3.

λ : Set of parameters (matrix of transition probabilities, matrix of output probabilities and initial state distribution) that defines an HMM (see $a_{i,j}$, $b_j(T_t)$ and I_i, and HMM in Glossary).

μ : The mean vector of a multivariate Gaussian distribution.

$N(T_t, \mu_j, C_j)$: Multivariate Gaussian distribution to model the output probability of state j. T_t is the observed parameter vector at time t, μ_j the mean vector, and C_j the covariance matrix (see $b_j(T_t)$ and HMM in Glossary).

NST : Stationarity coefficient (see section 4.4).

Q : Kullback-Leibler number employed by the EM algorithm (see EM in Glossary).

R : Reference pattern composed by a sequence of parameter vectors.

SD : Spectral Density comparison (see section 4.3).

$SsThr_m$: Threshold at channel m employed by spectral subtraction defined in (5.26), section 5.5.

T : Testing pattern composed by a sequence of parameter vectors.

$VarThr$: Threshold used by the weighting function defined in (5.30), section 5.7.

Contents

List of figures

List of tables

Chapter 1

Introduction

1.1 Speech recognition in noise

In speech technology, most of the research in the last 10 or 20 years has focused on automatic speech recognition, a fascinating topic mainly due to the fact it tries to emulate one of the most natural of human skills. Although a huge amount of work has been done by thousands of researchers in different countries, the human capability of decoding the acoustic linguistic information in variable contexts remains unreachable. However, the technology currently available allows systems to use speech as an input interface always assuming restrictions concerning the way in which the words are uttered, the speaker, the vocabulary and the noise. Due to the fact that the parameterisation process is based on physical measures of the acoustical signal (e.g. energy at the output of filters), automatic speech recognition systems (ASRS) are very sensitive to mismatches between training and testing conditions and noise robustness remains as the main problem to be solved in order to make ASRS successful in real applications.

This thesis addresses the problem of robustness of ASRS to additive and/or convolutional noise. Additive noise corresponds to an external process (e.g. background and line transmission noise) that is added to the clean speech. Convolutional noise denotes the effects introduced by the transmission channel response or replacement of microphones. The cancellation of both types of noise is theoretically and experimentally difficult and a generic solution is still not available, athough the results presented in this thesis alleviate the restrictions imposed by other methods and open a new research topic in speech recognition.

Other sorts of distortions such as Lombard effect (J.C.Junqua, 1989) (A.Wakao *et al.*, 1996), when the speaker is led to speak more loudly in the presence of background noise, and reverberation

are not treated in this thesis.

1.2 Reliability in noise cancelling

As discussed in the following chapters, the convolutional noise seems to be much easier to deal with than the additive one. This is due to the fact that the first one is reasonably constant and is supposed to equally corrupt the speech signal independently of the signal level. On the other hand, the additive noise can be non-stationary and clearly corrupts some segments of the signal more badly than others. Ordinary acoustic matching algorithms (e.g. DTW and HMM) give to all frames the same weight in the recognition process but, if some segments are more severely corrupted than others, this is conceptually wrong. Consequently, the classical concept of recognition algorithms should be revised in order to take into account the local or segmental signal-to-noise ratio (SNR) on a frame-by-frame basis. To do so, weighting matching algorithms (WMA) (N.B.Yoma *et al.*, 1995) (N.B.Yoma *et al.*, 1996d) (N.B.Yoma *et al.*, 1996a) (N.B.Yoma *et al.*, 1997b) (N.B.Yoma *et al.*, 1997a) (N.B.Yoma *et al.*, 1998a) (N.B.Yoma *et al.*, 1998b) were proposed and experiments showed that WMA can be effective in reducing the error rate, although the weighting coefficients should also be a function of the *reliability in noise cancelling*. The

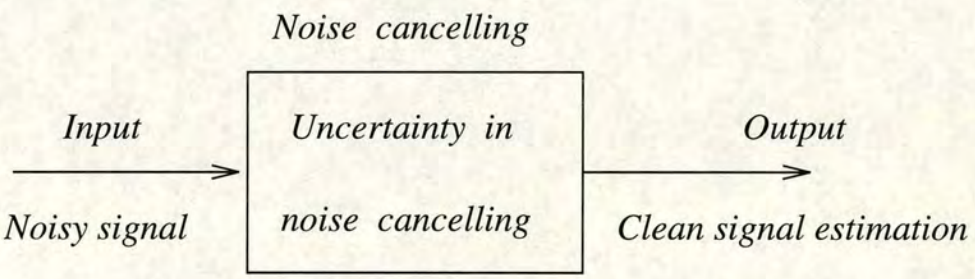


Figure 1.1: Noise cancelling technique seen as a system.

idea of reliability in noise cancelling rises from considering a noise removing technique as a system (see Fig.1.1) whose input is the noisy signal (always in the spectral domain in this thesis) and output is the estimation of the clean signal. Related to that system, it is possible to define the uncertainty (inverse of reliability) in noise cancelling as being the mean distance between the clean signal estimation and the original clean signal, which is unknown. The ordinary and weighted speech recogniton systems are shown in Fig. 1.2 and Fig. 1.3, respectively. As can

be seen, a) the ordinary recognition system is a special case of the weighted one and b)the idea presented in Fig. 1.3 should be able to be generalised to other fields of pattern recognition.
In the context of speech recognition, WMA should also alleviate the restriction concerning the stationarity of the additive noise due to the fact that WMA give to higher local SNR frames (less corrupted and more accurately estimated) a higher weight in the recognition process and variations on the corrupting signal should affect more severely those frames with lower local SNR and lower weight in the recognition algorithm. Finally, additive and convolutional noise cancelling techniques can also be used in the context of WMA and the method can be employed to cancel both types of distortion.

1.3 Robust speech pulse detection

Inaccurate detection of the endpoints is a major cause of errors in automatic speech recognition systems. Usual parameters which endpoint detecting techniques are based on such as energy levels, pitch, zero- and/or level-crossing rates, and timing may be insufficient for the correct detection of a speech pulse if the additive noise is present at a low SNR. In order to make speech pulse detection more robust to the background noise an end-point detector based on AR analysis of noise is proposed. AR analysis assumes that the additive noise is reasonably stationary over the observation/estimation period and strongly simplifies the complexity of the speech signal detector. The idea is that the AR filter trained with short only-noise intervals should be able to emphasize those components (in the frequency domain) of the speech signal that have lower energy in the noise.

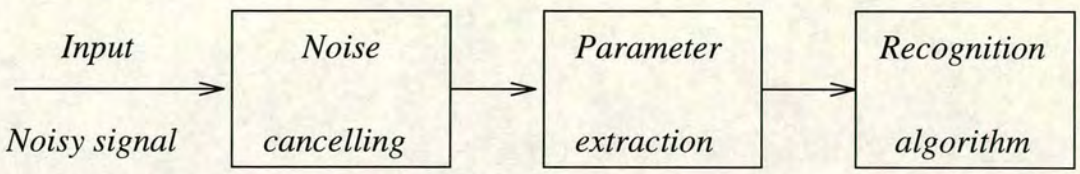


Figure 1.2: Ordinary speech recognition system.

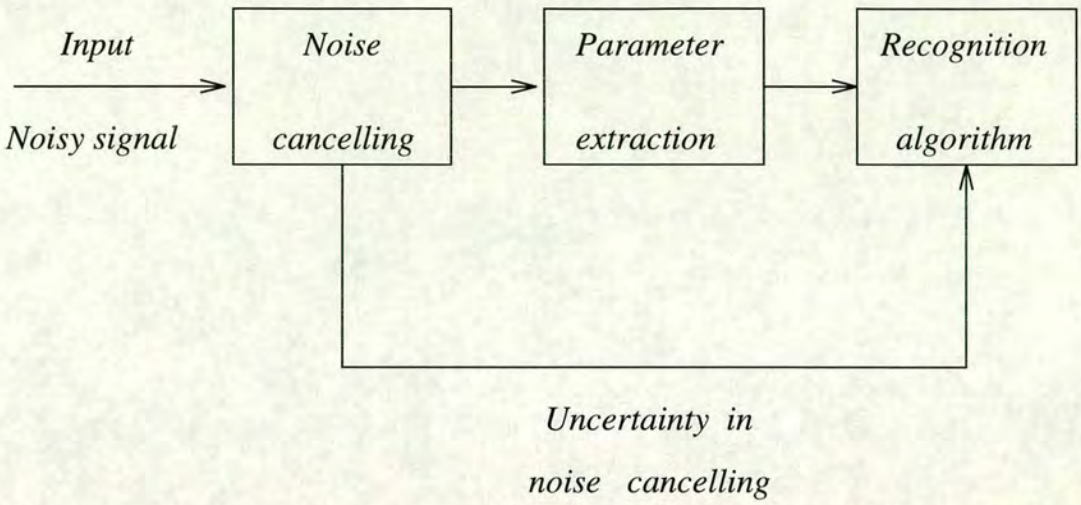


Figure 1.3: Weighted speech recognition system.

1.4 Thesis structure

This thesis is composed of six chapters in addition to this Chapter (Introduction) and Chapter 8 (Conclusions). In Chapter 2 a summary of acoustic pattern matching algorithms in the context of DTW (Dynamic Time Warping) and HMM (Hidden Markov Model) is given. Then, the problem of speech recognition in noise is discussed, techniques so far proposed to deal with the problem are highlighted and the idea of weighted matching algorithms (WMA) is presented as a solution to both types of distortions (additive and convolutional) that reduces the restrictions of previous methods.

In Chapter 3, a weighted DTW is introduced in the context of a noise reduction neural net LIN (Lateral Inhibition Net). Lateral inhibition is one of the processes responsible for the masking phenomena in different sensory systems, and it serves to sharpen a spatial input pattern by emphasizing its edges and peaks. The idea of the weighted matching algorithms was proposed to take into the account the fact that LIN should more easily remove noise from frames with high local SNR rather than low local SNR. Experiments related to the weighted DTW are reported in this chapter. The proposed weighted DP algorithm needs only one step, is compared to a two-step weighted DTW previously proposed and was proved effective in reducing the error rate for white Gaussian additive noise using as a weighting parameter the ratio between the estimated clean signal and the noisy signal energies. Experiments show that the improvement that results from weighting the information along the signal depends on the initial conditions of the LIN training

procedure and, consequently, the weighting coefficient should also take into consideration the response of the neural net. This is proved by means of computing the weighting coefficient using the uncertainty in noise cancelling estimated with a mean distortion curve characteristic of the neural net.

Adaptive autoregressive (AR) modelling of noise is proposed in Chapter 4 in order to reduce the influence of the corrupting signal in automatic speech pulse detection. Two forms of frame comparison are studied: spectral density comparison between noise and noisy speech signals; and non-stationarity measure. Finally, an end-point detector is proposed and tested on isolated digits corrupted by car and speech noises. The FIR filters employed in the autoregressive analysis are trained with the LMS algorithm during non-speech intervals.

Chapter 5 presents a novel model for additive noise, based on Mel filter banks (IIR and DFT), in which the hidden information of the clean signal energy is a function of the observed noisy energy signal, the noise energy (that can be approximately estimated) and the phase difference between the clean speech and noise signals. It is proposed that when the noise is added an uncertainty is introduced and the original clean signal cannot be recovered with 100% accuracy because the phase difference between corrupted and corrupting signals is unknown. Consequently, the hidden clean signal energy is treated as a stochastic variable and, assuming that the phase difference is uniformly distributed between $-\pi$ and π, it is proved that the spectral subtraction (SS) estimation corresponds to expected value of the clean signal energy given the noisy and noise signal energies. Using the same procedure, the uncertainty in noise cancelling, defined as being the mean quadratic distance between the clean signal estimation and the original clean signal, is estimated. This uncertainty corresponds to the variance of the hidden clean signal energy given the noisy and noise signal energies and is used to estimate the weighting coefficients for a modified weighted DTW. Results strongly confirmed that WMA can substantially reduce the error rate but the weighting function uses a threshold whose optimum value is still case dependent, although a wide range of only slightly sub-optimal values is achieved.

In Chapter 6, a weighted Viterbi algorithm (HMM) is proposed and applied in combination with a weighting function that does not need any free variable in isolated word recognition. This modified Viterbi algorithm is compared and combined with state duration modelling and it is shown that weighting the information along the signal leads to better results than the introduction

of temporal constraints in the recognition algorithm. Combined with temporal constraints, the weighted Viterbi algorithm results in a high recognition accuracy at SNR=18, 12 and 6dB without an accurate noise model. Also in this Chapter, the introduction of temporal constraints is discussed and the importance of modelling the state duration with a parametric distribution (e.g. gamma) is evaluated. Finally, the problem of connected digits is evaluated under the perspective of WMA.

The problem of additive and convolutional noise removal is addressed in Chapter 7. It is proposed that the effect of the transmission channel function can be removed after the additive noise has been removed by means of SS. Two convolutional techniques are addressed and applied in combination with SS: Cepstral Mean Normalization (CMN) and Maximum Likelihood estimation (MLE). When SS and CMN are applied together, it is shown that the weighted Viterbi algorithm with temporal constraints also leads to a substantial reduction in the error rate and a high recognition accuracy is achieved at SNR equal to 18, 12 and in some cases at 6dB when the speech signal is corrupted by additive noise and is distorted by a 6dB/oct spectral tilt. It is also proposed a novel strategy in which the additve noise should be cancelled firstly and the convolutional one should be estimated and/or removed using the output of the additive noise cancelling system. This is based on the fact that the additive noise could easily change with time and the convolutional one corresponds to the transmission channel characteristics that are supposed time invariant.

Finally, Chapter 8 summarizes the contributions and conclusions of this thesis. Future work is also addressed in this chapter.

Chapter 2

Acoustic pattern matching and noise robustness

2.1 Introduction

In this chapter, acoustic pattern matching is discussed in the context of DTW (Dynamic Time Warping) and HMM (Hidden Markov Model). Then, speech recognition in noise is addressed and the techniques previously proposed to deal with the problem are highlighted. Finally, a novel approach, termed weighted matching algorithms (WMA), is presented as a solution to improve the robustness of speech recognition systems based on DTW or HMM to both types of distortions (additive and convolutional) reducing the restrictions of previous methods.

2.2 Dynamic Time Warping (DTW)

DTW or Dynamic Programming (DP) matching (H.Sakoe & S.Chiba, 1978) (L.R.Rabiner & S.Levinson, 1981) (L.Rabiner & B.H.Juang, 1993) was introduced to compare two sequences or utterances taking into consideration that the duration of quasi-stationary intervals does not give any relevant phonetic information. DTW computes the sum (global distance) of local distances between reference and testing sequences (see Fig. 2.1) along the optimal alignment path. In this sense, two utterances belonging to the same word and speaker should give a low and similar overall distance independently of the articulation rate. The overall distance $D(T, R)$ between a testing T and a reference R sequence is given by,

$$D(T,R) = \min(\text{alignment path}) \frac{\sum_{k=1}^{K} d(p(k)) \cdot w(k)}{\sum_{k=1}^{K} w(k)} \tag{2.1}$$

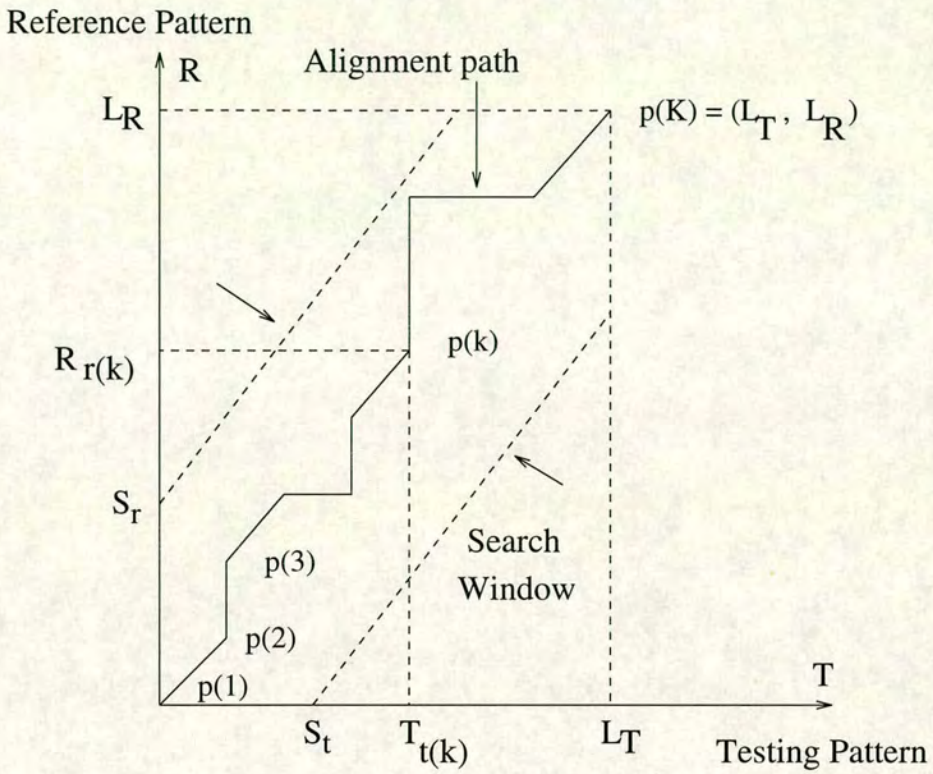


Figure 2.1: DTW and temporal alignment.

where the optimal alignment path is restricted to a search window (Fig. 2.1), which is generally symmetric but can also present different values for S_t and S_r. The local distances $d(p(k))$ are computed along the alignment path and $p(k)$ denotes which testing and reference frames are being compared:

$$p(k) = (t(k), r(k)) \tag{2.2}$$

where $t(k)$ and $r(k)$ are the indexes for the testing and reference utterances, respectively. The coefficients $w(k)$ give a higher weight to those segments of alignment path that are parallel to the diagonal line linking the first and last frames of both utterances. The denominator of 2.1 normalizes the overall distance $D(T, R)$ to the length of the alignment path.

Besides the restriction imposed by the search window, the optimal alignment path needs to satisfy a) end-point, and b) continuity and monotonicity constraints. As far as the end-point constraints are concerned, DTW, in its original form, sets that the starting and end-point of the

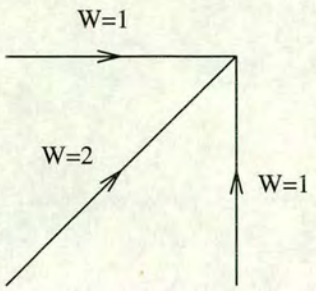


Figure 2.2: Local condition for DP matching.

aligment path should coincide with the extremes of both utterances:

$$t(1) = r(1) = 1 \tag{2.3}$$

and

$$t(K) = L_T \tag{2.4}$$

$$r(K) = L_R \tag{2.5}$$

where L_T and L_R are, respectively, the length of the testing and reference sequences. It is evident that the end-point constraints make DTW very sensitive to the end-point detection, which in turn is extremely dependent on the SNR for the case of speech corrupted by additive noise (the average length of the testing utterances tends to decrease as the SNR gets more severe). In order to counteract this effect, the end-point constraints on the DP algorithms can be relaxed by means of opening up the ends of the search region allowing the alignment path to start by comparing the first frame of the testing pattern with any of the first reference frames inside the search window, and to end by comparing the last test frame with any of the last reference frames inside the search window. Within the algorithm, continuity and monotonicity constraints (H.Sakoe & S.Chiba, 1978) (X.D.Huang *et al.*, 1990), represented by local conditions, determine for every pair $p(k) = (t(k), r(k))$ which possible previous pairs $p(k-1)$ are used in path computations. Two local conditions, shown in Figs. 2.2 and 2.3, were used in this research and they are

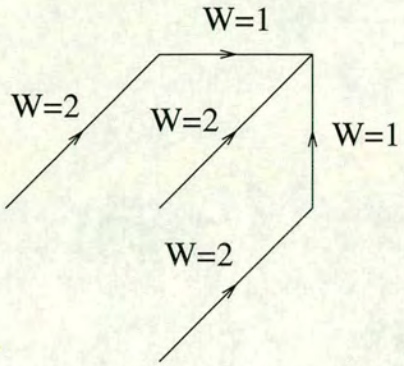


Figure 2.3: Local condition for DP matching.

represented respectively by.

$$p_1(k-1) = \begin{cases} (t(k)-1, r(k)) \\ (t(k)-1, r(k)-1) \\ (t(k), r(k)-1) \end{cases} \tag{2.6}$$

and

$$p_2(k-1) = \begin{cases} (t(k)-2, r(k)-1) \\ (t(k)-1, r(k)-1) \\ (t(k)-1, r(k)-2) \end{cases} \tag{2.7}$$

The DP programming matching computes for every allowed $p(k)$ the minimum accumulated distance $G(t,r)$ for every possible $p(k-1)$ given by (2.6) or (2.7) and chooses the smallest one. In other words, the local conditions shown in Figs 2.2 and 2.3 give, respectively, the following DP equations that are computed from the first to the last frame of both sequences inside the search window:

$$G_1(t,r) = \min \begin{pmatrix} G(t-1,r) + d(t,r) \\ G(t-1,r-1) + 2 \cdot d(t,r) \\ G(t,r-1) + d(t,r) \end{pmatrix} \tag{2.8}$$

and

$$G_2(t,r) = \min \begin{pmatrix} G(t-2,r-1)+2\cdot d(t-1,r)+d(t,r) \\ G(t-1,r-1)+2\cdot d(t,r) \\ G(t-1,r-2)+2\cdot d(t,r-1)+d(t,r) \end{pmatrix} \quad (2.9)$$

According to the principle of optimality (X.D.Huang *et al.*, 1990) (D.A.Pierre, 1986), the accumulated distance at $G(L_T, L_R)$ (see Fig.2.1) corresponds to the minimum global distance between the testing and reference utterances. The resulting alignment path is said to be optimum and is estimated by means of going backward from $p(K) = (L_T, L_R)$ until $p(1) = (1, 1)$.

2.3 Hidden Markov Models

Speech is an intrinsically stochastic process in the sense that the same word or phoneme may present inter and intra speaker random variations in terms of the acoustic characteristics of the speech signal: spectral density distributions and durations. Consequently, a deterministic approach (e.g. DTW) is in principle inadequate to model the speech process once the same phonetical unit (or word) may be physically represented by different signals. Hidden Markov Models (HMM) are the most popular and successful technique applied to speech recognition because it is able to store the information of many training utterances in the form of parameters of statistical distributions.

2.3.1 Definitions of HMM

An example of an HMM topology, 5-state left-to-right topology without skip-state transition, that could be used for word modelling is shown in Fig. 2.4:

1. The number of states is equal to 7 and they are denoted by $S = (s_1, s_2, s_3, ..., s_7)$: 5 emitting states and 2 non emitting ones. If the process is in one of the emitting states a frame or frames, represented by parameters of vectors (e.g. MFCC), are observed. This does not happen on the non-emitting states whose purpose is to allow to link two or more models as in the case of connected or continuous speech recognition.

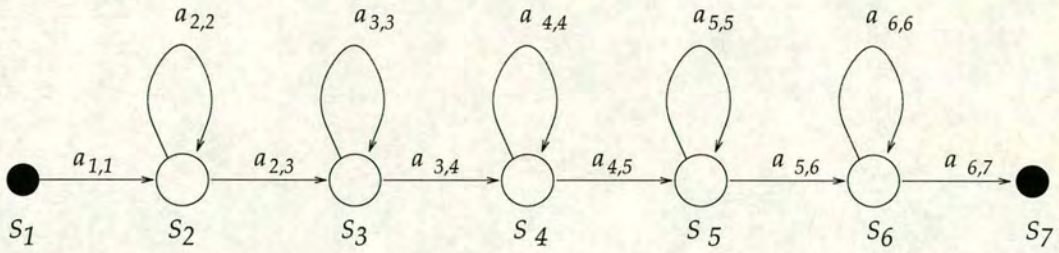


Figure 2.4: A left-to-right HMM without skip-state transition. The topology is composed by two non-emitting states (1 and 7) and five emitting ones(2-6).

2. To each emitting state it is possible to associate a probability distribution, or probability density function, that tries to model which frames are more or less likely on a given state. This output probability is denoted by:

$$b_j(T_t) = Pr(T_t|s_t = j) \tag{2.10}$$

which means the probability of observing the frame T_t at time t given that at time t the process is in state $s_t = j$. The set of all the output probabilities is indicated by the matrix B.

3. The transition between states is modelled by a_{ij} that denotes the probability of going (or staying if $i = j$) from state i to state j. The set of all the transition probabilities is indicated by the matrix A. In its original form, HMM assumes that a_{ij} are constant and independent of time t which leads to a geometric distribution for state durations. Although the HMM technique using a constant transition probability has led to good results, this geometric probability distribution is inadequate to model state durations (L.R.Rabiner *et al.*, 1989) and different approaches to include more accurate probability distribution for state and word time durations have been proposed (J.D.Ferguson, 1980) (L.R.Rabiner *et al.*, 1989) (M.J.Russell & R.K.Moore, 1985) (S.E.Levinson, 1986) (D.Burshtein, 1996) (K.Laurila, 1997).

4. The initial state distribution is defined by I:

$$I = \{I_i|I_i = Pr(s_1 = i)\}$$

and indicates the probability of the first state being i.

Given the topology and the number of states, an HMM is defined by λ,

$$\lambda = (A, B, I) \tag{2.11}$$

where λ is composed by the parameter matrices A, B and I.

2.3.2 Output probability distributions

The output probability distributions that are associated with each emitting state could be modelled by discrete (DHMM - Discrete Hidden Markov Models) or continuous distributions (CDHMM-Continuous density HMM). For the case of continuous distributions (the one used in this thesis) the parameters $b_j(T_t) = Pr(T_t|s_t = j)$ correspond to Probability Density Functions (PDF) although they are still referred to as "probabilities". The most common PDF used in CDHMM is the multivariate Gaussian due to the facts that (X.D.Huang *et al.*, 1990):

1. Any continuous PDF can be approximated by means of Gaussian mixture densities;
2. according to the Central Limit Theorem, a sum of independent random variables tends to a Gaussian distribution;
3. given a variance, Gaussian distribution is the one that presents the highest entropy.

Using a single Gaussian mixture, $b_j(T_t)$ is given by:

$$b_j(T_t) = N(T_t, \mu_j, C_j) \tag{2.12}$$

where $N(T_t, \mu_j, C_j)$ denotes a multivariate Gaussian distribution, μ_j the mean vector, and C_j the covariance matrix. The unimodal output densities may not be accurate enough to model interspeaker variability in medium and large vocabulary tasks and Gaussian mixture densities (L.R.Rabiner *et al.*, 1985) are needed:

$$b_j(T_t) = \sum_{k=1}^{K} c_{j,k} \cdot N(T_t, \mu_{j,k}, C_{j,k}) \tag{2.13}$$

where K is the number of Gaussians and $\mu_{j,k}$ and $C_{j,k}$ are the vector mean and covariance matrix of one Gaussian component. The coefficient $c_{j,k}$ is the weight associated to each mixture component.

2.3.3 Algorithms for HMM

From the operational point of view, there are three problems related to HMM. The first one concerns how to estimate the probability or likelihood, $Pr(T|\lambda)$, given a sequence of parameter vectors (T) and an HMM (λ), where

$$T = (T_1, T_2, T_3, ..., T_t, ...T_{L_T})$$

and L_T is the length of the observation sequence. For a given sequence of states S, the likelihood is given by:

$$Pr(T|S,\lambda) = b_{s(1)}(T_1) \cdot b_{s(2)}(T_2) \cdot b_{s(3)}(T_3) \cdot ...b_{s(L_T)}(T_{L_T}) \tag{2.14}$$

and the total likelihood could be estimated by means of summing (2.14) in all the possible state sequences (X.D.Huang *et al.*, 1990) (L.Rabiner & B.H.Juang, 1993) :

$$Pr(T|\lambda) = \sum_{all\ S} \prod_{t=1}^{L_T} a_{s(t-1),s(t)} \cdot b_{s(t)}(T_t) \tag{2.15}$$

Computationally, $Pr(T|\lambda)$ as defined by (2.15) is very expensive and $Pr(T|\lambda)$ is usually computed by means of the $forward - backward$ algorithm.

During the training procedure, an HMM is attributed to a word or phonetic class, and the parameters A, B and I are estimated using training examples of the phonetic class that the HMM belongs to. The training is generally done by means of the Baum-Welch algorithm that is a form of the EM (Expectation-Maximisation) technique.

In the recognition process, the testing sequence of parameter vectors is processed by trained models and the HMM (isolated word recognition) or sequence of HMM's (connected or continuous word recognition) with the highest likelihood is chosen as being the recognized word or sequence of words. During the test procedure, the likelihood is usually computed by means of the Viterbi algorithm that finds the optimum sequence of states and its likelihood given a testing utterance and a HMM or sequence of HMM's. The Viterbi algorithm is preferred over the Forward-Backward one for being computationally more efficient and, although the maximum likelihood does not correspond to $Pr(T|\lambda)$ (2.15), it yields good experimental results.

Forward-Backward algorithm

The Forward algorithm makes use of the variable $\alpha_t(i)$ defined as being the probability of the partial observation sequence to time t and state i which is reached at time t:

$$\alpha_t(i) = Pr(T_1, T_2, T_3, ..., T_t, S(t) = i|\lambda) \tag{2.16}$$

This probability is estimated inductively in the Forward algorithm (X.D.Huang *et al.*, 1990) (L.Rabiner & B.H.Juang, 1993) as follows:

STEP 1: $\alpha_1(i) = I_i \cdot b_i(T_1)$, for all states i;

STEP 2: Computing $\alpha()$ along the time axis, for $t = 2, 3, ..., L_T$ and all states j:

$$\alpha_t(j) = [\sum_i \alpha_{t-1}(i)a_{i,j}] \cdot b_j(T_t) \tag{2.17}$$

STEP 3: The probability of the sequence is given by

$$Pr(T|\lambda) = \sum_{i \epsilon S_F} \alpha_{L_T}(i) \tag{2.18}$$

where S_F denotes the set of possible final states for a given HMM.

Complementarily, the backward variable $\beta_t(i)$ is defined as being the probability of the partial observation sequence from $t+1$ to the final observation at L_T , given the state i at time t and the model λ:

$$\beta_t(i) = Pr(T_{t+1}, T_{t+2}, T_{t+3}, ..., T_{L_T}|S(t) = i, \lambda) \tag{2.19}$$

As in the Forward algorithm, $\beta_t(i)$ can also be inductively computed by means of the Backward algorithm:

STEP 1: $\beta_{L_T}(i) = \frac{1}{N_F}$, for all states $i \epsilon S_F$, otherwise $\beta_{L_T}(i) = 0$.

STEP 2: Estimate $\beta()$ along the time axes for $t = L_T - 1, L_T - 2, L_T - 3, ..., 1$ and all states j:

$$\beta_t(j) = [\sum_i a_{j,i} \cdot \beta_{t+1}(i) \cdot b_i(T_{t+1})] \tag{2.20}$$

Viterbi algorithm

Given an observation sequence, the state sequence cannot be uncovered but the maximum likelihood state sequence can be found by means of the Viterbi algorithm which also provides the likelihood for the estimated sequence. The Viterbi algorithm is very similar to the DTW algorithm previously discussed in section 2.2 and is generally employed during the recognition process where the likelihood of the optimal state sequence is used instead of the total likelihood that results from the Backward-Forward algorithm. The Viterbi algorithm is given by:

STEP 1 : Initialization. For each state i,

$$\delta_1(i) = I_i \times [b_i(T_1)]$$

$$\psi_1(i) = 0$$

STEP 2 : Recursion. From t=2 to L_T, for all states j,

$$\delta_t(j) = Max_i[\delta_{t-1}(i) \times a_{ij}] \times [b_j(T_t)]$$

$$\psi_t(j) = argmax_i[\delta_{t-1}(i) \times a_{ij}]$$

STEP 3: Termination. (* indicates the optimised results).

$$P^* = Max_{s \in S_f}[\delta_{L_T}(s)]$$

where L_T is the frame sequence length, s_F is the set of possible final states, $\delta_t(j)$ is the maximum likelihood at state j at time t that corresponds to the optimum state sequence from s(1) to s(t) = j, $\psi_t(j)$ denotes the state at time $t - 1$ in the optimum state sequence.

Training algorithm for HMM

The estimation of the parameters $\lambda = (A, B, I)$ is the most difficult task involving HMM's because it is a multi-dimensional optimisation problem without an analytical solution (X.D.Huang *et al.*, 1990) (L.Rabiner & B.H.Juang, 1993). However, an iterative algorithm based on the gradient technique is used to re-estimate the HMM parameters in order to increase the likelihood iteration-by-iteration until a local optimum is reached. This method, usually referred to as Baum-Welch

algorithm (L.E.Baum & J.E.Eagon, 1967), makes use of the information-theoretic function Q (i.e. Kullback-Leibler number (L.E.Baum *et al.*, 1970) (S.Kullback & R.A.Leibler, 1951) defined as

$$Q(\lambda, \hat{\lambda}) = \frac{1}{Pr(T|\lambda)} \cdot \sum_{all\ S} Pr(T, S|\lambda) \log Pr(T, S|\hat{\lambda}) \tag{2.21}$$

where $Q(\lambda, \hat{\lambda})$ is considered as a function of $\hat{\lambda}$ in the maximisation procedure. According to the EM algorithm (A.P.Dempster *et al.*, 1977) if

$$Q(\lambda, \hat{\lambda}) \geq Q(\lambda, \lambda)$$

then,

$$Pr(T|\hat{\lambda}) \geq Pr(T|\lambda)$$

The EM algorithm chooses $\hat{\lambda}$ that maximises $Q(\lambda, \hat{\lambda})$ in each iteration so it is guaranteed that the log-likelihood $\log Pr(T|\lambda)$ also increases iteration-by-iteration toward a local optimum. Although not employed in this thesis, the training algorithm can also rely on other criteria such as Maximum Mutual Information (MMI) (L.R.Bahl & et al., 1986) and Minimum Discrimination Information (MDI) (Y.Ephraim & L.R.Rabiner, 1990).

Speech recognition with HMM's

After having trained the HMM's (one per word or phonetic unit), the recognition process consists in finding the HMM with highest likelihood (for the isolated word case) or the most likely HMM sequence (for the connected or continuous word task). As far as connected or continuous speech recognition is concerned, algorithms such as Level Building and One-Pass (L.Rabiner & B.H.Juang, 1993) can be considered as being generalizations of the Viterbi algorithm for isolated words.

2.4 Speech recognition in noise

The performance of speech recognition systems degrades abruptly when the signal is corrupted by additive noise or distorted by the transmission channel (convolutional noise). This is the main problem faced by speech recognition systems in real applications and many techniques

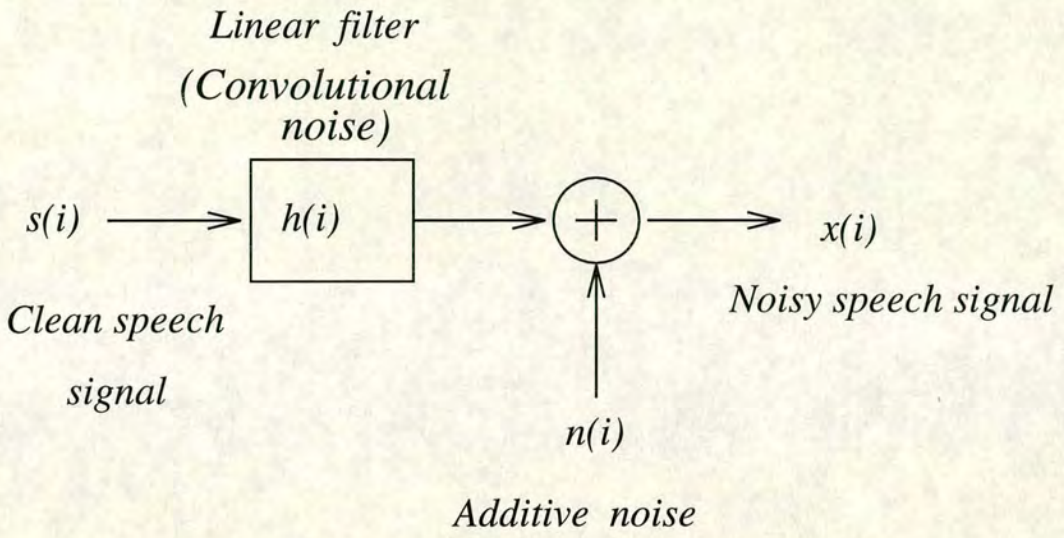


Figure 2.5: Model for the additive and convolutional noise.

have been proposed to address the issue although none of the methods is generic and completely effective in all the circumstances. The model for the corrupted speech signal adopted in this thesis is shown in Fig. 2.5 . The additive noise denotes the addition of an external corrupting signal to the speech, and can be modelled as an additive process in the linear domain either in the temporal or frequency domains. On the other hand, the convolutional distortion is caused by the insertion of the transmission channel (microphone and telephone line) and is generally an additive process in the logarithmic and cepstral domain. The name *convolutional* is due to the fact that the transmission channel could be modelled as a linear filter, h(i) in Fig. 2.5, and the distorted speech signal is the result of the *convolution* of the filter impulsive response with the speech signal in the temporal domain. In the frequency domain, this convolution becomes a multiplication, which in turn becomes a sum after applying the logarithmic function.

The techniques that have been proposed to address the problem of noise robustness can be divided in: firstly, inherently robust speech parameters; secondly, clean speech estimation; and finally, model-based techniques.

In inherently robust speech parameters, Cepstral Mean Normalization (CMN), Short-term Modified Coherence (SMC), Linear Discriminant Analysis (LDA) and Rasta filtering can be mentioned. CMN, which also can be applied in the logarithmic domain, consists in subtracting from the cepstral parameters the coefficient mean estimated in a reasonably long period of

time and is very effective to cancel the influence of the convolutional noise. However, CMN loses its effectiveness when additive noise is present. Short-term Modified Coherence (SMC) (D.Mansour & B.H.Juang, 1989) uses an all-pole modelling of the autocorrelation sequence and a spectral shaper and assumes that the noise affects only the first few autocorrelation parameters. This assumption is valid only for poorly correlated noises and the technique addresses only the additive noise case. Integrated Mel-scale with Linear Discriminant Analysis (IMELDA) (M.J.Hunt & C.Lefebre, 1989) gives more robust parameters than the ordinary MFCC, although the robustness of these features is still limited at lower SNR's. In order to improve the results, IMELDA should be employed using data collected in a particular noisy environment (O.Siohan, 1995) which substantially reduces the applicability of the technique. Rasta (H.Hermansky *et al.*, 1991) (J.Koehler *et al.*, 1994) was initially developed to address the convolutional noise problem assuming that the channel characteristic is stationary. Rasta applies high-pass or band-pass filtering to the temporal trajectory of spectral parameters assuming that the channel transfer function is mainly in low frequency components. In order to generalise the technique to the additive and convolutional noise case, Rasta-J (H.Hermansky *et al.*, 1993) was proposed and was shown useful to improve the robustness of speech recognition systems, but the method depends on a variable J which is case dependent and does not have an analytical solution.

As far as clean speech estimation is concerned, Spectral Subtraction (SS), Code-book Dependent Cepstral Normalization (CDCN) and State-Based Speech Enhancement have been proposed. Spectral Subtraction (SS) (S.F.Boll, 1979) is the most popular clean signal estimation technique and consists basically in subtracting the noise signal energy (estimated in non-speech intervals) from the noisy signal energy (Compernolle, 1989), although a more general form of SS was proposed in (M.Berouti *et al.*, 1979) in order to reduce the distortion in low SNR frames. SS provides a reasonable estimation for high SNR frames but loses accuracy when the speech and noise signal energies are similar. Code-book Dependent Cepstral Normalisation (CDCN) (A.Acero & R.Stern, 1990) makes use of a stochastic model of the speech process in the form of a code-book and estimates both additive and convolutional noises by means of the Maximum Likelihood criteria. However CDCN, and later versions such as the VTS (Vector Taylor Series approximation) algorithms (P.Moreno, 1996), is computationally expensive, makes intensive use of approximations due to the fact that the equations do not present an analytical solution,

and loses accuracy (the error rate increases more abruptly) when SNR$\leq$10dB. Besides that, the method was tested in utterances composed by several words which suggests that the ML estimation of the noises needs a reasonable amount of data, which in turn makes the technique vulnerable to variations in the dynamics of the additive noise. Finally, State-Based Speech Enhancement (SBSE) (C.W.Seymour & M.Niranjan, 1994) uses a model of the interfering additive noise and statistics of the clean speech signal to estimate the clean speech. The method performs initially a recognition pass using a set of HMM's in order to estimate the optimal frame-state sequence. SBSE was tested with only additive noise and, in order to improve the estimation of the alignment, parallel model combination was used which limits the applicability of the technique to very stationary additive noises.

The most popular model-based technique is Parallel Model Combination (PMC) (Gales, 1995) (Gales & S.Young, 1996) (Gales & S.Young, 1995). PMC is based on the fact that the best recognition results should be achieved when the training database is recorded under at the same conditions as the testing utterances and in this sense the HMM's trained with clean speech signals are combined with a HMM of noise in order to generate HMM's of noisy speech. The technique gives a high accuracy when the noise is corrupted by additive noise even at low SNR's. However, the technique presents some disadvantages (M.F.Gales, 1997): it is not easily used with some types of parametrizations such as Cepstral Mean Normalization (CMN); it has problems of adaptation speed for non-stationary noises; it is computationally expensive when compared to SS and CMN; and finally, it mainly addresses the problem of additive noise and the cancellation of the convolutional noise needs a previous knowledge about the channel response. The noise estimation is very important and PMC requires an accurate noise model to generate the noisy HMM's.

2.5 Weighted Matching Algorithms

It seems that convolutional noise is much easier to deal with than additive noise. The influence of the transmission channel response can be effectively cancelled by means of CMN or Rasta filtering but, in order to reduce the effect of the additive noise, several restrictions need to be assumed and the techniques that yield good results are more complex. Basically, two characteristics make the difference between both types of noise: firstly, stationarity seems to

be very natural for the convolutional noise but it is not for the additive one which results in the fact that the additive noise techniques are, in a higher or lower degree, sensitive to the variations on the corrupting signal dynamics; secondly, it is not unreasonable to suppose that the convolutional noise distorts the speech signal uniformly independently of the signal level, and this is completely untrue for the additive noise.

As can be seen in Fig. 2.6, additive noise corrupts some segments of the speech signal more severely than others and that leads to the fact that the classical concept of speech recognition algorithm where all the frames have the same influence in the recognition process should be revised to include the degree of corruption or reliability in noise cancelling frame-by-frame. Surprisingly, the fact that the local or segmental SNR strongly varies along the speech signal has not received much attention in the literature and there was only one paper where the subject was addressed. In (H.Kobatake & Y.Matsunoo, 1994) a two-step weighted DTW algorithm and weighting coefficients based on the short-time power or short-time autocorrelation were applied in combination with the Root-Power Sum distance. Although the weighted algorithm was shown to improve the recognition in some cases, the noise cancelling technique introduced an error rate for the clean signal, the weighting procedure was not modelled and the two-step DTW even increased the error rate depending on which weighting coefficient was used.

In this thesis, the problem of weighted matching algorithms (WMA) is studied, one-step weighted DTW and Viterbi (HMM) algorithms are proposed and tested, and the concept of reliability in noise cancelling is presented and used to estimate weighting coefficients. Using the terminology applied in section 2.4, WMA could be classified as a mixture of clean speech estimation and model-based technique. It is shown that weighting the information along the signal can substantially increase the performance of a novel noise cancelling neural net and spectral subtraction (SS). For the noise cancelling neural net the reliability is computed using a mean distortion curve and for SS a novel additive noise model is used to estimate the uncertainty (inverse of reliability) in noise cancelling. In the case of SS (an easily implemented technique) WMA lead to superior results even with a poor estimation for noise, without using any information about the speaker and keeping the restriction of stationarity concerning additive noise imposed by SS, which is weak when compared to other techniques such as CDCN and PMC because the method needs only the noise signal energy estimated in 100 or 200 ms of only-noise samples and can

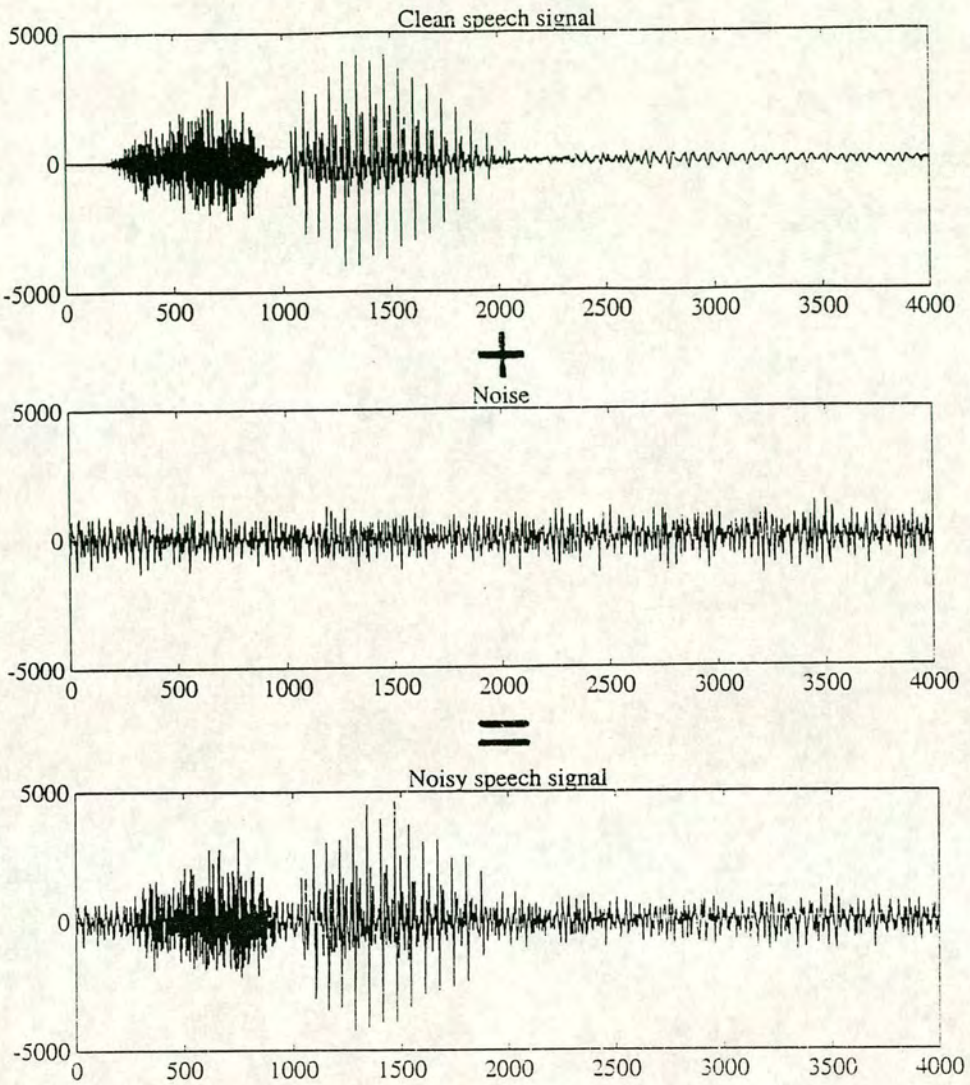


Figure 2.6: Speech signal plus additive noise.

easily capture the dynamics of the corrupting signal due to the fact that the noise energy can be re-estimated in the following non-speech interval. The main idea behind WMA is that the recognition, that is a decision process to choose the reference utterance or HMM with lowest distance or highest likelihood, should rely on those frames with higher energies or local SNR's and variations in the stationarity should not be so important specially at SNR$\geq$0 or 6dB.

The problem of end-point detection is also addressed and a method based on autoregressive analysis of noise is also proposed for robust speech pulse detection. The technique is shown to be effective in increasing the discrimination between the speech signal and background noise

and has not been reported in the literature before.

As far as additive and convolutional noise removal is concerned, in (H.Hermansky *et al.*, 1993) it is said that "results essentially confirm (A.Acero & R.Stern, 1990) which reports negative experience with cascading two systems, one dealing with the additive and the other with the convolutional noise". Actually, in this thesis it is shown that cascading SS and CMN leads to poor results (an average error rate equal to 24% at SNR= 6dB) when the speech signal is corrupted with additive noise and a 6dB/oct spectral tilt, and the recognition is done by means of the ordinary Viterbi (HMM) algorithm. However, results are substantially improved (an average error rate equal to 9% at SNR= 6dB) when the weighted Viterbi algorithm is used instead of the ordinary one (all these results were obtained using temporal constraints in the Viterbi matching; without these constraints the error rates were higher). Consequently, it is also proposed a novel strategy in which the additive noise should be cancelled firstly and convolutional one should be estimated and/or removed using the output of the additive noise cancelling system. Cancelling firstly the additive noise and then the convolutional one seems quite reasonable due to the fact that the former could easily change along time and the latter corresponds to the transmission channel characteristics that are supposed time invariant.

Finally, it is worth noting that reliability weighting could be considered as a formalization of a very important characteristic of the auditory perception which does not have to recover all the information of the corrupted speech signal and reduces the importance of the more noisy intervals to extract the information that is relevant to understand the message.

Chapter 3

LIN and Weighted Matching Algorithms

3.1 Introduction

In this chapter weighted matching algorithms are introduced in the context of a noise reduction neural net LIN (Lateral Inhibition Net). Lateral inhibition is one of the processes responsible for the masking phenomena in different sensory systems, and it serves to sharpen a spatial input pattern by emphasizing its edges and peaks. The conception of the neural net training procedure was inspired by the lateral inhibition process in the sense that it made use of moderately corrupted frames, where the spectral peaks of the clean speech signal should be preserved, as input patterns and clean frames as target patterns. The idea of the weighted matching algorithms was proposed to take into the account the fact that LIN should more easily remove noise from frames with high local SNR rather than low local SNR.

Two weighted matching algorithms have been proposed, one based on the DTW and another on the Viterbi algorithm for HMM, but only the experiments related to the modified DTW are reported in this chapter. The weighted DP algorithm needs only one step, in other words it estimates the optimal alignment path and the global distance (or likelihood) simultaneously and was proved effective in reducing the error rate for white Gaussian additive noise using as a weighting parameter the ratio between the estimated clean signal and the noisy signal energies (N.B.Yoma *et al.*, 1995). However, experiments showed that the improvement that resulted from weighting the information along the signal depended on the initial conditions of the LIN training procedure and, consequently, the weighting coefficient should also take into consideration the

response of the neural net (N.B.Yoma *et al.*, 1996d) (N.B.Yoma *et al.*, 1996a).

LIN in combination with the weighted DP algorithm strongly improved the recognition accuracy for speech signals corrupted by white Gaussian noise, but the high computational load of the backpropagation training algorithm made LIN inappropriate for real environments where it is necessary to capture variations in the stationarity of the corrupting signal. Nevertheless, weighted matching algorithms appeared to be a generic tool that could be used with other noise cancellation techniques. In some way, these algorithms emulate the temporal masking process in the sense that intervals with highest energy tend to mask the perception of those intervals with lowest energy. Weighted matching algorithms give speech recognition another dimension because the information extracted along the speech signal is not equally processed and those segments that provide more reliable information have more influence in the recognition process. This approach represents an important step toward a system that is able to recognize parts of a sentence despite the imprecision introduced by the corrupting signal, given that this imprecision is not uniformly distributed along the speech signal.

3.2 Lateral Inhibition Net (LIN): a Noise Cancellation Neural Net

Masking is basically the suppression of the lowest by the highest spectral components. Lateral inhibition is one of the processes responsible for the masking phenomena in different sensory systems and this concept was used to train the noise reduction neural network, LIN (Lateral Inhibition Net), employed in this research.

Given:

- E_i, the logarithm of the normalised energy at the output of the filter i in a bank of N filters;
- $F_i^c = (E_1^c, E_2^c, E_3^c, ..., E_N^c)$, frame *i* of clean signal;
- $F_i^n = (E_1^n, E_2^n, E_3^n, ..., E_N^n)$, frame *i* after it has noise added;

the lateral inhibition function (LI) can be set as:

$$LI(E_i) = E_i + f(E_1, E_2, E_3, ..., E_N) \tag{3.1}$$

where the function LI() was approximated with multilayer perceptrons with one hidden layer. Multilayer perceptrons were chosen because they can store the information of a large amount of training data, and produce a correct input-output mapping even when the input is slightly different from the examples used to train them (generalization) (S.Haykin, 1994) (D.E.Rumelhart *et al.*, 1987). Figure 3.1 shows the topology employed to approximate the equation (3.1).

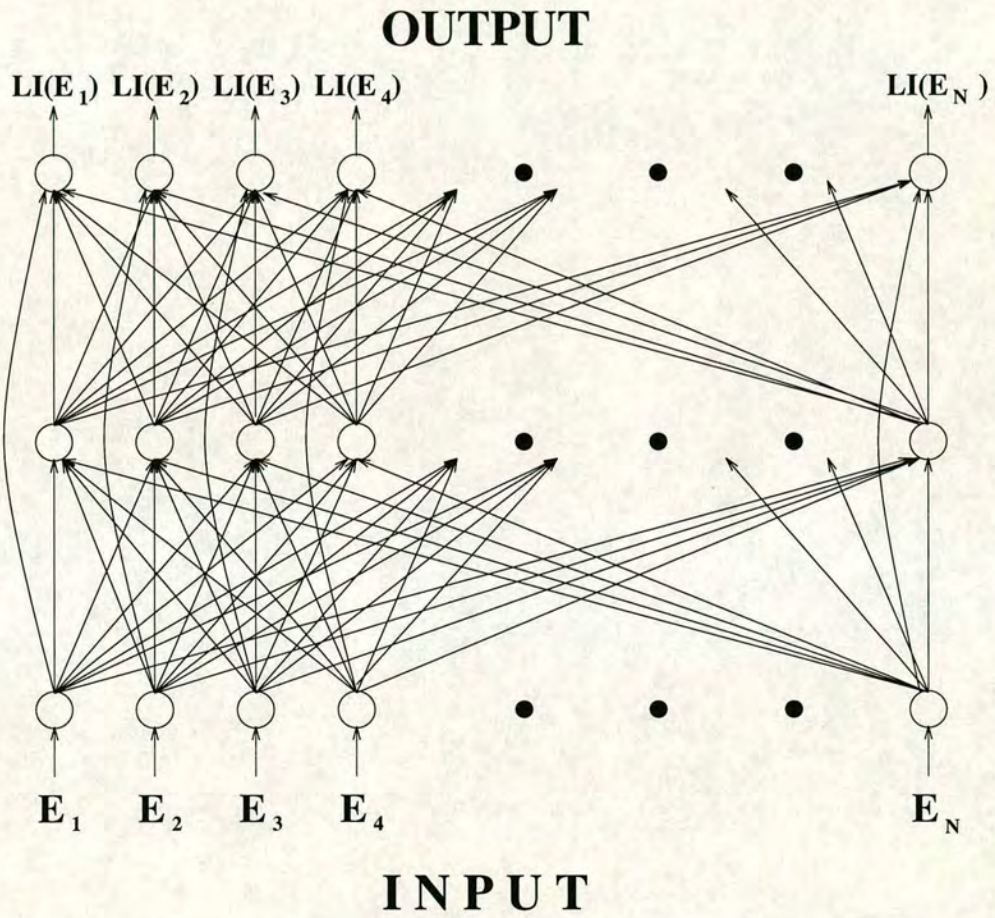


Figure 3.1: Multilayer perceptron to approximate the lateral inhibition function set by equation 3.1.

The output function for the hidden layer nodes was $\sigma(x) = 1/(1 + e^{-x})$ and the output function for input and output layers is linear. Each input node receives the energy of one filter and the same energy is fedforward to the output node in order to compound the equation (3.1). The number of input, hidden and output nodes were equal to the number of filters N.

The LIN was trained with the following conditions that define the lateral inhibition function

(N.B.Yoma *et al.*, 1995) (N.B.Yoma *et al.*, 1996a):

$$LI(F_i^n) \approx LI(F_i^c)$$

$$LI(F_i^c) \approx F_i^c$$

The first condition specifies that F_i^c and F_i^n should give approximately the same result after they are processed by LIN. The second condition settles that LI of a clean signal should give the same clean signal, so that the spectral information is preserved and no distortion should be introduced. All the weights of the neural net (except those on the feedforward connections from the inputs to the outputs which were always equal to 1) were estimated with the classical back-propagation algorithm (D.E.Rumelhart *et al.*, 1987) with cross-validation (S.Haykin, 1994). The training data were made up of input-reference pattern pairs. Initially, the reference patterns were frames of clean signal, F_i^c, and the input patterns were generated adding white Gaussian noise to F_i^c at 4 different SNR's (Clean, 18dB, 12dB, and 6dB). Therefore, each frame F_i^c originated 4 training input-reference pairs. In a modified version of the training algorithm, $LI(F_i^c)$ was used instead of F_i^c as reference patterns.

The training of the neural network was carried out frame by frame and not utterance by utterance, so the LIN should be able to recover the information from a noisy frame independently of the context. Moreover, the SNR training condition (SNR$\geq$6dB) guarantees the highest spectral components presented in the reference are preserved in the input training pattern, and, on the other hand, the generalization feature of neural nets should be able to mask the noises when the SNR is not included among the training conditions or even perhaps when the noise is poorly correlated but not white.

3.2.1 LIN input

In order to normalise the inputs between 0 and 1, firstly the maximum energy of the frame was determined. Then the energy of the other filters was computed in dB using the maximum energy as reference, and all components 50dB below this maximum energy were made equal to -50dB. Finally, the energies in dB were linearly transformed from the range [-50dB, 0db] to [0, 1].

3.2.2 Selection of frames for LIN training

Sounds that present low energy (typically fricatives) are the first to be masked by corrupting signals, and using these speech frames as training patterns could mean learning the neural network with an information that is lost even for moderate SNRs. In (N.B.Yoma *et al.*, 1995) was proposed the use of periodicity as a criterion to select training patterns. Periodicity was defined as:

$$\text{periodicity} = \frac{max[R_x(m)]}{R_x(0)}$$

where $R_x(m)$ is the autocorrelation of the speech signal and was computed with all m's in the range of fundamental periods. The main purpose of this coefficient was to choose voiced frames with high energies but it was observed that some frames, especially at the end of the utterances, presented a high periodicity coefficient and a very low energy. In the results reported in (N.B.Yoma *et al.*, 1996d) (N.B.Yoma *et al.*, 1996a), energy was used as discriminative parameter. Initially the maximum energy of the utterance was computed and then all the frames that were below a given threshold from the maximum energy were discarded. According to some preliminary experiments a suitable threshold would be 25dB.

3.2.3 LIN training algorithm

Initially the quadratic error at the back propagation algorithm was computed between the reference F_i^c and the output $LI(F_i^n)$, which should result in an estimation of the clean frame F_i^c (N.B.Yoma *et al.*, 1995). Given that F_i^{18dB} corresponds to noisy frame with local SNR equal to 18dB, F_i^{12dB} to noisy frame with local SNR equal to 12dB, F_i^{6dB} to noisy frame with local SNR equal to 6dB, figure 3.2 shows a two-dimensional interpretation of the LIN training algorithm. In recognition tests, reference (clean utterances) and testing patterns (noisy utterances) are processed by LIN, and hence in the acoustic pattern matching algorithm the local distances correspond to $d[LI(F_k^c), LI(F_i^n)]$ instead of $d[F_k^c, F_i^n]$, where k denotes a reference frame and i a test one. In the experiments reported here, the distance function d was the squared Euclidean metric.

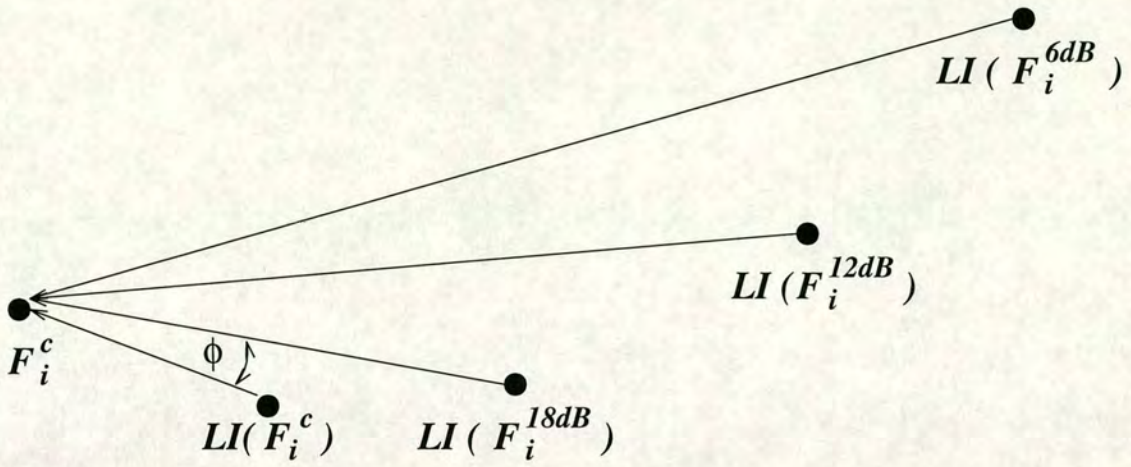


Figure 3.2: Two-dimensional interpretation of LIN training with the ordinary backpropagation algorithm where the reference is constant and equal to the clean frame.

3.3 Reliability in noise reduction

A noise cancelling neural net can be seen as a system that processes a noisy input and produces an output with the influence of noise reduced. Since there are several levels of distortions and the backpropagation training algorithm is essentially stochastic (the most common patterns have more influence in the weights re-estimation process), it is reasonable to suppose that the LIN efficacy depends on the input and each noisy frame could be associated to a reliability coefficient that attempts to measure how reliable is the result of LIN processing. Due to the fact that the noise cancelling depends on $d[LI(F_i^c), LI(F_i^n)]$ (the smaller this distance is, the better is the noise influence cancelling), the reliability coefficient could be related to this distortion by means of the curve shown in figure 3.3 (N.B.Yoma *et al.*, 1996d) (N.B.Yoma *et al.*, 1996a): if $d[LI(F_i^c), LI(F_i^n)]$ is smaller than a threshold δ, reliability will be 1.0; and if $d[LI(F_i^c), LI(F_i^n)] > \delta$, reliability will be inversely proportional to $d[LI(F_i^c), LI(F_i^n)]$. This curve is analytically described by the following function:

$$r = \begin{cases} 1 & \text{if } d[LI(F_i^c), LI(F_i^n)] \leq \delta \\ \frac{\delta}{d[LI(F_i^c), LI(F_i^n)]} & \text{if } d[LI(F_i^c), LI(F_i^n)] > \delta \end{cases} \quad (3.2)$$

It is interesting to highlight that LIN tends to preserve the highest energies and the position of local spectral peaks (see figure 3.4), or in other words, tends to preserve the phonetic information

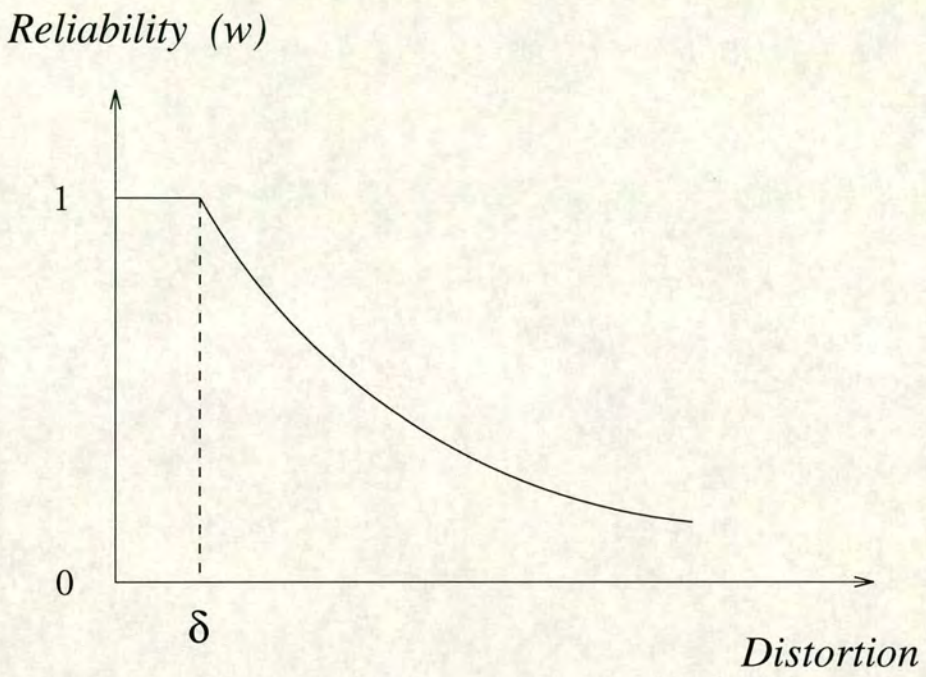


Figure 3.3: Reliability coefficient vs distortion.

of the frame. For this reason, if $d[LI(F_i^c), LI(F_i^n)]$ was low for any SNR, the recognition error would be also low independently of the noise level.

At the recognition procedure, the clean version F_i^c of the noisy testing frame F_i^n is not available but, due to the fact that the power spectral distribution of the corrupting signal is known (white Gaussian noise), F_i^c can be set as a function of F_i^n and the local SNR. After LIN has been trained, the training data-base could be used to approximate the relation between $d[LI(F_i^c), LI(F_i^n)]$, and F_i^n and the local SNR. Consequently, if the segmental SNR can be computed frame by frame and given that F_i^n is available, the reliability coefficient can be estimated frame by frame at the recognition process.

3.3.1 Local SNR estimation

If the noise is poorly correlated and uncorrelated with the speech signal, it is possible to estimate the power of the clean speech from the autocorrelation function of the noisy signal (N.B.Yoma *et al.*, 1995). Given that $R_x(m)$, $R_s(m)$ and $R_n(m)$ are the autocorrelation functions of the noisy speech, the clean speech and the noise signals, respectively, the following coefficient can be

computed frame by frame:

$$\eta = \frac{R_s(0)}{R_x(0)} = \frac{R_s(0)}{R_n(0) + R_s(0)} \tag{3.3}$$

$$\eta = \begin{cases} 1 & \text{if SNR} = \infty \\ 0 & \text{if SNR} = -\infty \end{cases}$$

The power of the speech signal $R_s(0)$ was estimated from $R_x(m)$ for $m \neq 0$, because $R_x(m) \cong R_s(m)$ if $m \neq 0$, applying some properties of the autocorrelation function: firstly, if the process is real, the autocorrelation function is even, that is, $R_s(m) = R_s(-m)$; secondly, the autocorrelation of any process is maximum at the origin (necessary condition) (A.Papoulis, 1991). In the results shown in this paper $R_s(0)$ was estimated from $R_x(1) = R_x(-1)$ and $R_x(2) = R_x(-2)$ by means of quadratic interpolation (N.B.Yoma *et al.*, 1995):

$$R_s(0) = \frac{4 \times R_x(1) - R_x(2)}{3} \tag{3.4}$$

The coefficient η can be computed frame by frame because it needs just the autocorrelation of the noisy signal at points m=0, 1 and 2. Observe that the estimation of the noise power in silence intervals is not needed and the method captures the dynamic of the speech and noise signals energy. Given that:

$$\text{SNR} = 10 \cdot \log\left(\frac{R_s(0)}{R_n(0)}\right)$$

the segmental SNR and the coefficient η are related by the following equation:

$$\eta = \frac{10^{\text{SNR}/10}}{1 + 10^{\text{SNR}/10}} \tag{3.5}$$

The more correlated is the speech signal, the more accurate is the local SNR estimation. If the speech signal is poorly correlated, the method loses accuracy.

3.3.2 Mean distortions

As an approximation, it can be assumed that the distortion $d[LI(F_i^c), LI(F_i^n)]$ depends exclusively on the local SNR. The mean-distortions for each SNR may be estimated at the LIN training procedure and, since the local SNR can be efficiently computed for correlated speech signals

as described in the previous section, $d[LI(F_i^c), LI(F_i^n)]$ could be estimated at the recognition process. Given:

- D_i^{snr}, the distortion $d[LI(F_i^c), LI(F_i^n)]$ for the frame F_i^n with local SNR equal to snr;
- $\overline{D_{snr}}$, the mean-distortion at local SNR equal to snr,

$\overline{D_{snr}}$ can be computed for some SNR's at the LIN training procedure and, by means of linear interpolation, it can be estimated for other values of SNR. Figure 3.5 shows the curve $\overline{D_{snr}}$ vs. SNR estimated with a LIN that was trained with the female speaker from the Noisex database. The limitation of this method concerns the fact that $d[LI(F_i^c), LI(F_i^n)]$ depends on F_i^n and not only on the segmental SNR.

For the results presented in this paper, $\overline{D_{snr}}$ was computed for SNR=18, 12, 6, 3 and 0dB by employing the LIN training database, after LIN had been trained. During the recognition procedure, the coefficient n was estimated by means of the autocorrelation function (3.3)(3.4) and the curve $\overline{D_{snr}} \times localSNR$ was mapped into the n domain by using the equation (3.5). The constant δ was made equal to 0.004, a value that was shown to be suitable according to some tests.

3.4 Modified Backpropagation Algorithm

In the ordinary neural net training algorithm, the quadratic error is computed between the reference F_i^c and the output $LI(F_i^n)$. However, the efficacy of LIN is related to the distortion $d[LI(F_i^c), LI(F_i^n)]$: the smaller $d[LI(F_i^c), LI(F_i^n)]$ is, the smaller should be the recognition error rate. As a consequence, it can be interesting to include the condition of minimization of $d[LI(F_i^c), LI(F_i^n)]$ in the training algorithm in a more explicit way. Figure 3.2 shows the ordinary backpropagation approach, where the target is the minimisation of the distances $d[F_i^c, LI(F_i^n)]$ instead of $d[LI(F_i^c), LI(F_i^n)]$. The minimisation of $d[F_i^c, LI(F_i^n)]$ leads to the reduction of $d[LI(F_i^c), LI(F_i^n)]$, but this distance also depends on the angle between $LI(F_i^c) - F_i^c$ and $LI(F_i^n) - F_i^c$ (see figure 3.2). In the modified algorithm, the clean signal F_i^c was replaced with $LI(F_i^c)$ as the reference for the noisy frames, and the quadratic error was computed between the reference $LI(F_i^c)$ and the output $LI(F_i^n)$.

In the ordinary LIN training algorithm (BLT-Backpropagation LIN Training), in each epoch the backpropagation minimizes the quadratic error of the following sequence of pairs reference-output:

1. F_i^c and $LI(F_i^c)$;

2. F_i^c and $LI(F_i^n)$, for all the local SNR's included in the training database.

In the modified training algorithm (MLT-Modified LIN Training), in each epoch the backpropagation minimizes the quadratic error of the following sequence of pairs reference-output:

1. F_i^c and $LI(F_i^c)$;

2. $LI(F_i^c)$ and $LI(F_i^n)$, for all the local SNR's included in the training database,

which is more coherent with the conditions that define the lateral inhibition function (see section 3.2). Figure 3.6 shows the two-dimensional interpretation of the MLT algorithm. It is interesting to highlight that the reference is not constant as in the ordinary backpropagation algorithm, but it is modified iteration by iteration because $LI(F_i^c)$ depends on LIN, and LIN's weights are re-estimated each time that a reference-output pair is presented to the training algorithm.

3.5 Weighted Matching Algorithms

Some modifications were included in matching algorithms in order to weight the reliability of the information extracted from testing frames. A weighting coefficient $w(t)$ ($w(t) = 1$, maximum reliability; $w(t) = 0$, minimum reliability) is associated to each testing frame in order to be employed in the modified vesions of the DTW and Viterbi (HMM) algorithms. In this chapter w was made equal to the coefficient n, related to the segmental SNR estimation (3.3), and to r `reliability` (section 3.3.2), in LIN processing. The main idea behind the modifications made on Viterbi (discussed in Chapter 6)) and DTW algorithms is that the influence of a frame on decisions must be proportional to its coefficient w. The proposed weighted DP algorithm was compared with the two-step DP algorithm proposed in (H.Kobatake & Y.Matsunoo, 1994).

3.5.1 DTW : modified DP equation

The same principle of weighting the importance of a frame according to $w(i)$ leads to a modified Dynamic Programming (DP) equation. For the local conditions shown in Fig. 2.2 in page 9, the proposed DP equation is given as follows :

$$G(i,j) = \min \left(\begin{array}{c} \frac{G(i,j-1)\times W(i,j-1)+d(i,j)\times w(i)}{W(i,j-1)+w(i)} \\ \\ \frac{G(i-1,j-1)\times W(i-1,j-1)+2\times d(i,j)\times w(i)}{W(i-1,j-1)+2\times w(i)} \\ \\ \frac{G(i-1,j)\times W(i-1,j)+d(i,j)\times w(i)}{W(i-1,j)+w(i)} \end{array} \right)$$

and

$$W(i,j) = \begin{cases} W(i,j-1) + w(i) \\ W(i-1,j-1) + 2 \times w(i) \\ W(i-1,j) + w(i) \end{cases}$$

This DP equation takes into account the weight $w(i)$ frame by frame, and the calculation of the overall distance, $G(i,j)$, is affected by $d(i,j)$ according to $w(i)$: if $w(i) = 1$ (high reliability or local SNR), the weight of $d(i,j)$ is maximum; if $w(i) = 0$ (very low reliability or local SNR), the importance of $d(i,j)$ is zero.

3.5.2 Two-step DP matching

This algorithm (H.Kobatake & Y.Matsunoo, 1994) consists of the following two-step processing. Firstly, the optimal alignment path $c_k = (i_k, j_k)$, $k = 1, 2, ..., K$ is obtained using the ordinary DP matching algorithm with symmetric weight, where i_k and j_k are the frame numbers of the testing and reference patterns respectively. The second step is the calculation of the global distance between the utterances weighted by $w(i_k)$ along the optimal path obtained at the first step.

3.6 Experiments

3.6.1 Database

The proposed methods were tested with speaker-dependent isolated words (English digits from 0 to 9) recognition experiments. The tests were carried out employing the two speakers (one female and one male) from the Noisex database. The isolated clean words were automatically end detected and generated the database used in this research. For each speaker, the 100 training clean utterances (10 repetitions per digit) generated 10 reference sets (set of repetition 1 of each word, set of repetition 2 of each word, etc). The 100 testing clean utterances were used to create the noisy database by adding white noise at 5 global-SNR levels: clean speech, +18dB, +12dB, +6dB, +3dB and 0dB. The global-SNR was defined as in (Ghitza, 1987). Firstly, the total energy E of the clean word was computed. Then, the mean energy per sample E_t was determined dividing E by the number of samples of the signal. Finally, E_t was used to set the variance of the zero mean white Gaussian noise to be added.

3.6.2 Pre-processing

Before the Gaussian noise was added, the speech signals were low pass filtered, using a 10th order Tchebychev filter with cut off frequency equal to 3700 Hz, and down sampled from 16000 to 8000 samples/sec. The band from 300 to 3400 Hz was covered with 14 Mel 2nd order IIR digital filters. The energy of each filter was an input of LIN as explained in section 3.2.1. After LIN processing 10 cepstral coefficients were computed.

3.6.3 Training the neural network

For each speaker, the frames from the set of repetition 1 of the training database generated the input-reference pattern pairs used by the LIN training algorithm to estimate the weights. The frames from the set of repetition 2 of the training database generated the input-reference pattern pairs used for evaluation of the performance of LIN. Several training conditions (learning rate and initial weights) were tested and the one that gave the best recognition results on the test data was chosen. For each speaker, the LIN training variables were kept constant in order to compare the MLT and BLT algorithms at the same conditions.

3.6.4 Results

The results presented in this chapter were achieved with 1000 recognition tests for each SNR: 10 reference sets $\times$ 100 testing utterances. The following configurations were tested: the ordinary DTW algorithm with cepstral coefficient without (DP-C) and with (DP-L) LIN processing; the proposed weighted DP algorithm with LIN processing, (DPW-$\overline{D}$) with the mean-distortions method for reliability estimation and (DPW-n) with local SNR estimation; and finally, the Two-step DP matching with LIN, (DP2-$\overline{D}$) with the mean-distortions method for reliability estimation and (DP2-n) with local SNR estimation. Table 3.1 shows the number of iterations required by each algorithm. The recognition error rates are presented in Tables 3.2 and 3.3 for the female speaker, and in Tables 3.4 and 3.5 for the male one.

3.7 Discussion

3.7.1 LIN efficacy in noise cancelling

LIN showed a substantial reduction in error rates even without reliability weighting. LIN with the ordinary DTW algorithm ($DP - L$) practically eliminated the influence of the noise at SNR=18 and 12 dB, and resulted in a mean reduction of 87, 70 and 48% at SNR=6, 3 and 0dB, respectively. Moreover, the error introduced for testing clean signal was almost zero.

3.7.2 Comparison between weighting coefficients

As can be seen in Tables 3.2 and 3.3 (female speaker) and Tables 3.4 and 3.5 (male speaker), the reliability coefficient estimated with the mean-distortions method gave a greater reduction in the error rate than the SNR weighting in all noisy conditions. When LIN was trained by means of the MLT algorithm, the reduction due to reliability weighting was as high as 100, 84 and 57% at SNR=12, 6 and 3dB, respectively, while the SNR estimation resulted in a much smaller reduction in most of the cases and even in an increase in the error rate in other cases.

The proposed one-step weighted algorithm showed almost the same performance as the two-step one with the reliability coefficient, but resulted in a poorer improvement when the SNR estimation was used as a weighted parameter. This must be due to the fact that in the one-step algorithm the influence of a frame on decisions must be proportional to its coefficient w, and the

Table 3.1: Number of iterations needed to train LIN.

Speaker	*Female*	*Male*
BLT	6132	7403
MLT	3869	1702

Table 3.2: Recognition error rate (%) for the female speaker. LIN was trained with the BLT algorithms

SNR	*Cln*	*18dB*	*12dB*	*6dB*	*3dB*	*0dB*
DP-C	0.1	3.5	31.9	67.0	70.6	75.6
DP-L	0.1	0.1	1.2	11.5	31.9	53.5
DPW-$\overline{\text{D}}$	0.2	0.1	0.1	4.0	17.2	33.3
DPW-π	0.1	0.4	1.0	6.4	26.0	43.9
DP2-$\overline{\text{D}}$	0.1	0.0	0.1	3.5	15.9	32.1
DP2-π	0.1	0.2	0.4	4.0	20.6	38.2

reliability coefficient includes not only the information concerning the segmental SNR, but also the LIN characteristic in the form of the mean-distortion curve (figure 3.5), and provides a more accurate estimation about the reliability of the information extracted from each frame.

3.7.3 Comparison beween MLT and BLT algorithms

According to Tables 3.2-3.5, the reliability coefficient as a weighting parameter gave the best results, with the MLT algorithm for the female speaker and with the BLT algorithm for the male one. The error rate was kept below 1.5% at SNR=6dB and below 10% at SNR=3dB for both speakers.

Some preliminary experiments showed that the best results were achieved with the combination of MLT and reliability coefficient weighting. This could be the result of firstly the weakening of the learning constraints imposed by MLT, and secondly the better matching between these constraints and the estimation of $d[LI(F_i^c), LI(F_i^n)]$ required by the reliability coefficient computation. In the MLT algorithm, the approximation between $LI(F_i^c)$ and $LI(F_i^n)$ (figure 3.6) seemed to be more natural than the approximation between F_i^c and $LI(F_i^n)$ in BLT (figure 3.2).

However, further tests showed that the BLT algorithm could lead, depending on LIN training conditions, to better results than the MLT one (male speaker).

Table 3.3: Recognition error rate (%) for the female speaker. LIN was trained with the MLT algorithm

SNR	*Cln*	*18dB*	*12dB*	*6dB*	*3dB*	*0dB*
DP-C	0.1	3.5	31.9	67.0	70.6	75.6
DP-L	0.1	0.2	0.6	5.9	10.6	24.5
DPW-$\overline{\text{D}}$	0.1	0.0	0.0	0.7	6.1	17.9
DPW-η	0.1	0.1	0.3	3.0	9.5	30.1
DP2-$\overline{\text{D}}$	0.1	0.0	0.0	0.5	5.6	17.6
DP2-η	0.1	0.1	0.1	2.3	6.5	23.6

Table 3.4: Recognition error rate (%) for the male speaker. LIN was trained with the BLT algorithm

SNR	*Cln*	*18dB*	*12dB*	*6dB*	*3dB*	*0dB*
DP-C	0.0	16.8	49.9	65.1	69.4	74.6
DP-L	0.3	0.4	1.6	9.8	21.9	41.8
DPW-$\overline{\text{D}}$	0.1	0.1	0.1	1.3	6.3	24.6
DPW-η	0.5	0.5	3.4	10.4	20.6	43.4
DP2-$\overline{\text{D}}$	0.1	0.1	0.2	1.2	6.6	25.2
DP2-η	0.3	0.1	1.7	8.2	17.2	38.9

3.8 Conclusions

The combination of LIN and weighted DP algorithms proved to be effective in reducing the influence of white Gaussian noise, and the error introduced for testing clean signal was almost zero. The reliability coefficient gave better results than the SNR estimation as a weighting parameter and this must arise from the fact that this coefficient takes into account not only the local SNR estimation but also the characteristic response of LIN in the form of the mean-distortion curve (figure 3.5). The weighted DP algorithms helped to reduce the error rate, but its improvement decreased when the SNR became more severe. The proposed one-step DP matching was also shown to be effective in reducing the error rate, and led to approximately the same error rates as the two-step matching (H.Kobatake & Y.Matsunoo, 1994) when the reliability weighting was used.

Weighting the information along the speech signal seems quite obvious when defined in the context of LIN, which was trained with moderately corrupted frames and assuming that the

Table 3.5: Recognition error rate (%) for the male speaker. LIN was trained with the MLT algorithm

SNR	*Cln*	*18dB*	*12dB*	*6dB*	*3dB*	*0dB*
DP-C	0.0	16.8	49,9	65.1	69.4	74.6
DP-L	0.0	0.6	2.7	9.2	22.6	38.2
DPW-$\overline{\text{D}}$	0.1	0.0	0.0	2.2	7.8	24.1
DPW-n	0.5	0.8	3.3	11.5	22.0	36.1
DP2-$\overline{\text{D}}$	0.1	0.0	0.0	2.3	7.8	24.8
DP2-n	0.3	0.0	0.9	8.5	17.7	31.6

speech signal peaks were reasonably preserved after the noise being added. In other words, weighting matching algorithms tried to take into account the fact that frames with low SNR should not be reliably processed by LIN and should have a low weight in the recognition procedure. Nevertheless, the reliability coefficient as a weighting parameter seems to be a generic approach and could also be employed with other noise cancelling techniques. In Chapter 5, spectral subtraction (SS) is analysed under the perspective of reliability in noise cancelling and tested in combination with the weighted DP algorithm. SS is an easily implemented technique and is able to capture quite well the dynamic of the corrupting signal. Before continuing the study on reliability based weighting, the problem of speech pulse detection is addressed in the next chapter where the autoregressive analysis of noise is proposed to improve the discrimination between speech and noise when the corrupting signal is highly correlated, a condition easily found in practical applications.

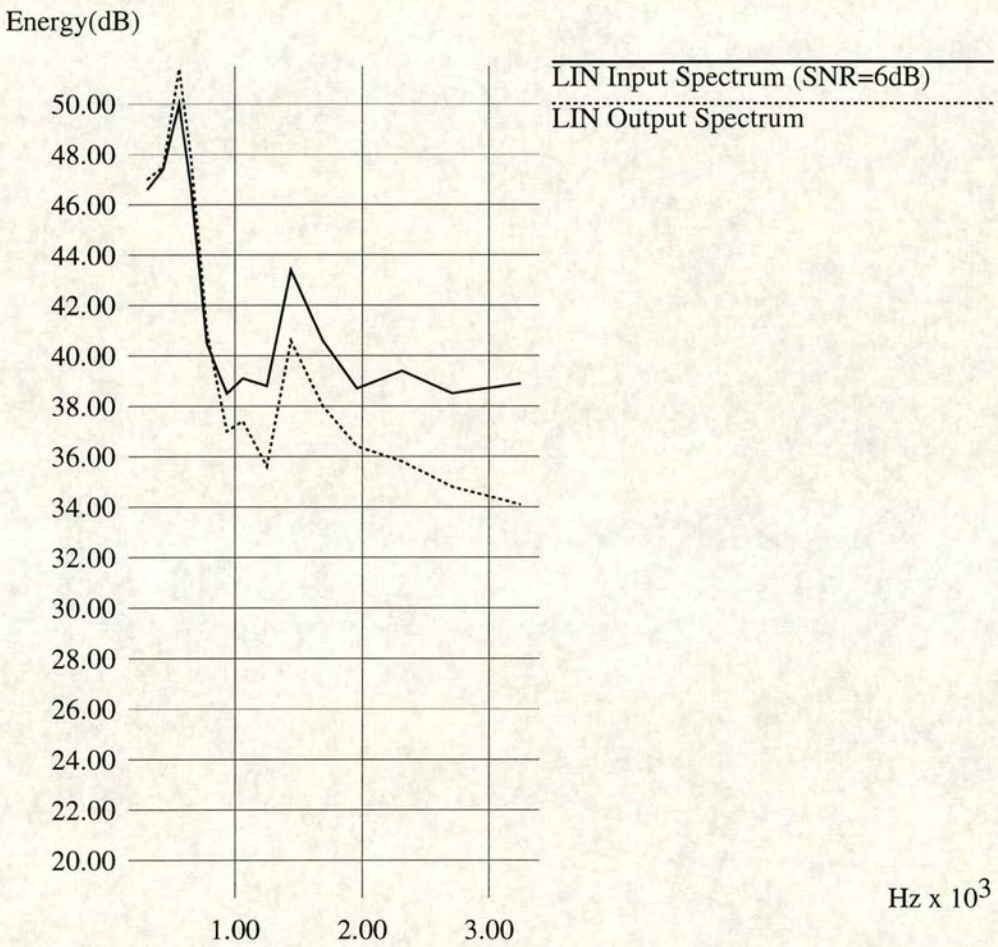


Figure 3.4: Noisy frame with local SNR equal to 6dB before and after LIN processing. The frame corresponds to the vowel. Observe that the highest component tends to be preserved and the position of the second formant does not change.

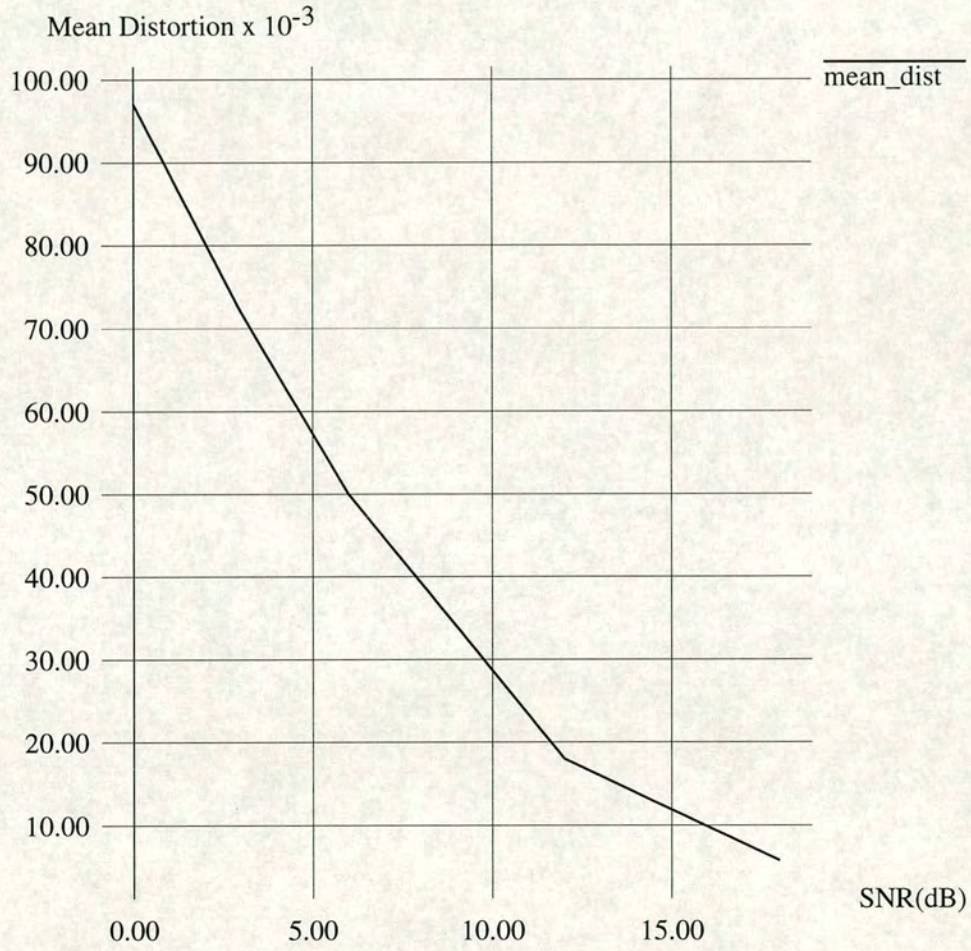


Figure 3.5: Mean Distance vs SNR for the female speaker.

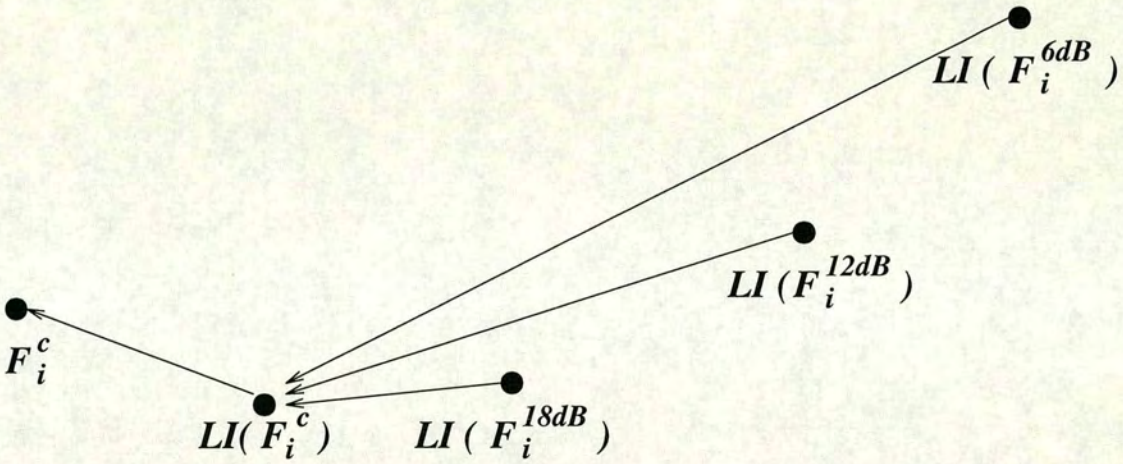


Figure 3.6: Two-dimensional interpretation of the modified LIN training algortihm (MLT).

Chapter 4

Robust speech pulse detection using autoregressive analysis of noise

4.1 Introduction

The inaccurate detection of the endpoints is a major cause of errors in automatic speech recognition systems. Most of the endpoint detecting techniques are based on energy levels, pitch, zero- and/or level-crossing rates, and timing (B.Mak *et al.*, 1992). However, in many real environments the speech signal is corrupted by additive and coloured (correlated) noise and these parameters may be insufficient for the correct detection of a speech pulse if the signal to noise ratio (SNR) is low. Firstly, the short-term energies resulting from fluctuations in the noise can be even as high as the ones presented by the speech pulses. Secondly, if the noise is correlated and mainly concentrated in low frequencies (below 1000 Hz), where the speech signal has also most of its energy, pitch and zero- or level-crossing rates cannot discriminate between speech and silence intervals. Finally, the increase of the noise level tends to reduce the speech pulses' width and the efficacy of timing constraints. In order to make speech recognition experiments more realistic, an end-point detector based on AR analysis of noise is proposed. AR analysis assumes that the additive noise is reasonably stationary and strongly simplifies the complexity of the speech signal detector.

In this chapter adaptive autoregressive modelling of noise is proposed in order to reduce the influence of the corrupting signal and two forms of frame comparison for speech pulse detection are studied: spectral density comparison between noise and noisy speech signals; and non-stationarity measure. Finally, an end-point detector is proposed and tested on isolated digits

corrupted by the car and speech noises which were addressed in this chapter.

The FIR filters employed in the autoregressive analysis are trained with the LMS algorithm during non-speech intervals. The spectral density comparison is made between noisy speech frames and an estimation of noise also in non-speech intervals. In contrast, non-stationarity measures are based on spectral distances between contiguous frames and do not require noise estimation. Preliminary experiments have shown that the AR analysis generally increases the discrimination between speech and noise and that spectral density comparison and non-stationarity measures might be more effective than energy to indicate the presence of a speech pulse at low SNR's.

4.2 AR analysis of the noise signal

It is assumed that the noise $n(i)$ could be described by an AR process of order M, or in another words, it would satisfy the following equation (S.Haykin, 1991):

$$H_A(z)N(z) = W(z) \tag{4.1}$$

where $N(z)$ and $W(z)$ are the z-transform of the noise and a white-noise process, respectively, and $H_A(z)$ is defined as:

$$H_A(z) = 1 + \sum_{k=1}^{M} a_k z^{-k} \tag{4.2}$$

If the noise is reasonably stationary, its autoregressive filter $H_A(Z)$ estimated in non-speech intervals may be used to increase the energy gap between the noise and the noisy speech signals (N.B.Yoma *et al.*, 1996b) (N.B.Yoma *et al.*, 1996c). Since the speech signal is intrinsically non-stationary and has components in all the considered band (250-3200 Hz), its spectral density and that of the noise are likely to differ along time, even if the noise is correlated and mainly concentrated in low frequencies (below 1000 Hz). Consequently, it is expected that the attenuation caused by $H_A(Z)$ will be lower on average for the speech than for the corrupting signal. The filter $H_A(Z)$ is transversal or FIR and its coefficients can be estimated by means of the classical LMS algorithm. If the coefficients a_i are replaced with c_i, where $c_i = -a_i$, the tap weights adaptation is given by:

$$c_k(i+1) = c_k(i) + \eta n(i-k)e(i) \tag{4.3}$$

where η is the learning rate and $e(i)$ corresponds to the prediction error:

$$e(i) = n(i) - \sum_{k=1}^{M} c_k n(i-k) \tag{4.4}$$

4.3 Spectral Density Comparison

If the noise is assumed to be reasonably stationary, the noise spectral density could be considered constant between two consecutive silence periods and could be useful to detect speech pulses. In the results presented in this chapter, the spectral estimation was made with a 14 channel Mel-filter bank, the same used in recognition experiments, but neither logarithmic compression nor normalization was applied. The Spectral Density Comparison coefficient ($SD(i)$) for a frame i is defined, in the Euclidean metric context, as:

$$SD(i) = 20 \times \log\left(\frac{\sqrt{\sum_{k=1}^{14}(E_k^n - E_{i,k})^2}}{\sqrt{\sum_{k=1}^{14}(E_k^n)^2}}\right) \tag{4.5}$$

where $S_i = (E_{i,1}, E_{i,2}, E_{i,3}, ..., E_{i,14})$ and $S^n = (E_1^n, E_2^n, E_3^n, ..., E_{14}^n)$ correspond, respectively, to the spectral estimation of frame i and that of the noise; E_k^n and $E_{i,k}$ represent the filter k output energies.

4.4 Stationarity Coefficient

If the noise is reasonably stationary its statistical properties are constant or change slowly, or might even present fast but small variations along time. In order to employ these features of the corrupting signal in speech pulse detection, the non-stationarity coefficient ($NST(i)$) for a frame i is defined, in the Euclidean metric context, as:

$$NST(i) = 20 \times \log\left(\frac{\sqrt{\sum_{k=1}^{14}(E_{i,k} - E_{i-1,k})^2}}{\sqrt{\sum_{k=1}^{14}(E_k^n)^2}}\right) \tag{4.6}$$

where $S_{i-1} = (E_{i-1,1}, E_{i-1,2}, E_{i-1,3}, ..., E_{i-1,14})$ and $S_i = (E_{i,1}, E_{i,2}, E_{i,3}, ..., E_{i,14})$ correspond, respectively, to the spectral estimations of two contiguous frames.

4.5 End-point detection

In order to automatically detect isolated utterances (digits) in the presence of background noise, an end-point detector was developed using the AR analysis of the corrupting signal. The idea was not to develop a very accurate end-point detector but to detect speech signal in the presence of background noise in order to make recognition tests more realistic. Actually, it was noticed that a high accuracy in the detection of the speech signal is out of consideration at moderate and low SNR's, and that the problem is more basic and concerns how to detect at least part of the utterance in noisy conditions.

The end-point detector employs energy, spectral estimation and temporal constraints (L.Lamel *et al.*, 1981). Neither zero or level crossings (M.H.Savoji, 1989) nor pitch (B.Mak *et al.*, 1992) are used. Both energy and spectral estimation are computed after AR analysis of noise. Temporal constraints concern rise, fall and duration of a speech pulse and separation between two speech pulses in an utterance. It is supposed that every utterance was composed by one or two speech pulses, although a more generic case would be easily included, and that the separation between two consecutive speech pulses in an utterance could not exceed a given threshold (e.g. 500 ms). The assumption about the number of speech pulses per word relied on the facts that the end-point detector ran in an isolated digits task (Noisex-92 database (A.Varga *et al.*, 1992)) and that digits are reasonably short utterances. Given that $s(l)$, $n(l)$ and $x(l)$ are the clean speech, the noise and the resulting noisy signal, respectively, the additive condition in the temporal domain may be set as:

$$x(l) = s(l) + n(l) \tag{4.7}$$

where $x(l)$, $s(l)$ and $n(l)$ are noisy speech, clean speech and noise signals, respectively. The mean frame energy at frame i, mfe_i, is defined as

$$mfe_i = \sum_{l=1}^{L} x(l)^2 \tag{4.8}$$

where L is the number of samples per frame. The frame power in dB at frame i, fp_i, is given by

$$fp_i = 10 \cdot \log(mfe_i) - np_e \tag{4.9}$$

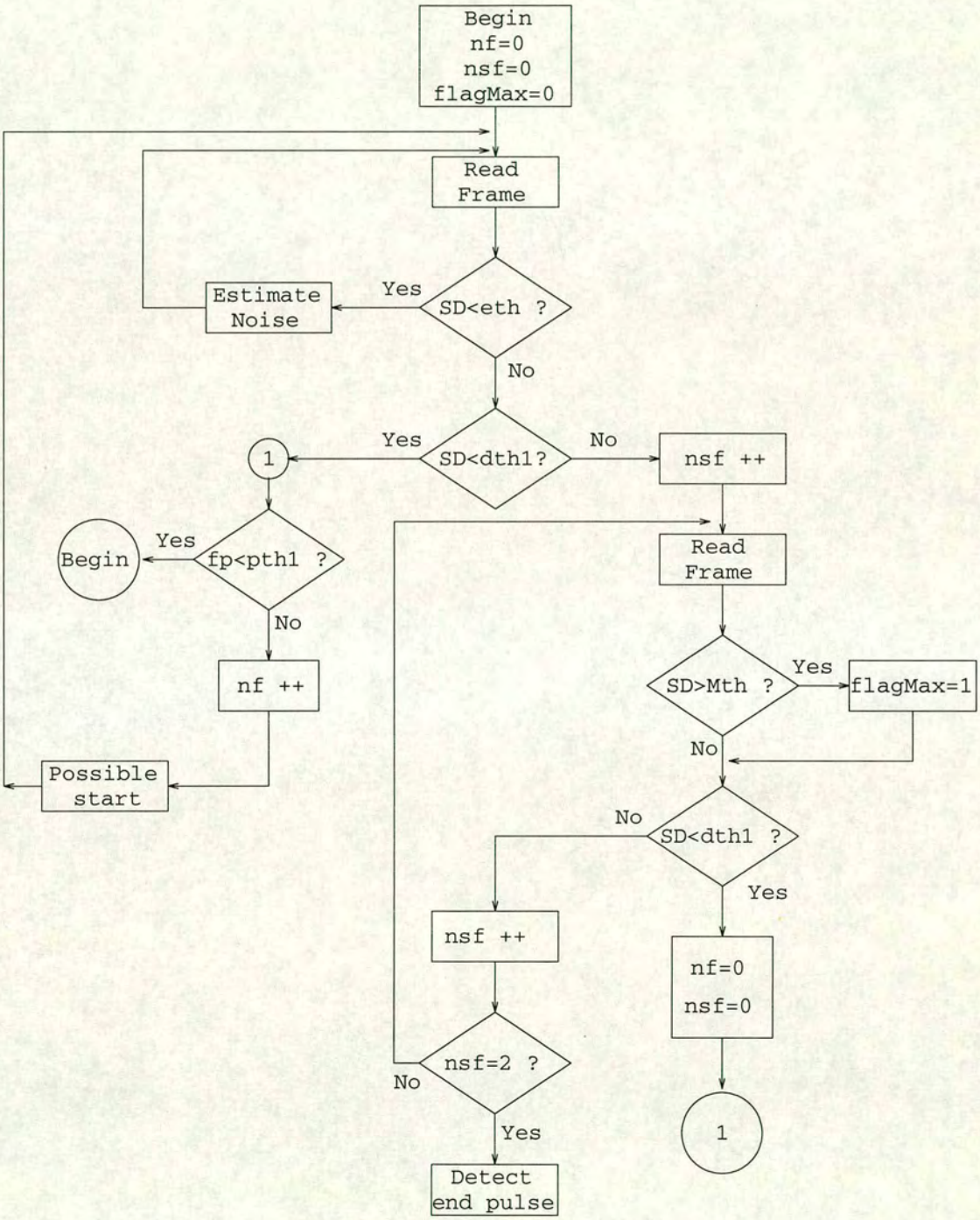


Figure 4.1: Algorithm for detection of speech pulse start.

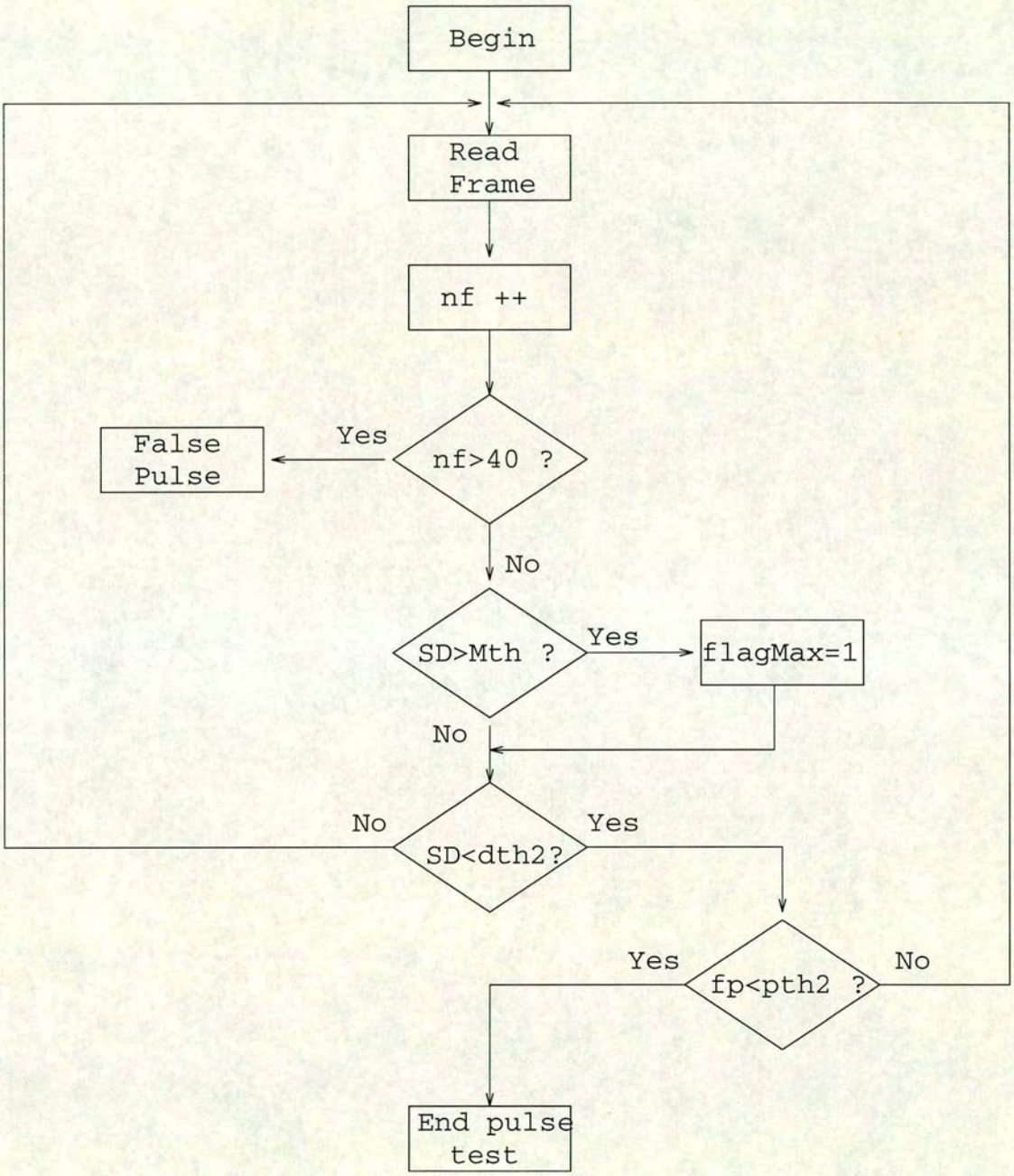


Figure 4.2: Algorithm for detection of speech pulse end.

where np_e is the noise power estimation that is initially computed in a short non-speech interval (eg 200 ms) by means of

$$np_e = \frac{1}{I}\sum_{i=1}^{I} 10 \cdot \log(mfe_i) \tag{4.10}$$

where I is the number of non-speech frames used in a first estimation of np_e.

The start of a speech pulse is detected using the algorithm shown in Fig.4.1. Frames are read every 25 ms without overlapping and SD is computed according to (4.5). In order to capture the dynamics of the corrupting signal, if SD is lower than a threshold for estimation eth the noise power $\overline{np_e}$ and spectral density S_e^n are re-estimated according to:

$$np_e = \alpha \cdot 10 \cdot \log(fp_i) + (1-\alpha) \cdot np_{e-1} \tag{4.11}$$

and

$$S_e^n = \alpha \cdot S_i^n + (1-\alpha) \cdot S_{e-1} \tag{4.12}$$

where e and $e-1$ denote the current and previous estimation respectively, α is a constant between 0 and 1 and S_i^n is the spectral estimation at the frame i composed by only-noise samples. As a result, the current noise power and spectral estimation are a weighted arithmetic mean between the previous estimation and the noise power and spectral vector at the current frame i, if the frame i is classified as a non-speech one ($SD < eth$). The threshold eth should not be high to avoid speech frames being used to re-estimate the noise. The scheme could capture slow variations in the noise stationarity and should make the noise estimation more reliable.

Returning to the algorithm shown in Fig.(4.1) to detect the start of a speech pulse, if SD is lower than a detection threshold dth1, it is considered that a speech pulse has not probably started, but in order to make this decision more reliable the frame power envelope fp is compared to a power threshold pth1: if fp is higher than pth1, the frame is stored as a possible start of a speech signal. In order to confirm this start, SD needs to be higher than dth1 for at least 2 frames, and if it does not in the next frame the first of the frames showing $fp > pth1$ is discarded. Simultaneously, SD is compared with a peak threshold Mth as part of the constraints imposed to a speech pulse: SD needs to be higher than Mth for at least one frame. If $SD > Mth$,

flagMax is set to 1 and this condition will be tested after an end of the possible speech pulse has been detected by means of the algorithm shown in Fig.(4.2).

A speech pulse is considered ended if SD is lower than dth2 and fp lower than pth2 for at least one frame. If these conditions do not occur in less than 1 sec (maximum duration allowed for an utterance) the speech is taken as invalid. After the end of the speech signal being detected, two constraints are applied in order to validate a speech pulse. Firstly, the vocal tract coarticulation speed requires a speech pulse to be longer than a given threshold (e.g. 75 or 100ms); in other words, peaks shorter than this threshold are considered variation in the stationarity of the corrupting signal. Secondly, SD needs to be higher than Mth at least once along the speech signal; this condition is checked by means of flagMax = 1.

The complete end-point algorithm is composed of two speech pulse detectors assuring that the maximum separation between two contiguous pulses is not higher than a threshold (0.5 sec). If this threshold is not respected the utterance is considered composed by only one pulse. This end-point detector was tested on isolated digits of the Noisex database using car and speech noises. The algorithm is quite simple when compared with other ones found in the literature and the use of AR analysis on noise made it work reasonably well in noisy conditions at SNR=18, 12, 6 and 0dB. It was observed that a tuning of the thresholds (specially for dth1 and dth2) may be needed at the most severe conditions, although an empirical approach was enough for the purposes of this research. Initially, a suitable value for dth1 and dth2 (dth1 and dth2 were made equal in these experiments) was estimated at SNR=18, 12 and 6 dB. However, this configuration allowed one utterance out of 100 to be missed at SNR=0dB and another value for dth2 and dth1 had to be found in order to detect all the utterances at this SNR. It was observed that without AR analysis of noise, the parameter tuning problem was much more acute and was even necessary at SNR=12 and 6dB. Moreover, without AR analysis the fluctuations of the corrupting noise are much higher and a more complex algorithm is required in order to discern the noisy speech signal from the background noise (see Figs. 4.4-4.6).

A general effect of additive noise on the end-point detection procedure is to shorten the utterance due to the fact that the corrupting signal tends to completely cover or hide those intervals showing the lowest energies. AR analysis considerably improved the discrimination between the corrupting and speech signals but in many cases did not avoid the shortening effect of the

detected utterance. This problem could be minimised by adding at the begining and end of the detected utterance a few frames under the assumption that these extra frames would contain speech signal hidden by the background noise. However, these extra frames present a low local or segmental SNR and, in the context of weighted matching algorithms, have a low weight in the recognition process. In the experiments shown in Chapter 5 with the weighted Dynamic Programming (DP) algorithm the shortening effect of the detected utteraces was counteracted by means of end point relaxation opening up the ends of the search region.

4.6 Results

In all the experiments the signal was divided in 25ms frames without overlapping. Each frame was processed with Hamming window before the frame energy and spectral estimation being computed. The spectral estimation was made with a 14 channel Mel-filter bank covering the band from 300 to 3400 Hz. The AR filters were estimated by means of LMS algorithm in short non-speech intervals (180-430 ms) before the utterances start. The learning rate was made equal to $0.1/(M \times \mathtt{noise\_power})$, where M is the filter order. The FIR taps were set to 0 at the beginning of the iterative procedure. In order to determine the optimum prediction order, several configurations were tested and the one that gave the lowest prediction error was chosen. If the prediction error was similar for two M's, the lowest M was chosen.

4.6.1 Noisex database

In the experiments done with the Noisex-92 database (A.Varga *et al.*, 1992), the signals were low pass filtered by using a 10th order Tchebychev filter with cut off frequency 3700 Hz, down sampled from 16000 to 8000 samples/sec, and high-pass filtered by employing a 4th order Tchebychev filter with cut off frequency 120 Hz and a minimum attenuation equal to 25dB.
Three noises from Noisex-92 were considered (car, speech and Lynx) and for each case one AR FIR filter was trained by means of 10 noise-only sample frames (250 ms). The clean and noisy speech signals belonged to the male speaker of Noisex-92 database. Table 1 shows the optimum number of taps for each filter and the ratio G between the attenuation gain on clean speech signals and the attenuation gain on the training noise signal after the AR FIR filter being estimated. This quotient G gives an idea of the energy gap increase between noise and speech due to the

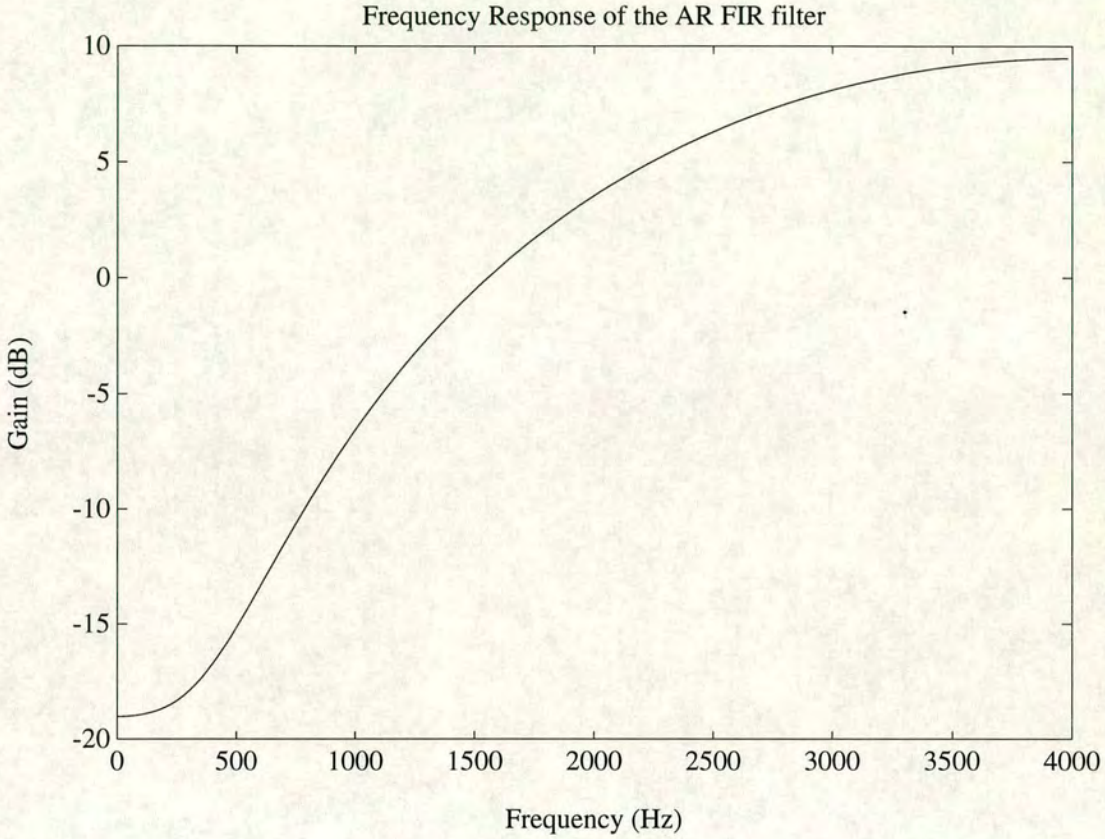


Figure 4.3: Frequency response of the AR FIR filter for the car noise of Noisex database

Table 4.1: Optimum AR FIR order and quotient G between the energy attenuation gains on clean speech signal and training noise.

NOISE	*Car*	*Speech noise*	*Lynx*
Optimum FIR Order	2	2	4
G (dB)	13.1	6.6	5.3

AR FIR filter. The clean signals corresponded to 10 utterances (one per digit) automatically end detected. Fig. 4.3 shows the frequency response of the AR filter for the car noise. Figures 4.4-4.6 present the power envelope, spectral comparison and non-stationarity coefficients before and after processing the signal with the AR FIR filter. The power envelope corresponds to the difference between the mean frame energy (dB) and the noise power energy estimation (dB) according to (4.9).

Results are shown in Figs. 4.4, 4.5 and 4.6 for the digits "one" at SNR=0dB, "six" also at SNR=0dB and "one" at SNR=-6dB.

4.6.2 Telephone database

The autoregressive analysis of noise was also tested on the BT-Subscriber database recorded over the telephone line across UK using 1000 callers. The database is composed by phonetically balanced sentences and only signals that were labelled as 'noisy' were considered. Two cases of data recorded on the telephone line were studied: a) speech signal with background noise composed mainly by low frequencies (presumably from the power net); and b) speech signal corrupted by low and high frequency components. The data correspond to 5 and 4 sec of continuous speech sampled at 8 kHz and high-pass filtered, as done with the Noisex database, by employing a 4th order Tchebychev filter with cut off frequency 120 Hz and a minimum attenuation equal to 25dB. The speech signal was manually speech/no-speech segmented and the power envelope before and after AR analysis were compared. The AR filter was estimated with non-speech signal available at the beginning of the data: 430 ms for the low frequency noise, and 180 ms for the low and high frequency components one. The learning rate was made equal to $0.1/(M \times \mathrm{noise\_power})$ and the FIR taps were set to 0 at the beginning of the iterative procedure. In order to determine the optimum prediction order the same criterion used with the Noisex database was followed. The optimum filter order was $M = 2$ and $M = 4$ for the low frequency noise and the other one, respectively. However the results achieved with $M = 2$ for the noise with low and high frequency components were similar to the ones with $M = 4$.
Figures 4.7 and 4.8 show the frequency response of the AR filter for the noises considered in this section.

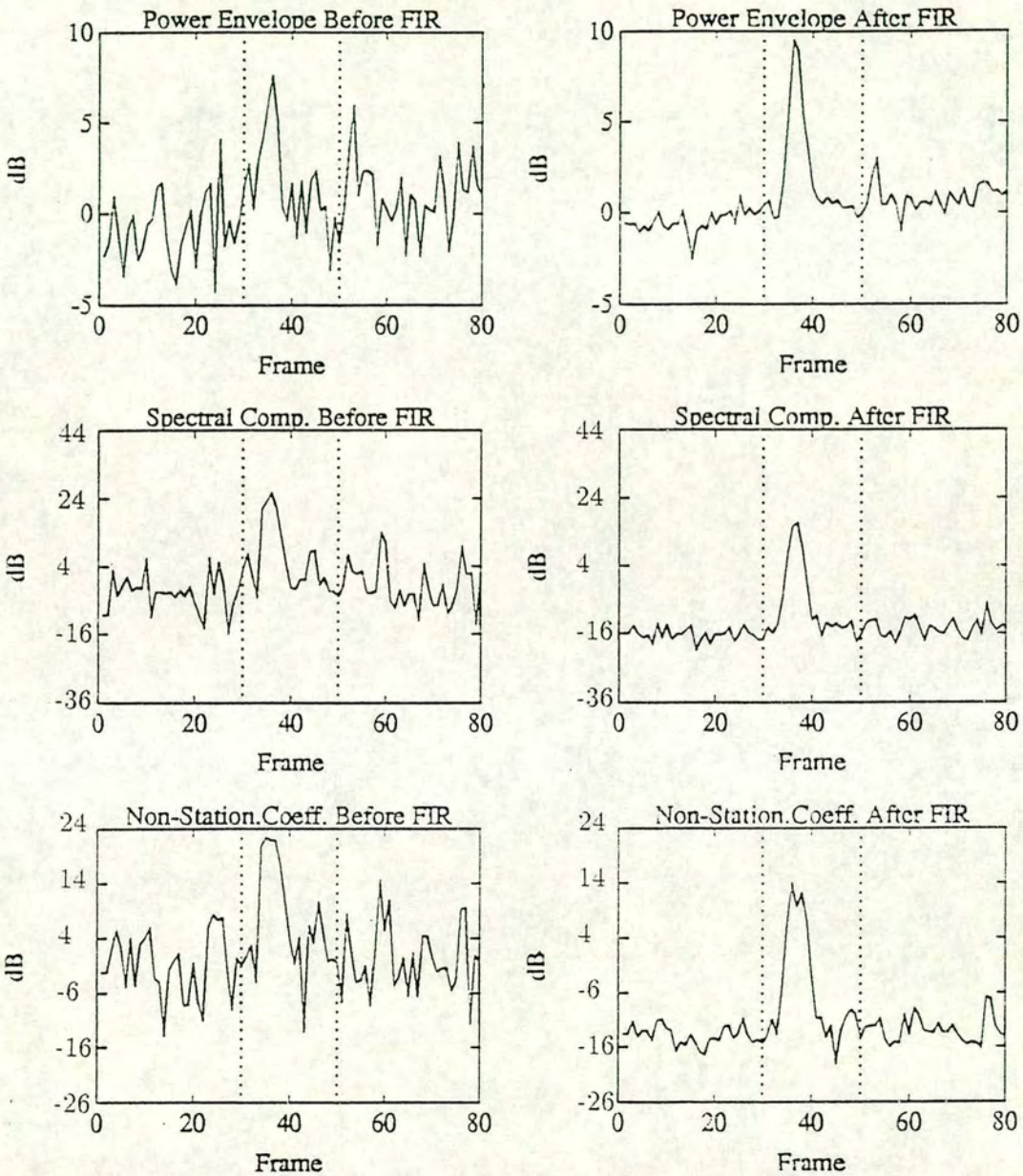


Figure 4.4: Power envelope, spectral comparison and non-stationarity coefficients before and after processing the signal with the AR FIR filter. The utterance correponds to the digit 'one' in car noise with SNR equal to 0dB. The dotted vertical lines denote the endpoints of the speech signal.

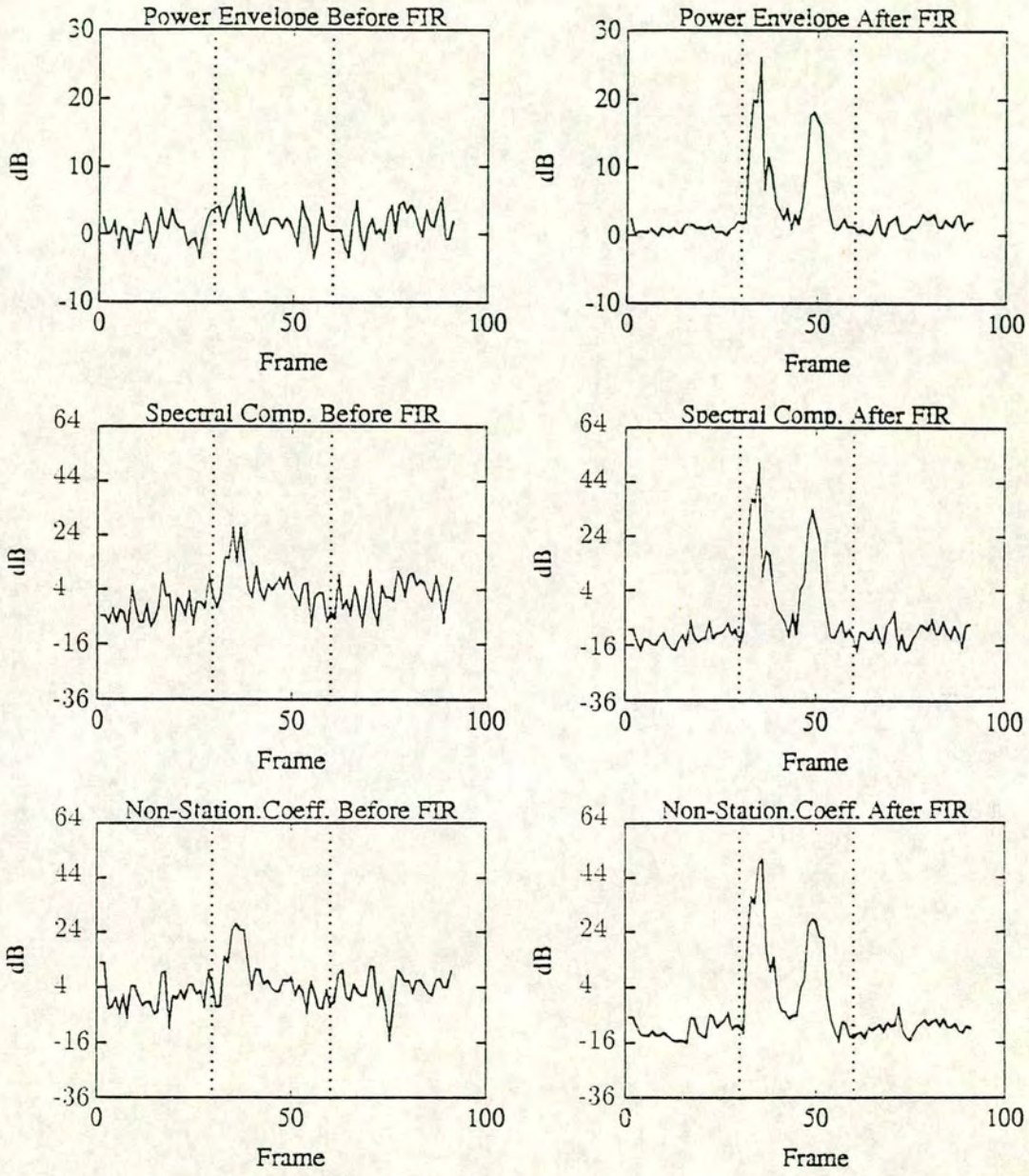


Figure 4.5: Power envelope, spectral comparison and non-stationarity coefficients before and after processing the signal with the AR FIR filter. The utterance correponds to the digit 'six' in car noise with SNR equal to 0dB The dotted vertical lines denote the endpoints of the speech signal.

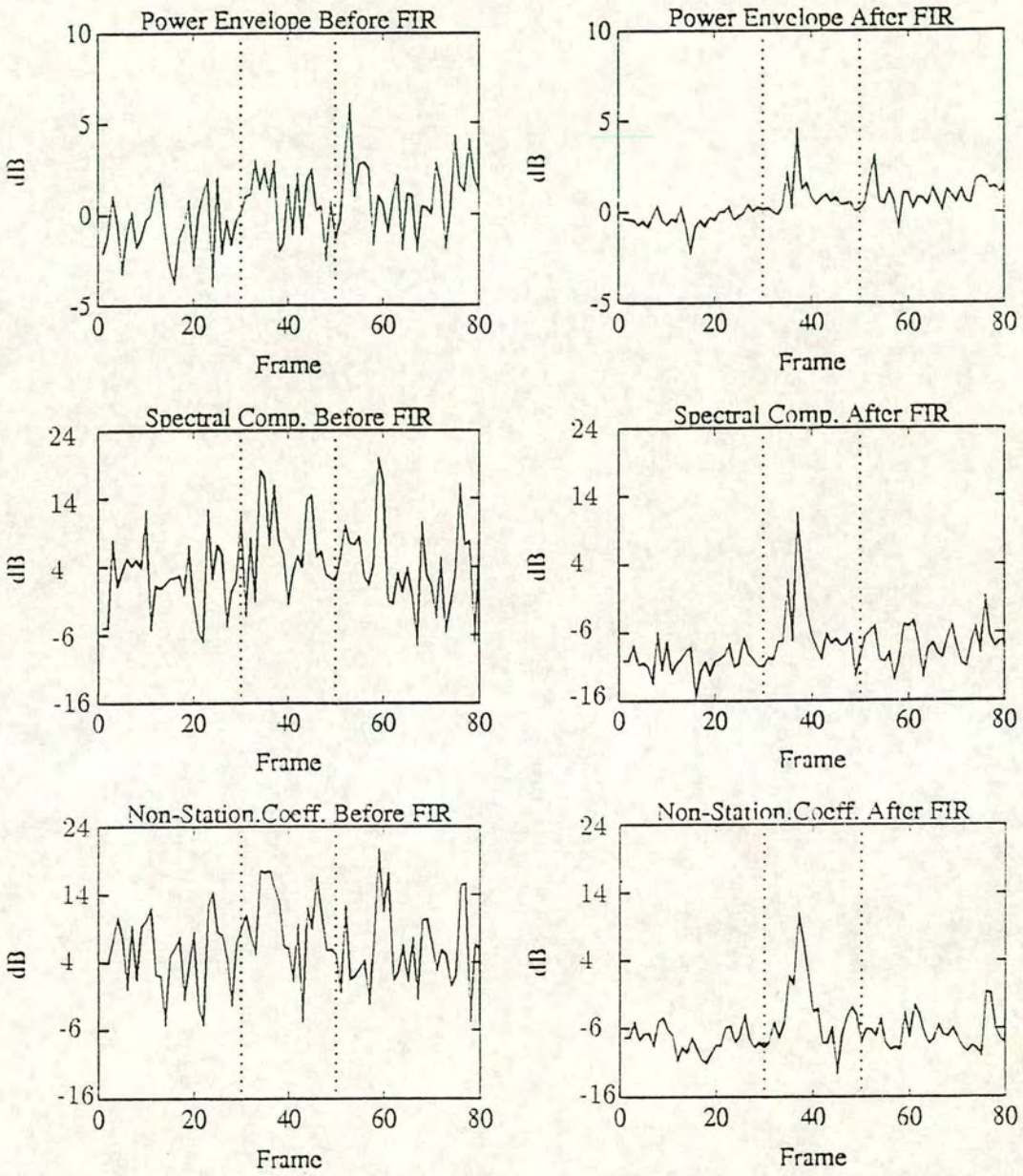


Figure 4.6: Power envelope, spectral comparison and non-stationarity coefficients before and after processing the signal with the AR FIR filter. The utterance correponds to the digit "one" in car noise and SNR equal to -6dB. The dotted vertical lines denote the endpoints of the speech signal.

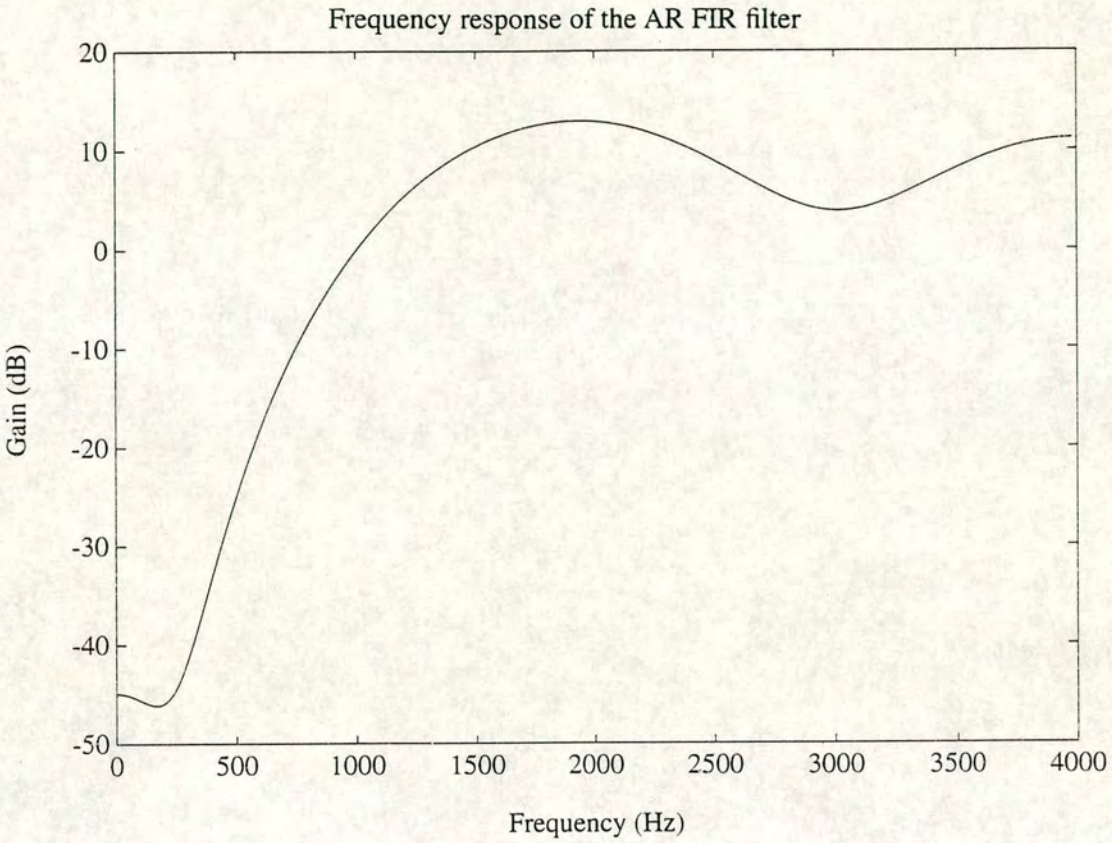


Figure 4.7: Frequency response of the AR FIR filter for the low frequency noise.

Figures 4.9 and 4.10 present the power envelope before and after the signal being processed by the AR filter, and the manual speech/non-speech segmentation for both noises.

4.7 Discussion

According to the results with Noisex database (section 4.6.1) , the AR analysis led to a higher attenuation on average for the noises than for speech signals (see Table 3.1). According to Figs. 4.4-4.6, the AR FIR filters increased the discrimination between the speech signal and backgound noise in the power, spectral comparison and non-stationarity coefficient domains. When compared with the power envelope, spectral comparison and non-stationarity coefficients increased slightly the difference between speech and non-speech pulses before FIR processing but gave similar results after FIR at SNR equal to 0dB (Figs. 4.4 and 4.5). According to Fig.

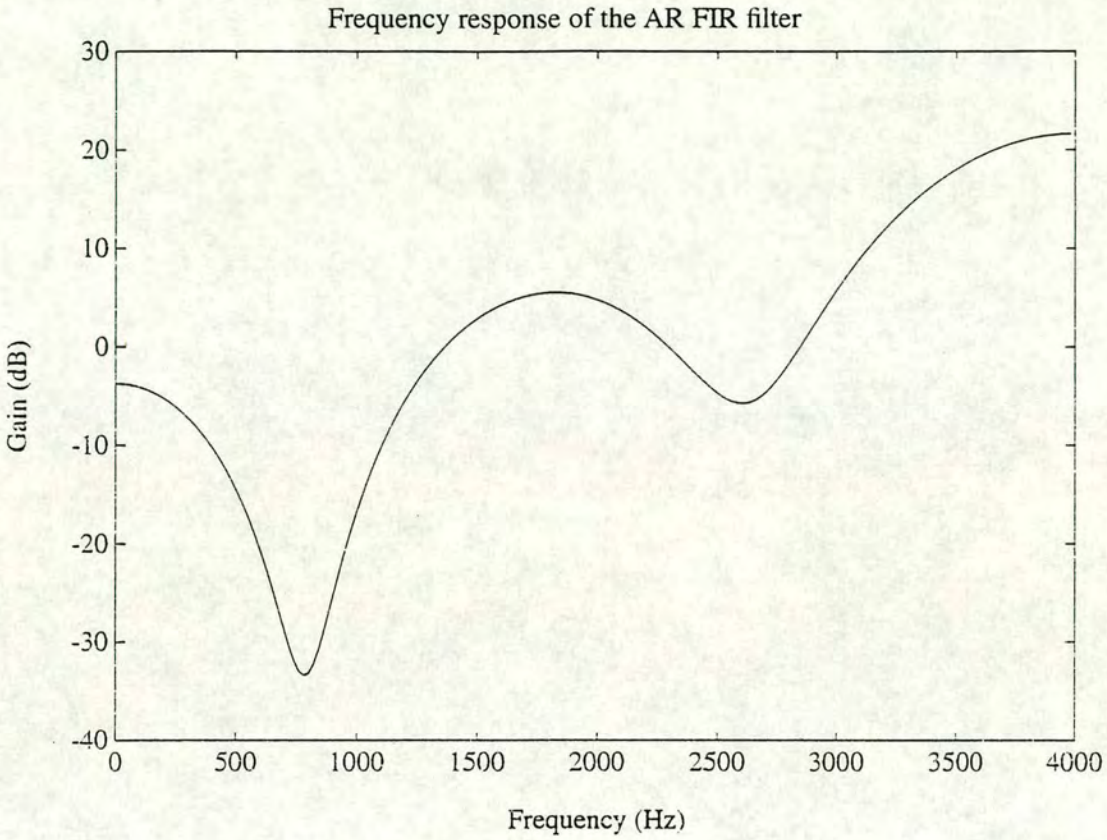


Figure 4.8: Frequency response of the AR FIR filter for the low and high frequency noise.

4.6 (SNR=-6dB), these coefficients increased the difference between speech and non-speech pulses after FIR processing but the improvement achieved before AR analysis was not enough to highlight the speech signal from the backgound.

According to Fig. 4.4 and 4.5, the improvement due to the AR filter was more important for the utterance 'six' rather than for the utterance 'one'. As can be seen in Fig. 4.5, the speech pulses were almost completely masked in the power envelope domain before FIR processing, but after the AR filter the speech signal was even more accurately detected than for the utterance 'one' (Fig. 4.4), that was slightly more evident than the utterance 'six' without FIR processing. This must be caused by the fact that the word 'six' has considerable energy in highest frequencies and could be more easily discriminated from the background noise, which was mainly concentrated in low frequencies.

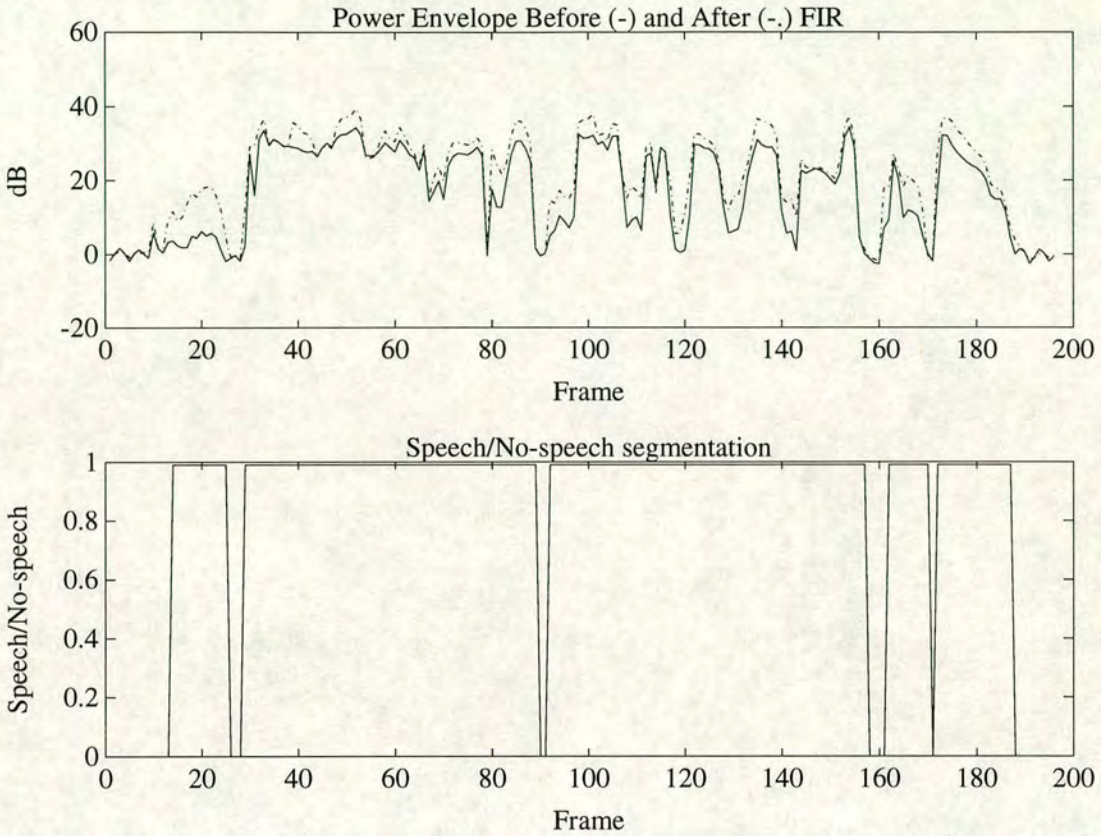


Figure 4.9: Power envelope before and after AR analysis, and manual speech/non-speech segmentation for the low frequency noise.

The estimation of M, number of taps of the AR filter (4.2), deserves a brief discussion. In all the examples considered, the optimum M was 2 or 4. Higher values for M (until 15) were tested and no substantial variation on the prediction error was observed, although a minimum one was clearly discernable in some cases. A priori, increasing the AR FIR order should improve the noise signal model, but the LMS algorithm firstly tries to cancel the highest components and, if the signal keeps its stationarity, tries to compensate the lowest ones (J.M.Romano, 1996). Nevertheless, this may take a considerable amount of only-noise samples which is not exactly the case of this research where just short intervals (e.g. 250 ms) are available to train the FIR structure. In order words, the low variation with M of the prediction error may be due to the fact that the lowest components needed more samples to be compensated and that the noises were not perfectly stationary. If the non-speech intervals where the AR filter could be trained are generally short (e.g. 200-400ms), the LMS algorithm is able to cancel just the highest component (and

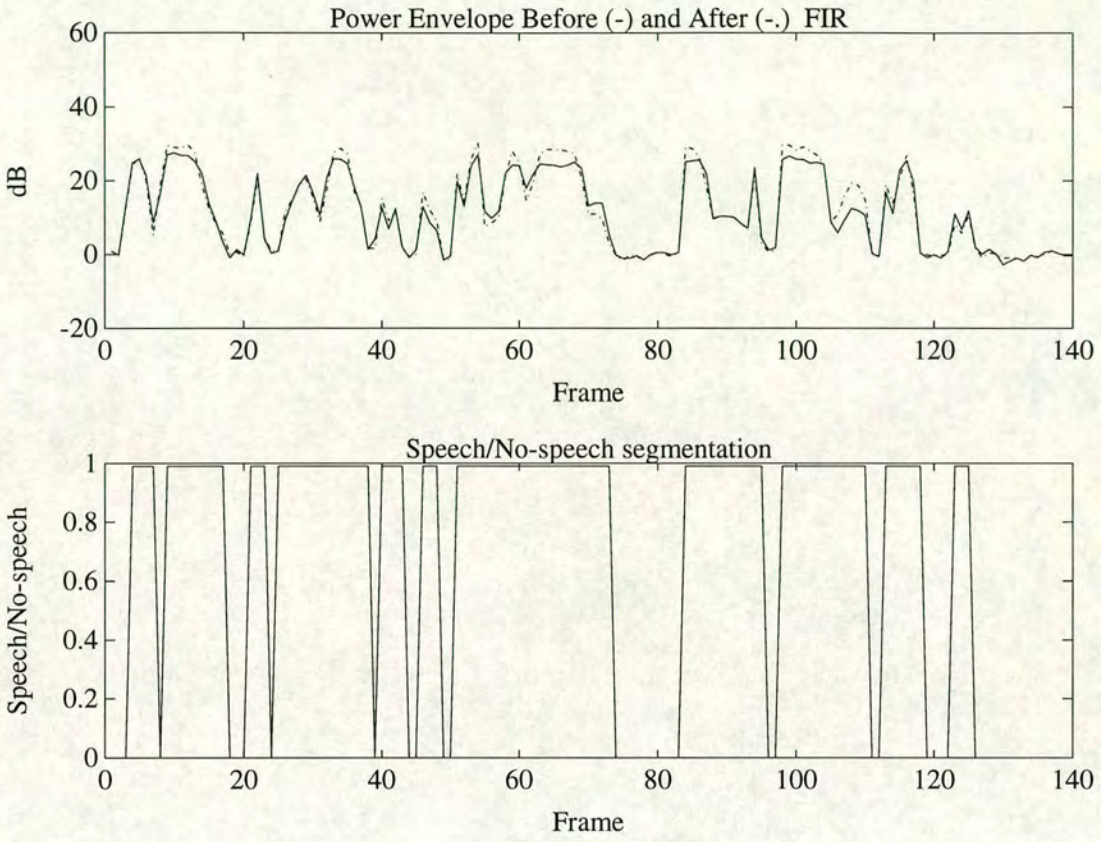


Figure 4.10: Power envelope before and after AR analysis, and manual speech/non-speech segmentation for the low high frequency components noise.

the second one when it is the case) and it is not worth using a number of taps higher than 2 or 4. Moreover, a big difference was not observed between the results with M =2 and M =4 so it could be concluded that if the AR can improve the discrimination between noisy speech and background noise, M =2 should be enough.

As can be seen from the results with the telephone signals (section 4.6.2), one noise is clearly composed by only low frequency components (Fig. 4.7) and the other one shows energies around 700 and 2500 Hz (Fig. 4.8). According to Fig 4.9, in the case of a low-frequency noise the AR analysis substantially improved the discrimination between background noise and speech signal, specially at those low-energy intervals at the begining and end of the utterances. As shown in Fig. 4.10, in the case of low and high-frequency noise the AR analysis did not introduce any significant effect on the discrimination between speech and noise signals. Some intervals were slightly enhanced and others minimally attenuated, but the overall effect of the AR filter could

be considered neutral.

4.8 Conclusion

As can be seen in the results with the Noisex database, the observed improvements were due mainly to the AR analysis, although spectral comparison and non-stationarity coefficient might be useful at low SNR's. The AR FIR filters needed a low number of taps, the LMS algorithm seems to be fast enough to capture slow variations of the noise characteristics and only one microphone is necessary.

AR analysis relies on the fact that the noise is reasonably stationary. Under this assumption, it was shown in sections 4.6.1 and 4.6.2 that if the noise is mostly composed by low frequencies, a case easily found in practical applications, AR analysis can strongly increase the discrimination between noisy speech and corrupting signals. On the other hand, according to section 4.6.2 AR analysis apparently does not introduce any artifact if the noise is composed by a mixture of low and high frequencies. If the noise is restricted to high frequency components, a case that was not studied in this research and not easily found in real applications, the AR filter would attenuate low-energy high-frequency unvoiced speech signals and highlight mainly high-energy low-frequency voiced intervals. This fact would help to shorten the detected utterances at mild SNR's but AR analysis would still be useful to discriminate speech and noise at more severe SNR's.

As far as the utterance detector algorithm is concerned, further work is needed to automatically estimate the thresholds according to the SNR and to include more general cases such as connected and continuous speech. However, the end-point detector described in this chapter was effective enough to detect isolated words in noisy backgrounds and is not going to be discussed again in this thesis due to the fact that it is not the main subject of this research.

The end-point detector with AR analysis of noise was employed to automatically detect isolated words (digits) in the context of the Noisex database to run recognition experiments shown in Chapter 5, where spectral subtraction (SS) is analysed under the perspective of reliability in noise cancelling and tested in combination with a weighted DP algorithm.

Chapter 5

Spectral subtraction and reliability in noise cancelling

5.1 Introduction

Due to the fact that the intervals with highest energies are less corrupted by additive noise, it is reasonable to suppose that these intervals provide more reliable information for speech recognition than those intervals with lower energies. In (H.Kobatake & Y.Matsunoo, 1994) and (N.B.Yoma *et al.*, 1995) were proposed two weighted matching algorithms to take into account the local SNR. Both algorithms were tested with poorly correlated and white Gaussian noises but with different noise cancellation techniques. In (H.Kobatake & Y.Matsunoo, 1994) a two-step weighted DTW algorithm and weighting coefficients based on the short-time power or short-time autocorrelation were proposed and applied in combination with the Root-Power Sum distance. Although the weighted algorithm was shown to improve the recognition in some cases, the noise cancelling technique introduced an error rate for the clean signal, the weighting procedure was not modelled and the two-step DTW even increased the error rate depending on which weighting coefficient was used. In Chapter 3 a one-step weighted DP algorithm was proposed and compared with the two-step one. They were tested in combination with a noise cancellation neural net, and shown effective in reducing the error rate. However, further experiments showed that the improvements due to the weighted Dynamic Programming algorithms depended on the neural net training conditions, and suggested that the weighting coefficient $w(t)$ should take into account not only the segmental SNR but the characteristics of the noise reduction method (Chapter 3). Following this idea, in (N.B.Yoma *et al.*, 1996d) (N.B.Yoma *et al.*, 1996a)

was proposed the use of a weighting parameter based on reliability in noise cancelling. This parameter takes into account not only the local SNR but also the characteristic response of the noise cancellation method in the form of a mean distortion curve (section 3.3.2).

The contributions of this chapter concern: a) a model for additive noise based on Mel filter banks (IIR and DFT); b) analysis of spectral subtraction (SS) in terms of reliability in noise cancelling; and c) combination of weighted matching algorithms with spectral subtraction technique. The approach here covered has not been found in the literature and seems to be generic and interesting from the practical applications point of view. The techniques were tested with DTW recognition algorithms on an isolated word recognition task. DTW was used because it is a simple and generic algorithm which allows noise cancelling techniques to be compared without the need for extensive tuning of the modelling. In Chapter 6 the techniques explored here will also be employed by a weighted Viterbi (HMM) algorithm.

5.2 Model for additive noise using IIR filters

Given that $s(i)$, $n(i)$ and $x(i)$ are the clean speech, the noise and the resulting noisy signal, respectively, the additiveness condition may be set as:

$$x(i) = s(i) + n(i) \tag{5.1}$$

The signal was processed by 14 Mel IIR filters. At the output of filter m the noisy signal is given by:

$$x_m(i) = s_m(i) + n_m(i) \tag{5.2}$$

and its mean energy in a frame by:

$$\overline{x_m^2} = \overline{s_m^2} + \overline{n_m^2} + \overline{2 s_m n_m} \tag{5.3}$$

where $\overline{x_m^2(i)} = \frac{1}{N}\sum_{i=1}^{N} x_m^2(i)$, $\overline{s_m^2(i)} = \frac{1}{N}\sum_{i=1}^{N} s_m^2(i)$, $\overline{n_m^2(i)} = \frac{1}{N}\sum_{i=1}^{N} n_m^2(i)$, $\overline{2 s_m(i) n_m(i)} = \frac{1}{N}\sum_{i=1}^{N} 2 s_m(i) n_m(i)$ and N is the length of the frames in number of samples.

If the speech signal and the noise are uncorrelated, $E[\overline{2 s_m\ n_m}] = 0$ in a long term analysis, where $E[]$ corresponds to the expected value. However, the condition $\overline{2 s_m\ n_m} = 0$ may not be

satisfied in a short term analysis (i.e. a 25 ms frame) and the noise is certainly not perfectly stationary. Consequently, once the noise is added the clean signal energy, $\overline{s_m^2}$, becomes a hidden information and cannot be recovered with a 100% accuracy. As a result, $\overline{s_m^2}$ should be treated as a stochastic variable and could be associated to a variance that indicates how accurate the estimation of the clean signal energy is.

Initially, the signals $s_m(i)$ and $n_m(i)$ are considered sinusoidal components with frequency f_m, the central frequency of filter m, with a phase difference ϕ. Under these assumptions,

$$\overline{x_m^2} = \frac{a_{s_m}^2}{2} + \overline{n_m^2} + a_{s_m} a_{n_m} \cos(\phi) \tag{5.4}$$

where a_{s_m} and a_{n_m} are the amplitudes of the speech signal and noise components respectively: $\overline{s_m^2} = a_{s_m}^2/2$ and $\overline{n_m^2} = a_{n_m}^2/2$.

5.2.1 Correction of the sinusoidal model for IIR filters

The sinusoidal model for additive noise represented by equation (5.4) assumes that the components $s_m(i)$ and $n_m(i)$ at the output of filter m have frequency f_m and a phase difference ϕ in a given frame. These assumptions are not perfectly accurate in practice. Firstly, the 14 mel filters are not highly selective, which reduces the validity of the assumption of coherence between both components. Secondly, the phase ϕ between $s_m(i)$ and $n_m(i)$ is not necessarily constant and a few discontinuities in the phase difference may occur, although many of them are unlikely in a short term analysis (i.e. a 25 ms frame). However, the sinusoidal model represents the fact that there is a variance in the short term analysis and specifies the relation between this variance and the clean and noise signal levels. Due to the lack of coherence between $s_m(i)$ and $n_m(i)$ and to the discontinuity in the phase difference, the variance predicted by the model is higher than the the real one for the same frame length, and a correction should be included in (5.4). According to (5.4) and considering that the random variable ϕ was uniformly distributed between $-\pi$ and π:

$$Var[\overline{x_m^2(i)} | \overline{s_m^2(i)}, \overline{n_m^2(i)}] = 0.5 a_{s_m}^2 a_{n_m}^2$$

because

$$Var[\cos(\phi)] = E[\cos^2(\phi)] - E^2[\cos(\phi)]$$

and

$$E[\cos^2(\phi)] = \frac{1}{2\pi}\int_{-\pi}^{\pi} \cos^2(\phi)d\phi = 0.5$$

$$E[\cos(\phi)] = \frac{1}{2\pi}\int_{-\pi}^{\pi} \cos(\phi)d\phi = 0$$

In order to estimate the correction of the sinusoidal model, the coefficient r_m defined as

$$r_m = \frac{\overline{2s_m(i)n_m(i)}}{a_{s_m} a_{n_m}} \tag{5.5}$$

was computed with clean speech and only-noise frames. According to (5.4), $Var[r_m|\overline{s_m^2(i)}, \overline{n_m^2(i)}]$ should be equal to 0.5 but due to the lack of coherence between $s_m(i)$ and $n_m(i)$ and to the discontinuity in the phase difference,

$$Var[r_m|\overline{s_m^2(i)}, \overline{n_m^2(i)}] < 0.5$$

and a correction factor c_m needs to be included in (5.4):

$$\overline{x_m^2(i)} = \frac{a_{s_m}^2}{2} + \overline{n_m^2(i)} + a_{s_m} a_{n_m}\sqrt{c_m}\cos(\phi) \tag{5.6}$$

where c_m is defined as

$$c_m = 2Var[r_m|\overline{s_m^2(i)}, \overline{n_m^2(i)}]$$

With the sinusoidal model for additive noise represented by (5.6), the variance (or uncertainty) of the hidden information $\overline{s_m^2(i)}$ given the observed information $\overline{x_m^2(i)}$ is estimated. Solving (5.6) for a_{s_m} and using $\overline{s_m^2(i)} = a_{s_m}^2/2$:

$$\begin{aligned}\overline{s_m^2(i)} = {} & a_{n_m}^2 c_m\cos^2(\phi) + \overline{x_m^2(i)} - \overline{n_m^2(i)} - \\ & a_{n_m}\sqrt{c_m}\cos(\phi)\sqrt{a_{n_m}^2 c_m\cos^2(\phi) + 2(\overline{x_m^2(i)} - \overline{n_m^2(i)})}\end{aligned} \tag{5.7}$$

The equation above sets $\overline{s_m^2(i)}$ as a function of ϕ, $\overline{n_m^2(i)}$ and $\overline{x_m^2(i)}$:

$$\overline{s_m^2(i)} = \overline{s_m^2}(\phi, \overline{n_m^2(i)}, \overline{x_m^2(i)}) \tag{5.8}$$

Replacing a_{n_m} with $\sqrt{2 \cdot \overline{n_m^2}}$, the function $\overline{s_m^2}(\phi, \overline{n_m^2}, \overline{x_m^2})$ can be written as:

$$\overline{s_m^2}(\phi, \overline{n_m^2}, \overline{x_m^2}) = 2 \cdot A^2 \cdot \cos^2(\phi) + B - 2 \cdot A \cdot \cos(\phi) \cdot \sqrt{A^2 \cdot \cos^2(\phi) + B} \quad (5.9)$$

where $A = \sqrt{\overline{n_m^2} c_m}$ and $B = \overline{x_m^2} - \overline{n_m^2}$.

5.3 Model for additive noise using DFT filters

In the last section a model for additive noise using IIR filters (N.B.Yoma *et al.*, 1997b) was proposed. The low selectivity of the IIR filters makes the system more vulnerable to convolutional distortions and the use of a DFT filter bank is desirable because it provides an infinite rejection outside the filter band. In this section, the equivalent additive noise model for the case of DFT filters is presented.

Again, given that $s(i)$, $n(i)$ and $x(i)$ are the clean speech, the noise and the resulting noisy signal, respectively, the additiveness condition in the temporal domain may be set as:

$$x(i) = s(i) + n(i)$$

The signal was processed by 14 DFT Mel filters. If $S(k)$, $N(k)$ and $X(k)$ correspond to the FFT transform of $s(i)$, $n(i)$ and $x(i)$ at the point k, and ϕ_k is the phase difference between $S(k)$ and $N(k)$, the additiveness condition is then set by:

$$X(k) = S(k) + N(k) \quad (5.10)$$

According to the cosine rule,

$$|X(k)|^2 = |S(k)|^2 + |N(k)|^2 + 2 \cdot |S(k)| \cdot |N(k)| \cdot \cos(\phi_k) \quad (5.11)$$

The energy at the output of the filter m, $\overline{x_m^2}$, is computed by means of:

$$\overline{x_m^2} = \sum_{k \in \text{filter } m} G(m, k) \cdot |X(k)|^2 \quad (5.12)$$

where $G(m, k)$ is the set of weights that define the filter m. If $|X(k)|^2$ in (5.12) is replaced with

the expression given in (5.11), $\overline{x_m^2}$ can be set as

$$\overline{x_m^2} = \overline{s_m^2} + \overline{n_m^2} + \sum_{k \in \text{filter } m} 2 \cdot G(m,k) \cdot |S(k)| \cdot |N(k)| \cdot \cos(\phi_k) \tag{5.13}$$

where: $\overline{s_m^2}$ and $\overline{n_m^2}$ are the filter m mean frame energy of the clean speech and noise signal, respectively.

Assuming that the phase difference $\phi(k) = \phi$, $N(k)$ and $X(k)$ are considered constant inside each one of the 14 DFT Mel filters indexed by m:

$$\overline{x_m^2} = \overline{s_m^2} + \overline{n_m^2} + 2 \cdot \sqrt{\overline{s_m^2}} \cdot \sqrt{\overline{n_m^2}} \cdot \cos(\phi) \tag{5.14}$$

5.3.1 Correction of the additive noise model for DFT filters

The model for additive noise represented by (5.14) assumes that the components $|S(k)|$ and $|N(k)|$ and the phase difference ϕ are constant inside every filter in a given frame. These assumptions are not perfectly accurate in practice. Firstly, the 14 DFT mel filters are not highly selective, which reduces the validity of the assumption of low variation of these parameters inside the filters. Secondly, the phase ϕ between $|S(k)|$ and $|N(k)|$ is not necessarily constant and a few discontinuities in the phase difference may occur, although many of them are unlikely in a short term analysis (i.e. a 25 ms frame). However, this model represents the fact that there is a variance in the short term analysis and specifies the relation between this variance and the clean and noise signal levels. Due to these approximations the variance predicted by the model is higher than the the real one for the same frame length, and a correction should be included. Using (5.14) and considering that ϕ was uniformly distributed between $-\pi$ and π:

$$Var[\frac{\overline{x_m^2}}{2} | \overline{s_m^2}, \overline{n_m^2}] = 0.5 \cdot \overline{s_m^2} \cdot \overline{n_m^2}$$

because, as stated above (section 5.2.1), $Var[\cos(\phi)] = 0.5$.

In order to estimate the correction of the model, the coefficient r_m defined as

$$r_m = \frac{\overline{x_m^2} - \overline{s_m^2} - \overline{n_m^2}}{2 \cdot \sqrt{\overline{s_m^2}} \sqrt{\overline{n_m^2}}} \tag{5.15}$$

was computed with clean speech and only-noise frames. According to (5.14), $Var[r_m | \overline{s_m^2}, \overline{n_m^2}]$ should be equal to 0.5 but due to the approximations this variance is lower than 0.5 and a

correction factor c_m needs to be included in (5.14):

$$\overline{x_m^2} = \overline{s_m^2} + \overline{n_m^2} + 2 \cdot \sqrt{c_m} \cdot \sqrt{\overline{s_m^2}} \cdot \sqrt{\overline{n_m^2}} \cdot \cos(\phi) \quad (5.16)$$

where c_m is defined as

$$c_m = 2Var[r_m|\overline{s_m^2}, \overline{n_m^2}]$$

Solving (5.16) for $\overline{s_m^2}$

$$\overline{s_m^2}(\phi, \overline{n_m^2}, \overline{x_m^2}) = 2 \cdot A^2 \cdot \cos^2(\phi) + B - 2 \cdot A \cdot \cos(\phi) \cdot \sqrt{A^2 \cdot \cos^2(\phi) + B} \quad (5.17)$$

where $A = \sqrt{\overline{n_m^2} c_m}$ and $B = \overline{x_m^2} - \overline{n_m^2}$.

5.4 Channel variance

With the model for additive noise represented by (5.9) or (5.17), the variance (or uncertainty) of the hidden information $\overline{s_m^2}$ given the observed information $\overline{x_m^2}$ is estimated in the logarithmic domain considering that the random variables ϕ and $\overline{n_m^2}$ are uncorrelated, ϕ is uniformly distributed between $-\pi$ and π and that $\overline{n_m^2}$ is concentrated near its mean $E[\overline{n_m^2}]$. The variance $Var[\log(\overline{s_m^2})|\overline{x_m^2}]$ is given by:

$$Var[\log(\overline{s_m^2})|\overline{x_m^2}] = E[\log^2(\overline{s_m^2})|\overline{x_m^2}] - E^2[\log(\overline{s_m^2})|\overline{x_m^2}] \quad (5.18)$$

where

$$E[\log^2(\overline{s_m^2})|\overline{x_m^2}] \simeq \frac{1}{2\pi} \int_{-\pi}^{\pi} \log^2[\overline{s_m^2}(\phi, E[\overline{n_m^2}], \overline{x_m^2})] d\phi \quad (5.19)$$

and

$$E[\log(\overline{s_m^2})|\overline{x_m^2}] \simeq \frac{1}{2\pi} \int_{-\pi}^{\pi} \log[\overline{s_m^2}(\phi, E[\overline{n_m^2}], \overline{x_m^2})] d\phi \quad (5.20)$$

As shown in next section (N.B.Yoma *et al.*, 1997a), $E[\log(\overline{s_m^2})|\overline{x_m^2}] \simeq \log[\overline{x_m^2} - E[\overline{n_m^2}]]$ and the integral for estimating $E[\log^2(\overline{s_m^2})|\overline{x_m^2}]$ was computed by means of Simpson's rule with the interval $(-\pi, \pi)$ divided in 100 regular partitions and replacing the difference $\overline{x_m^2} - E[\overline{n_m^2}]$ in (5.9) or (5.17) with $r(Est(\overline{s_m^2}), SsThr)$ (5.26). The constant 100 is to make $Var[\log(\overline{s_m^2})|\overline{x_m^2}]$ compatible with the fact that the log energies are considered in dB.

Analysing (5.9) and (5.17), it can be seen that the variance of $\log[\overline{s_m^2}]$ given $\overline{x_m^2}$ should be a function of the ratio $(\overline{x_m^2} - E[\overline{n_m^2}])/E[\overline{n_m^2}]$ because any gain, that results in a multiplicative constant in the linear domain, becomes additive in the log one and it is cancelled. Fig. 5.1 shows $1/Var[\log(\overline{s_m^2})|\overline{x_m^2}]$ vs the ratio $(\overline{x_m^2} - E[\overline{n_m^2}])/E[\overline{n_m^2}]$ for some values of c_m. The variance of $\log[\overline{s_m^2}]$ was computed as explained in section 5.4. In fact, as can be seen in Fig. 5.1, $1/Var[\log(\overline{s_m^2})|\overline{x_m^2}]$ seems to be directly proportional to $(\overline{x_m^2} - E[\overline{n_m^2}])/E[\overline{n_m^2}]$, which suggests that $1/Var[\log(\overline{s_m^2})|\overline{x_m^2}]$ could be approximated by

$$Var[\log(\overline{s_m^2})|\overline{x_m^2}] \approx \frac{c_m \cdot E[\overline{n_m^2}]}{2.7 \cdot (\overline{x_m^2} - E[\overline{n_m^2}])} \tag{5.21}$$

Although (5.21) offers an efficient way to compute $Var[\log(\overline{s_m^2})|\overline{x_m^2}]$, in all the experiments this variance was estimated by means of (5.18) and directly computing the integral in (5.19) using the Simpson's rule.

5.5 Spectral subtraction

The models for additive noise, represented by (5.9) for the IIR filter bank or by (5.17) for the DFT one, set that given the observed information of the noisy signal $\overline{x_m^2}$ and the noise estimation $\overline{n_m^2}$ there is an uncertainty about the hidden clean signal $\overline{s_m^2}$ due to lack of knowledge about the phase difference between the corrupting and clean signal. In others words, once the noise is added $\overline{s_m^2}$ cannot be recovered with 100% accuracy but (5.20) gives the clue to minimize the risk in estimating $\overline{s_m^2}$.

Given the model for additive noise represented by (5.9) or (5.17), the expected value of $\log[\overline{s_m^2}(\phi, \overline{n_m^2}, \overline{x_m^2})]$ given the observed information $\overline{x_m^2}$ would be:

$$E[\log(\overline{s_m^2})|\overline{x_m^2}] = \tag{5.22}$$
$$E[\log(2 \cdot \frac{A^2}{B} \cdot \cos^2(\phi) + 1 - 2 \cdot \frac{A}{B} \cdot \cos(\phi) \cdot \sqrt{A^2 \cdot \cos^2(\phi) + B})|\overline{x_m^2}] + E[\log(B)|\overline{x_m^2}]$$

Taking square root of argument of log in (5.22), and assuming that the random variables ϕ and $E[\overline{n_m^2}]$ are uncorrelated, ϕ is uniformly distributed between $-\pi$ and π and that $\overline{n_m^2}$ is concentrated

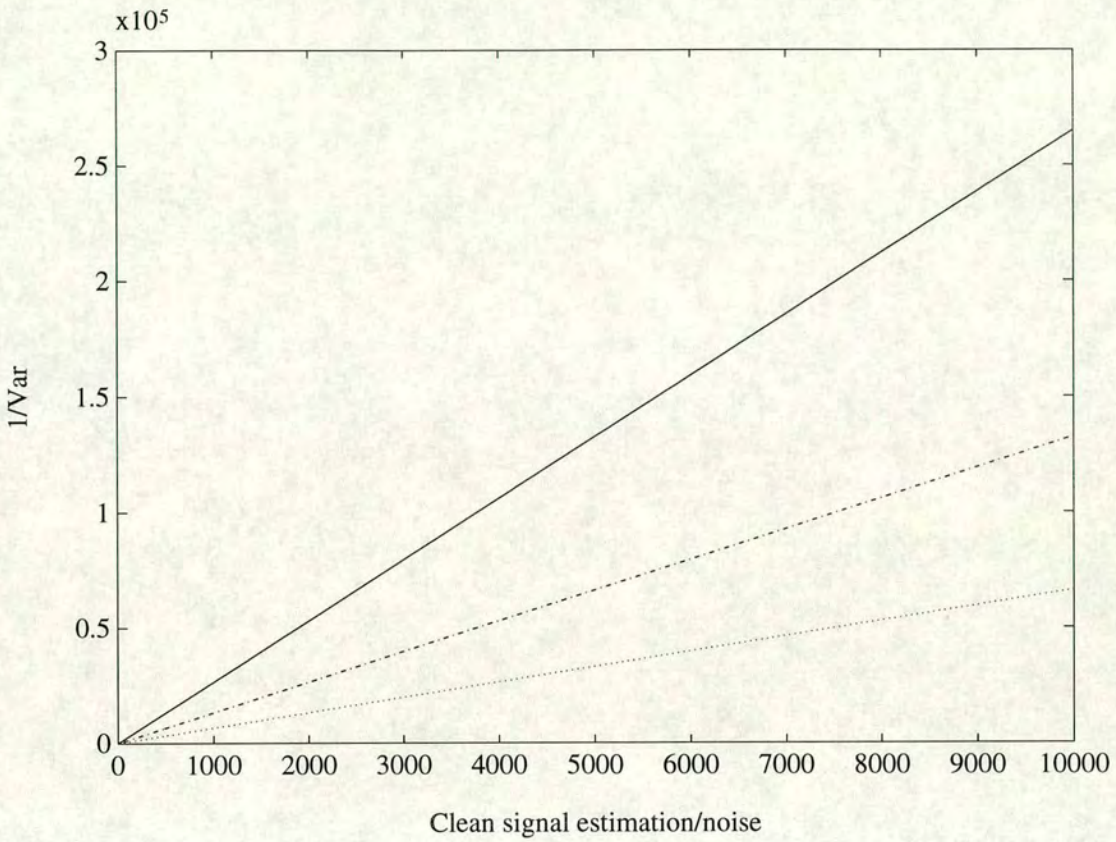


Figure 5.1: Inverse of the channel variance vs the clean signal estimation normalized to the noise energy: (-), $c_m = 0.1$; (-.), $c_m = 0.2$; (..), $c_m = 0.4$. The channel or uncertainty variance was estimated according to (5.18) using numerical integration, and the clean signal estimation corresponds to the difference between the noisy signal energy and the the noise energy estimation in a channel. This graphic suggests that the channel variance could be approximated by (5.21).

near its mean $E[\overline{n_m^2}]$, $E[\log(\overline{s_m^2})|\overline{x_m^2}]$ can be written as:

$$E[\log(\overline{s_m^2})|\overline{x_m^2}] \simeq \frac{1}{\pi}\int_{-\pi}^{\pi} \log[\sqrt{\frac{(\bar{A})^2 \cdot \cos^2(\phi)}{\bar{B}} + 1} - \frac{\bar{A}}{\sqrt{\bar{B}}} \cdot \cos(\phi)]d\phi + \log(\bar{B}) \qquad (5.23)$$

where $\bar{A} = \sqrt{E[\overline{n_m^2}] \cdot c_m}$ and $\bar{B} = \overline{x_m^2} - E[\overline{n_m^2}]$. Splitting domain of integration into $[-\pi, 0]$ and $[0, \pi]$, replacing the variable ϕ with $u = -\frac{\bar{A}}{\sqrt{\bar{B}}} \cdot \cos(\phi)$ and noting symmetry, the integral in (5.23) becomes

$$\frac{1}{\pi}\int_{-\pi}^{\pi} \log[\sqrt{\frac{(\bar{A})^2 \cdot \cos^2(\phi)}{\bar{B}} + 1} - \frac{\bar{A}}{\sqrt{\bar{B}}} \cdot \cos(\phi)]d\phi = \tag{5.24}$$

$$\frac{2 \cdot \sqrt{\bar{B}}}{\bar{A} \cdot \ln(10) \cdot \pi} \int_{-\frac{\bar{A}}{\sqrt{\bar{B}}}}^{\frac{\bar{A}}{\sqrt{\bar{B}}}} \frac{\sinh^{-1}(u)}{\sqrt{1 - \frac{\bar{B}}{(\bar{A})^2} \cdot u^2}} du = 0$$

because the functions $\sinh^{-1}(u)$ and $\sqrt{1 - \frac{\bar{B}}{(\bar{A})^2} \cdot u^2}$ are odd and even respectively. Consequently,

$$E[\log(\overline{s_m^2})|\overline{x_m^2}] \simeq \log(\overline{x_m^2} - E[\overline{n_m^2}]) \tag{5.25}$$

This result means, according to the model for additive noise, that the expected value of the hidden information $\log(\overline{s_m^2})$ is equal to the log of the SS estimation $Est[\overline{s_m^2}]$ if SS is defined as being

$$Est[\overline{s_m^2}] = \overline{x_m^2} - E[\overline{n_m^2}]$$

This result will also be used to cancel convolutional noise as discussed in Chapter 7.

Due to the fact that $\overline{2s_m\ n_m} = 0$ may not be true in a short term analysis and that the noise energy presents fluctuations, $Est(\overline{s_m^2})$ may be negative in those channels with low SNR. In order to avoid negative magnitude estimates a rectifying function r()is applied:

$$r(Est(\overline{s_m^2}), SsThr_m) = \begin{cases} Est(\overline{s_m^2}) & \text{if } Est(\overline{s_m^2}) \geq SsThr_m \\ SsThr_m & \text{if } Est(\overline{s_m^2}) < SsThr_m \end{cases} \tag{5.26}$$

where $SsThr_m$ could be estimated according to (Compernolle, 1989):

$$10 \cdot \log(SsThr_m) = CPE_m - DYN_m \tag{5.27}$$

where CPE_m is the Channel Peak Energy and DYN_m is the dynamic range of the speech signal at channel m in quiet environment. CPE is defined as being the $3 \cdot 5$ upper percentile on the speech distribution of the channel m histogram.

5.6 Weighted Matching Algorithms

Some modifications were included in matching algorithms in order to weight the reliability of the information extracted from testing frames. A weighting coefficient $w(t)$ ($w(t) = 1$, maximum reliability; $w(t) = 0$, minimum reliability) is associated to each testing frame in order to be employed in the modified vesions of the DTW and Viterbi (HMM) algorithms (N.B.Yoma *et al.*, 1995) (N.B.Yoma *et al.*, 1996a). The main idea behind the modifications made on Viterbi (HMM) and DTW algorithms is that the influence of a frame on decisions must be proportional to its coefficient $w(t)$. The proposed one-step weighted DP algorithm was compared with the two-step DP algorithm proposed in (H.Kobatake & Y.Matsunoo, 1994). This chapter concerns experiments with weighted DTW algorithms only. The equivalent version for the Viterbi algorithm (HMM) is tested in Chapter 6.

The proposed one-step DP equation that corresponds to the local condition shown in Fig.2.3 in page 10 is given as follows :

$$G(t,r) = \min \left(\begin{array}{c} \frac{G(t-2,r-1)W(t-2,r-1)+2w(t-1)d(t-1,r)+w(t)d(t,r)}{W(t-2,r-1)+2w(t-1)+w(t)} \\ \\ \frac{G(t-1,r-1)W(t-1,r-1)+2w(t)d(t,r)}{W(t-1,r-1)+2w(t)} \\ \\ \frac{G(t-1,r-2)W(t-1,r-2)+2w(t)d(t,r-1)+w(t)d(t,r)}{W(t-1,r-2)+3w(t)} \end{array} \right)$$

and

$$W(t,r) = \begin{cases} W(t-2,r-1) + 2w(t-1) + w(t) \\ W(t-1,r-1) + 2w(t) \\ W(t-1,r-2) + 3w(t) \end{cases}$$

The local condition shown in Fig.2.2, which the DP equation in section 3.5.1 is based on, does not include any temporal restriction concerning the length of the reference and testing utterances. In contrast, the local constraints represented by Fig.2.3, which is used for the above weighted

DP equation, imposes that the lower and upper bounds of the reference utterance length must be, respectively, half and twice the testing utterance size.

This DP equation takes into account the weight $w(t)$ frame by frame, and the calculation of the overall distance, $G(t,r)$, is affected by the local distance $d(t,r)$ according to $w(t)$: if $w(t) = 1$ (high reliability or local SNR), the weight of $d(t,r)$ is maximum; if $w(t) = 0$ (very low reliability or local SNR), the importance of $d(t,r)$ is zero.

5.6.1 Two-step DP matching

As discussed in Chapter 3, the algorithm proposed in (H.Kobatake & Y.Matsunoo, 1994) consists of the following two-step processing. Firstly, the optimal alignment path $c_k = (t_k, r_k), k = 1, 2, ..., K$ is obtained using the ordinary DP matching algorithm, where t_k and r_k are the frame numbers of the testing and reference patterns respectively. The second step is the calculation of the global distance between the utterances weighted by $w(t_k)$ along the optimal path obtained at the first step.

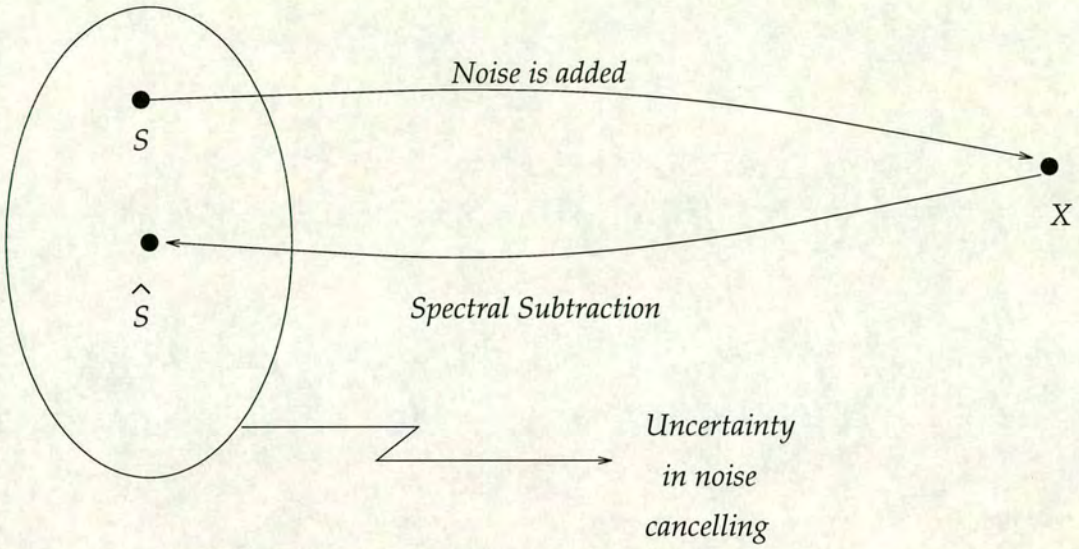
5.7 Reliability in noise cancelling

The additive noise models given in sections 5.2 and 5.3 set the hidden clean signal $\overline{s_m^2}$ as a function of ϕ, $\overline{x_m^2}$ and $\overline{n_m^2}$. In other words, for every pair $\overline{x_m^2}$ and $\overline{n_m^2}$ there is a set of possible $\overline{s_m^2}$ (Fig.5.2) whose variance can be estimated by means of (5.18). When the noise is added an uncertainty is introduced (due to the lack of knowledge about the phase difference between the clean and noise signals) and the original clean signal energy cannot be recovered with 100 % accuracy. The distortion $d(\phi, \overline{x_m^2}, \overline{n_m^2})$ related to the estimation of $\overline{s_m^2}$ can be set as:

$$d(\phi, \overline{x_m^2}, \overline{n_m^2}) = \sum_{m=1}^{14} \left(\log(\overline{s_m^2}(\phi, \overline{x_m^2}, \overline{n_m^2})) - E[\log(\overline{s_m^2}|\overline{x_j^2})] \right)^2 \tag{5.28}$$

where ϕ_m denotes the phase difference in filter m and, as shown in section 5.5, $E[\log(\overline{s_m^2})|\overline{x_m^2}]$ is approximately equal to $\log(\overline{x_m^2} - E[\overline{n_m^2}])$. Consequently, the expected value of $d(\phi, \overline{x_m^2}, \overline{n_m^2})$ is given by:

$$E[d(\phi, \overline{x_m^2}, \overline{n_m^2})] = \sum_{m=1}^{14} Var[\log(\overline{s_m^2})|\overline{x_m^2}] \tag{5.29}$$



S, clean speech frame;

X, noisy signal frame;

$\hat{S}$, *estimation of the clean signal.*

Figure 5.2: Interpretation of reliability in noise cancelling.

The expected value of $d(\phi, \overline{x_m^2}, \overline{n_m^2})$ is a measure of the uncertainty about the clean signal information and it is reasonable to suppose that the reliability related to SS in a channel would be inversely proportional to $E[d(\phi, \overline{x_m^2}, \overline{n_m^2})]$. This is coherent with the result shown in Chapter 3 where, in the context of the noise cancelling neural net LIN, the reliability in noise cancelling should also be inversely proportional to the mean distortion at a given segmental SNR.

However, the inverse of the variance weighting presents some problems: firstly, it goes very high when the noise is low (theoretically, it goes to infinite if there is no noise); and secondly, it is observed that the recognition error remains low, even zero, for high SNR's and starts increasing after the noise reaching a given level. In order to counteract these deficiences, the weighting coefficient w, to be used by the weighted algorithms DP (N.B.Yoma *et al.*, 1996a) (N.B.Yoma *et al.*, 1997b) and that attempts to measure how reliable is the result of the noise cancelling

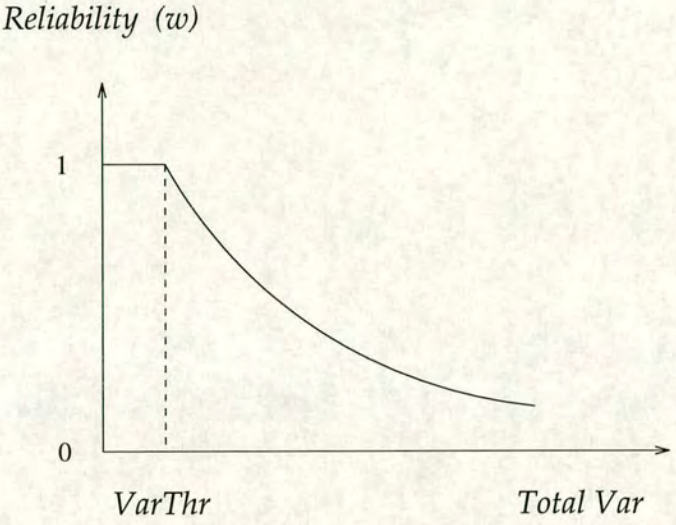


Figure 5.3: Reliability coefficient vs variance.

method in a frame, was defined as (Fig. 1):

$$w = \begin{cases} 1 & \text{if } \mathtt{TotalVar} \leq \mathtt{VarThr} \\ \frac{\mathtt{VarThr}}{\mathtt{TotalVar}} & \text{if } \mathtt{TotalVar} > \mathtt{VarThr} \end{cases} \tag{5.30}$$

where

$$\mathtt{TotalVar} = \sum_{m=1}^{14} 100 \cdot \mathrm{Var}[\log(\overline{s_m^2})|\overline{x_m^2}] \tag{5.31}$$

and $\mathrm{E}[d(\phi, \overline{x_m^2}, \overline{n_m^2})]$ was replaced with TotalVar to be coherent with the papers published about this research; the constant 100 is due to the fact that the log energies are considered in dB.

5.8 Inverse of the channel variance weighting

In the last section it was discussed that reliability in noise cancelling by SS, defined as being the inverse of the sum of the channel variance, could be used to weight the recognition algorithm, and a threshold VarThr was introduced to take into consideration the fact that the information provided by a noisy frame above a given segmental SNR is as reliable as the one provided by a clean frame. It was observed that, for the task considered (digits), the recognition error rate remains low at high global SNR's (18 or 12 dB) and starts to increase more abruptly at SNR=6dB which indicates that most frames should give a highly reliable information. Consequently, frames

with local SNR around 18 or 12 dB should have weight equal to 1 in the recognition algorithm and an approximate estimation for VarThr could be done by means of (5.18) considering that the noisy signal energy $\overline{x_m^2}$ is 9 dB (between 12 and 6 dB) above the noise power estimation $E[\overline{n_m^2}]$ in all the channels:

$$VarThr = \sum_{m=1}^{14} Var[\log(\overline{s_m^2})|\overline{x_m^2} = E[\overline{n_m^2}] + 9dB] \tag{5.32}$$

which gave 8.5 and 9.1 for the car and speech noises, respectively.

Nevertheless, according to some preliminary experiments, the optimal VarThr resulted much higher than the one predicted by (5.32) and a modification had to be included in order to improve the results and bring the optimal VarThr closer to 10. The SS threshold $SsThr_m$ was kept constant at all the global SNR's and (5.21) indicates that the maximum $Var[\log(\overline{s_m^2})|\overline{x_m^2}]$ should be inversely proportional to $SsThr/E[\overline{n_m^2}]$ and if $SsThr_m$ is low when compared to $E[\overline{n_m^2}]$, $Var[\log(\overline{s_m^2})|\overline{x_m^2}]$ may be too high at low segmental SNR's (when the model loses accuracy). This is counteracted by setting an upper bound to the channel variance that was initially made equal to $Var[\log(\overline{s_m^2})]$, which is estimated on the clean signal. This means that the highest uncertainty about the hidden clean signal information should not be higher than the variance of $\log(\overline{s_m^2})$. In fact, using $Var[\log(\overline{s_m^2})]$ as an upper threshold for $Var[\log(\overline{s_m^2})|\overline{x_m^2}]$ improved the results and made the error rate at VarThr =10 closer to the optimum one in many cases.

At global SNR's equal to 18, 12 and sometimes at 6 dB, frames showing one or more channel energies equal to SsThr coexist with high local SNR frames whose channels gave SS estimation well above SsThr. However, if only one filter presents the estimated energy as being SsThr it does not mean that the rest of the channels give a poorly reliable information, and if MaxVar is too high those frames with only one filter energy equal to SsThr may have a too low weight in the recognition algorithm. This is counteracted by increasing the optimal VarThr whose optimal value resulted higher than 10 in some cases. In contrast, at global SNR=18dB almost every frame presents at least one channel whose estimated energy is below than SsThr which makes MaxVar less important.

5.8.1 Maximum distortion

In this section, the additive noise model is used to estimate the uncertainty or mean distortion when the difference between the noisy signal energy and the noise estimation is less than $SsThr_m$. The variance $Var[\log(\overline{s_m^2})|\overline{x_m^2}]$ according to (5.28) could be interpreted as being the average distortion in the log domain between the possible clean signal energy $\overline{s_m^2}(\phi, \overline{n_m^2}, \overline{x_m^2})$ and the expected value of this energy given the observed noisy energy (section 5.4). Using this interpretation, this distortion is given by

$$d(\phi, \overline{n_m^2}, \overline{x_m^2}) \mid_{(\overline{x_m^2} - E[\overline{n_m^2}]) < SsThr} = \left(\log[\overline{s_m^2}(\phi, \overline{n_m^2}, \overline{x_m^2})] \mid_{(\overline{x_m^2} - E[\overline{n_m^2}]) < SsThr} - \log(SsThr_m)\right)^2 \quad (5.33)$$

Assuming that the clean signal energy is uniformly distributed between $\log(\overline{s_m^2}(\phi, \overline{n_m^2}, \overline{x_m^2}) \mid_{(\overline{x_m^2} - E[\overline{n_m^2}]) = SsThr}$ and $\log(min(\overline{s_m^2}))$, where $min(\overline{s_m^2})$ is the minimum energy of the speech signal, the mean distortion is given by

$$E[d(\phi, \overline{n_m^2}, \overline{x_m^2}) \mid_{(\overline{x_m^2} - E[\overline{n_m^2}]) < SsThr}] = \frac{1}{3} \cdot Var[\log(\overline{s_m^2})|\overline{x_m^2}] + \frac{100}{3} \cdot \log^2(SsThr) + \frac{100}{3} \cdot \log^2(min(\overline{s_m^2})) - \frac{200}{3} \log(SsThr) \cdot \log(min(\overline{s_m^2})) \quad (5.34)$$

where, as in the section 5.5, it is considered that the random variables ϕ and $\overline{n_m^2}$ are uncorrelated, ϕ is uniformly distributed between $-\pi$ and π and that $\overline{n_m^2}$ is concentrated near its mean $E[\overline{n_m^2}]$. This expected $E[d(\phi, \overline{n_m^2}, \overline{x_m^2}) \mid_{(\overline{x_m^2} - E[\overline{n_m^2}]) < SsThr_m}]$ is a measure of the uncertainty about the clean signal energy when SS estimation is equal to $SsThr_m$ and was used as an upper bound for $Var[\log(\overline{s_m^2})|\overline{x_m^2}]$.

5.9 Experiments

The proposed methods were tested using an IIR and a DFT filter bank with speaker-dependent isolated word (English digits from 0 to 9) recognition experiments in slightly different contexts. The tests were carried out employing the two speakers (one female and one male) from the Noisex database (A.Varga *et al.*, 1992).

The signals were low pass filtered by using a filter with cut off frequency 3700 Hz, down

sampled from 16000 to 8000 samples/sec, and high-pass filtered by employing a filter with cut off frequency 120 Hz. The data signal was divided in 25ms frames with 12.5ms overlapping. Each frame was processed with a Hamming window before the spectral estimation. The band from 300 to 3400 Hz was covered with 14 Mel 2nd order IIR or DFT filters. At the output of each channel the energy was computed, SS and the log function were applied. Finally, 10 cepstral coefficients were computed.

In the Noisex database, all the 100 testing utterances for a given speaker-noise-SNR configuration are in sequence and separated by approximatly 1 sec in a single file. In this chapter, the noise energy was estimated only once using 200 ms of non-speech samples at the beginning of the utterance sequences and the noise estimation was kept constant for all the utterances of a given speaker-noise-SNR.

The experiments were done firstly by means of DTW algorithms instead of HMM because DTW allows the direct comparison of two utterances and the training procedure consists just in choosing the reference patterns. In this sense, due to the fact that one reference pattern corresponds to one utterance, 10 reference sets were used and the number of recognition experiments was multiplied by 10. In contrast, HMM is a stochastic system and is very dependent on the size of the training database and experiments with the Noisex database generally use only one set of HMM's trained by means of all the training utterances. Consequently, the DTW systems allow more precise comparisons because the results were achieved with 1000 recognition tests for each SNR instead of only 100.

The following configurations were tested: the ordinary DTW algorithm (H.Sakoe & S.Chiba, 1978) without (DTW) and with SS ($DTW - SS$); the proposed one-step weighted DP algorithm (N.B.Yoma *et al.*, 1995) with SS ($1SW - SS$); the two-step DP matching (H.Kobatake & Y.Matsunoo, 1994) ($2SW - SS$) also with SS; and finally, the proposed one-step DP algorithm with SS but without reliability in noise cancelling weighting, $w(t) = 1$ ($1S - SS$).

5.9.1 Experiments with IIR filter bank

For historical reasons, the reliability in noise cancelling weighting was initially tested in the context of IIR filter bank (section 5.2) and with isolated word automatically end detected using an algorithm based on autoregressive analysis of noise (Chapter 4) (N.B.Yoma *et al.*, 1996b). As

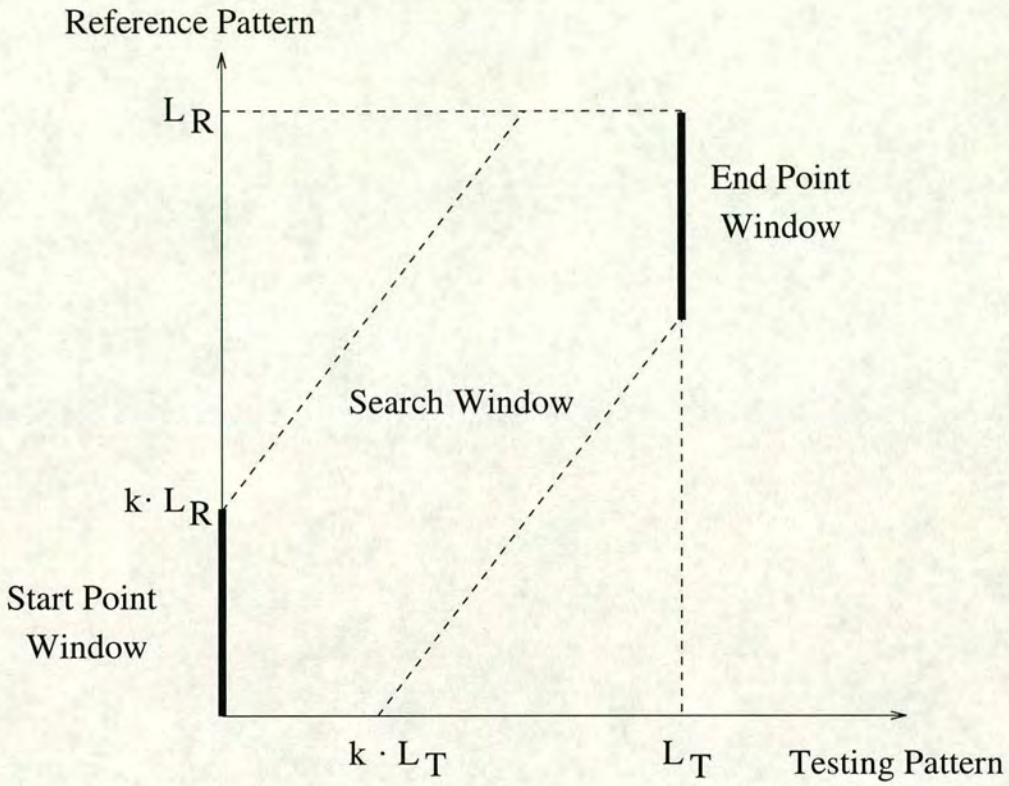


Figure 5.4: End-point constraints relaxation.

a result of the automatic end-point detection the average length of the testing utterances decreases as the SNR gets more severe. In order to counteract this effect, the endpoint constraints on the DP algorithms were relaxed by means of opening up the ends of the search region allowing the alignment path to start by comparing the first frame of the testing pattern with any of the first reference frames inside the search window, and to end by comparing the last test frame with any of the last reference frames inside the search window (see Fig.5.4). Due to the fact that the length of the testing utterances presented a high variation, the sides of the search window were made proportional to the utterance length.

In these first experiments no attempt was made to estimate a suitable value for SsThr under the assumption that the weighting procedure should make SS less dependent on this threshold due to the fact that frames with low segmental SNR should have low weight in the recognition algorithm. For each configuration several search window widths k (Fig.5.4) were tested and the one that gave minimum error rate was chosen to achieve the results shown in Tables 5.1 and 5.2 where SsThr was made equal to 0.05 and VarThr equal to 600. According to section 5.8,

Table 5.1: Recognition error rate (%) for speech signal corrupted by car noise. The recognition experiments were done by the IIR filter bank and *VarThr* was made equal to 600. For every configuration, the search window width *k* that gave the minimum error rates was chosen.

SNR	*18dB*	*12dB*	*6dB*	*0dB*
DTW	1.0	5.0	21.6	44.9
DTW-SS	0.5	1.3	6.0	21.0
1SW-SS	0.0	0.1	0.8	5.3
2SW-SS	0.1	0.4	3.4	14.7
1S-SS	0.0	0.1	1.9	8.9

Table 5.2: Recognition error rate (%) for speech signal corrupted by speech noise. The recognition experiments were done by the IIR filter bank and *VarThr* was made equal to 600. For every configuration, the search window width *k* that gave the minimum error rates was chosen.

SNR	*18dB*	*12dB*	*6dB*	*0dB*
DTW	0.8	6.5	31.3	61.0
DTW-SS	2.9	6.7	18.4	54.4
1SW-SS	0.0	0.3	1.7	16.6
2SW-SS	0.3	2.6	8.9	37.9
1S-SS	0.3	1.2	7.1	31.0

VarThr should be around 10 but, as can be seen in Fig. 5.5, this was not observed at SNR=6 and 0dB. This must have been the result of the low SsThr_m. The higher $\overline{x_m^2}$ is when compared to $E[\overline{n_m^2}]$, the better this approximation is. However, when $\overline{x_m^2}$ gets closer to $E[\overline{n_m^2}]$, the model loses accuracy and $\mathrm{Var}[\log(\overline{s_m^2})|\overline{x_m^2}]$ assumes incorrect high values when SsThr is low when compared to $E[\overline{n_m^2}]$. This is compensated by means of increasing VarThr. As can be seen in Tables 5.1 and 5.2, the one step algorithm in combination with the noise cancellation reliability weighting gave the lowest error rate. This reduction in the error rate was due to a) the ability of the one step algorithm in normalising the overall distance to the length of the alignment path, and b) the information provided by the noise cancellation reliability coefficient. The ordinary DTW does not take into consideration which point of the start window the optimal alignment path begins and was very sensitive to the search window width. Consequently, when k was increased (Fig. 5.4), DTW-SS and 2SW-SS increased the error rate after reaching an optimum search window. On the other hand, the DP equation shown in section 5.6 computes the overall

weight $W(i,j)$ step-by-step and was almost independent to the alignment path length. As a result, 1SW-SS should be compared with 1S-SS in order to separate the improvement due to the alignment path normalisation and the one due to the noise cancelling reliability weighting.

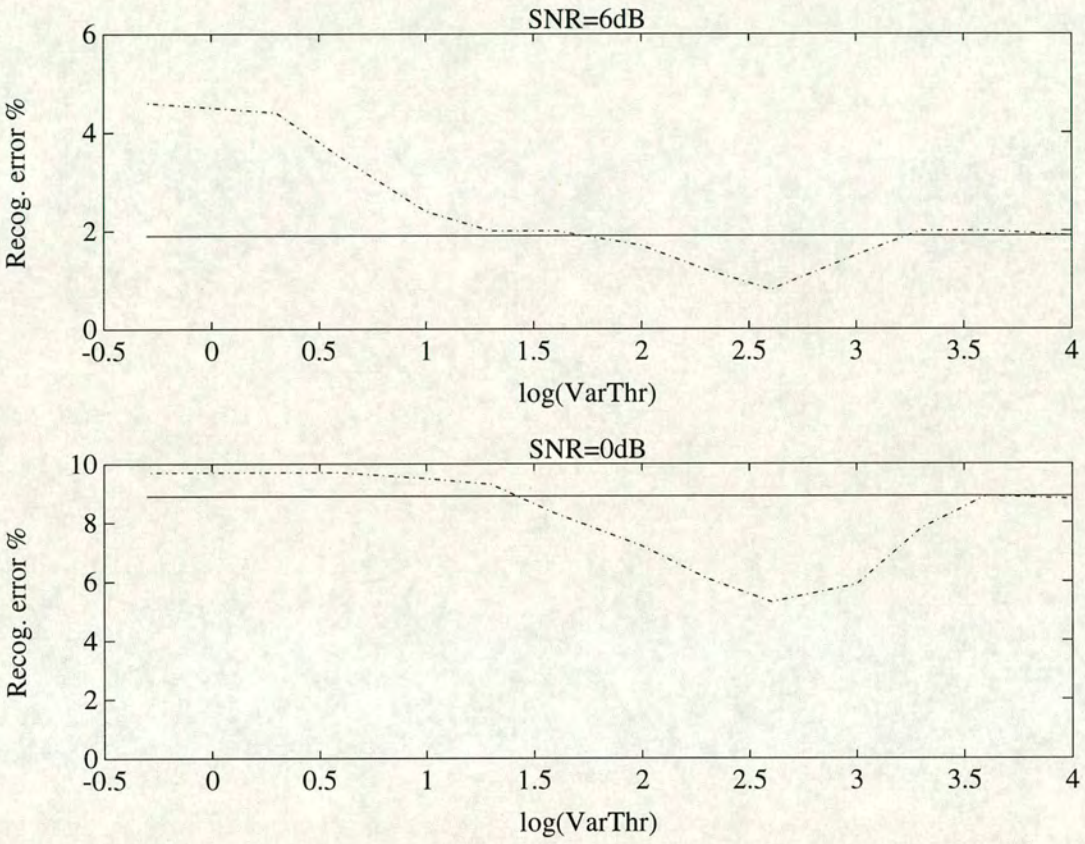


Figure 5.5: Recognition error rate vs *VarThr* for the car noise at global SNR=6 and 0dB. The experiments were done with the IIR filter bank, SS and the one-step weighted algorithm: (-), without weighting *1S-SS*; and (-.), with weighting *1SW-SS*. The threshold *SsThr* was made equal to 0.05. *log()* denotes logarithm to base 10.

When compared with 1S-SS, 1SW-SS showed reductions of 58% and 40% in the error rate at SNR=6dB and 0dB for the car noise. At SNR=18dB and 12dB both configurations gave error rate equal to 0 and 0.1%, repectively. For the speech noise, 1SW-SS presented reductions of 75%, 76% and 46% at SNR=12, 6 and 0dB. At SNR=18dB the error rate went from 0.3% to 0. As can be seen, the improvement due to the reliability weighting was higher for the speech noise than for the car one. This must result from the facts that the speech noise is less stationary than the car noise so the estimation of noise energy is less accurate, and that the reliability coefficient is also a function of the local SNR so low energy intervals have low weight in the pattern matching process. Therefore, noise cancellation reliability weighting made the SS process more robust to variations in the noise stationarity.

To conclude, these first experiments, done using IIR Mel filter bank, were enough to show that weighting the information along the signal as described in this chapter can substantially improve the performance of SS. However, the estimation of SsThr, which is related to the maximum $Var[\log(\overline{s_m^2})|\overline{x_m^2}]$, needs further discussion since the dependence of the error rate on the parameter VarThr was not the expected one at SNR=6 and 0 dB (Fig. 5.5) and the optimal VarThr was higher than the one predicted in section 5.8.

5.9.2 Experiments with DFT filter bank

The experiments using the additive noise model for DFT filters (section 5.3) were done without automatic end point detection in order to eliminate any effect introduced by the discriminative selection of speech intervals with higher energies. When the utterances were automatically end detected, the one step algorithm gave lower error rates than the two-step one (section 5.9.1). The two-step algorithm is very sensitive to the search window width, and in order to better compare both algorithms and not to take into consideration the ability of the one step algorithm in normalising the overall distance to the length of the alignment path, the automatic end point detection was removed.

As explained in section 5.8, the arbitrary low value for SsThr resulted in a too high maximum $Var[\log(\overline{s_m^2})|\overline{x_m^2}]$, and in a very discernible and high optimal VarThr. According to section 5.8 the optimum VarThr should be equal to 5 or 10. As can be seen in (5.21), $Var[\log(\overline{s_m^2})|\overline{x_m^2}]$ is inversely proportional to $\overline{x_m^2} - E[\overline{n_m^2}]$ and the experiments of this section were done by

means of estimating SsThr according to (5.27). CPE was defined as the 3% upper percentile on $\log(\overline{s_m^2})$, which was supposed to have a Gaussian distribution. $E[\log(\overline{s_m^2})]$ and $Var[\log(\overline{s_m^2})]$ were computed on the training clean utterances for every speaker, and DYN was considered equal to 50 dB, a value used in (Compernolle, 1989) and that was also verified as being valid according to measures on the Noisex database. Using (5.27), SsThr resulted around 20 dB for all the channels m. Additionally, in order to avoid too high $Var[\log(\overline{s_m^2})|\overline{x_m^2}]$, the result given by (5.18) was bounded above by (5.34). Results are presented in Figs. 5.6- 5.11 and in Tables 5.3-5.7 for the car, speech, Lynx, operation room and factory noises of the Noisex database.

Figs. 5.6-5.11 present the error rate vs VarThr at SNR=18, 12, 6 and 0dB for the five noises considered. When compared to Fig. 5.5, Fig. 5.6 shows that estimating SsThr according to (Compernolle, 1989) and setting an upper limit to $Var[\log(\overline{s_m^2})|\overline{x_m^2}]$ seem to improve the results and reduce the dependence of the weighting algorithm on VarThr. At VarThr =10, log(VarThr) =1, the error rate is near the lowest one at SNR=12, 6 and 0 dB, and there is a wide range of sub-optimal values for VarThr. Something similar is observed in Figs. 5.7, 5.8 and 5.9, but not in Fig. 5.10, which indicates that the weighting procedure requires further revision.

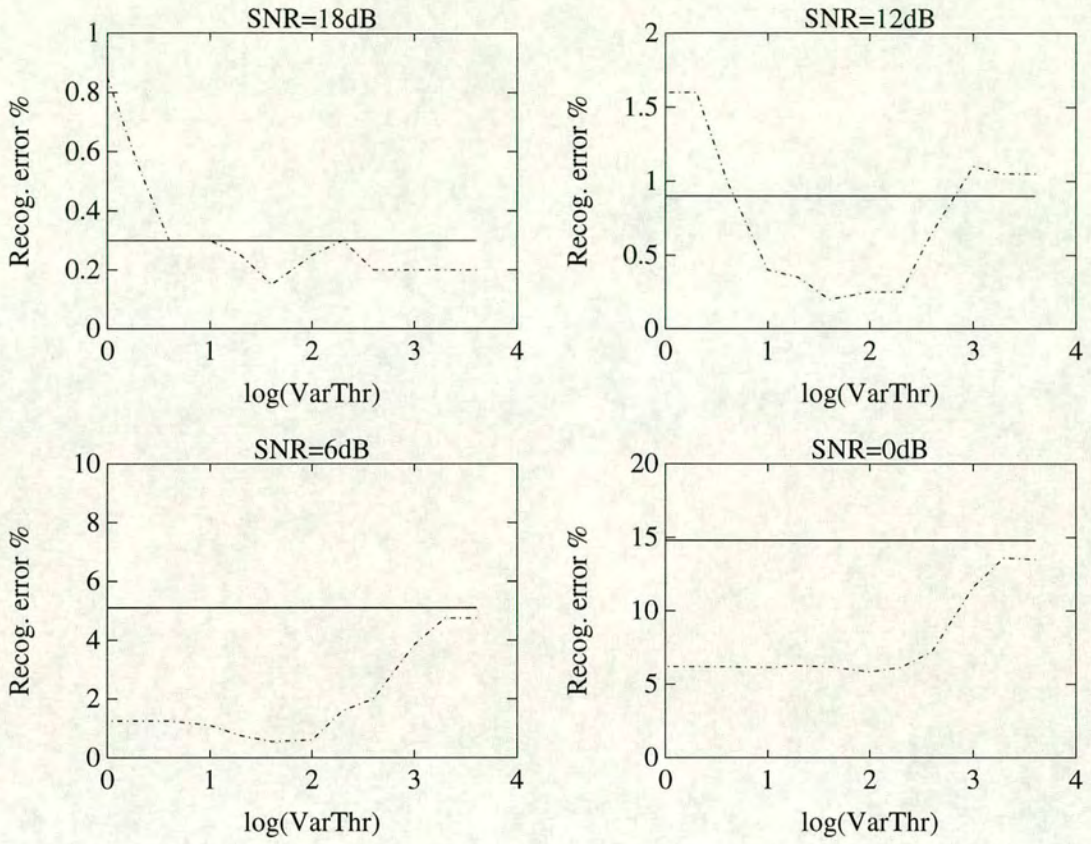


Figure 5.6: Recognition error rate vs *VarThr* for the car noise at global SNR=18, 12, 6 and 0dB. The experiments were done with the DFT filter bank and SS: (-), without weighting *DTW-SS*; and (-.), with weighting *1SW-SS*. *log()* denotes logarithm to base 10.

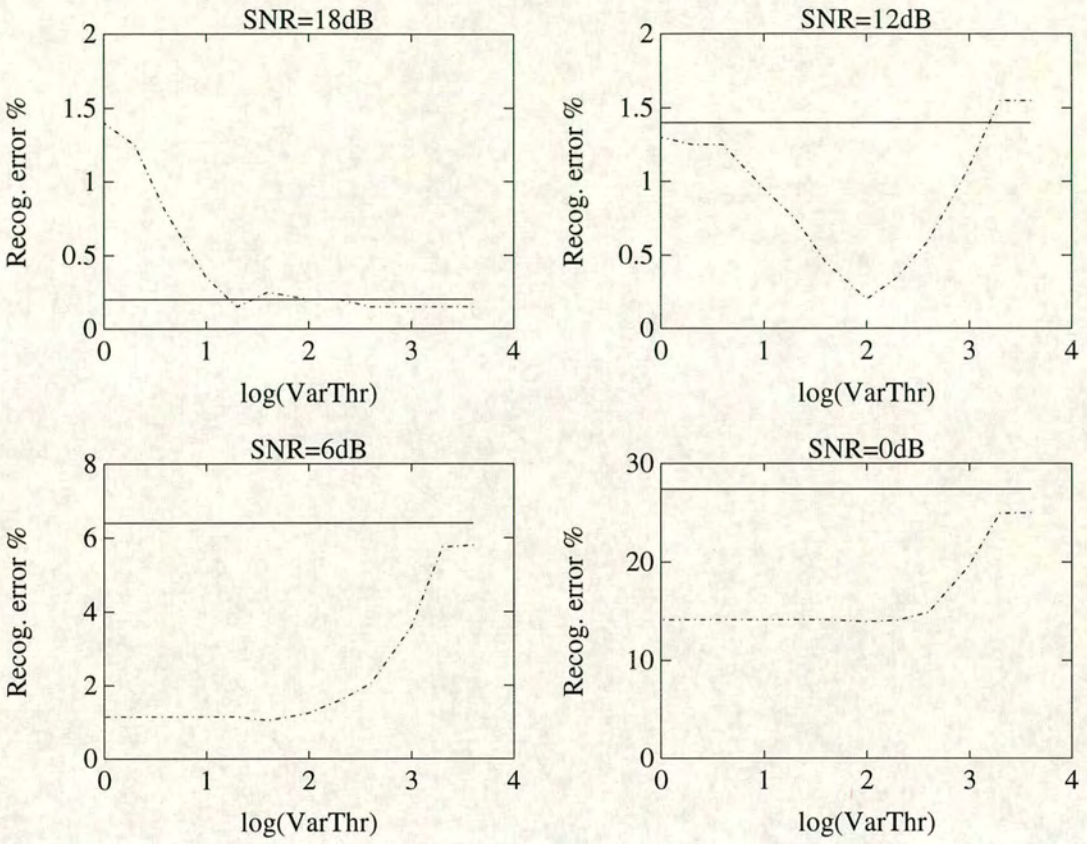


Figure 5.7: Recognition error rate vs *VarThr* for the speech noise at global SNR=18, 12, 6 and 0dB. The experiments were done with the DFT filter bank and SS: (-), without weighting *DTW-SS*; and (-.), with weighting *1SW-SS*. *log()* denotes logarithm to base 10.

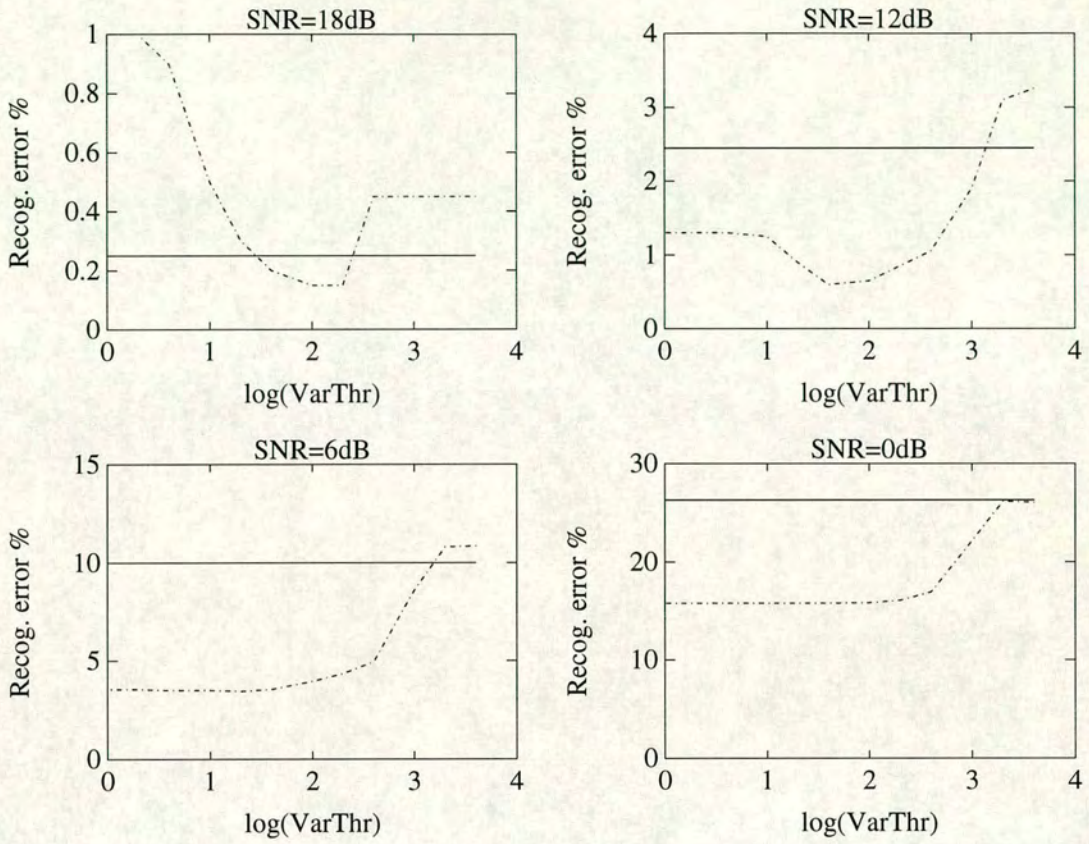


Figure 5.8: Recognition error rate vs *VarThr* for the Lynx noise at global SNR=18, 12, 6 and 0dB. The experiments were done with the DFT filter bank and SS: (-), without weighting *DTW-SS*; and (-.), with weighting *1SW-SS*. *log()* denotes logarithm to base 10.

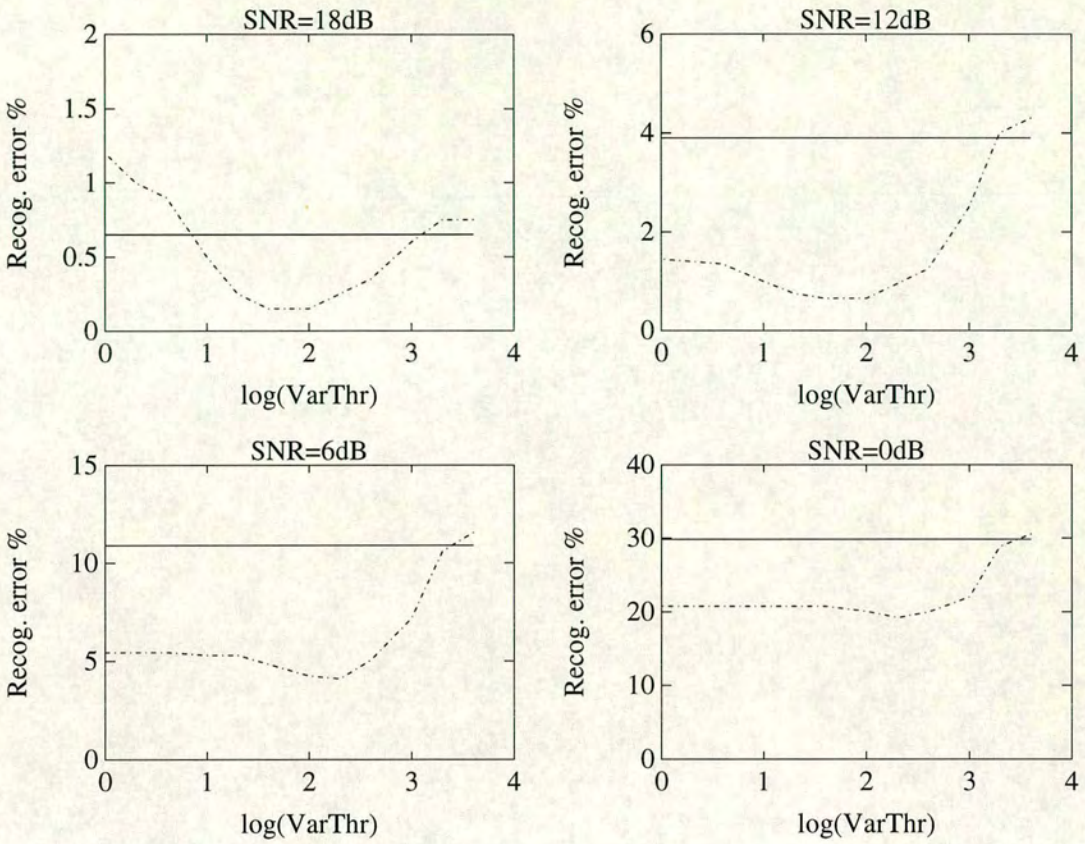


Figure 5.9: Recognition error rate vs *VarThr* for the operation room noise at global SNR=18, 12, 6 and 0dB. The experiments were done with the DFT filter bank and SS: (-), without weighting *DTW-SS*; and (-.), with weighting *1SW-SS*. *log()* denotes logarithm to base 10.

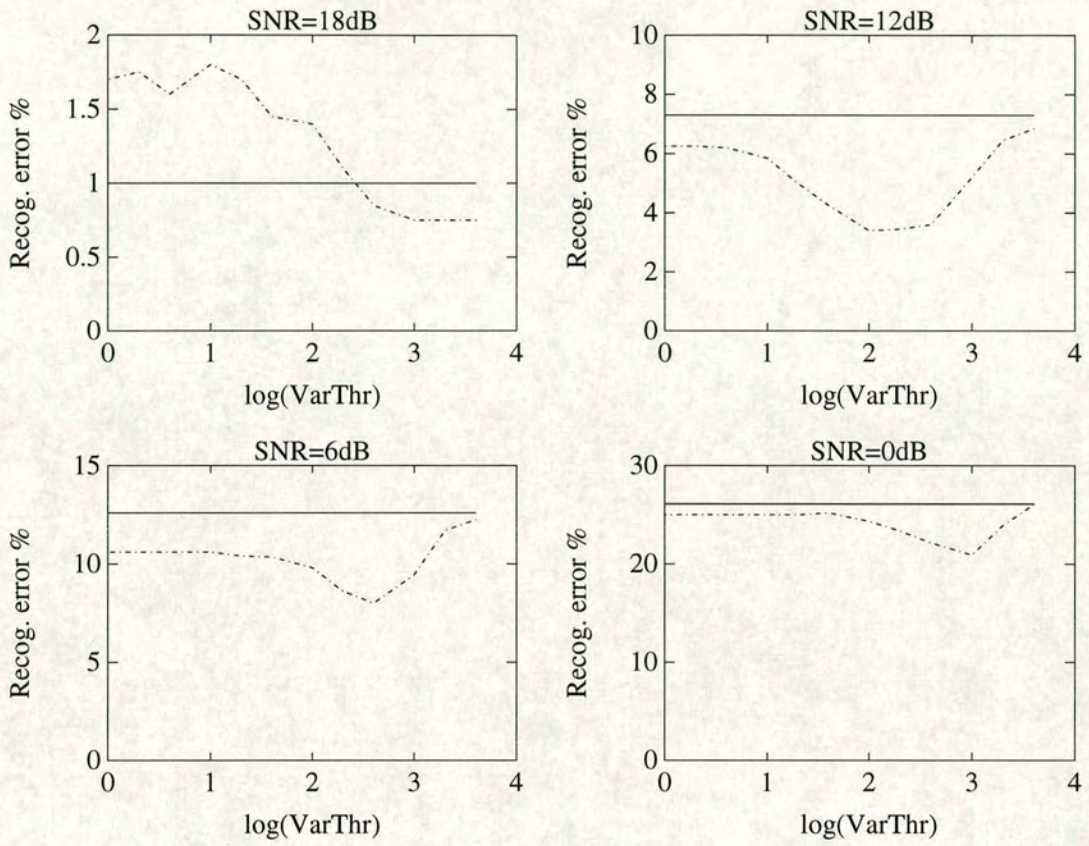


Figure 5.10: Recognition error rate vs *VarThr* for the factory noise at global SNR=18, 12, 6 and 0dB. The experiments were done with the DFT filter bank and SS: (-), without weighting *DTW-SS*; and (-.), with weighting *1SW-SS*. *log()* denotes logarithm to base 10.

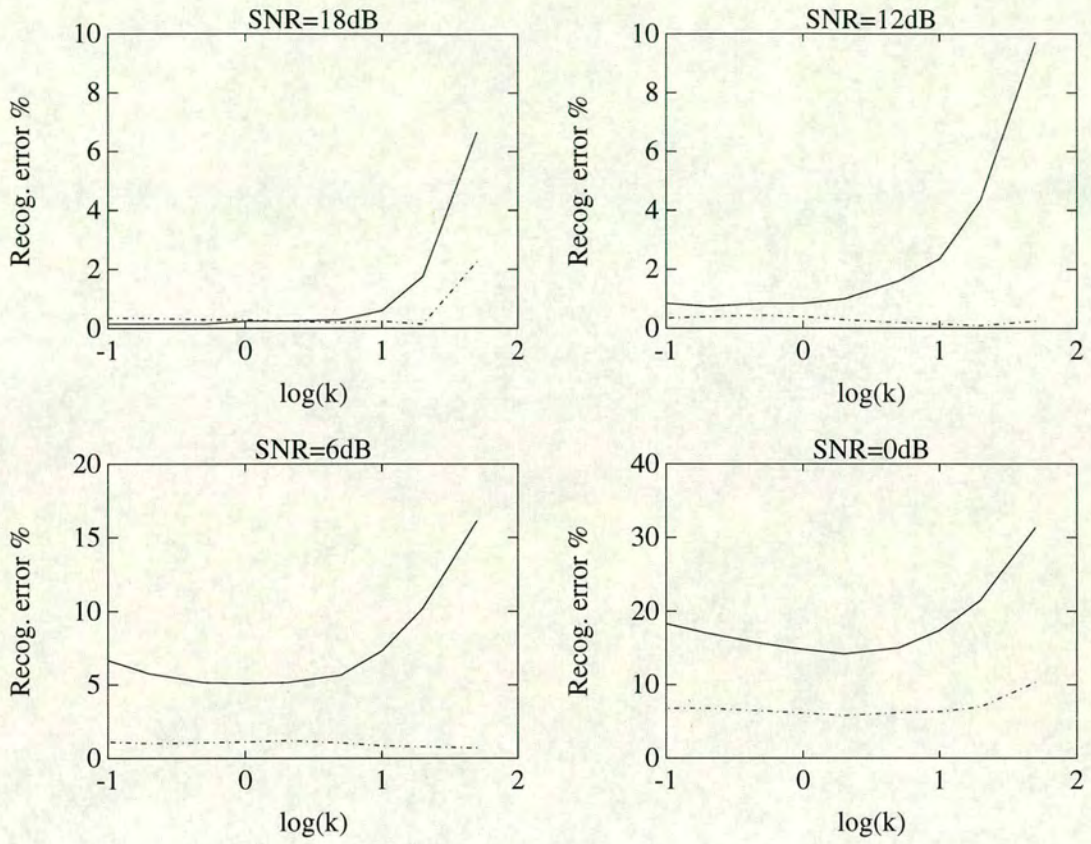


Figure 5.11: Study of sensitivity of SS to the threshold *SsThr* by means of multiplying the threshold used in Figs.5.7-5.11 by *k*. Experiments were done with the car noise at global SNR=18, 12, 6 and 0dB, using SS and DTW algorithms: (-), ordinary DTW; and (-.), weighted DP equation shown in section 5.6. The parameter *VarThr* was the same used in Tables 1-4. *log()* denotes logarithm to base 10.

Table 5.3: Recognition error rate (%) for speech signal corrupted by additive noise (car). The threshold *VarThr* was made equal to 10.

SNR	*18dB*	*12dB*	*6dB*	*0dB*
DTW-SS	0.3	0.9	5.1	14.8
1SW-SS	0.3	0.4	1.1	6.2
2SW-SS	0.1	0.7	2.1	6.7

Table 5.4: Recognition error rate (%) for speech signal corrupted by additive noise (speech). The threshold *VarThr* was made equal to 10.

SNR	*18dB*	*12dB*	*6dB*	*0dB*
DTW-SS	0.2	1.4	6.4	27.4
1SW-SS	0.4	1.0	1.2	14.2
2SW-SS	0.2	1.8	2.9	14.0

In order to study the influence on the accuracy of the determination of the SS threshold, SsThr was multiplied by a constant k. Fig. 5.11 presents the error rate vs k at SNR=18, 12, 6 and 0 dB for the car noise. As can be seen, the weighted DP algorithm strongly reduced the dependence of SS on the threshold SS, estimated according to (Compernolle, 1989), with better results.

As can be seen from tables 5.3-5.7, weighting the information along the signal substantially increased the performance of SS, an easily implemented technique, even with a poor estimation for the noise and without using any information about the speaker. The improvement depended on the noise and SNR and all the results with the weighted algorithms, $1SW-SS$ and $2SW-SS$, correspond to VarThr equal to 10, which is not necessarily the optimum one according to Figs. 5.6-5.10. As far the weighted algorithms are concerned, the one-step DP ($1SW-SS$) algorithm performed better that the two-step DTW ($2SW-SS$). This must be due to the fact that the

Table 5.5: Recognition error rate (%) for speech signal corrupted by additive noise (Lynx). The threshold *VarThr* was made equal to 10.

SNR	*18dB*	*12dB*	*6dB*	*0dB*
DTW-SS	0.3	2.5	10.0	26.3
1SW-SS	0.5	1.3	3.5	15.8
2SW-SS	0.4	2.5	4.9	16.2

Table 5.6: Recognition error rate (%) for speech signal corrupted by additive noise (operation room). The threshold *VarThr* was made equal to 10.

SNR	*18dB*	*12dB*	*6dB*	*0dB*
DTW-SS	0.7	3.9	10.9	29.9
1SW-SS	0.5	1.0	5.3	20.75
2SW-SS	0.6	2.5	6.8	21.3

Table 5.7: Recognition error rate (%) for speech signal corrupted by additive noise (factory). The threshold *VarThr* was made equal to 10.

SNR	*18dB*	*12dB*	*6dB*	*0dB*
DTW-SS	1.4	8.1	15.6	29.3
1SW-SS	1.6	3.7	9.7	25.9
2SW-SS	1.5	6.1	11.9	25.8

weighting coefficient is used to detemine the alignment in the one-step technique but is not in the two-step DTW.

5.10 Discussion and conclusion

The interaction between clean signal speech and additive noise was modelled in the context of IIR and DFT Mel filters. As a result a model for additive noise was proposed with several implications from the theoretical and practical point of view: firstly, a) this model suggests that the clean signal information should be treated as a stochastic variable; secondly, b) when the noise is added an uncertainty is introduced due to the lack of knowledge about the phase difference between the clean signal and noise, and the original clean signal energy cannot be recovered with 100% accuracy; then, c) it is shown that the expected value of the hidden clean signal energy leads to a definition for SS. The uncertainty in noise cancelling is defined as being the variance of the hidden clean signal energy in the log domain given the noisy energy and the noise estimation. This uncertainty variace is estimated using the additve noise model and used to weight the result of the SS along the signal in weighted DTW algorithms. Two weighted algorithms were tested: the proposed one-step DTW and the two-step one (H.Kobatake & Y.Matsunoo, 1994). Experiments with several types of noises showed that the weighted algorithms dramatically

reduced the error rate in all the SNR's and the proposed one-step weighted DTW led to better results than the two-step one.

Results strongly confirmed firstly the weighted algorithm approach and secondly, the reliability in noise cancelling weighting. However, the weighting function uses a threshold, VarThr, whose optimum value is still case dependent although a wide range of sub-optimal values is achieved by means of setting an upper threshold to the uncertainty variance. This indicates that a further revision on the weighting function is needed but this will done in the next chapter using HMM.

It is worth mentioning that the results here reported concern speaker dependent experiments but the proposed method does not not use any *a priori* information about the speaker and the approach should be easily generalised to the speaker independent case.

Finally, this chapter formalised the revision of the classical concept of acoustic matching algorithms where all the frames have the same weight, and it is proposed that the reliability in noise cancelling should be considered in a frame-by-frame basis. This approach seems generic and has important implications from the practical application point of view.

Chapter 6

Weighted Matching Algorithms in the context of HMM

6.1 Introduction

In the previous chapter it was shown that weighting the information along the signal can substantially improve the recognition accuracy when the speech signal is corrupted by additive noise using spectral subtraction (SS), an easily implemented technique. This means that no model about the corrupting signal is needed except a rough estimation of the noise energy. The experiments were done using a one-step weighted DTW and the noise energy in the filter bank was poorly estimated only once using 200 ms of non-speech signal. Those results revealed that the classical concept of matching algorithm where all the frames have the same weight should be revised in order to take into consideration the reliability in noise cancelling frame by frame. It was shown that once the noise is added, an uncertainty is introduced and the original signal cannot be recovered with 100 % accuracy, and the reliability (inverse of uncertainty) in noise cancelling is dependent on the segmental SNR.

In this chapter, a weighted Viterbi algorithm is proposed and applied in combination with a weighting function that does not need any free variable in isolated word recognition. This modified Viterbi algorithm is compared and combined with state duration modelling and it is shown that weighting the information along the signal leads to better results than the introduction of temporal constraints in the recognition algorithm, even requiring a low computational load. In combination with temporal constraints, the weighted Viterbi algorithm resulted in a high recognition accuracy at SNR equal to 18dB (error rate less than 1%), at 12dB (error rate less

than 3%) and in some cases at 6dB (error rate less than 10%) without a noise model. The approach here covered seems to be generic and interesting from the practical applications point of view. The author believes that weighted matching algorithm approach could be applied to other problems of robust processing such as speaker verification and speaker adaptation in noisy conditions. Also in this chapter, the introduction of temporal constraints is discussed and the importance of modelling the state duration with a parametric distribution (e.g. gamma) is evaluated.

A widely accepted idea is that the best recognition results are achieved when the training database is recorded in the same conditions as the testing utterances. This is the principle which Parallel Model Combination (PMC) (Gales, 1995) (Gales & S.Young, 1996) (Gales & S.Young, 1995) relies on. In this approach, the HMMs trained with clean speech signals are combined with an HMM of noise in order to generate HMMs of noisy speech. Results using the Noisex database revealed that PMC is able to give a high recognition accuracy at SNR equal to 18dB (error rate less than 1%), at 12dB (error rate less than 1%), at 6dB (error rate less than 5%) and at 0dB (error rate less than 10%) (Gales, 1995). However, the technique presents some disadvantages (M.F.Gales, 1997): a) it requires explicit and accurate models of both the additive and channel distortion ; b) it is not easily used with some types of parameterisation such as Cepstral Mean Normalization (CMN); c) it has problems of adaptation speed for non-stationary noises; d) it is computationally expensive; and finally, e) it mainly addresses the problem of additive noise and the cancellation of the convolutional noise needs a previous knowledge about the channel response. In contrast, Weighted Matching Algorithms (WMA) could be considered as a formalization of a very important characteristic of the auditory perception which does not have to recover all the information of the corrupted speech signal and reduces the importance of the more noisy intervals to extract the information that is relevant to understand the message. WMA tried to explore the fact that the segmental SNR widely varies along the signal, it is computationally efficient, it is able to capture well the dynamics of the corrupting signals and considerably improve the robustness of the recognition algorithm with a minimum knowledge about the additive noise and, as will be discussed in the next chapter, no information about the transmission channel is needed. As mentioned before, WMA in combination with temporal constraints can give a high recognition accuracy (error rate less than 10%) at SNR>6dB but gives

poorer results than PMC at SNR equal to 0dB. Nevertheless, as explained in the next chapter, WMA allows the use of an easily implemented technique (CMN) to cancel the transmission channel response.

As it was initially said, WMA was compared and combined with temporal constraints and successfully applied to the problem of isolated word recognition using word modelling. However, when applied to the problem of connected digits in the context of the token passing algorithm (S.J.Young *et al.*, 1989) using HTK (S.J.Young *et al.*, 1989) , the weighted Viterbi algorithm without temporal constraints was proved to be able to improve the recognition accuracy although the reduction in the error rate was much lower than for the isolated case. This must be a consequence of the fact that the temporal constraints are much more important for the connected or continuous recognition than for the isolated case. The reformulation of the connected algorithm to include reliability weighting and temporal constraints is out of the scope of this thesis and is seen as a continuation of the work here presented. Finally, the results with isolated words are encouraging and indicate that the same order of improvement could be achieved in more complex applications.

6.2 Speech recognition with HMM for isolated words

In isolated word recognition for small vocabularies (Fig.6.1)every word is modelled using a single HMM (X.D.Huang *et al.*, 1990) and a common topology is the left-to-right without skip-state transition one (Fig. 6.2). During the recognition procedure, as explained in Chapter 2, a test utterance is divided in overlapped frames and cepstral coefficients based on spectral analysis (eg Mel filter bank) are computed in every frame. Finally, the sequence of parameter vectors is processed by all the HMMs (one per word of the vocabulary) and the HMM with highest likelihood corresponds to the recognized word. This procedure is illustrated in Fig. 6.1.

For the optimal sequence of states S^*, the one that results from the Viterbi alignment (X.D.Huang *et al.*, 1990), the likelihood is given by:

$$Pr(T|\lambda) = Pr(T|S^*,\lambda) \cdot Pr(S^*|\lambda) \tag{6.1}$$

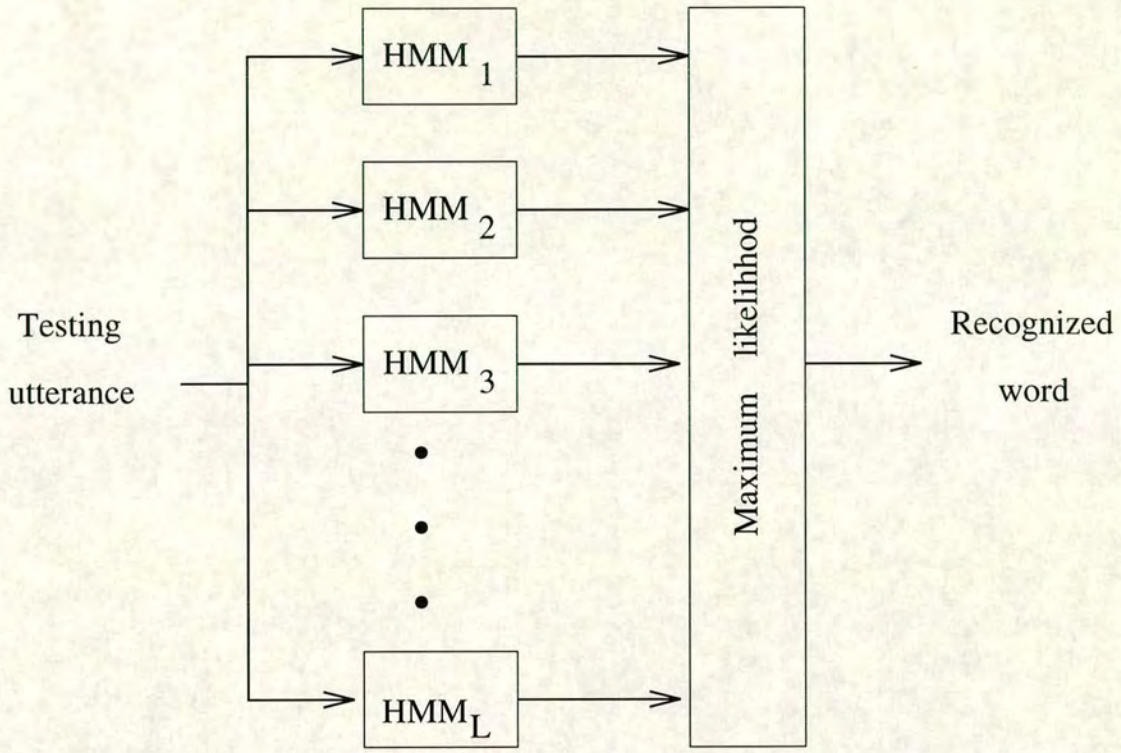


Figure 6.1: Word recognition using one HMM per word. The testing utterance is processed by all the HMMs and the one with highest likelihood corresponds to the recognized word.

where

$$Pr(T|S^*,\lambda) = b_{s(1)}(T_1) \cdot b_{s(2)}(T_2) \cdot b_{s(3)}(T_3) \cdot \ldots \cdot b_{s(t)}(T_t) \cdot \ldots \cdot b_{s(L_T)}(T_{L_T}) \qquad (6.2)$$

and

$$Pr(S^*|\lambda) = a_{s(1),s(2)} \cdot a_{s(2),s(3)} \cdot a_{s(3),s(4)} \cdot \ldots \cdot a_{s(t-1),s(t)} \cdot \ldots \cdot a_{s(L_T-1),s(L_T)} \qquad (6.3)$$

where T denotes the test utterance, T_t the parameter vector at time t, λ is a model, $b_i(T_t)$ is the observation probability of state i and $a_{i,j}$ corresponds to the transition probability between states i and j. As discussed in Chapter 5 in the context of DTW, the noise corrupts some segments of the speech signal more severely than others and weighting the local distances using the reliability in noise cancelling strongly reduced the error rate in all SNR's using a poor estimation for the noise. The same idea could be applied to a HMM recognizer where the error resulting from the additive noise leads to a distortion in the observation probability. Once the noise is added, an uncertainty is introduced and the clean frame cannot be recovered with 100 % accuracy mainly because

the phase difference between noise and clean signal is not known. This uncertainty depends on the local SNR and the reliability of the information given by the observation probability is not constant along the signal.

The one-step DP algorithm discussed in Chapter 5 is based on the arithmetic mean concept but since $Pr(T|S^*, \lambda)$ is represented by the product of observation probabilities it is reasonable to suppose that the weighting procedure in the HMM context should be represented by the geometric mean. If the output probability is modelled using a single multivariate Gaussian distribution the geometric mean becomes an arithmetic one in the $\log(Pr(T|S^*, \lambda))$ domain.

6.3 Weighted Viterbi algorithm

In the previous chapter it was suggested that the hidden clean information of the speech signal is a function of the observed noisy signal energy $\overline{x_m^2}$, the noise energy $\overline{n_m^2}$ and the phase difference ϕ_m between the clean signal and noise in channel m:

$$\overline{s_m^2}(\phi_m, \overline{n_m^2}, \overline{x_m^2}) = 2 \cdot A^2 \cdot \cos^2(\phi) + B - 2 \cdot A \cdot \cos(\phi) \cdot \sqrt{A^2 \cdot \cos^2(\phi) + B} \quad (6.4)$$

where $A = \sqrt{\overline{n_m^2} c_m}$, $B = \overline{x_m^2} - \overline{n_m^2}$ and c_m is a correction coefficient. Using (6.4) and assuming that the random variables ϕ and $\overline{n_m^2}$ are uncorrelated, ϕ is uniformly distributed between $-\pi$ and π and that $\overline{n_m^2}$ is concentrated near its mean $E[\overline{n_m^2}]$, it is possible to show that

$$E[\log(\overline{s_m^2})|\overline{x_m^2}] \simeq \log(\overline{x_m^2} - E[\overline{n_m^2}]) \quad (6.5)$$

and compute

$$Var[\log(\overline{s_m^2})|\overline{x_m^2}] = E[\log^2(\overline{s_m^2})|\overline{x_m^2}] - E^2[\log(\overline{s_m^2})|\overline{x_m^2}] \quad (6.6)$$

The result presented by (6.5) suggests that the expected value of the hidden information $\log(\overline{s_m^2})$ is approximately equal to the log of the spectral subtraction (SS) estimation ($Est(\overline{s_m^2})$) if $Est(\overline{s_m^2}) = \overline{x_m^2} - E[(\overline{n_m^2})]$ where $E[\overline{n_m^2}]$ is the mean noise energy estimation made in non-speech

intervals. In order to avoid negative magnitude estimates a rectifying function is applied:

$$\mathrm{Est}(\overline{s_m^2}) = \begin{cases} \overline{x_m^2} - \mathrm{E}[\overline{n_m^2}] & \text{if} \quad \overline{x_m^2} - \mathrm{E}[\overline{n_m^2}] \geq \mathrm{SsThr}_m \\ \mathrm{SsThr}_m & \text{if} \quad \overline{x_m^2} - \mathrm{E}[\overline{n_m^2}] < \mathrm{SsThr}_m \end{cases} \tag{6.7}$$

where SsThr_m is a constant estimated according to section 5.5. In some experiments, SS defined as (6.7) was employed to compute the uncertainty variance and to estimate the clean signal energy, although better results were achieved when the clean signal energy was estimated with the more general definition for SS (M.Berouti *et al.*, 1979):

$$\mathrm{Est}(\overline{s_m^2}) = \begin{cases} \overline{x_m^2} - \alpha \cdot \mathrm{E}[\overline{n_m^2}] & \text{if} \quad \overline{x_m^2} - \alpha \cdot \mathrm{E}[\overline{n_m^2}] \geq \beta \cdot \mathrm{E}[\overline{n_m^2}] \\ \beta \cdot \mathrm{E}[\overline{n_m^2}] & \text{if} \quad \overline{x_m^2} - \alpha \cdot \mathrm{E}[\overline{n_m^2}] < \beta \cdot \mathrm{E}[\overline{n_m^2}] \end{cases} \tag{6.8}$$

The variance $\mathrm{Var}[\log(\overline{s_m^2})|\overline{x_m^2}]$ is an estimation of the uncertainty related to noise cancelling and was used to weight the matching algorithm and it was proved, by means of a modified version of the DTW algorithm, that weighting the information along the signal could substantially reduce the error rate when the clean signal was corrupted by additive noise using a poor estimation of the corrupting signal. The frame weighting function was defined as being

$$w = \begin{cases} 1 & \text{if } \mathrm{TotalVar} \leq \mathrm{VarThr} \\ \frac{\mathrm{VarThr}}{\mathrm{TotalVar}} & \text{if } \mathrm{TotalVar} > \mathrm{VarThr} \end{cases} \tag{6.9}$$

where

$$\mathrm{TotalVar} = \sum_{m=1}^{14} \mathrm{Var}[\log(\overline{s_m^2})|\overline{x_m^2}] \tag{6.10}$$

The estimated maximum distortion, according to (5.34) on page 77, when SS estimation is equal to SsThr is used as an upper bound for $\mathrm{Var}[\log(\overline{s_m^2})|\overline{x_m^2}]$. As an alternative, a less precise upper bound could be $\mathrm{Var}[\log(\overline{s_m^2})]$ which is estimated on the clean signal.

The weighting function set by (6.9) implies that the reliability related to SS in a channel would be inversely proportional to $\mathrm{Var}[\log(\overline{s_m^2})|\overline{x_m^2}]$. However, the inverse variance weighting presents some problems: firstly, it goes very high when the noise is low; and secondly, it is observed that the recognition error remains low for high SNR's and starts increasing more abruptly from a given

noise level. Equation (5.34) tries to counteract these deficiences by means of the discontinuity introduced at VarThr. In the context of the weighted DTW, (5.34) strongly reduced the error rate in all the SNR's but the optimal threshold TotalVar was case dependent.

The reliability coefficient can be included in the Viterbi algorithm (X.D.Huang *et al.*, 1990) by raising the output probability of observing the frame T_t to the power of $w(t)$, where t is the time index. This modification leads to the following algorithm:

STEP 1 : Initialization. For each state i,

$$\delta_1(i) = \pi_i \times [b_i(T_1)]^{w(1)}$$

$$\psi_1(i) = 0$$

STEP 2 : Recursion. From t=2 to L_T, for all states j,

$$\delta_t(j) = Max_i[\delta_{t-1}(i) \times a_{ij}] \times [b_j(T_t)]^{w(t)}$$

$$\psi_t(j) = argmax_i[\delta_{t-1}(i) \times a_{ij}]$$

STEP 3: Termination. (* indicates the optimised results).

$$P^* = Max_{s \in Sf}[\delta_{L_T}(s)]$$

where L_T is the frame sequence length and s_F is the set of possible final states (for the definition of $\delta_t(j)$ and $\psi_t(j)$ refer to page 16). Consequently, the influence of the probability $b_i(T_{t-1})$ in the decision $Max_i[\delta_{t-1}(i) \times a_{ij}] = Max_i[Max_h[\delta_{t-2}(h) \times a_{hi}] \times [b_i(T_{t-1})]^{w(t-1)} \times a_{ij}]$ at STEP 2 depends on $w(t-1)$: if $w(t-1) = 1$ (high reliability), the influence of $b_i(T_{t-1})$ is maximum; if $w(t-1) = 0$ (very low reliability), the influence of $b_i(T_{t-1})$ is zero because $[b_i(T_{t-1})]^0 = 1$ for all states i.

If the output probability is a multivariate Gaussian pdf:

$$b_i(T_t) = \frac{1}{(2\pi)^{\frac{d}{2}}|C_{\lambda,i}|^{\frac{1}{2}}} \cdot e^{-\frac{1}{2}\cdot(T_t-\mu_{\lambda,i})' C_{\lambda,i}^{-1}(T_t-\mu_{\lambda,i})} \qquad (6.11)$$

where λ denotes an HMM, d is the dimension of vector T_t, $\mu_{\lambda,i}$ is the mean vector of state i and $C_{\lambda,i}$ is the d by d covariance matrix and $(T_t - \mu_{\lambda,i})'$ is the transpose of $(T_t - \mu_{\lambda,i})$. If the

logarithmic function is applied to (6.1),

$$\log\{Pr(T|\lambda)\} = -\sum_{t=1}^{L_T}\log\left\{(2\pi)^{\frac{d}{2}}|C_{\lambda,s(t)}|^{\frac{1}{2}}\right\}$$
$$-\sum_{t=1}^{L_T}\left\{\frac{1}{2}\cdot(T_t-\mu_{\lambda,s(t)})'C^{-1}_{\lambda,s(t)}(T_t-\mu_{\lambda,s(t)})\right\}+\log\{Pr(S^*|\lambda)\} \quad (6.12)$$

The first sum does not depend on the testing sequence so does not have any discriminative value. The second one corresponds to the sum of the Euclidean distances, along the optimal alignment path, weighted by the variances of every state. The similarity with the DTW algorithm is evident except for the variances of every coefficient. If the weighting process is included in the Viterbi algorithm, (6.12) becomes

$$\log\{Pr(T|S^*,\lambda)\} = -\cdot\sum_{t=1}^{L_T}w(t)\cdot\log\left\{(2\pi)^{\frac{d}{2}}|C_{\lambda,s(t)}|^{\frac{1}{2}}\right\}$$
$$-\sum_{t=1}^{L_T}w(t)\cdot\left\{\frac{1}{2}\cdot(T_t-\mu_{\lambda,s(t)})'C^{-1}_{\lambda,s(t)}(T_t-\mu_{\lambda,s(t)})\right\}+\log\{Pr(S^*|\lambda)\} \quad (6.13)$$

whose second term is also similar to the global distance between two utterances according to the weighted version of the DTW algorithm discussed in Chapter5. It is worth mentioning that the alignment without weighting is not necessarily the same as the one with weighting because the modified Viterbi algorithm is also only one step.

As can be seen in (6.13), the weight $w(t)$, which is between 0 and 1, tends to compress the range of variation of the output probability $b_i(T_t)$: if $w(t)$ is close to 0 (low reliability), $[b_i(T_t)]^{w(t)}$ will be close to 1 regardless of $b_i(T_t)$ and this probability loses discriminability. Consequently, the importance of $Pr(S^*|\lambda)$ to discriminate between two models increases in those segments where the local SNR is lower. However, at least for the application here considered (digits), the transition probabilities offer a low discriminative ability. Moreover, the transition probabilities are related to the modelling of state durations which is not well achieved using the geometric distribution of the ordinary HMM topology. In other words, the weighting procedure should enhance the need of more realistic state duration distribution. For the case of isolated digit, the term $Pr(S^*|\lambda)$ should not present big changes for different state alignment and the weighted Viterbi alignment strongly reduces the error rate although the state alignment is not necessarily the same with and without weighting, and the recognition tends always to rely on those frames

with higher segmental SNR. However, for connected digits recognition, the state and word alignment is used to decide the sequence of words and to achieve this it is necessary to segment properly the speech signal in terms of where one word should start and finish. Consequently, for connected recognition the role of the temporal constraints seems to be dramatically more important than for the isolated case to make the weighted algorithm successful.

6.4 Revision of the weighting function

Preliminary experiments with the modified version of the Viterbi algorithm were done using (6.9) were TotalVar was computed using the uncertainty variances in the logarithm domain as defined in section 5.4 (N.B.Yoma *et al.*, 1998a) . Results mainly confirmed the previous experiments with the weighted DP equation (N.B.Yoma *et al.*, 1997b) (N.B.Yoma *et al.*, 1997a), but due to the effect of high $Var[\log(\overline{s_m^2})|\overline{x_m^2}]$ caused when the SS estimation is equal to SsThr another weighting function was defined in the cepstral domain using the HMM variances (N.B.Yoma *et al.*, 1998b).

6.4.1 Mapping from the log to the cepstral domain

The cepstral coefficients are estimated by means of,

$$c_n = \sum_{m=1}^{M} E_m \cdot \cos(n, m) \tag{6.14}$$

where E_m is the logarithm of the energy at the output of the filter m that results from the SS estimation, M is the number of filters and $\cos(n, m)$ denotes

$$\cos(n, m) = \cos[\frac{\pi \cdot n}{M} \cdot (m - 0.5)]$$

The uncertainty variance of the cepstral coefficient c_n given the observed energy is given by (see Appendix A),

$$Var[c_n|X] = \sum_{m=1}^{M} Var[\log(\overline{s_m^2})|\overline{x_m^2}] \cdot \cos^2(n, m) + 2 \cdot \sum_{m}^{M} \sum_{i>m}^{M} \left(R_{m,i} \cdot \sqrt{Var[\log(\overline{s_m^2})|\overline{x_m^2}]} \cdot \sqrt{\cdot Var[\log(\overline{s_i^2})|\overline{x_i^2}]} \right) \cdot \cos(n, m) \cdot \cos(n, i) \tag{6.15}$$

where $X = [\overline{x_1^2}, \overline{x_2^2}, \overline{x_3^2}, ..., \overline{x_M^2}]$ is the observed noisy signal energies along the M filters and $R_{m,i}$

is the correlation coefficient (A.Papoulis, 1991) between the components $\log(\overline{s_m^2}(\phi_m, \overline{n_m^2}, \overline{x_m^2}))$ and $\log(\overline{s_i^2}(\phi_i, \overline{n_i^2}, \overline{x_i^2}))$. If $R_{m,i}$ is supposed equal to zero for $m \neq i$, $Var[c_n|X]$ can be re-written as

$$Var[c_n|X] = \sum_{m}^{M} Var[\log(\overline{s_m^2})|\overline{x_m^2}] \cdot \cos^2(n, m) \tag{6.16}$$

The components $\log(\overline{s_m^2}(\phi_m, \overline{n_m^2}, \overline{x_m^2}))$ and $\log(\overline{s_i^2}(\phi_i, \overline{n_i^2}, \overline{x_i^2}))$ are clearly correlated specially when m and i are close. However, although the uncorrelated condition was a rough aproximation, it was enough to lead to good results.

6.4.2 Modified weighting function

The weighting function that (6.9) attempts to model could also be approximated by

$$w = \frac{VarThr}{VarThr + TotalVar} \tag{6.17}$$

Experiments with the weighted version of DTW showed that (6.9) and (6.17) lead to similar results although the optimal threshold $VarThr$ is not necessarily the same for both functions. As in (6.9), the optimal $VarThr$ depended on the noise and speaker as a result of the high $TotalVar$ caused when at least one of the SS estimations is equal to $SsThr_m$. In order to counteract this limitation and avoid the threshold $VarThr$, a weighting function based on (6.17) was proposed using the variances of the HMMs. In the experiments here reported, each word was modelled using an 8-state left-to-right topology without skip-state transition (Fig 6.2), with a single multivariate Gaussian density per state and a diagonal covariance matrix, and the modified frame weighting function is defined as

$$w = \frac{1}{D} \sum_{n=1}^{D} \frac{\sigma_{\lambda,i,n}^2}{\sigma_{\lambda,i,n}^2 + Var[c_n|X]} \tag{6.18}$$

where $\sigma_{\lambda,i,n}^2$ is the variance of coefficient n, state i and model λ. The function shown in (6.18) compares the uncertainty variance of coefficient n with the variance of the coefficient n in a phonetic class or state of a HMM. Moreover, if uncertainty variance is high for one coefficient, w is not necessarily low because the weight is the sum of terms $\frac{\sigma_{\lambda,i,n}^2}{\sigma_{\lambda,i,n}^2 + Var[c_n|X]}$. Finally, if the signal is clean, $Var[c_n|X]$ is zero for all n and $w = 1$.

6.5 Temporal constraints

When noisy testing utterances are processed by HMMs trained in clean conditions, the error introduced by the state output probabilities leads to an unreasonable optimal alignment path where some states may be active for too many frames and others for too few ones. Every state could be associated to a phoneme (or part of one) and, due to the limitations of the vocal tract articulation rate, stationary segments (eg vowels) can not be shorter than 50 or 75 ms. Moreover, in normally uttered speech, excessively long phonemes are unlikely and bounding or modelling state durations in order to impose restrictions to the optimal alignment path resulting from the Viterbi algorithm seems an interesting approach to reduce the error rate specially when the speech signal is corrupted by noise. However, the transition probability is represented by a constant in the ordinary HMM topologies and this leads to a geometric probability density for state duration which is not accurate for most cases.

Many techniques have been proposed to include state duration modelling in HMM. In (J.D.Ferguson, 1980) the state duration probability is estimated during the Baum-Welch algorithm and the method requires a high computational load and a large amount of training data. Parametric state duration distributions, Poisson (M.J.Russell & R.K.Moore, 1985) and gamma (S.E.Levinson, 1986), were used in order to reduce the amount of training data although a high computational load was still required. In (L.R.Rabiner *et al.*, 1989) it was proposed a backtracking procedure where the duration contribution to the standard Viterbi metric is added after collecting possible candidate paths. The disadvantage of this approach is that the correct alignment path may not be one of these candidates. A significant improvement of the error rate when the speech signal was corrupted by additive noise was reported in (K.Laurila, 1997) by means of introducing the state duration constraints in the training procedure, using the state sequences that are likely to happen and fulfill the temporal restrictions. This implementation requires the Forward and Viterbi algorithm to be modified, which was not interesting for the purpose of this research given the evironment in which the tests were done.

In order to include temporal constraints in the HMM recognizer, the procedure suggested by (D.Burshtein, 1996) was followed, where the state durations are modelled using gamma distributions. Every state was associated to a gamma distribution whose parameters were estimated

using the training database after the HMMs have been trained. The discrete gamma distribution is given by (D.Burshtein, 1996) :

$$d(\tau) = K \cdot e^{-\alpha \cdot \tau} \cdot \tau^{p-1} \tag{6.19}$$

where $\tau = 0, 1, 2, \ldots$ is the duration of a given state in number of frames, $\alpha > 0$, $p > 0$ and K is a normalizing term. This distribution was proved to fit better the empirical (state and word) duration distributions than the Gaussian or geometric functions (D.Burshtein, 1996) . After training the HMMs, the optimal state sequence was estimated for every training utterance using the Viterbi algorithm and the parameters α and p were estimated for every state in each model by means of:

$$\alpha = \frac{E(\tau)}{Var(\tau)} \tag{6.20}$$

and

$$p = \frac{E^2(\tau)}{Var(\tau)} \tag{6.21}$$

where $E(\tau)$ and $Var(\tau)$ are, respectively, the mean and variance of the state duration directly computed using Viterbi alignment. Beside $E(\tau)$ and $Var(\tau)$, $min(\tau)$ and $max(\tau)$ were also estimated.

Instead of using the duration metric suggested in (D.Burshtein, 1996), the transition probabilities were defined as

$$a_{i,i}^{(\tau)} = Prob(s_{t+1} = i | s_t = s_{t-1} = \ldots = s_{t-\tau+1} = i) \tag{6.22}$$

and

$$a_{i,j}^{(\tau)} = Prob(s_{t+1} = j | s_t = s_{t-1} = \ldots = s_{t-\tau+1} = i) \tag{6.23}$$

Using these definitions for the transition probabilities, $a_{i,i}^{(\tau)}$ and $a_{i,j}^{(\tau)}$ can be estimated by

$$a_{i,i}^{(\tau)} = \frac{D_i(\tau) - d_i(\tau)}{D_i(\tau)} \tag{6.24}$$

and

$$a_{i,j}^{(\tau)} = \frac{d_i(\tau)}{D_i(\tau)} \tag{6.25}$$

where $D_i(\tau)$ is the probability of state i being active for $t \geq \tau$:

$$D_i(\tau) = \sum_{t=\tau}^{t_{max}} d(t) \tag{6.26}$$

In order to include the possible min and max durations, the transition probabilities were modified to:

$$a_{i,i}^{\tau} = \begin{cases} 1 & \text{if } \tau < t_{min} \\ 0 & \text{if } \tau \geq t_{max} \\ \frac{D_i(\tau) - d_i(\tau)}{D_i(\tau)} & \text{otherwise} \end{cases} \tag{6.27}$$

and

$$a_{i,i+1}^{\tau} = \begin{cases} 0 & \text{if } \tau < t_{min} \\ 1 & \text{if } \tau \geq t_{max} \\ \frac{d_i(\tau)}{D_i(\tau)} & \text{otherwise} \end{cases} \tag{6.28}$$

where $t_{min} = 0.8 \cdot min(\tau)$ and $t_{max} = 1.5 \cdot max(\tau)$. The constants 0.8 and 1.5 introduce a tolerance to the min and max duration for every state.

The recognition experiments were speaker dependent using isolated words (digits). In some cases it was observed that the variation in state duration was very low, which resulted in a low $Var(\tau)$ which in turn caused a low recognition accuracy (error rate higher than 20% at SNR equal to 18dB). To counteract this, a threshold was introduced to set a floor for $Var(\tau)$. According to some experiments, a suitable value for this threshold would be 4.

The gamma function matches better the state or word duration distribution than the Gaussian or geometric densities (D.Burshtein, 1996)). However, this fact does not mean that the gamma density fuction is the best from the recognition point of view. Accurately modelling time duration should improve the recognition accuracy but, on the other hand, the information given by the duration of phonemes is usually less important than the spectral (or cepstral) information

and even the perception can hardly decide if a vowel lasts, as an example, for 200 or 300 ms. Moroever, what seems relevant to reduce the recognition error rate is to avoid unreasonable optimal alignments that result in a too long duration for some states and in a too short one for others. In order to evaluate the contribution of the gamma modelling in the error rate, experiments were done using the restrictions for possible max and min durations but keeping the geometric distribution of the ordinary HMM topology:

$$a_{i,i}^{\tau} = \begin{cases} 1 & \text{if } \tau < t_{min} \\ 0 & \text{if } \tau > t_{max} \\ a_{i,i} & \text{otherwise} \end{cases} \tag{6.29}$$

$$a_{i,i+1}^{\tau} = \begin{cases} 0 & \text{if } \tau < t_{min} \\ 1 & \text{if } \tau > t_{max} \\ a_{i,i+1} & \text{otherwise} \end{cases} \tag{6.30}$$

where $a_{i,i}$ and $a_{i,i+1}$ are transition probabilities estimated during the training algorithm. Both sets of transition probabilities, (6.27)(6.28) and (6.29)(6.30), specify a maximum and minimum state duration, although (6.27)(6.28) better fits the empirical state duration distributions.

6.6 Experiments with isolated words

As in the last chapter, the tests were carried out employing the two speakers (one female and one male), and five noises from the Noisex database (car, speech, Lynx, operation room and factory) (A.Varga *et al.*, 1992). The experiments were speaker dependent and the vocabulary was composed by English digits from 0 to to 9. The signals were downsampled to 8000 samples/sec. The signal was divided in 25ms frames with 12.5ms overlapping. Each frame was processed with a Hamming window before the spectral estimation. The band from 300 to 3400 Hz was covered with 14 Mel DFT filters. At the output of each channel the energy was computed, SS (either defined by (6.7) or (6.8)) was applied and the log of the energy was estimated. When SS was defined as in (6.7), the threshold $SsThr_m$ was estimated according to (Compernolle, 1989)

and was approximately equal to 20dB for all the channels. When SS was defined as in (6.8), the overestimation parameter for SS $\alpha = 2.0$ and the noise spectral floor $\beta = 0.01$ (T.Claes *et al.*, 1996). In every frame 10 cepstral coefficients were computed.

In these experiments the noise estimation was made only once using just 200ms of non-speech signal and was kept constant for all the experiments at the same global SNR.

The threshold $SsThr_m$ used to compute the maximum distortion, according to (5.34) on page 77, was the same used for SS according to (6.7) and was approximately equal to 20dB for all the channels. Each word was modelled using an 8-state left-to-right HMM without skip-state transition (Fig. 6.2), with a single multivariate Gaussian density per state and a diagonal covariance matrix. The HMMs and the state duration distributions were estimated by means of

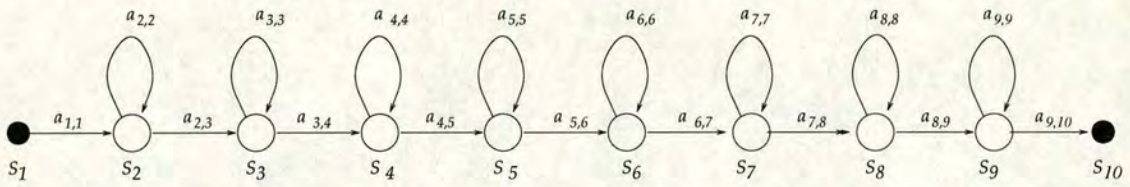


Figure 6.2: Eight-state left-to-right HMM without skip-state transition.

the clean signal training utterances. In the experiments HTK V.2.0 with modifications to include the temporal constraints and reliability weighting in the testing procedure was used for the HMM experiments.

6.6.1 Temporal constraints

In order to test the validity of the state duration modelling with gamma distribution from speech recognition point of view, three experiments were done: the ordinary Viterbi algorithm Vit; the Viterbi algorithm with max and min state duration plus state duration distribution with gamma pdf $Vit - Mm - Gamma$; and finally, the Viterbi algorithm with max and min state duration plus the ordinary geometric distribution $Vit - Mm - Geom$. In all the experiments SS according to (6.7) was used to estimate the clean signal. Results are shown in Tables 6.1-6.5. As can be seen, the introduction of temporal constraints $Vit - Mm - Gamma$ and $Vit - Mm - Geom$ substantially reduced the error rate when compared to the odinary Viterbi algorithm with all the noises and at all the SNR's. However, the state duration modeling using the gamma distribution did not improve the recognition accuracy when compared with the ordinary geometric one using

Table 6.1: Comparison of temporal constraints. Recognition error rate(%) for speech corrupted by additive noise (car).

SNR	*18dB*	*12dB*	*6dB*	*0dB*
Vit	1.5	16.5	66	86
Vit-Mm-Gamma	0	4.5	26	59
Vit-Mm-Geom	0	5.5	27	56.5
W2-Vit	0	0	8.5	40
W2-Vit-Mm-Gamma	0	0	3	18.5

Table 6.2: Comparison of temporal constraints. Recognition error rate(%) for speech corrupted by additive noise (speech).

SNR	*18dB*	*12dB*	*6dB*	*0dB*
Vit	4.5	38.5	78.5	90
Vit-Mm-Gamma	0.5	14	46	78
Vit-Mm-Geom	0.5	15.5	45.5	77
W2-Vit	0	2	22.5	65.5
W2-Vit-Mm-Gamma	0	1	11.5	42

the same max and min durations for every state. This result suggests that: a) the gamma distribution, although it fits better the duration of states, does not necessarily lead to better results; and b) the restrictions imposed by the max and min durations are the main factor responsible for the reduction in the error rate.

6.6.2 Weighting coefficients

In order to compare the weighting functions given by (6.9) and (6.18), experiments were done using the modified version of the Viterbi algorithm in combination with temporal constraints (gamma distribution plus max and min durations). The following configurations were tested: the ordinary Viterbi algorithm plus temporal constraints, Vit – Mm – Gamma; the weighted

Table 6.3: Comparison of temporal constraints. Recognition error rate(%) for speech corrupted by additive noise (Lynx).

SNR	*18dB*	*12dB*	*6dB*	*0dB*
Vit	15	48	78.5	90.5
Vit-Mm-Gamma	2	26.5	55	82.5
Vit-Mm-Geom	1.5	25	53.5	81.5
W2-Vit	1	5.5	22.5	46.5
W2-Vit-Mm-Gamma	0	4	11.5	42

Table 6.4: Comparison of temporal constraints. Recognition error rate(%) for speech corrupted by additive noise (operation room).

SNR	*18dB*	*12dB*	*6dB*	*0dB*
Vit	7	31	69	89.5
Vit-Mm-Gamma	1	17	36.5	71
Vit-Mm-Geom	1	16.5	38	70.5
W2-Vit	0	2.5	17.5	49.5
W2-Vit-Mm-Gamma	0	1	13.5	36

Table 6.5: Comparison of temporal constraints. Recognition error rate(%) for speech corrupted by additive noise (factory).

SNR	*18dB*	*12dB*	*6dB*	*0dB*
Vit	5	38	73	86
Vit-Mm-Gamma	0.5	10.5	34	63
Vit-Mm-Geom	0.5	11.5	35	62
W2-Vit	0.5	6	22.5	42.5
W2-Vit-Mm-Gamma	0.5	2	12	35

version of the Viterbi algorithm using (6.9) as weighting function plus temporal constraints, $W1 - Vit - Mm - Gamma$; and the weighted Viterbi algorithm using (6.18) as weighting function plus temporal constraints, $W2 - Vit - Mm - Gamma$. In all the experiments SS according to (6.7) was used to estimate the clean signal. The results are presented in Figs. 6.3-6.7 where the error rate is plotted vs log($VarThr$) for SNR=18, 12, 6 and 0dB. The error rate with $Vit - Mm - Gamma$ and $W2 - Vit - Mm - Gamma$ are represented by a straight line which emphasize the fact that these configurations do not depend on the free variable $VarThr$ as does the weighting function (6.9) $W1 - Vit - Mm - Gamma$. As can be seen, (6.18) led to lower or equal error rate than the lowest error rate with (6.9) at the optimal $VarThr$ in most cases, without the need of a free variable. In those cases when $W2 - Vit - Mm - Gamma$ was worse than $W1 - Vit - Mm - Gamma$ at the optimal $VarThr$, the difference between the error rate given by (6.18) and the minimum one given by (6.9) was very small when compared with the difference between $W2 - Vit - Mm - Gamma$ and $Vit - Mm - Gamma$. These results indicate that the function (6.18) is a reasonable approximation for the weighting coefficient to be used with the modified Viterbi algorithm proposed in section 6.3, although the uncorrelated condition assumed during the mapping from the logarithmic domain is a rough approximation.

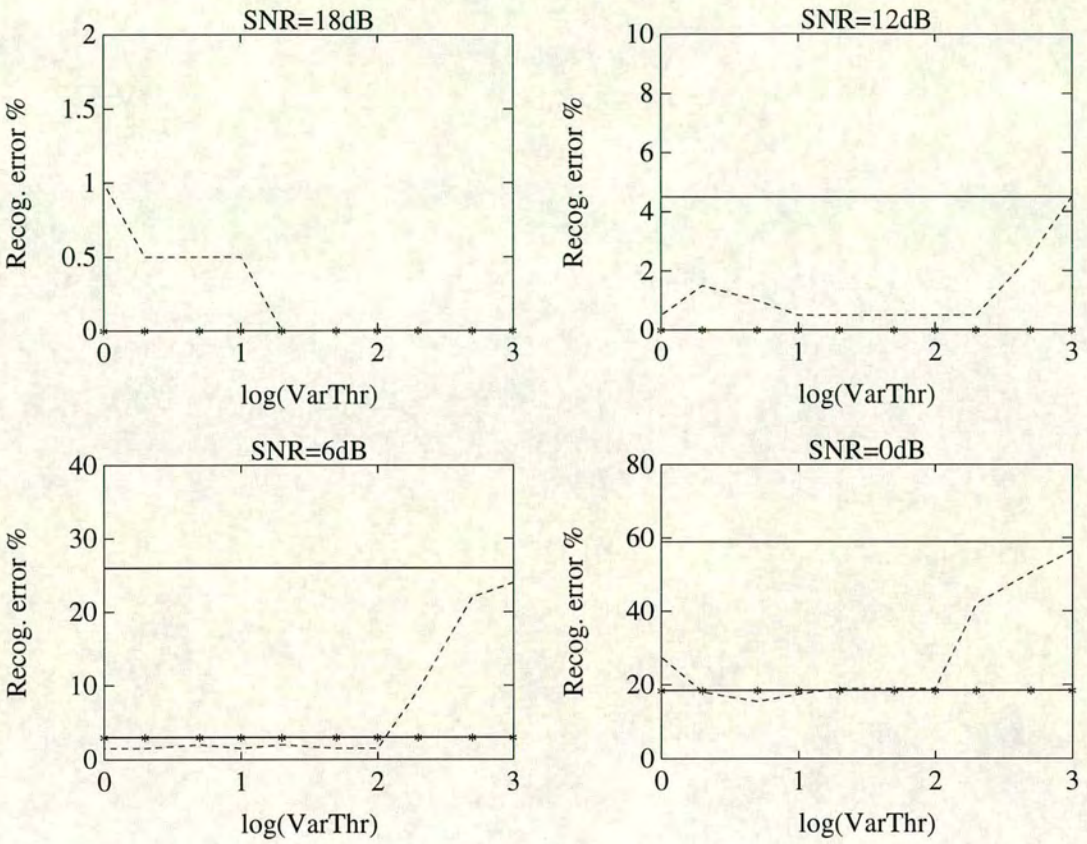


Figure 6.3: Recognition error rate(%) for speech signal corrupted by additive noise (car noise): (–), *Vit-Mm-Gamma* ; (- -), *W1-Vit-Mm-Gamma* ; and (-*-), *W2-Vit-Mm-Gamma* .

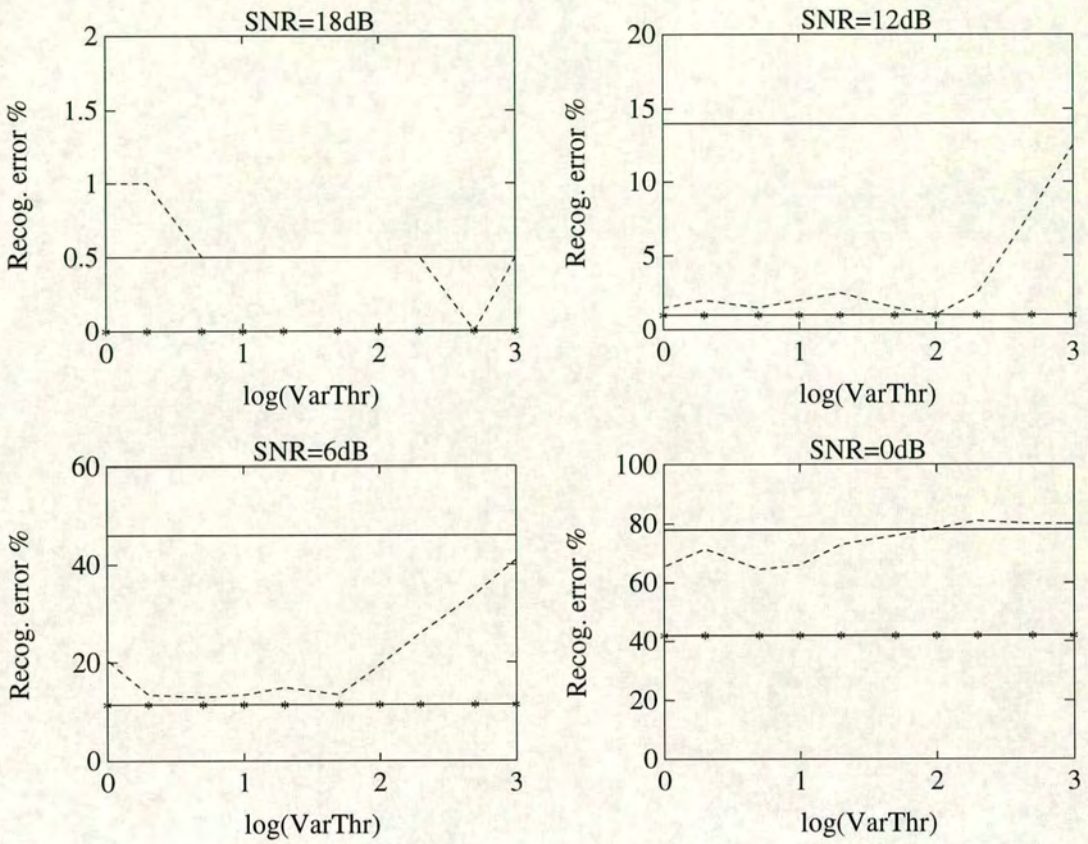


Figure 6.4: Recognition error rate(%) for speech signal corrupted by additive noise (speech noise): (–), *Vit-Mm-Gamma*; (- -), *W1-Vit-Mm-Gamma* ; and (-*-), *W2-Vit-Mm-Gamma* .

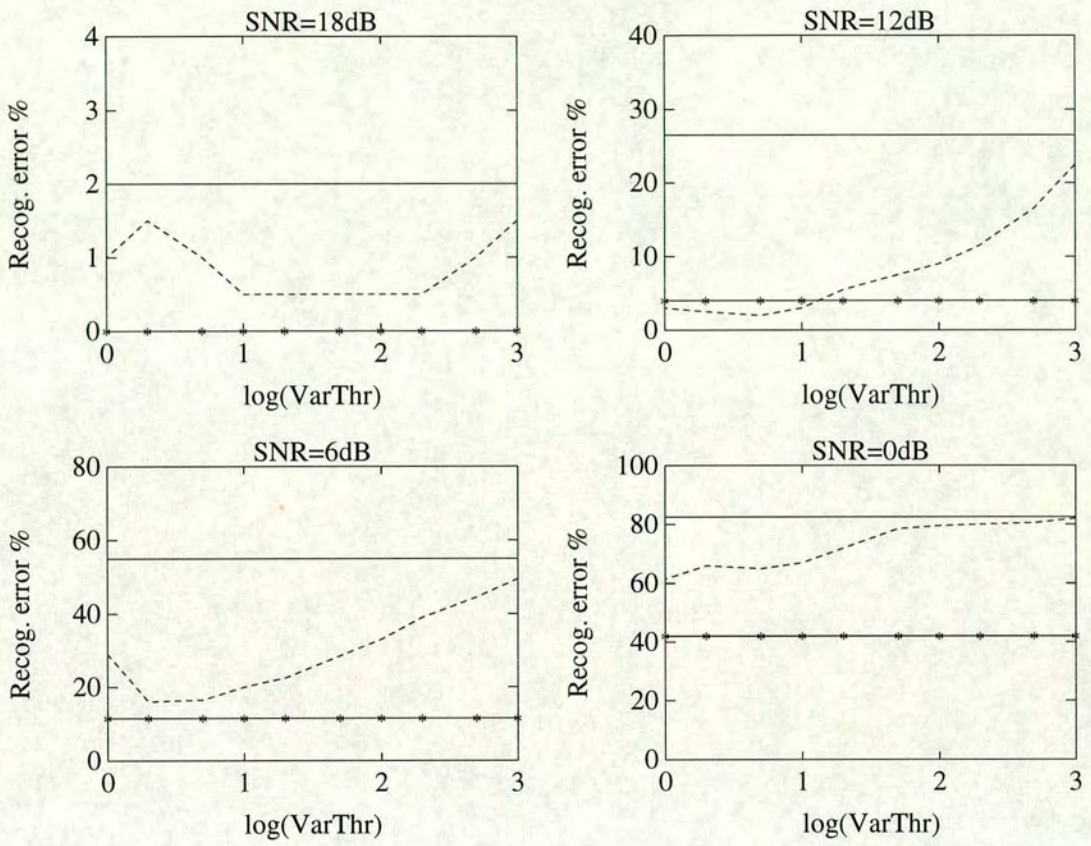


Figure 6.5: Recognition error rate(%) for speech signal corrupted by additive noise (Lynx noise): (–), *Vit-Mm-Gamma* ; (- -), *W1-Vit-Mm-Gamma* ; and (-*-), *W2-Vit-Mm-Gamma* .

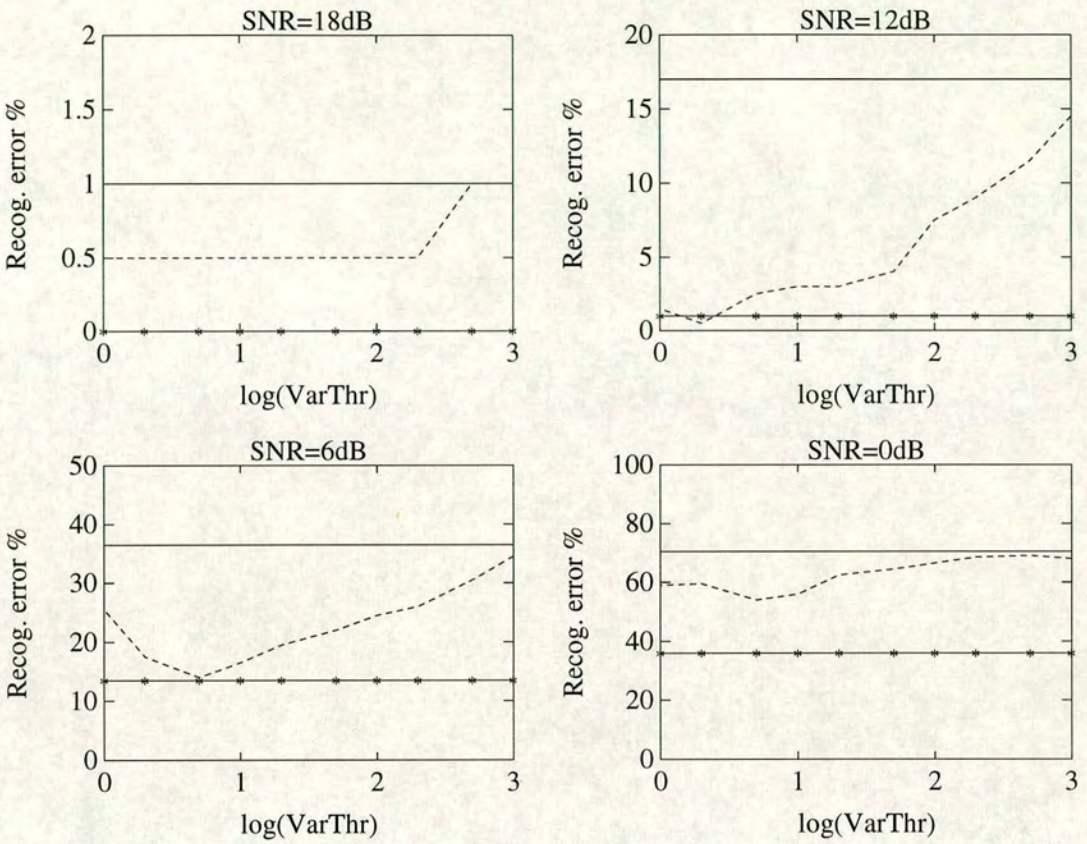


Figure 6.6: Recognition error rate(%) for speech signal corrupted by additive noise (operation room noise): (–), *Vit-Mm-Gamma* ; (- -), *W1-Vit-Mm-Gamma* ; and (-*-), *W2-Vit-Mm-Gamma* .

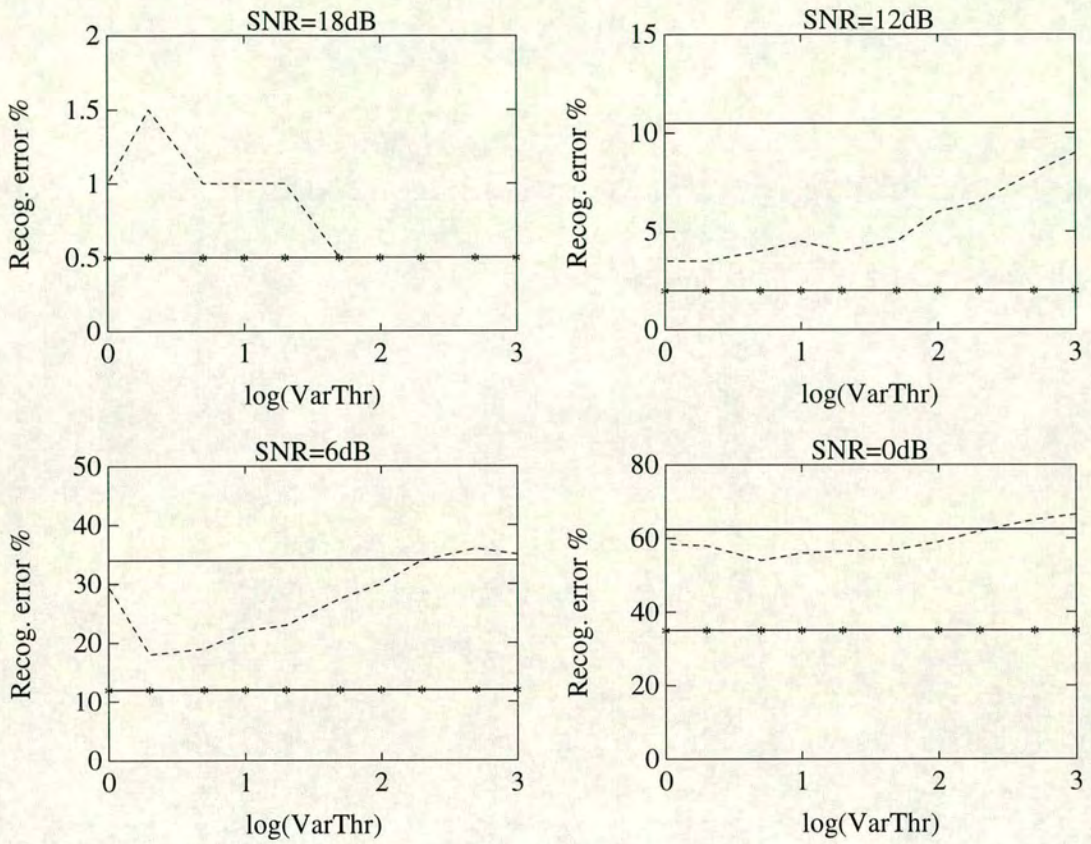


Figure 6.7: Recognition error rate(%) for speech signal corrupted by additive noise (factory noise): (–), *Vit-Mm-Gamma* ; (- -), *W1-Vit-Mm-Gamma* ; and (-*-), *W2-Vit-Mm-Gamma* .

6.6.3 Temporal constraints vs weighted algorithm

As mentioned above, unreasonable state sequences resulting from the Viterbi alignment is a major cause of the fast degradation of recognition systems in noisy conditions. This could be counteracted by means of including state duration modelling either in the training or testing procedure. Another way to overcome this natural deficiency of HMMs is the weighting procedure applied to the Viterbi algorithm proposed in this chapter. The effect of the additive noise is to introduce an inaccuracy in the output probabilities that takes the form of a noise in the $b_i(T_t)$ domain, and raising the output probability by a number less or equal than 1 reduces the noise in $b_i(T_t)$ and the recognition tends to rely on those frames with higher segmental SNR. In section 6.6.1 it was shown that the introduction of temporal restrictions substantially reduced the error rate at all the SNR's, and results in section 6.6.2 with the Viterbi algorithm using temporal constraints suggest that the weighting function represented by (6.18) gave better results that (6.9) without the need of a free variable. However, an interesting comparison is to compare the weighting procedure with the introduction of the temporal constraints done by means of duration modelling. Results for Viterbi algorithm with temporal constraints and for the weighted version of the Viterbi algorithm using (6.18) as weighting function are presented in Tables 6.1- 6.5. All the experiments were done using using SS as in (6.7). As can be seen, the weighted version of the Viterbi algorithm using (6.18) as weighting function $W2-Vit$ gave better results at all the SNR's than the Viterbi algorithm with temporal constraints.

6.6.4 Weighting with and without temporal constraints

As can be seen in Tables 6.1- 6.5, the weighted Viterbi algorithm gave better results with $W2-Vit-Mm-Gamma$ than without $W2-Vit$ temporal constraints in all the cases, although the improvement depended on the noise and SNR. This result confirms the relevance of including temporal constraints in the recognition process despite the fact that the the main reduction in the error rate was due to the weighted algorithm.

6.6.5 Comparison of SS techniques

As discussed in the last chapter, the SS definition according to (6.7) could be seen as a cosequence of the model for additive noise due to the fact that the expected value of the hidden clean signal

Table 6.6: Comparison of SS techniques: *SS1*, SS according to (6.7); and *SS2*, SS according to (6.8). Recognition error rate(%) for speech corrupted by additive noise (car).

SNR	*18dB*	*12dB*	*6dB*	*0dB*
SS1	0	0	3	18.5
SS2	0	0	0	13

Table 6.7: Comparison of SS techniques: *SS1*, SS according to (6.7); and *SS2*, SS according to (6.8). Recognition error rate(%) for speech corrupted by additive noise (speech).

SNR	*18dB*	*12dB*	*6dB*	*0dB*
SS1	0	1	11.5	42
SS2	0	0	4.5	38

information in the logarithmic domain is equal to the difference between the noisy signal energy and the noise energy estimation in each channel of the filter bank. However, a more general SS defined as in (6.8) (M.Berouti *et al.*, 1979) has been adopted by other authors (T.Claes & Compernolle, 1996) (T.Claes *et al.*, 1996) (S.V.Vaseghi *et al.*, 1994). Tables 6.6-6.10 present the results using SS according to (6.7), SS1, and to (6.8), SS2. Experiments were done using the configurtation $W2 - Vit - Mm - Gamma$ (the modified Viterbi algorithm using temporal constraints and (6.18) as weighting function). For SS2, the overestimation parameter $\alpha = 2.0$ and the noise spectral floor $\beta = 0.01$, a configuration found in other papers. The reliability variance was estimated according to the same procedure followed in the previous sections. As can be seen in Tables 6.6 and 6.7, the more general form for SS given by (6.8) gave better results than (6.7). This must be a consequence of the ability of (6.8) in reducing the spectral noise peaks (M.Berouti *et al.*, 1979) . However, Tables 6.8-6.10 show that the general SS gave the same or slightly worse results than the simpler SS equation (6.7). This could be due to the fact that (6.8) also uses the noise energy as a reference for the lower bound of the SS estimation and the accuracy of the noise energy estimation may be more important for some noises than others, although no test was done with other values for α and β. As mentioned before, the noise was

Table 6.8: Comparison of SS techniques: *SS1*, SS according to (6.7); and *SS2*, SS according to (6.8). Recognition error rate(%) for speech corrupted by additive noise (Lynx).

SNR	*18dB*	*12dB*	*6dB*	*0dB*
SS1	0	4	11.5	42
SS2	0.5	3	15	47

Table 6.9: Comparison of SS techniques: *SS1*, SS according to (6.7); and *SS2*, SS according to (6.8). Recognition error rate(%) for speech corrupted by additive noise (operation room).

SNR	*18dB*	*12dB*	*6dB*	*0dB*
SS1	0	1	13.5	36
SS2	0	1	14.5	43

Table 6.10: Comparison of SS techniques: *SS1*, SS according to (6.7); and *SS2*, SS according to (6.8). Recognition error rate(%) for speech corrupted by additive noise (factory).

SNR	*18dB*	*12dB*	*6dB*	*0dB*
SS1	0.5	2	12	35
SS2	0	4	14.5	42.5

estimated only once using just 200 ms of non-speech signal in order to make the test conditions more severe.

6.7 Preliminary experiments with connected words

In order to initially evaluate the weighted algorithm in the problem of connected word recognition, tests were done using triplets (three digits in sequence), both speakers of Noisex database and car noise. The pre-processing is the same as in the isolated case and SS was done according to (6.7). The word models initially trained with isolated digits were re-estimated by means of embedded training. The HMM estimation and recognition were also done using HTK V2.0 but without temporal constraints. A restrictive grammar that allows three digits in sequence was employed and the recognition algorithm is able to segment the speech signal and find the sequence of digits in one step. Results for the ordinary Viterbi algorithm, Vit, and the weighted algorithm using (6.9) as weighting function, $W1 - \mathrm{Vit}$, are presented in Table 6.11. The results for $W1 - \mathrm{Vit}$ correspond to $\mathrm{VarThr} = 100$ (the optimal value). The weighting function (6.18), the one that gave the best results for the isolated case without the need of a free variable,

Table 6.11: Recognition error rate(%) with connected words (triplets): *Vit*, ordinary Viterbi algorithm; and *W1-Vit*, weighted Viterbi algorithm with (6.9) as weighting function. The speech signal is corrupted by additive noise (car).

SNR	*18dB*	*12dB*	*6dB*	*0dB*
Vit	1.4	4	13.5	41.5
W1-Vit	0.7	2	8.4	33.7

degraded the recognition accuracy for the connected digits without temporal constraints but the function (6.9) showed that weighting the information along the signal could also lead to good results for this case, although the improvement was much lower than for the isolated digits task. This must be due to the fact that the weighting procedure tends to enhance the importance of the state duration modelling, which is much more important for the connected than for the isolated recognition task. As discussed above, a different state alignment does not necessarily lead to a recognition error for the isolated word task because the weighted Viterbi algorithm tries to emphasize the importance of high local SNR frames in the recognition procedure and the transition probabilities are not highly discriminative, although the best results were achieved in combination with temporal constarints. In contrast, a different optimal state alignment in connected word recognition may lead to a different word segmentation which in turn can easily result in a word recognition error.

6.8 Discussion and conclusion

In this chapter the weighting procedure was applied to HMM recognizers using word modelling. Initially, for the isolated word recognition task, the weighted version of the Viterbi algorithm strongly reduced the error rate at all the SNR's. The ordinary Viterbi algorithm with temporal constraints also reduced the error rate but the improvement was smaller than with the weighted algorithm. However, the best results were achieved when weighting procedure was applied in combination with state duration modelling. It is interesting to highlight that weighting the information along the signal requires a low computational load and was more effective than the introduction of the temporal constraints. In other words, the weighted Viterbi algorithm was more robust to unlikely alignments because the recognition tends always to rely on those frames with higher segmental SNR. A weighting function (6.18) was proposed and was proved to lead to slightly better results than (6.9) with the optimal threshold $VarThr$ without the need of a free variable. Also in the context of isolated digits recognition task, it was shown that the introduction of temporal constraints in HMM recognizers is essential to try to achieve the same recognition accuracy observed with the weighted DTW algorithm in the last chapter. The experiments with temporal constraints also suggest that: a) the gamma distribution, although it fits better the duration of states, does not necessarily lead to better results; and b) the restrictions

imposed by the max and min durations are the main factor responsible for the reduction in the error rate. As far as SS techniques are concerned, the general SS represented by (6.8) does not necessarily lead to better results than the simpler SS (6.7) in the context of WMA although no experiments were done using a more accurate noise estimation. In combination with temporal constraints, the weighted Viterbi algorithm resulted in a high recognition accuracy at SNR equal to 18dB (error rate less than 1%), at 12dB (error rate less than 3%) and in some cases at 6dB (error rate less than 10%) without an accurate noise model for the car and speech noises. For more complex noises (Lynx, operation room, and Factory) this approach gave an improvement as high as 90 and 80 % in the error rate at SNR=12 and 6dB.

Finally, preliminary experiments with connected word recognition (triplets) revealed that a) WMA could also be applied to this task and b) temporal constraints are very important in the connected word task to improve the results in the context of the weighted Viterbi algorithm.

Reliability in noise cancelling seems to be a very generic and interesting approach for pattern recognition. In the context of speech recognition, weighting the information along the signal led to a substantial reduction in the error rate firstly with DTW and secondly for HMM in isolated word task with speech signal corrupted by several sort of additive noises. Results in Table 6.11 indicate that WMA could also be applied to the case of connected or continuous speech but in order to improve the results it is necessary either a) to include temporal constraints in the connected words algorithm or b) to reformulate the recognition procedure for the connected or continuous case. Both alternatives are out of the scope of this work and are considered as topics for future work, which also include: the improvement of the weighting function and logarithm-cepstral domain mapping for the uncertainty variances; the introduction of the delta and acceleration coefficients in the WMA context.

In the next chapter the problem of speech recognition when the signal is corrupted by additive and convolutional noises is addressed and it is shown that both noises could be cancelled by means of SS and Cepstral Mean Normalization (CMN) in combination with WMA. Both techniques are easily implemented and the results are interesting from the practical application point of view.

Chapter 7

Additive and convolutional noise removal

7.1 Introduction

In the previous chapters reliability in noise cancelling was used to weight the information along the signal firstly in the context of DTW or then with HMM. Several additive noises were considered and in all the cases a substantial improvement in the recognition accuracy was observed when SS was combined with the weighted DTW and Viterbi algorithm.

In this chapter, the problem of additive and convolutional noise removal is addressed and it is proposed that the effect of the transmission channel function can be removed after the additive noise has been removed by means of SS. Two convolutional techniques are addressed and applied in combination with SS: Cepstral Mean Normalization (CMN) and Maximum Likelihood estimation (MLE). When SS and CMN are applied together, it is shown that the weighted Viterbi algorithm with temporal constraints also leads to a substantial reduction in the error rate and a high recognition accuracy is achieved at SNR equal to 18dB (error rate less than 1%), at 12dB (error rate less than 3%) and in some cases at 6dB (error rate less than 10%) when the speech signal is corrupted by additive noise and is distorted by a 6dB/oct spectral tilt. The tilt used by the Noisex database is a flat frequency response up to a break point frequency of 250Hz followed by a 3dB/oct tilt above 250Hz. Due to the fact that in this thesis the data was downsampled from 16 to 8kHz and the band width was reduced from 8 to 4kHz, the 3dB/oct tilt did not introduce a high recognition error and in order to make the testing conditions more severe the tilt was increased to 6dB/oct.

Additive and convolutional noises are the main problems to be solved in order to make speech recognition successful in real applications (telephony, car, office, etc) and the results presented in this chapter suggest that the weighted Viterbi algorithm allows to achieve a low error rate at moderate SNR's using no information about the transmission channel function and a simple estimation of the additive noise made in short non-speech intervals, which in turn allows the method to capture reasonably well the dynamics of the corrupting signal. Moreover, the simplicity of the restrictions concerning the additive and convolutional noises makes the method here proposed interesting from the practical application point of view.

7.2 Influence of the transmission channel

It is well known that the mismatch between training and testing conditions increases the error rate of recognition systems to unacceptable levels. This mismatch is generally modelled as being the result of two types of distortion (Fig. 2.5): additive, $n(i)$, and convolutional, $h(i)$, noise. The additive noise, addressed in the previous chapters, corresponds to the addition of an external corrupting signal to the speech, and can be modelled as an additive process in the linear domain either in the temporal or frequency domain. The corrupting and corrupted signals are generally considered uncorrelated. On the other hand, the convolutional distortion is caused by the change or insertion of the transmission channel (microphone and telephone line) and it is generally modelled as an additive process in the logarithmic and cepstral domain. It is called convolutional because the transmission channel could be modelled as a filter, generally assumed linear, and the distorted speech signal is the result of the *convolution* of the filter impulsive response with the speech signal in the temporal domain. In the frequency domain, this convolution becomes a multiplication, which in turn becomes a sum after applying the logarithmic function.

Both distortions are conceptually very diffferent. As far as stationarity is concerned, nothing could be said about the additive noise except it can be stationary or non-stationary. However, all the methods that address the additive noise problem need to assume that it is stationary at least during a word or utterance (or between two consecutive non-speech intervals) and the ability in capturing the noise dynamic is a main issue for noise removal techniques. Moreover, as discussed in the previous chapters, the additive distortion corrupts some segments of the speech

signal more severely than others and in order to explore this characteristic WMA (Weighted Matching Algorithms), proposed in this research, suggests that the classical acoustic pattern matching process where all the frames have the same weight should be revised in order to include the reliability in noise cancelling frame-by-frame. In contrast, the convolutional noise is assumed constant with the time because it depends on the physical characteristic of the transmission channel which are supposed not to change in short periods of time. Besides this, the convolutional distortion equally corrupts all the frames and the concept of local SNR loses sense in this case. If the gain introduced by the transmission channel is considered constant inside each one of the 14 DFT Mel filters the convolutional distortion can be represented by

$$H = [H_1, H_2, H_3, ..., H_m, ...H_{14}]$$

and the distortion at the output of every filter is simply given by

$$\log(H_m \cdot \overline{s_m^2}) \simeq \log(\overline{s_m^2}) + H_m^l \tag{7.1}$$

where $H_m^l = \log(H_m)$. This and the fact that H is constant with time (or at least varies slowly) make the convolutional noise much easier to deal with than the additive noise when the latter is not present. It is interesting to highlight that due to the fact that the cepstral transform is linear, the convolutional distortion is also modelled by means of an additive constant in the cepstral domain.

7.3 Convolutional noise cancelling

The techniques that address the problem of convolutional noise assume that the distortion is additive in the log domain, equally distorts all the frames and it is constant with the time. If there is only convolutional noise, the techniques work reasonably well in practical applications which shows that the assumptions about the distortion introduced by the transmission channel are a good approximation for the problem.

The most popular technique is Cepstral Mean Normalization. It consists in subtracting from every Mel-cepstral coefficient the coefficient mean estimated in an interval of speech signal:

$$c'_{t,n} = c_{t,n} - \overline{c_{t,n}} \tag{7.2}$$

where

$$\overline{c_{t,n}} = \frac{\sum_{k=1}^{L_S} c_{k,n}}{L_S} \tag{7.3}$$

where $c_{t,n}$ denotes the cepstral coefficient n at time t, $c'_{t,n}$ is the cepstral coefficient after having the mean $\overline{c_{t,n}}$ subtracted, and L_S is the length in frames of the speech signal available to compute the coefficient mean. The main advantage of CMN is that it can efficiently cancel the transmission channel influence that can be represented as a constant in the logarithmic or cepstral domain, given that L_S is big enough in order to reliably estimate $\overline{c_{t,n}}$.

Another technique that has widely been used in telephony applications is Rasta filtering (H.Hermansky *et al.*, 1991) (J.Koehler *et al.*, 1994) (H.Hermansky *et al.*, 1993) and it consists in band-pass filtering the trajectory of log energies along the time. Given that the convolutional distortion is time invariant (or varies slowly), it is reasonably well cancelled in practical applications because Rasta removes the dc component of the spectral coefficients trajectory. The disadvantages of the method are: it may increase the dependence of the data on its previous context (J.Koehler *et al.*, 1994); tends to cancel sustained phonemes, although this is not very relevant for practical applications; and does not necessarily lead to better results than Cepstral Mean Normalization (CMN) (Veth & Boves, 1996) despite the fact of being computationally more expensive. However, Rasta does not have the limitation concerning the length of the speech sample that CMN does.

MLE (Maximum Likelihood estimation) (A.Acero & R.Stern, 1990) (P.Moreno, 1996) (B.Raj *et al.*, 1996) is another technique that has been applied to remove the distortion caused by transmission channels generally in combination with additive noise. The method uses stochastic model (code-book) of the clean speech signal and the maximum likelihood criteria to estimate the convolutional (and also the additive) noise by means of the EM algorithm (see Appendix B). The main disadvantages of MLE are the high computational complexity, the fact that the EM algorithm employs the gradient technique and the global optimum is not guaranteed, and finally the fact that the code-book needs to reliably represent the speech process.

In this thesis, CMN and MLE were implemented and employed to cancel firstly only convolutional noise and finally in combination with WMA and SS to cancel both convolutional and additive noise. Rasta filtering was not addressed in this research because it is not directly applied

with WMA.

7.4 Additive and convolutional noise cancellation

When the speech signal is corrupted by both types of noise all the approaches discussed in the last section lose their effectiveness, and they need to be generalized or be applied in combination with other techniques. CMN loses its applicability and its behaviour is hard to predict when additive noise is also present (Gales, 1995) (Gales & S.Young, 1995).

Rasta filtering in its original form can not be used when the speech signal is corrupted by additive and convolutional noises and a modified version, J-Rasta (H.Hermansky *et al.*, 1993), was proposed and shown to improve the robustness to both noises. However, J-Rasta depends on a constant that cannot be analytically estimated and is case dependent which reduces the applicability of the method.

The problem of additive and convolutional noise estimation and removal using MLE has been the subject of a sequence of papers (A.Acero & R.Stern, 1990) (F.H.Liu *et al.*, 1992) (P.Moreno, 1996) (B.Raj *et al.*, 1996). The disadvantages are the same as the ones discussed in the last section plus the fact that the additive noise needs to be considered stationary during the utterance where the algorithm is running, and if there is some non-stationarity the technique may converge to a wrong solution. CDCN (A.Acero & R.Stern, 1990) (F.H.Liu *et al.*, 1992) and VTS (P.Moreno, 1996) (B.Raj *et al.*, 1996) estimate both additive and convolutional distortions using stochastic model of the speech process and in order to reliably estimate the noises a minimum amount of data should be necessary. In other words, the longer is the speech interval used to run the EM algorithm, the better is the noise estimation. This is perfectly coherent with the fact that the distortion introduced by the transmission channel is time invariant but does not agree with the additive noise whose stationarity is always a critical issue in practical applications.

7.5 SS and convolutional noise cancellation

In Chapter 5 a model for additive noise was proposed and it was suggested that the hidden clean information of the speech signal is a function of the observed noisy signal energy $\overline{x_m^2}$, the noise energy $\overline{n_m^2}$ and the phase difference ϕ_m between the clean signal and noise in channel m.

Using (5.17 in page 68) and assuming that the random variables ϕ and $\overline{n_m^2}$ are uncorrelated, ϕ is uniformly distributed between $-\pi$ and π and that $\overline{n_m^2}$ is concentrated near its mean $E[\overline{n_m^2}]$, it is possible to show that (section 5.5)

$$E[\log(\overline{s_m^2})|\overline{x_m^2}] \simeq \log(\overline{x_m^2} - E[\overline{n_m^2}]) \tag{7.4}$$

This result means, according to the model for additive noise, that the expected value of the hidden information $\log(\overline{s_m^2})$ is equal to the log of the SS estimation if SS is defined as being $\overline{x_m^2} - E[\overline{n_m^2}]$. If the gain introduced by the transmission channel is considered constant inside each one of the 14 DFT Mel filters and constant with the time, the expected value of the hidden clean signal information when the signal is also distorted by the convolutional distortion is given by

$$E[\log(H_m \cdot \overline{s_m^2})|\overline{x_j^2}] \simeq E[\log(\overline{s_m^2})|\overline{x_j^2}] + H_m^l \tag{7.5}$$

Therefore, the convolutional distortion could be effectively removed by means of CMN (an easily implemented technique) after the additive noise being firstly cancelled using SS. This result, although simple, is very interesting and important from the practical point of view. It suggests that the additive noise should be cancelled before the convolutional one or, in other words, the result of the additive noise removal (SS) should be used by a convolutional noise removal technique (e.g. CMN). This fact is extremely coherent with the nature of both types of noises: the additive one can be stationary and the other one could be considered constant with time. It means that the noise energy estimation needs to be subtracted from the noisy energy before applying the convolutional noise removal, and the noise energy estimation could be estimated at least between two non-speech intervals which allows to follow reasonably well the dynamic of the corrupting signal. However, if SS and CMN are applied in sequence using the common recognition algorithms the error rate is still poor and the explanation for this is that although the convolutional noise corrupts equally all the frames, the additive noise does not, and a substantial improvement was observed when the ordinary matching algorithms were replaced with WMA's. This is also coherent with (H.Hermansky *et al.*, 1993) where it is said that "results essentially confirm (A.Acero & R.Stern, 1990) which reports negative experience with cascading two systems, one dealing with the additive and other with the convolutional

noise". Actually, in the following sections it is shown that cascading SS with CMN leads to poor results when the speech signal is corrupted with additive noise and a 6dB/oct spectral tilt, and the recognition is done by means of the ordinary Viterbi (HMM) algorithm. However, results are substantially improved when the weighted Viterbi algorithm is used instead of the ordinary one.

7.6 Experiments with only convolutional noise cancelling

The tests were carried out with isolated digits employing both speakers of Noisex database (one female and one male), with signal corrupted only by convolutional noise that was introduced with a 40-tap FIR filter whose spectral response is approximatly flat until 250 Hz and then falls with a 6dB/oct slope (Fig. 7.1). The same pre-processing that was used in Chapter 6 was applied

Figure 7.1: Frequency response of the FIR filter used to introduce the convolutional distortion.

here. At the output of each channel the log energy was computed and 10 cepstral coefficients

Table 7.1: Convolutional noise removal with CMN. Recognition error rate(%) for signal distorted by a 6dB/Oct spectral tilt.

Noise removal	*Without CMN*	*With CMN*
Vit	13	0
Vit-Mm-Gamma	5	0

were estimated in every frame. Each word was modelled using an 8-state left-to-right HMM without skip-state transition (Fig. 6.2), with a single multivariate Gaussian density per state and a diagonal covariance matrix. CMN and MLE were tested firstly with the ordinary Viterbi algorithm, Vit, and then the temporal constraints using max and min duration and gamma distribution were introduced $Vit - Mm - Gamma$. HTK V.2.0 with modifications to include the temporal constraints in the testing procedure was used for the experiments.

7.6.1 Convolutional noise cancellation with CMN

CMN was implemented according to (7.2) and (7.3) and the coefficient mean was initially computed using one utterance per word of the vocabulary (digits) every time. It means that the speech sample that was used to estimate the means was composed by ten words. Results are presented in Table 7.1. As can be seen in Table 7.1, the temporal constraints helped to reduce the effect of the spectral tilt, but CMN completely removed the convolutional noise with and without state duration modelling.

7.6.2 Convolutional noise cancellation with MLE

In these experiments the transmission channel response was estimated in the cepstral domain by means of the Maximum Likelihood criteria using a variant of the well-known EM algorithm (X.D.Huang *et al.*, 1990) (T.K.Moon, 1996). Due to the fact that the ML estimation was used to estimate exclusively the convolutional distortion H^c, the re-estimation procedure had an analytical solution. If the same method is used to estimate both additive and convolutional noises the algorithm does not have a closed form solution and vector Taylor series approximation are needed (P.Moreno, 1996) (B.Raj *et al.*, 1996). The recursive algorithm to estimate H^c is given by:

1. Initial values for $Pr[cw_j] = \frac{1}{L}$ (for $j = 1, 2, ..., L$) and $H_i^c = 0.0$ (for $i = 1, 2, ..., d$) ;

2. Compute $Pr[cw_j|T_t, \varphi_j, H^c]$:

$$Pr[cw_j|T_t, \varphi_j, H^c] = \frac{f[T_t|cw_j, \varphi_j, H^c] \cdot Pr[cw_j]}{\sum_{j=1}^{L} f[T_t|cw_j, \varphi_j, H^c] \cdot Pr[cw_j]} \quad (7.6)$$

3. Re-estimate $Pr[cw_j]$:

$$\hat{Pr}(cw_j) = \frac{1}{L_T} \sum_{t=1}^{L_T} Pr(cw_j|T_t, \varphi_j, H^c) \quad (7.7)$$

4. Re-estimate H^c:

$$\hat{H}_i^c = \frac{\sum_{j=1}^{L} \sum_{t=1}^{L_T} Pr[cw_j|T_t, \varphi_j, H^c] \cdot \frac{T_t - \mu_{j,i}}{\sigma_{j,i}^2}}{\sum_{j=1}^{L} \sum_{t=1}^{L_T} \frac{Pr[cw_j|T_t, \varphi_j, H^c]}{\sigma_{j,i}^2}} \quad (7.8)$$

5. Stop if convergence has been reached, otherwise go to Step 2.

where $T_t = [T_{t,1}, T_{t,2}, T_{t,3}, ..., T_{t,d}]$ denotes the testing frame in the cepstral domain and d is the number of cepstral coefficients; L_t is the size in frames of the speech sample used to estimate H^c; L is the size of the codebook generated with clean signals; $Pr[cw_j]$ is the *a priori* probability of the codeword cw_j; $f[T_t|cw_j, \varphi_j, H^c]$ is a multivariate Gaussian distribution representing the category j conditional pdf; φ_j denotes the mean vector $\mu_j = [\mu_{j,1}, \mu_{j,2}, ..., \mu_{j,d}]$ and the d-by-d diagonal covariance matrix whose diagonal elements are $\sigma_{j,d}^2$.

Two codebooks (one for each speaker) with 32 codewords each were generated using the classical LBG algorithm (Y.Linde *et al.*, 1980) and clean speech signal from both speakers of the Noisex database (one male and one female).

The convolutional noise was the same as in section 7.6.1 (Fig. 7.1). The deduction of the algorithm is presented in the Appendix B and it was assumed that the transmission channel gain is considered constant inside each one of the filters, does not depend on the signal level and does not affect the covariance matrices. Consequently, the mean of all the codewords is shifted by H^c: $\mu_j' = \mu_j + H^c$. After H^c has been estimated it was subtracted from T_t. The same pre-processing configuration adopted in section (7.6.1) was used here and d = 10, and H^c was computed using one utterance per word of the vocabulary (digits) every time. In other words, L_T corresponded to the number of frames in ten words. Results are presented in Table 7.2. According to Table 7.2, the EM algorithm discussed in this section completely removed the convolutional noise with

Table 7.2: Convolutional noise removal with ML. Recognition error rate(%) for signal distorted by a 6dB/Oct spectral tilt.

Noise removal	*Without MLE*	*With MLE*
Vit	13	0
Vit-Mm-Gamma	5	0

and without state duration modelling, although with a much higher computational complexity than CMN.

7.7 Reliability in convolutional noise cancelling

Although the convolutional noise is independent of the signal level, the additive noise is not and a reliability (based on the uncertainty variance) in additive noise cancelling is defined. However, the convolutional distortion removal, according to the approach here proposed, employs the result of the SS estimation and also needs to be analysed under the perspective of reliability in noise removal.

If frames are supposed uncorrelated, the same assumption made by HMM, the uncertainty variance after CMN (7.2) (7.3) is given by

$$Var[c'_{t,n}|X,N] = (1-\frac{1}{L_T})^2 \cdot Var[c_{t,n}|X_t,N] + \frac{1}{L_T^2} \cdot \sum_{k=1,k\neq t}^{L_T} Var[c_{k,n}|X_k,N] \quad (7.9)$$

where

$$X_t = [\overline{x^2_{t,1}}, \overline{x^2_{t,2}}, \overline{x^2_{t,3}}, ..., \overline{x^2_{t,M}}]$$

is the set of observed energies of the noisy signal in a Mel filter bank at time t, M is the number of filters,

$$X_t = [X_1, X_2, X_3, ..., X_t, ..., X_{L_T}]$$

is the observation sequence of the noisy signal energy and

$$N = [E[\overline{n_1^2}], E[\overline{n_2^2}], E[\overline{n_3^2}], ..., E[\overline{n_M^2}]$$

is the noise energy estimation.

On the other hand, the convolutional noise cancelling using MLE can be defined as

$$c'_{t,n} = c_{t,n} - H^c_n \tag{7.10}$$

where H^c_n is computed by means of an iterative algorithm (EM) which in turn uses non-linear functions (i.e. Gaussian pdf's). These characteristics make the estimation of H^c_n difficult to model in terms of the uncertainty variance and in the experiments with SS/MLE the weighting function included only the reliablity in additive noise cancelling.

7.8 Experiments with additive and convolutional noise cancelling using SS and CMN

These experiments were also carried out with isolated digits employing the two speakers (one female and one male), and five noises from the Noisex database (car, speech, Lynx, operation room and factory) (A.Varga *et al.*, 1992). The same pre-processing that has been previously used was adopted here. At the output of each channel the energy was computed, SS according to (6.7) was applied, and the log of the energy was estimated. The $SsThr_m$ threshold for SS was estimated as explained in chapters 5 and 6 (Compernolle, 1989). In every frame 10 cepstral coefficients were computed. When CMN was applied after SS, the coefficient mean was computed using one utterance per word of the vocabulary (digits) every time.

In the Noisex database, all the utterances for given speaker-noise-SNR configuration are in sequence and separated by approximately 1 sec in a single file. In the last two chapters, the noise energy was estimated only once using 200 ms of non-speech samples at the beginning of the files and the noise estimation was kept constant for all the utterances of a given speaker-noise-SNR. This procedure must have made the testing conditions more severe when the problem of additive noise was addressed, but in order to test the convolutional and additive noise removal techniques the noise was estimated in 100 ms of non-speech samples at the beginning of every utterance. The convolutional noise experiments were performed with the spectral tilt composed by a flat frequency response up to a break point frequency of 200Hz followed by a +6dB/oct tilt above 250Hz (Fig. 7.1) applied to the noisy signals.

The uncertainty variance was estimated according to (7.9) and the weighting coefficient was computed using (6.18). Each word was modelled using an 8-state left-to-right HMM without

Table 7.3: SS and CMN. Recognition error rate(%) for speech corrupted by additive noise (car) and spectral tilt.

SNR	*18dB*	*12dB*	*6dB*	*0dB*
Vit-Mm-Gamma	27.5	40.0	50.0	57.5
SS-W2-Vit-Mm-Gamma	17	22	27	31
SS-CMN-W2-Vit-Mm-Gamma	0.5	0.5	2.5	19.5
SS-CMN-Vit-Mm-Gamma	0.5	4.5	14.5	44

Table 7.4: SS and CMN. Recognition error rate(%) for speech corrupted by additive noise (speech) and spectral tilt.

SNR	*18dB*	*12dB*	*6dB*	*0dB*
Vit-Mm-Gamma	31	44	50.5	60.5
SS-W2-Vit-Mm-Gamma	19	23	24.5	41.5
SS-CMN-W2-Vit-Mm-Gamma	0	0.5	8	32
SS-CMN-Vit-Mm-Gamma	3	6.5	31	69

skip-state transition (Fig. 6.2), with a single multivariate Gaussian density per state and a diagonal covariance matrix. All the experiments used the Viterbi algorithm with max and min state duration plus state duration distribution with gamma pdf (section 6.6.1). HTK V.2.0 with modifications to include the temporal constraints and reliability weighting in the testing procedure was used for the HMM experiments.

When CMN was applied, the training utterances were also processed with CMN. Consequently, two sets of HMM's were used: one to run experiments with SS only and another to test SS/CMN. The following configurations were employed: firstly, the ordinary Viterbi algorithm with temporal constraints, $Vit - Mm - Gamma$; secondly, SS and the weighted version of the Viterbi algorithm were introduced, $SS - W2 - Vit - Mm - Gamma$; then, CMN was applied after SS, $SS - CMN - W2 - Vit - Mm - Gamma$; and finally, the weighted Viterbi algorithm was replaced with the ordinary one, $SS - CMN - Vit - Mm - Gamma$. Results are shown in Tables 7.3-7.7 As can be seen in Tables 7.3-7.7, the convolutional noise introduced a high error rate at all the SNR's when no noise cancelling method was applied $Vit - Mm - Gamma$. The introduction of SS and weighting $SS - W2 - Vit - Mm - Gamma$ tends to improve the results but a high improvement is achieved when CMN is applied in sequence with SS, $SS - CMN - W2 - Vit - Mm - Gamma$. In order to evaluate the contribution of the weighting procedure the weighted Viterbi algorithm was removed and compared to

Table 7.5: SS and CMN. Recognition error rate(%) for speech corrupted by additive noise (Lynx) and spectral tilt.

SNR	*18dB*	*12dB*	*6dB*	*0dB*
Vit-Mm-Gamma	31	38	52	72.5
SS-W2-Vit-Mm-Gamma	17	19	25.5	45
SS-CMN-W2-Vit-Mm-Gamma	1.5	3	11	36
SS-CMN-Vit-Mm-Gamma	2.0	7.5	32.5	69

Table 7.6: SS and CMN. Recognition error rate(%) for speech corrupted by additive noise (operation room) and spectral tilt.

SNR	*18dB*	*12dB*	*6dB*	*0dB*
Vit-Mm-Gamma	26	39.5	50	58.5
SS-W2-Vit-Mm-Gamma	17.5	23	30.5	40.5
SS-CMN-W2-Vit-Mm-Gamma	0	2	13	38
SS-CMN-Vit-Mm-Gamma	0.5	7	23.5	60

$SS-CMN-W2-Vit-Mm-Gamma$, $SS-CMN-Vit-Mm-Gamma$ resulted in a substantial increase of the error rate.

Results with SS and CMN basically confirm (7.5) and a high recognition accuracy is achieved at SNR equal to 18dB (error rate less than 1%), at 12dB (error rate less than 3%) and in some cases at 6dB (error rate less than 10%), with several types of additive noises and a strong convolutional distortion. However, a higher error rate was observed when SS and CMN try to cancel both types of distortions than when SS is used to remove only the additive noise. The SS threshold $SsThr_m$ has an important role in the estimation of the coefficient means, specially at those frames with low local SNR, and the effect of the rectifying function can not be accurately modelled because in principle, if it is assumed that the noise may be non-stationary, there are not enough samples of noise. Moreover, the introduction of the noise energy distribution increases the computational complexity of the additive model. In (N.B.Yoma *et al.*, 1997a), a low $SsThr_m$

Table 7.7: SS and CMN. Recognition error rate(%) for speech corrupted by additive noise (factory) and spectral tilt.

SNR	*18dB*	*12dB*	*6dB*	*0dB*
Vit-Mm-Gamma	22.5	38.5	45.5	54
SS-W2-Vit-Mm-Gamma	16	24.5	30	42
SS-CMN-W2-Vit-Mm-Gamma	0.5	2.5	11	30
SS-CMN-Vit-Mm-Gamma	1	4	17	53

was adopted and the uncertainty variance in the log domain was used to weight the estimation of the coefficient mean employed by CMN. This procedure was shown to substantially increase the recognition accuracy for that case, but it did not give the same result when $SsThr_m$ was estimated according to (Compernolle, 1989) as done here. Due to the fact that the estimation according to (Compernolle, 1989) gave better result than an arbitrary low value for $SsThr_m$ without the weighted arithmeic mean, the problem of the rectifying function compensation was not considered in this thesis. Nevertheless, for being an important problem to improve the performance of SS in combination with CMN, the interaction between the SS threshold and CMN will be addressed in a future work.

Finally, it is worth emphasizing that the weighted Viterbi algorithm was effective and dramatically reduced the error when SS and CMN were combined resulting in a low computational complexity when compared with PMC and CDCN (or VTS). This is a very interesting result from the practical application point of view because SS and CMN are easily implemented techniques, no information about the convolutional distortion was required and a minimum information about the additive noise was needed.

7.9 Experiments with additive and convolutional noise cancelling using SS and ML estimation for H^c

In these experiments the transmission channel response was estimated in the cepstral domain by means of the same algorithm discussed in section 7.6.2 for ML estimation after SS. The experiments with SS/MLE were done using the same pre-processing and environment as the one employed for SS/CMN in section 7.8 and the same convolutional noise (6dB/oct spectral tilt) was introduced in the noisy signal. The same codebooks with 32 codewords were also employed here and the EM algorithm estimated H^c using one utterance per word of the vocabulary (digits). As explained in section 7.7, the weighting function did not take into consideration the uncertainty in the estimation of H^c due to the complexity of the EM algorithm.

The following configurations were employed: SS in sequence with MLE using the ordinary Viterbi algorithm with temporal constraints, SS – MLE – Vit – Mm – Gamma; and finally, the weighted Viterbi algorithm was applied, SS – MLE – W2 – Vit – Mm – Gamma. Results are presented in Tables 7.8-7.12 which also includes the error rates with no noise cancelling and

Table 7.8: SS and MLE. Recognition error rate(%) for speech corrupted by additive noise (car) and spectral tilt.

SNR	*18dB*	*12dB*	*6dB*	*0dB*
Vit-Mm-Gamma	27.5	40	50	57.5
SS- MLE -Vit-Mm-Gamma	0	0.5	17	37
SS- MLE - W2-Vit-Mm-Gamma	1	0.5	6.5	33
SS- CMN - W2-Vit-Mm-Gamma	0.5	0.5	2.5	19.5

Table 7.9: SS and MLE. Recognition error rate(%) for speech corrupted by additive noise (speech) and spectral tilt.

SNR	*18dB*	*12dB*	*6dB*	*0dB*
Vit-Mm-Gamma	31	44	50.5	60.5
SS- MLE -Vit-Mm-Gamma	0	4.5	27	63
SS- MLE - W2-Vit-Mm-Gamma	0	1.5	20.5	40
SS- CMN - W2-Vit-Mm-Gamma	0	0.5	8	32

with SS/CMN, $SS - CMN - W2 - Vit - Mm - Gamma$ (Tables 7.3-7.7), for comparison.

As can be seen in Tables 7.8-7.12, SS/MLE was able to substantially reduce the effect of both additive and convolutional noises confirming (7.5). However, the improvement due to the weighted Viterbi algorithm was much lower than for SS/CMN. This should be due to the fact that the the ML estimation of H^c was not modelled under the perspective of reliability and the weighting function did not take into consideration the uncertainty variance of the EM estimation of the convolutional noise.

7.10 Discussion and conclusions

In this chapter, the problem of additive and convolutional noise cancelling was addressed. It is proposed that the convolutional distortion could be removed after the additive noise has been cancelled by means of SS. This result, although simple, has not been reported before and

Table 7.10: SS and MLE. Recognition error rate(%) for speech corrupted by additive noise (Lynx) and spectral tilt.

SNR	*18dB*	*12dB*	*6dB*	*0dB*
Vit-Mm-Gamma	31	38	52	72.5
SS- MLE -Vit-Mm-Gamma	0	4.5	27	63
SS- MLE - W2-Vit-Mm-Gamma	2.5	3.5	20.5	48
SS- CMN - W2-Vit-Mm-Gamma	1.5	3	11	36

Table 7.11: SS and MLE. Recognition error rate(%) for speech corrupted by additive noise (operation room) and spectral tilt.

SNR	*18dB*	*12dB*	*6dB*	*0dB*
Vit-Mm-Gamma	26	39.5	50	58.5
SS- MLE -Vit-Mm-Gamma	0	5.5	22	54
SS- MLE - W2-Vit-Mm-Gamma	1.5	3.5	19	46
SS- CMN - W2-Vit-Mm-Gamma	0	2	13	38

Table 7.12: SS and MLE. Recognition error rate(%) for speech corrupted by additive noise (factory) and spectral tilt.

SNR	*18dB*	*12dB*	*6dB*	*0dB*
Vit-Mm-Gamma	22.5	38.5	45.5	54
SS- MLE -Vit-Mm-Gamma	0	1.5	18.5	48
SS- MLE - W2-Vit-Mm-Gamma	0.5	3.0	15	37
SS- CMN - W2-Vit-Mm-Gamma	0.5	2.5	11	30

has important implications from the practical application point of view. Firstly, SS needs just the noise energy estimation that can be done in two consecutive non-speech intervals and can capture reasonably well the dynamic of the corrupting signal. Secondly, the weighted Viterbi algorithm assures that a high recognition accuracy can be achieved at moderate SNR's even if the noise estimation is not accurate (i.e. done in short nonspeech intervals). Finally, cancelling firstly the additive noise and the the convolutional one is coherent with the nature of both types of distortions since stationarity is a critical issue to tackle the first one but the second can be considered constant with the time.

Two convolutional noise removal techniques were addressed: CMN (Cepstral Mean Normalization) and MLE (maximum likelihood estimation) of the convolutional distortion. Both methods were shown to work well when there was not additive noise although MLE requires a clean speech model and a much higher computational complexity than CMN. However, it is worth mentioning that the EM algorithm has a closed solution because it is used to estimate only the convolutional noise. Afterwards, both techniques were applied after SS when the speech signal was corrupted by several types of noises and results basically confirmed the validity of the additive-convolutional noise cancelling approach, although different results were observed specially at SNR=6dB. The weighted Viterbi algorithm was shown to substantially reduce the error rate when SS was applied in combination with CMN (easily implemented techniques) and a

high recognition accuracy was achieved at SNR equal to 18dB (error rate less than 1%), at 12dB (error rate less than 3%) and in some cases at 6dB (error rate less than 10%) using no information about the convolutional noise and a simple estimation of the corrupting signal energy in a Mel filter bank. However, when CMN was replaced with MLE, the weighted Viterbi algorithm did not lead to the same order of reduction in the error rate and this must have resulted from the fact that the reliability in the estimation of the convolutional noise was not included in the weighting function due to the complexity of the EM algorithm.

When compared to SS/MLE, SS/CMN with the weighted algorithm gave better results with a much lower computational complexity and without a clean speech model. Nevertheless, the performance of CMN (and MLE) is related to the SS threshold ($SsThr_m$) but a conventional estimation for this threshold was adopted and the problem was not addressed in this research.

Chapter 8

Conclusions

8.1 Summary of results

This thesis has investigated the problem of automatic speech recognition in noise (additive and convolutional) under the perspective of Weighted Matching Algorithms (WMA). The basic idea is that additive noise corrupts some segments of the speech signal more severely than others and WMA revises the classical concept of acoustic pattern matching in order to include the segmental signal to noise ratio (SNR) frame-by-frame. WMA is a new topic with important implications from the practical and theoretical point of view, has proved to alleviate the restrictions imposed by previous techniques and can set new bounds for robust speech recognition in more complex applications. Moreover, WMA are plausible of being generalized to other fields of robust pattern recognition given the generality of the approach. The problem of end-point detection has also been addressed and a method based on autoregressive analysis of noise is also proposed for robust speech pulse detection. The technique has been shown to be effective in increasing the discrimination between the speech signal and background noise.

All the experiments in this thesis were done using the Noisex-92 database mainly with isolated word (digits) recognition although the applicability of WMA to the problem of connected words was also evaluated in Chapter 6. Five types of noise presenting different degree of difficulty (stationarity and spectral distribution) were addressed despite the fact that some of these noises are not commonly tackled in the literature probably because they represent a more difficult problem. Although the tests were speaker dependent, the combination of WMA with spectral subtraction (SS), Chapters 5-7, does not need any information about the speaker and should

be easily generalised to the speaker independent case. While the correction coefficients of the additive noise model (sections 5.2.1 and 5.3.1) are actually estimated using only-noise samples and the speaker's clean speech signal, these coefficients did not show much variation across the noises and the two speakers (one male and one female) of the Noisex database. Consequently, a speaker independent system could use the average of correction coefficients estimated for a set of speakers and noises.

Initially, a weighted DP algorithm was proposed and tested with a novel noise cancelling neural net (LIN) and proved to be effective in reducing the error rate when the weighting coefficient was defined as being the ratio between the estimated clean and noisy signal energies for white Gaussian noise. However, it was noticed that the improvement due to the weighted algorithm depended on the neural net training condition and this suggested that the weighting coefficient should also take into consideration the efficacy or uncertainty (inverse of reliability) in noise removal of LIN. Uncertainty can be defined as being the mean quadratic distance between the clean signal estimation and the original clean signal and, in the context of LIN was estimated using mean distortion curve.

Due to the fact that LIN presented a slow training and certainly a low adaptability, the replacement of the neural net with a conventional noise cancelling technique became imperative and SS was chosen due to its simplicity and ability to follow changes in the dynamics of the corrupting signal. In order to be used in combination with WMA, SS was modelled in terms of reliability in noise cancelling and to do so a novel model for additive noise was proposed. This model uses a bank of Mel filters (IIR or DFT) and sets the hidden information of the clean signal energy as function of the observed noisy signal energy, the noise energy (that can be approximately estimated) and the phase difference between the clean speech and noise signals. It is proposed that when the noise is added an uncertainty is introduced and the original clean signal cannot be recovered with 100% accuracy because the phase difference between corrupted and corrupting signals is unknown. Consequently, the hidden clean signal energy is treated as a stochastic variable and it is proved that the SS itself corresponds to the expected value of the clean signal energy given the noisy and noise signal energies. Similarly, the uncertainty in noise cancelling

was estimated as being the variance of the hidden clean signal energy given the noisy and noise signal energies, and was used to estimate the weighting coefficients for another one-step weighted DTW. The weighted one-step algorithm here proposed had superior performance to a two-step one previously presented and substantially reduced the error rate, but the weighting function used a threshold whose optimum value was still case dependent, although a wide range of acceptable sub-optimal values was achieved. Another consequence of the additive noise model, used later in Chapter 7, suggested that the convolutional noise could be removed after the additive one has been cancelled by means of SS.

In sequence, the weighted Viterbi (HMM) algorithm was proposed and applied with a weighting function that does not need any free variable (see section 6.4), and compared and combined with state duration modelling. The new weighting function uses the uncertainty and HMM's variances and led to better results than the previous weighting function. It was shown that weighting the information along the signal led to better results than the introduction of temporal constraints in the recognition algorithm and, with temporal constraints, the weighted Viterbi algorithm resulted in a high recognition accuracy at SNR equal to 18dB (error rate less than 1%), at 12dB (error rate less than 3%) and in some cases at 6dB (error rate less than 10%) without an accurate noise model. Experiments with temporal constraints suggest that the gamma distribution, although it fits better the duration of states, does not necessarily lead to better results and that the restrictions imposed by the max and min durations are the main factor responsible for the reduction in the error rate. It is interesting to mention that temporal constraints could also be speaker independent and hence independent of the testing environment. Tests with connected digits without temporal constraints showed that WMA could be useful to reduce the error rate although further work is needed to integrate WMA and temporal constraints in the context of connected or continuous word recognition.

As far as additive and convolutional noises are concerned, both can be simultaneously cancelled using WMA. It is proposed that the convolutional distortion can be removed after the additive noise has been cancelled by means of SS. This result, although simple, has not been reported before and, as explained before, has important implications from the practical application point

of view. Cancelling firstly the additive noise and then the convolutional one is coherent with the nature of both types of distortions since stationarity is a critical issue to tackle the first one but the second can be considered constant along the time. Two convolutional noise removal techniques were addressed: CMN (Cepstral Mean Normalization) and MLE (maximum likelihood estimation) of the convolutional distortion. Although the EM algorithm has a closed solution because it is used to estimate only the convolutional noise, it is hugely more complex than CMN. When the speech signal was corrupted by several types of additive noise and spectral tilt, CMN and EM were applied after SS and results basically confirmed the validity of the additive-convolutional noise cancelling approach, although different results were observed specially at SNR=6dB. The weighted Viterbi algorithm was shown to substantially reduce the error rate when SS was applied in combination with CMN (easily implemented techniques) and a high recognition accuracy (as explained above) was achieved at SNR=12dB and in some cases at 6dB using no information about the convolutional noise and a simple estimation of the corrupting signal energy in a Mel filter bank. However, the weighted Viterbi algorithm did not show the same improvement with MLE, and this must have resulted from the fact that the reliability in the estimation of the convolutional noise was not included in the weighted function due to the complexity of the EM algorithm. Compared to SS/MLE, SS/CMN with the weighted algorithm gave better results with a much lower computational complexity and without a clean speech model.

Finally, it seems reasonable to suppose that WMA could be applied with other noise removing methods as long as they are properly modelled in terms of reliability in noise cancelling (refer to section 7.7).

8.2 Future work

Results presented in this thesis are encouraging and, due to the fact that WMA is a new topic and had not been studied before, there is a lot of room for generalizations. Firstly, the study of the applicability of WMA and temporal constraints to the problem of connected or continuous speech recognition and wordspotting needs to be done.

All the experiments in this thesis were carried out with static Mel frequency cepstral coefficients

(MFCC). However, static parameters are not enough to deal with bigger vocabularies and the use of delta and delta2 parameters becomes imperative. Consequently, parameters delta and delta2 should be studied under the perspective of WMA and reliability in noise cancelling in order to address the problem of speech recognition with a higher number of words.

The complexity of the vocabulary also determines the complexity of the HMM modelling. The results here presented concern HMM with single Gaussian mixture which becames insufficient when the vocabulary increases, and hence the weighting function needs to be generalized to Gaussian density mixtures.

As far as convolutional noise is concerned, there are two topics that deserve attention. The first topic concerns the SS threshold that affects the convolutional noise estimation and/or removing, specially in those bands with low segmental SNR, and hence the estimation of this threshold needs to be more deeply studied. The second one is related to the assumptions about the convolutional noise: given that the microphone response depends on the distance to the speaker, the stationarity condition should be revised in some cases; moreover, the channel frequency response also depends on the signal level, although this does not seem very relevant when the speech signal is also corrupted by additive noise because the recognition will rely on those frames with higher local SNR. Due to these reasons and to the fact that the convolutional noise was artificially introduced with FIR filter, experiments with more realistic environments should be done in the future.

Appendix A

Uncertainty variance in the cepstral domain

A.1 Cepstral transform

Given a Mel filter bank composed by M channels, Mel-Frequency Cepstral Coefficients (MFCC) are defined as:

$$c_n = \sum_{m=1}^{M} E_m \cdot \cos[\frac{\pi \cdot n}{M} \cdot (m - 0.5)] \quad (A.1)$$

where E_m is the logarithm of the energy at the output of the filter m and D is the number of cepstral coefficients.

MFCC are the parametrisation used by many speech recognition systems due to the facts that they give good discrimination with a smaller number of coefficients when compared to the number of filters, that they are less correlated than the log energies and justify better the use of diagonal variance matrix in HMM's. Moreover, the insertion of a transmission channel is represented by a constant in the logarithmic and MFCC domain because the cepstral transform is linear. This allows the use of the technique Cepstral Mean Normalization (CMN) to cancel the convolutional noise.

A.2 Uncertainty variance in the log domain

According to Chapter 5, the variance (or uncertainty) of the hidden information $\overline{s_m^2}$ given the observed information $\overline{x_m^2}$ is estimated in the logarithmic domain considering that the random

variables ϕ_m (phase difference between noise and clean signal) and $\overline{n_m^2}$ (noise energy in the channel m) are uncorrelated, ϕ_m is uniformly distributed between $-\pi$ and π and that $\overline{n_m^2}$ is concentrated near its mean $E[\overline{n_m^2}]$. The variance $Var[\log(\overline{s_m^2})|\overline{x_m^2}]$ is given by:

$$Var[\log(\overline{s_m^2})|\overline{x_m^2}] = E[\log^2(\overline{s_m^2})|\overline{x_m^2}] - E^2[\log(\overline{s_m^2})|\overline{x_m^2}] \quad (A.2)$$

where

$$E[\log^2(\overline{s_m^2})|\overline{x_m^2}] \simeq \frac{1}{2\pi}\int_{-\pi}^{\pi} \log^2[\overline{s_m^2}(\phi, E[\overline{n_m^2}], \overline{x_m^2})]d\phi \quad (A.3)$$

and

$$E[\log(\overline{s_m^2})|\overline{x_m^2}] \simeq \frac{1}{2\pi}\int_{-\pi}^{\pi} \log[\overline{s_m^2}(\phi, E[\overline{n_m^2}], \overline{x_m^2})]d\phi \quad (A.4)$$

and $E[\log(\overline{s_m^2})|\overline{x_m^2}] \simeq \log[\overline{x_m^2} - E[\overline{n_m^2}]]$

A.3 Mapping from the logarithmic to the cepstral domain

The additive model proposed in Chapter 5 sets the clean signal information as a function of the noisy energy $\overline{x_m^2}$, the noise energy $\overline{n_m^2}$ and the phase difference between the clean signal and noise: $\overline{s_m^2}(\phi_m, \overline{x_m^2}, \overline{n_m^2})$. Consequently, the cepstral coefficients can be expressed as,

$$c_n(\Phi, X, N) = \sum_{m=1}^{M} \log\left(\overline{s_m^2}(\phi_m, \overline{x_m^2}, \overline{n_m^2})\right) \cdot \cos[\frac{\pi \cdot n}{M} \cdot (m - 0.5)] \quad (A.5)$$

where

$$X = [\overline{x_1^2}, \overline{x_2^2}, \overline{x_3^2}, ..., \overline{x_M^2}]$$

$$N = [\overline{n_1^2}, \overline{n_2^2}, \overline{n_3^2}, ..., \overline{n_M^2}]$$

$$\Phi = [\phi_1, \phi_2, \phi_3, ..., \phi_M]$$

are, respectively, the observed noisy signal energies, the noise energy and the phase difference along the M filters.

Assuming that the random variables ϕ_m and $\overline{n_m^2}$ are uncorrelated, ϕ_m is uniformly distributed between $-\pi$ and π and that $\overline{n_m^2}$ is concentrated near its mean $E[\overline{n_m^2}]$, the expected value of

$c_n(\Phi, X, N)$ given X is

$$E[c_n|X] = \sum_{m=1}^{M} E\left(\log[\overline{s_m^2}]|\overline{x_m^2}\right) \cdot \cos[\frac{\pi \cdot n}{M} \cdot (m - 0.5)] =$$

$$\sum_{m=1}^{M} \log\left(\overline{x_m^2} - E[\overline{n_m^2}]\right) \cdot \cos[\frac{\pi \cdot n}{M} \cdot (m - 0.5)] \quad (A.6)$$

The uncertainty variance of the cepstral coeffcient c_n given the observed energy is given by,

$$Var[c_n|X] = E[c_n^2|X] - E^2[c_n|X] \quad (A.7)$$

From the definition for the cepstral transform (A.1), the product $c_n \cdot c_n$ can be written as,

$$c_n \cdot c_n = \sum_{m}^{M} \left(\log[\overline{s_m^2}(\phi_m, \overline{x_m^2}, \overline{n_m^2})]\right)^2 \cdot \cos^2(n, m) +$$

$$2 \cdot \sum_{m}^{M} \sum_{i>m}^{M} \left(\log[\overline{s_m^2}(\phi_m, \overline{x_m^2}, \overline{n_m^2})]\right) \cdot \left(\log[\overline{s_i^2}(\phi_i, \overline{x_i^2}, \overline{n_i^2})]\right) \cdot \cos(n, m) \cdot \cos(n, i) \quad (A.8)$$

where $\cos[\frac{\pi * n}{M} \cdot (m - 0.5)]$ was replaced with $\cos(n, m)$ for simplification. Using (A.7) and (A.8), $Var[c_n|X]$ can be written as,

$$Var[c_n|X] = \sum_{m}^{M} Var[\log(\overline{s_m^2})|\overline{x_m^2}] \cdot \cos^2(n, m) +$$

$$2 \cdot \sum_{m}^{M} \sum_{i>m}^{M} E\left(\log[\overline{s_m^2}(\phi_m, \overline{x_m^2}, \overline{n_m^2})] \cdot \log[\overline{s_i^2}(\phi_i, \overline{x_i^2}, \overline{n_i^2})]|\overline{x_m^2}, \overline{x_i^2}\right) \cdot \cos(n, m) \cdot \cos(n, i)$$

$$-2 \cdot \sum_{m}^{M} \sum_{i>m}^{M} E\left(\log[\overline{s_m^2}]|\overline{x_m^2}\right) \cdot E\left(\log[\overline{s_i^2}]|\overline{x_i^2}\right) \cdot \cos(n, m) \cdot \cos(n, i) \quad (A.9)$$

or

$$Var[c_n|X] = \sum_{m}^{M} Var[\log(\overline{s_m^2})|\overline{x_m^2}] \cdot \cos^2(n, m) +$$

$$2 \cdot \sum_{m}^{M} \sum_{i>m}^{M} \left(R_{m,i} \cdot \sqrt{Var[\log(\overline{s_m^2})|\overline{x_m^2}]} \cdot \sqrt{Var[\log(\overline{s_i^2})|\overline{x_i^2}]}\right) \cdot \cos(n, m) \cdot \cos(n, i) \quad (A.10)$$

where $R_{m,i}$ is the correlation coefficient (A.Papoulis, 1991) between the components $\log[\overline{s_m^2}(\phi_m, \overline{x_m^2}, \overline{n_m^2})]$ and $\log[\overline{s_i^2}(\phi_i, \overline{x_i^2}, \overline{n_i^2})]$. This correlation coefficient $R_{m,i}$ depends on the clean speech and noise signals, and the phase difference. To avoid the estimation of $R_{m,i}$ experiments were done assuming that $\log[\overline{s_m^2}(\phi_m, \overline{x_m^2}, \overline{n_m^2})]$ and $\log[\overline{s_i^2}(\phi_i, \overline{x_i^2}, \overline{n_i^2})]$ are

uncorrelated. Although the uncorrelated condition is a rough approximation it was enough to lead to good results. Another option could be to consider $R_{m,i}$ as being inversely proportional to $|m - i|$:

$$R_{m,i} = \frac{1}{|m - i|} \tag{A.11}$$

It is interesting to mention that if $R_{m,i}$ is monotonically decreasing with $|m - i|$ as in (A.11) then,

$$\sum_{n=1}^{D} Var[c_n|X] \approx \sum_{n=1}^{D}\sum_{m=1}^{M} Var[\log(\overline{s_m^2})|\overline{x_m^2}] \cdot \cos^2(n, m) \tag{A.12}$$

Appendix B

Maximum Likelihood estimation of convolutional noise

B.1 Stochastic model of the speech signal process using a code-book

Initially, a code-book is built using clean speech signal from one speaker (or several speakers as done in (A.Acero & R.Stern, 1990) (P.Moreno, 1996) for a speaker independent system). Inside each code-word the mean and variance are computed, and the distribution of frames in the cells is supposed to be Gaussian:

$$f(s|cw_j, \varphi_j) = \frac{1}{(2\pi)^{\frac{D}{2}}|C_j|^{\frac{1}{2}}} \cdot e^{-\frac{1}{2}\cdot(s-\mu_j)^t C_j^{-1}(s-\mu_j)} \quad \text{(B.1)}$$

where

$$s = [s_1, s_2, s_3, \ldots, s_D]$$

is a clean signal vector composed by D cepstral coefficients (D is also the dimension of the code-book); cw_j is the code-word j whose Gaussian function parameters are represented by φ_j, which in turn is composed by the mean vector

$$\mu_j = [\mu_{j,1}, \mu_{j,2}, \mu_{j,3}, \ldots, \mu_{j,D}]$$

and C_j^{-1}, the inverse of the $D - by - D$ covariance matrix that is supposed diagonal

$$C_j = \begin{pmatrix} \sigma^2_{j,1} & 0 & 0 & 0 & \dots & 0 \\ 0 & \sigma^2_{j,2} & 0 & 0 & \dots & 0 \\ 0 & 0 & \sigma^2_{j,2} & 0 & \dots & 0 \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ 0 & 0 & 0 & 0 & \dots & \sigma^2_{j,D} \end{pmatrix}$$

where $\sigma^2_{j,i}$ is the variance of the coeffiicient i in the code-word cw_j; and $(s - \mu_j)^t$ denotes the transpose of $(s - \mu_j)$. The speech model is composed by the code-words, their parameters (means and variances) that define the Gaussian distributions and the probability $Pr(cw_j)$ of each code-word. Consequently, the pdf associated to the frame s given the clean speech signal model is

$$f(s|\varphi) = \sum_{j=1}^{L} f(s|cw_j, \varphi_j) \cdot Pr(cw_j) \tag{B.2}$$

where φ denotes all the means and variances of the code-book and L is the number of code-words. Given that T is a testing sequence of frames whose phonetic identity is unknown

$$T = [T_1, T_2, T_3, \dots, T_t, \dots, T_{L_T}]$$

where L_T is the length of the sequence in frames and

$$T_t = [T_{t,1}, T_{t,2}, T_{t,3}, \dots, T_{t,D}]$$

corresponds to a D cepstral coefficient frame. If T is corrupted by only convolutional noise, this distortion can be modelled by

$$T_t = T'_t + H^c \tag{B.3}$$

where T'_t corresponds to the signal before being distorted by the transmission channel and

$$H^c = [H^c_1, H^c_2, H^c_3, ..., H^c_D]$$

is the convolutional noise in the cepstral domain. This models assumes that the gain introduced by the transmission channel is considered constant inside each one of the 14 DFT Mel filters, does not depend on the signal level and is time invariant.

The likelihood of T given the clean speech model (B.2) and H^c is defined as:

$$f(T|\varphi_j, H^c) = \prod_{t=1}^{L_T} f(T_t|\varphi_j, H^c) \tag{B.4}$$

Likelihood is generally used in the log domain in order to transfor the product in a sum:

$$l(T|\varphi_j, H^c) = \log\{f(T|\varphi_j, H^c)\} = \sum_{t=1}^{L_T} \log\{f(T_t|\varphi_j, H^c)\} \tag{B.5}$$

Maximum Likelihood estimation of the convolutional distortion consists in estimating the vector H^c that maximises $l(T|\varphi_j, H^c)$ assuming that the distorted testing frame sequence T was generated by a *noisy* model whose covariance matrix is the same that is used by the clean speech one B.2 but whose mean vectors are shifted by H^c:

$$C^n_j = C_j \tag{B.6}$$

$$\mu^n_j = \mu_j + H^c \tag{B.7}$$

where C^n_j and μ^n_j are the covariance matrix and mean vector of code-word cw_j in the noisy speech model. Given this model, H^c could be estimated by means of applying to (B.5) the gradient operator with respect to H^c and then equalling to zero the partial derivatives. However, this procedure leads to a system of non-linear equations and the problem is solved using the EM algorithm (A.P.Dempster *et al.*, 1977) (X.D.Huang *et al.*, 1990) (T.K.Moon, 1996).

B.2 The Expectation-Maximization algorithm

The class (or code-word) from which each frame T_t was originated is not known (unobservable) and only the frame coefficients resulting from the spectral analysis on signal are observed. The observable data is denominated incomplete, and the data composed by the observable and unobservable data is called complete. In the problem here addressed, the observed data corresponds to T and the unobserved data is represented by

$$y = [y_1, y_2, y_3, \ldots, y_t, \ldots, y_T]$$

where y_t corresponds to the hidden number that refers to the code-word or density of the observed frame, and the measure space of y (i.e. the L code-words) is denoted by Y.

The EM (Expectation and Maximisation) (A.P.Dempster *et al.*, 1977) (X.D.Huang *et al.*, 1990) (T.K.Moon, 1996) maximises the log-likelihood of the incomplete data T by means of iteratively maximising the expectation of the log-likelihood of the complete data (X.D.Huang *et al.*, 1990). If $f(y_t|\Phi)$ and $f(T_t|\Phi)$ are members of a parametric family of pdf defined on Y and on the measure space for T respectively for parameter Φ, the function Q is defined as

$$Q(\Phi, \hat{\Phi}) = E\left\{\log\left(f(T_t, y_t|\hat{\Phi})\right) | T_t, \Phi\right\} \tag{B.8}$$

If y_t is a discrete random vector, the Q function can be re-written as

$$Q(\Phi, \hat{\Phi}) = \sum_{y_t \in Y} \frac{f(T_t, y_t|\Phi)}{f(T_t|\Phi)} \log\left\{f(T_t, y_t|\hat{\Phi})\right\} \tag{B.9}$$

It can be proved that (A.P.Dempster *et al.*, 1977) if

$$Q(\Phi, \hat{\Phi}) \geq Q(\Phi, \Phi)$$

then,

$$\log\left(f(T_t|\hat{\Phi})\right) \geq \log\left(f(T_t|\Phi)\right)$$

The EM algorithm chooses $\hat{\Phi}$ that maximises $Q(\Phi, \hat{\Phi})$ in each iteration so it is guaranteed that the log-likelihood $\log(f(T_t|\hat{\Phi}))$ also increases iteration-by-iteration toward a local optimum.

In the problem here discussed, the observed and unobserved data have L_T components each and the Q function can be defined as

$$Q(\Phi, \hat{\Phi}) = \sum_{t=1}^{L_T} \sum_{y_t=1}^{L} \frac{f(T_t, y_t|\Phi)}{f(T_t|\Phi)} \log\left\{f(T_t, y_t|\hat{\Phi})\right\} \tag{B.10}$$

or

$$Q(\Phi, \hat{\Phi}) = \sum_{t=1}^{L_T} \sum_{j=1}^{L} Pr(cw_j|T_t, \varphi_j^n) \log\left(\hat{Pr}(cw_j) f(T_t|cw_j, \hat{\varphi}_j^n)\right) \tag{B.11}$$

where y_t was replaced with j for simplicity because the summand was summed over all y_t (i.e. $1 \leq y_t \leq L$) and the range of possible values of y_t does not depend on t. The parameters φ_j^n of code-word cw_j correspond to the noisy speech code-book, and due to the fact that the clean and noisy speech parameter models are related according to (B.6) and (B.7), the re-estimation model procedure consists only in re-estimating the convolutional distortion H^c. The expression for $Q(\Phi, \hat{\Phi})$ as given by (B.11) can be decomposed in two terms that, after inverting the order of the summations and replacing φ_j^n with φ_j and H^c, are given by

$$A = \sum_{j=1}^{L} \left\{ \sum_{t=1}^{L_T} Pr(cw_j|T_t, \varphi_j, H^c) \right\} \log\left(\hat{Pr}(cw_j)\right) \tag{B.12}$$

and

$$B = \sum_{j=1}^{L} \sum_{t=1}^{L_T} Pr(cw_j|T_t, \varphi_j, H^c) \log\left(f(T_t|cw_j, \varphi_j, \hat{H^c})\right) \tag{B.13}$$

Maximising (B.11) respect to $\hat{\Phi}$ is equivalent to maximise A with respect to $\hat{Pr}(cw_j)$ and B to $\hat{H^c}$.

B.2.1 Maximising A

The probabilities $\hat{Pr}(cw_j)$ are estimated by means of maximising A with the Lagrange method (P.E.Gill *et al.*, 1981) (D.A.Pierre, 1986) . Defining,

$$\alpha_j = \sum_{t=1}^{L_T} Pr(cw_j|T_t, \varphi_j, H^c)$$

this problem is equivalent to choose $\hat{Pr}(cw_j)$ that maximizes

$$A = \sum_{j=1}^{L} \alpha_j \log\left(\hat{Pr}(cw_j)\right) \tag{B.14}$$

given the constraint

$$\sum_{j=1}^{L} \hat{Pr}(cw_j) = 1 \tag{B.15}$$

Firstly, the augmented objective function f_a is defined as being

$$f_a = A - \lambda \cdot \sum_{j=1}^{L} \hat{Pr}(cw_j) \tag{B.16}$$

where λ is denominated Lagrange multiplier*. Applying to f_a the partial derivative with respect to $\hat{Pr}(cw_j)$ and equalling to zero,

$$\frac{\alpha_j}{\hat{Pr}(cw_j)} - \lambda = 0 \tag{B.17}$$

Multiplying by $\hat{Pr}(cw_j)$ both terms of (B.17) and using (B.15), λ can be written as

$$\lambda = \sum_{j=1}^{L} \alpha_j \tag{B.18}$$

and substituting (B.18) in (B.17) $\hat{Pr}(cw_j)$ can be estimated by

$$\hat{Pr}(cw_j) = \frac{1}{L_T} \sum_{t=1}^{L_T} Pr(cw_j|T_t, \varphi_j, H^c) \tag{B.19}$$

B.2.2 Maximising B

The re-estimation of $\hat{H}^c$ is done by means of applying to B the gradient operator and equalling the partial derivatives to zero:

$$\nabla_{\hat{H}^c}(B) = \sum_{j=1}^{L} \sum_{t=1}^{L_T} Pr(cw_j|T_t, \varphi_j, H^c) \nabla_{\hat{H}^c} \log\left(f(T_t|cw_j, \varphi_j, \hat{H}^c)\right) = 0 \tag{B.20}$$

*The Lagrange multiplier is generally denoted by λ. This should not be confused with the set of parameters that defines an HMM in the notation of Chapters 2 and 6.

which, using (B.1), can be re-written as

$$\nabla_{\hat{H}^c}(B) = \sum_{j=1}^{L} \sum_{t=1}^{L_T} Pr(cw_j|T_t, \varphi_j, H^c) C_j^{-1} [T_t - (\mu_j + \hat{H}^c)] = 0 \qquad (B.21)$$

where C_j^{-1} and μ_j are the inverse of the covariance matrix and the mean vector of code-word cw_j of the clean speech model. Solving (B.21) for $\hat{H}^c$

$$\hat{H}_i^c = \frac{\sum_{j=1}^{L} \sum_{t=1}^{L_T} Pr[cw_j|T_t, \varphi_j, H^c] \cdot \frac{T_t - \mu_{j,i}}{\sigma_{j,i}^2}}{\sum_{j=1}^{L} \sum_{t=1}^{L_T} \frac{Pr[cw_j|T_t, \varphi_j, H^c]}{\sigma_{j,i}^2}} \qquad (B.22)$$

B.3 EM algorithm for the convolutional noise estimation

It is interesting to highlight that if the ML method is applied to estimate only the convolutional noise, the EM algorithm has an analytical solution. However, if the ML technique is used to estimate both additive and convolutional distortions, the re-estimation expressions have an approximate form due to the expansion in series of the logarithmic function (P.Moreno, 1996) (B.Raj *et al.*, 1996) .

Finally, the ML estimation of the convolutional noise using the EM algorithm is given by:

1. Initial values for $Pr[cw_j] = \frac{1}{L}$ (for $j = 1, 2, ..., L$) and $H_i^c = 0.0$;

2. Compute $Pr[cw_j|T_t, \varphi_j, H^c]$:

$$Pr[cw_j|T_t, \varphi_j, H^c] = \frac{f[T_t|cw_j, \varphi_j, H^c] \cdot Pr[cw_j]}{\sum_{j=1}^{L} f[T_t|cw_j, \varphi_j, H^c] \cdot Pr[cw_j]} \qquad (B.23)$$

3. Re-estimate $Pr[cw_j]$ according to (B.19);

4. Re-estimate H^c according to (B.22);

5. Stop if convergence has been reached, otherwise go to Step 2.

Appendix C

Publications by the author

C.1 Journal papers

IEE Proceedings- "Lateral inhibition net and weighted matching algorithm for speech recognition in noise". IEE Proceedings Vision, Image and Signal Processing, Vol. 143, No. 5, October 1996, pp. 324-330.

IEE Electronics Letters- "Robust speech pulse detection using adaptive noise modelling". IEE Electronics Letters, Vol. 32, No. 15, July 1996, pp. 1350-1352.

IEEE Transactions- "Improving Performance of Spectral Subtraction in speech recognition using a model for additive noise" Accepted for publication in IEEE Transactions on Speech and Audio Processing

C.2 Conference papers

Eurospeech'95- "Improved Algorithms for Speech Recognition in Noise using Lateral Inhibition and SNR Weighting". Proceedings International Conference Eurospeech'95, pp.461-464.

ICSLP'96- "Use of a Reliability coefficient in noise cancelling by Neural Net and Weighted Matching Algorithms". Proceedings International Conference for Spoken Language Processing, ICSLP'96, pp. 2297-2300.

IVTTA'96- "Robust speech pulse detection using adaptive noise modelling and non-statonarity

measure" International Workshop on Interactive Voice technology for Telecommunications Applications, IVTTA 96, pp. 69-72.

ICASSP'97- "Weighted Matching Algorithms and Reliability in Noise Cancelling by Spectral Subtraction". Proceedings ICASSP 97, vol.2, pp. 1171-1174.

Eurospeech'97- "Spectral Subtraction and Mean Normalization in the context of Weighted Matching Algorithms". Proceedings International Conference Eurospeech'97, pp.1411-1414.

ICASSP'98- "Weighted Viterbi algorithm and state duration modelling for speech recognition in noise ". Accepted for publication in proceedings ICASSP'98.

Lateral inhibition net and weighted matching algorithms for speech recognition in noise

N.B.Yoma
F.McInnes
M.Jack

Indexing terms: Speech recognition, Lateral inhibition net, Noise

Abstract: The authors address the problem of speech recognition with signals corrupted by white Gaussian additive noise at moderate SNR. The energy of the noise is not required. A technique based on a lateral inhibition process approximation with a multilayer neural net (the lateral inhibition net (LIN)) and neural net processing efficacy weighting in acoustic pattern matching algorithms is proposed. In the recognition procedure, the local SNR is computed by means of the autocorrelation function and is employed to estimate the efficacy of LIN in noise cancelling which is taken into account as a weight in a pattern matching algorithm. A general criterion based on weighting the frame influence in decisions according to the reliability in noise reduction is suggested, and modified versions of both HMM and DTW algorithms have been designed. To be more coherent with the conditions that define LIN, a modification in the backpropagation algorithm is also proposed.

1 Introduction

Many of the techniques that have been proposed to solve the noise sensitivity of automatic speech recognition systems (ASRS) are based on the estimation of noise at intervals where there are no speech signals. This restriction could be accepted in some real applications of isolated word recognition, but it is very inappropriate for general real environments and especially for continuous speech recognition, where the time separation between two consecutive silence intervals can be much larger than in the isolated word case. The noise signal can change in energy and/or spectral distribution and the noise estimation can become obsolete between two silence intervals. In addition, the efficacy of noise cancelling methods cannot be the same along the speech signals, first, because the local SNR is not constant, and secondly, because the response of the noise reduction system can also depend on the characteristics of the input speech.

IEE Proceedings online no. 19960758

Paper received 13th December 1995

The authors are with the Centre for Communication Interface Research, University of Edinburgh, 80 South Bridge, Edinburgh EH1 1HN, UK

This paper describes a method to improve the noise robustness of ASRS to white Gaussian additive noise at moderate SNR by emulating spectral lateral inhibition with a neural net, and the noise reduction efficacy weighting in acoustic pattern matching algorithms. The noise power is not required by this approach. Four problems have been addressed. These are: first, the approximation of the lateral inhibition function with multilayer neural nets (LIN); secondly, frame-by-frame computation of the SNR; thirdly, estimation of the effectiveness of the LIN processing; and finally reliability weighting in acoustic pattern matching algorithms. The backpropagation algorithm was modified to be more coherent with the LIN definition.

The conception of the neural net training procedure inspired by lateral inhibition had as its main purpose a possible generalisation of the LIN structure to other types of noise. Because lateral inhibition is basically the attenuation of the lowest by the highest energies, this mechanism could reduce the influence of any noise if the local SNR preserves the highest components of the speech signal.

The local SNR estimation proposed herein does not need the noise power estimation in silence intervals and can be efficiently computed frame by frame, although the method loses accuracy if the speech signal is poorly correlated. Furthermore, the evaluation of the efficacy of a noise cancelling method seems to be a generic approach and can be applied to other techniques.

The DTW algorithm based on the dynamic programming equation proposed in this research (DPW) is just one-step, and has similar performance to the two-step DTW previously proposed in [4]. The modified Viterbi algorithm for HMM has not been previously reported. In addition, the modified DTW and HMM algorithms are sufficiently generic to be employed with other noise cancellation techniques.

2 LIN: a noise cancellation neural net

Masking is basically the suppression of the lowest by the highest spectral components. Lateral inhibition is one of the processes responsible for the masking phenomena in different sensory systems and this concept was used to train the noise reduction neural network, LIN, employed in this research.

Given that: • E_j is the logarithm of the normalised energy at the output of the filter j in a bank of N filters. • $F_i^c = (E_1^c, E_2^c, E_3^c, ..., E_N^C)$ is frame i of the clean signal; and • $F_i^n = (E_1^n, E_1^n, E_1^n, ..., E_N^n)$ is frame i after it

has noise added, then the lateral inhibition function (LI) can be set as

$$LI(E_i) = E_i + f(E_1, E_2, E_3, \ldots, E_N) \qquad (1)$$

where the function $LI()$ was approximated with multilayer perceptrons with one hidden layer. Multilayer perceptrons were chosen because they can store the information from a large amount of training data, and produce a correct input–output mapping even when the input is slightly different from the examples used to train them (generalisation). Fig. 1 shows the topology employed to approximate eqn. 1.

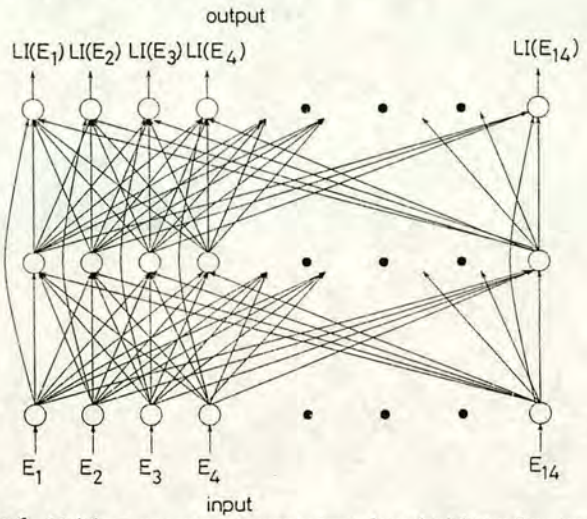


Fig. 1 *Multilayer perceptron to approximate lateral inhibition function set by eqn. 1*

The output function for the hidden layer nodes was $\sigma(x) = 1/(1 + e^{-x})$ and the output function for input and output layers is linear. Each input node receives the energy of one filter and the same energy is fedforward to the output node to compound eqn. 1. The number of input, hidden and output nodes was equal to the number of filters, N.

The LIN was trained with the following conditions that define the lateral inhibition function:

$$LI(F_i^n) \approx LI(F_i^c) \qquad LI(F_i^c) \approx F_i^c$$

The first condition specifies that F_i^c and F_i^n should give approximately the same result after they are processed by LIN. The second condition settles that LI of a clean signal must give the same clean signal, so that the spectral information is preserved and no distortion is introduced.

All the weights of the neural net (except those on the feedforward connections from the inputs to the outputs which were always equal to 1) were estimated with the classical backpropagation algorithm [1] with cross-validation [2]. The training data were made up of input–reference pattern pairs. Initially, the reference patterns were frames of clean signal, F_i^c, and the input patterns were generated adding white Gaussian noise to F_i^c at four different SNRs (clean, 18 dB, 12dB, and 6dB). Therefore, each frame F_i^c originated four training input–reference pairs. In a modified version of the training algorithm, $LI(F_i^c)$ was used instead of F_i^c as reference patterns.

The training of the neural network was carried out frame by frame and not utterance by utterance, so the LIN should be able to recover the information from a noisy frame independently of the context. Moreover, the SNR training condition (SNR ≥ 6dB) guarantees that the highest components are preserved from the reference to the input training pattern, and, on the other hand, the generalisation feature of the neural nets should be able to mask the noises when the SNR is not included among the training conditions or even perhaps when the noise is poorly correlated but not white.

2.1 LIN input

To normalise the inputs between 0 and 1, first the maximum energy of the frame was determined. Then the energy of the other filters was computed in decibels using the maximum energy as reference, and all components 50dB below this maximum energy were made equal to –50dB. Finally, the energies in dB were linearly transformed from the range [–50dB, 0db] to [0, 1].

2.2 LIN training database

Sounds that present low energy (typically fricatives) are the first to be masked by corrupting signals, and using these speech frames as training patterns could mean learning the neural network with information that is lost even for moderate SNRs. In [7] the use of periodicity as a criterion to select training patterns was proposed. Periodicity was defined as

$$\text{periodicity} = \frac{\max[R_x(m)]}{R_x(0)}$$

where $R_x(m)$ is the autocorrelation of the speech signal and was computed with all ms in the range of fundamental periods. The main purpose of this coefficient was to choose voiced frames with high energies but it was observed that some frames, specially at the end of the utterances, presented a high periodicity coefficient and a very low energy. In the results reported in this paper, energy was used as a discriminative parameter. Initially the maximum energy of the utterance was computed and then all the frames that were below a given threshold from the maximum energy were discarded. According to some preliminary experiments a suitable threshold would be 25dB.

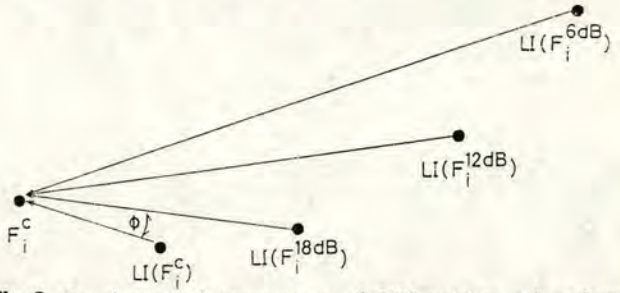


Fig. 2 *Two-dimensional interpretation of LIN training with ordinary back-propagation algorithm*
Reference is constant and equal to the clean frame

3 LIN and reliability in noise reduction

Initially the quadratic error at the backpropagation algorithm was computed between the reference F_i^c and the output $LI(F_i^n)$, which should result in an estimation of the clean frame F_i^c. Given that F_i^{18db} corresponds to a noisy frame with local SNR equal to 18dB, F_i^{12db} to a noisy frame with local SNR equal to 12dB, F_i^{6db} to noisy frame with local SNR equal to 6dB, Fig. 2 shows a two-dimensional interpretation of the LIN training algorithm. In recognition tests, reference (clear utterances) and testing patterns (noisy utterances) are processed by LIN, and hence in the acoustic pattern

matching algorithm the local distances correspond to $d[LI(F_k^c), LI(F_i^n)]$ instead of $d[F_k^c, F_i^n]$, where k denotes a reference frame and i a test one. In the experiments reported here, the distance function d was the Euclidean metric.

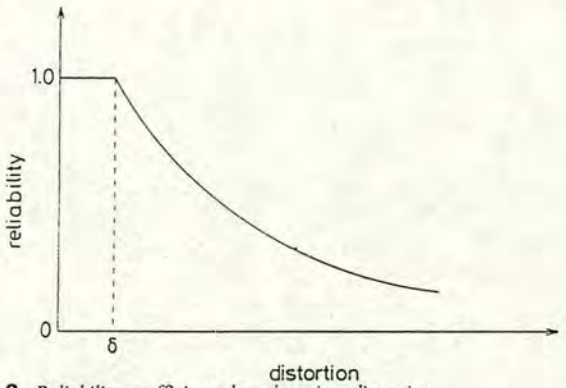


Fig. 3 *Reliability coefficient plotted against distortion* $d[LI(F_i^c), LI(F_i^n)]$

A noise cancelling neural net can be seen as a system that processes a noisy input and produces an output with the influence of noise reduced. Since there are several levels of distortions and the backpropagation training algorithm is essentially stochastic (most common patterns have more influence in the weights re-estimation process), it is reasonable to suppose that the LIN efficacy depends on the input and each noisy frame could be associated to a reliability coefficient that attempts to measure how reliable is the result of LIN processing. As the noise cancelling depends on $d[LI(F_i^c), LI(F_i^n)]$ (the smaller this distance, the better is the noise influence cancelling), the reliability coefficient could be related to this distortion by means of the curve shown in Fig. 3. If $d[LI(F_i^c), LI(F_i^n)]$ is smaller than a threshold δ, the reliability will be 1.0; and if $d[LI(F_i^c), LI(F_i^n)] > \delta$, the reliability will be inversely proportional to $d[LI(F_i^c), LI(F_i^n)]$. This curve is analytically described by the following function:

$$r = \begin{cases} 1 & \text{if } d[LI(F_i^c), LI(F_i^n)] \leq \delta \\ \dfrac{\delta}{d[LI(F_i^c), LI(F_i^n)]} & \text{if } d[LI(F_i^c), LI(F_i^n)] > \delta \end{cases}$$

It is interesting to highlight that LIN tends to preserve the highest energies and the position of local spectral peaks (see Fig. 4), or in other words, tends to preserve the phonetic information of the frame. For this reason, if $d[LI(F_i^c), LI(F_i^n)]$ was low for any SNR, the recognition error would be also low independently of the noise level.

At the recognition procedure, the clean version F_i^c of the noisy testing frame F_i^n is not available but, because the power spectral distribution of the corrupting signal is known (white Gaussian noise), F_i^c can be set as a function of F_i^n and the local SNR. After LIN has been trained, the training database could be used to approximate the relation between $d[(LI(F_i^c), LI(F_i^n)]$, and F_i^n and the local SNR. Consequently, if the segmental SNR could be computed frame by frame and given that F_i^n is available, the reliability coefficient could be estimated frame by frame at the recognition process.

3.1 Local SNR estimation

If the noise is poorly correlated and uncorrelated with the speech signal, it is possible to estimate the power of the clean speech from the autocorrelation function of the noisy signal [7]. Given that $R_x(m)$, $R_s(m)$ and $R_n(m)$ are the autocorrelation functions of the noisy speech, the clean speech and the noise signals, respectively, the following coeffficient can be computed frame by frame:

$$n = \frac{R_s(0)}{R_x(0)} = \frac{R_s(0)}{R_n(0) + R_s(0)} \tag{2}$$

$$n = \begin{cases} 1 & \text{if } SNR = \infty \\ 0 & \text{if } SNR = -\infty \end{cases}$$

where $R_s(0)$ was estimated by means of applying some properties of the autocorrelation function and quadratic interpolation [7]

$$R_s(0) = \frac{4 \times R_x(1) - R_x(2)}{3} \tag{3}$$

The coefficient n can be computed frame by frame because it needs just the autocorrelation of the noisy signal at points m = 0, 1 and 2. Observe that the estimation of the noise power in silence intervals is not needed and the method captures the dynamic of the speech and noise signals energy. Given that

$$SNR = 10 \log \left(\frac{R_s(0)}{R_n(0)} \right)$$

the segmental SNR and the coefficient n are related by

$$n = \frac{10^{SNR/10}}{1 + 10^{SNR/10}} \tag{4}$$

The more correlated is the speech signal, the more accurate is the local SNR estimation. If the speech signal is poorly correlated, the method loses accuracy.

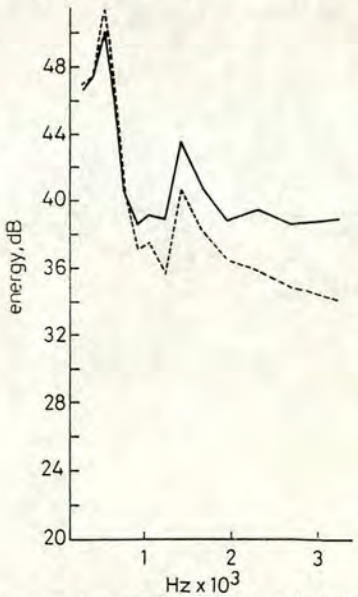


Fig. 4 *Noisy frame with local SNR equal to 6dB before and after LIN processing*
Frame corresponds to vowel ε
Highest component tends to be preserved and position of second format does not change
— LIN input spectrum (SNR = 6dB)
..... LIN output spectrum

3.2 Mean distortions

As an approximation, it can be assumed that the distortion $d[LI(F_i^c), LI(F_i^n)]$ depends exclusively on the local SNR. The mean distortions for each SNR can be estimated at the LIN training procedure and, once the local SNR can be efficiently computed for correlated speech signals [7], $d[LI(F_i^c), LI(F_i^n)]$ could be estimated at the recognition process. Given: • D_i^{snr}, the distortion $d[LI(F_i^c), LI(F_i^n)]$ for the frame F_i^n with local SNR equal to *snr*; and • $\overline{D_{snr}}$, the mean-distortion at local SNR equal to *snr*; then $\overline{D_{snr}}$ can be computed for

some SNRs at the LIN training procedure and, by means of linear interpolation, it can be estimated for other values of SNR. Fig. 5 shows the curve $\overline{D_{snr}}$ against SNR estimated with a LIN that was trained with the female speaker. The limitation of this method concerns the fact that $d[LI(F_i^c), LI(F_i^n)]$ depends on F_i^n and not only on the segmental SNR.

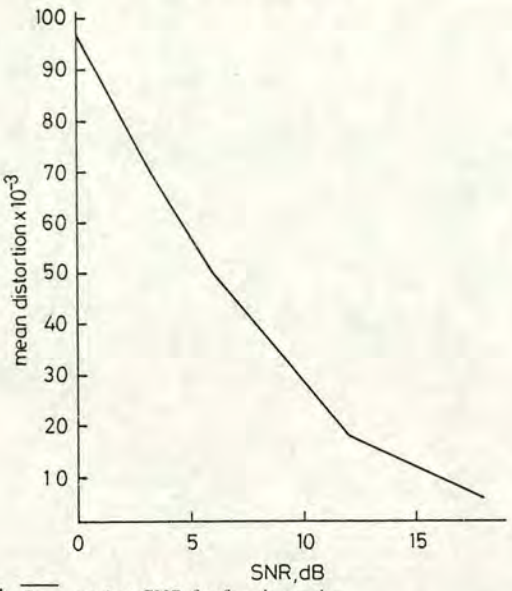


Fig. 5 $\overline{D_{snr}}$ *against SNR for female speaker*

For the results presented in this paper, $\overline{D_{snr}}$ was computed for SNR = 18, 12, 6, 3 and 0dB by employing the LIN training database, after LIN had been trained. During the recognition procedure, the coefficient n was estimated by means of the autocorrelation function eqns. 2 and 3 and the curve $\overline{D_{snr}} \times localSNR$ was mapped into the n domain using eqn. 4. The constant δ was made equal to 0.004, a value that was shown to be suitable according to some tests.

4 Modified backpropagation algorithm

In the ordinary neural net training algorithm, the quadratic error is computed between the reference F_i^c and the output $LI(F_i^n)$. However, the efficacy of LIN is related to the distortion $d[LI(F_i^c), LI(F_i^n)]$: the smaller $d[LI(F_i^c), LI(F_i^n)]$ is, the smaller should be the recognition error rate. As a consequence, it can be interesting to include the condition of minimisation of $d[LI(F_i^c), LI(F_i^n)]$ in the training algorithm in a more explicit way. Fig. 2 shows the ordinary backpropagation approach, where the target is the minimisation of the distances $d[F_i^c, LI(F_i^n)]$ instead of $d[LI(F_i^c), LI(F_i^n)]$. The minimisation of $d[F_i^c, LI(F_i^n)]$ leads to the reduction of $d[LI(F_i^c), LI(F_i^n)]$, but this distance also depends on the angle between $LI(F_i^c) - F_i^c$ and $LI(F_i^n) - F_i^c$ (see Fig. 2). In the modified algorithm, the clean signal F_i^c was replaced with $LI(F_i^c)$ as the reference for the noisy frames, and the quadratic error was computed between the reference $LI(F_i^c)$ and the output $LI(F_i^n)$.

At the ordinary LIN training algorithm (BLT-backpropagation LIN training), in each epoch the backpropagation minimises the quadratic error of the following sequence of pairs reference–output: (1) F_i^c and $LI(F_i^c)$; (2) F_i^c and $LI(F_i^n)$, for all the local SNRs included in the training database.

At the modified training algorithm (MLT-modifled LIN training), in each epoch the backpropagation minimises the quadratic error of the following sequence of pairs reference–output: (1) F_i^c and $LI(F_i^c)$; (2) $LI(F_i^c)$ and $LI(F_i^n)$, for all the local SNRs included in the training database, which is more coherent with the conditions that define the lateral inhibition function (see Section 2). Fig. 6 shows the two-dimensional interpretation of the MLT algorithm. It is interesting to note that the reference is not constant, as in the ordinary backpropagation algorithm, but is modified iteration by iteration because $LI(F_i^c)$ depends on LIN, and LIN's weights are re-estimated each time that a reference–output pair is presented to the training algorithm.

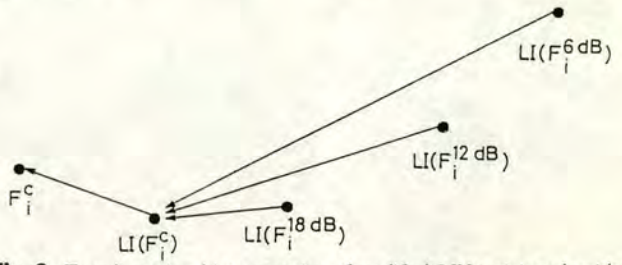


Fig. 6 *Two-dimensional interpretation of modified LIN training algorithm (MLT)*

5 Weighted matching algorithms

Some modifications were included in matching algorithms to weight the reliability of the information extracted from testing frames. A weighting coefficient $w(t)$ ($w(t) = 1$, maximum reliability; $w(t) = 0$, minimum reliability) is associated with each testing frame employed in the modified versions of the DTW and Viterbi (HMM) algorithms. In this paper w was made equal to the coefficient n, related to the segmental SNR estimation (Section 3.1), and to r, reliability in LIN processing. The main idea behind the modifications made on the Viterbi (HMM) and DTW algorithms is that the influence of a frame on decisions must be proportional to its coefficient w. The proposed weighted DP algorithm was compared with the two-step DP algorithm proposed in [4]. The modified Viterbi algorithm has not yet been tested.

5.1 HMM: modified Viterbi algorithm

The reliability coefficient can be included in the Viterbi algorithm [3] by raising the output probability of observing the frame O_t to the power of $w(t)$. This modification leads to the following algorithm:

Step 1: Initialisation. For each state i,

$$\delta_1(i) = \pi_i \times [b_i(O_1)]^{w(1)}$$

$$\psi_1(i) = 0$$

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

$$\delta_t(j) = \max_i[\delta_{t-1}(i) \times a_{ij}] \times [b_j(O_t)]^{w(t)}$$

$$\psi_t(j) = \arg\max_i[\delta_{t-1}(i) \times a_{ij}]$$

Step 3: Termination. (* indicates the optimised results).

$$P^* = \max_{s \in Sf}[\delta_T(s)]$$

Consequently, the influence of the probability $b_i(O_{t-1})$ in the decision $\text{Max}_i[\delta_{t-1}(i) \times a_{ij}] = \text{Max}_i[\text{Max}_h[\delta_{t-2}(h) \times a_{hi}] \times [b_i(O_{t-1})]^{w(t-1)} \times a_{ij}]$ at Step 2 depends on $w(t-1)$: if $w(t-1) = 1$ (high reliability), the influence of $b_i(O_{t-1})$ is maximal; if $w(t-1) = 0$ (very

low reliability); the influence of $b_i(O_{t-1})$ is zero because $[b_i(O_{t-1})]^0 = 1$ for all states i.

5.2 DTW: modified DP equation

The same principle of weighting the importance of a frame according to $w(i)$ leads to a modified dynamic programming (DP) equation. The proposed DP equation is

$$G(i,j) = \min \left(\begin{array}{c} \frac{G(i,j-1)\times W(i,j-1)+d(i,j)\times w(i)}{W(i,j-1)+w(i)} \\ \frac{G(i-1,j-1)\times W(i-1,j-1)+2\times d(i,j)\times w(i)}{W(i-1,j-1)+2\times w(i)} \\ \frac{G(i-1,j)\times W(i-1,j)+d(i,j)\times w(i)}{W(i-1,j)+w(i)} \end{array} \right)$$

$$W(i,j) = \begin{cases} W(i,j-1)+w(i) \\ W(i-1,j-1)+2\times w(i) \\ W(i-1,j)+w(i) \end{cases}$$

This DP equation takes into account the weight $w(i)$ frame by frame, and the calculation of the overall distance, $G(i, j)$, is affected by $d(i, j)$ according to $w(i)$: if $w(i) = 1$ (high reliability or local SNR), the weight of $d(i, j)$ is maximal; if $w(i) = 0$ (very low reliability or local SNR), the importance of $d(i, j)$ is zero.

5.3 Two-step DP matching

This algorithm [4] consists of the following two-step processing. First, the optimal alignment path $c_k = (i_k, j_k)$, $k = 1, 2, ..., K$ is obtained using the ordinary DP matching algorithm with symmetric weight, where i_k and j_k are the frame numbers of the testing and reference patterns, respectively. The second step is the calculation of the global distance between the utterances weighted by $w(i_k)$ along the optimal path obtained at the first step.

6 Experiments of word recognition

6.1 Database

The proposed methods were tested with speaker-dependent isolated word (English digits from 0 to 9) recognition experiments. The tests were carried out employing the two speakers (one female and one male) from the Noisex database. The isolated clean words were automatically end detected and generated the database used in this research. For each speaker, the 100 training clean utterances (ten repetitions per digit) generated ten reference sets (set of repetition 1 of each word, set of repetition 2 of each word etc). The 100 testing clean utterances were used to create the noisy database by adding white noise at five global-SNR levels: clean speech, +18dB, +12dB, +6dB, +3dB and 0dB. The global-SNR was defined as in [5]. First, the total energy E of the clean word was computed. Then, the mean energy per sample E_t was determined dividing E by the number of samples of the signal. Finally, E_t was used to set the variance of the zero mean white Gaussian noise to be added.

6.2 Preprocessing

Before the Gaussian noise was added, the speech signals were lowpass filtered, using a 10th-order Tchebychev filter with cut-off frequency equal to 3700Hz, and down sampled from 16000 to 8000 samples/second. The band from 300 to 3400Hz was covered with 14 Mel second-order IIR digital filters. The energy of each filter was an input of LIN as explained in Section 2.1. After LIN processing ten cepstral coefficients were computed.

6.3 Training the neural network

For each speaker, the frames from the set of repetition 1 of the training database (Section 6.1) generated the input–reference pattern pairs used by the LIN training algorithm to estimate the weights. The frames from the set of repetition 2 of the training database generated the input–reference pattern pairs used to evaluate the performance of the LIN. Several training conditions (learning rate, initial weights and database) were tested and the one that gave the best results on the test data was chosen. For each speaker, the LIN training variables were kept constant to compare the MLT and BLT algorithms at the same conditions.

6.4 Results

The results presented in this paper were achieved with 1000 recognition tests for each SNR: ten reference sets × 100 testing utterances. The following configurations were tested: the ordinary DTW algorithm with cepstral coefficient without (DP-C) and with (DP-L) LIN processing; the proposed weighted DP algorithm with LIN processing, (DPW-$\bar{D}$) with the mean-distortions method for reliability estimation and (DPW-n) with local SNR estimation; and finally, the two-step DP matching with LIN, (DP2-$\bar{D}$) with the mean-distortions method for reliability estimation and (DP2-n) with local SNR estimation. Table 1 shows the number of iterations required by each algorithm. The recognition error rates are presented in Table 2 for the female speaker, and in Table 3 for the male one.

Table 1: Number of iterations needed to train LIN

Speaker	Female	Male
BLT	6132	7403
MLT	3869	1702

7 Discussion

7.1 LIN efficacy in noise cancelling

The LIN showed a substantial reduction in error rates even without reliability weighting. With the ordinary DTW algorithm (*DP-L*) the LIN practically eliminated the influence of the noise at SNR = 18 and 12dB, and resulted in a mean reduction of 87, 70 and 48% at SNR = 6, 3 and 0dB, respectively. Moreover, the error introduced for testing the clean signal was almost zero.

7.2 Comparison between weighting coefficients

As can be seen in Table 2 (female speaker) and Table 3 (male speaker), the reliability coefficient estimated with the mean-distortions method gave a greater reduction in the error rate than the SNR weighting in all noisy conditions. When the LIN was trained by means of the MLT algorithm, the reduction due to reliability weighting was as high as 100, 84 and 57% at SNR = 12, 6 and 3dB, respectively, while the SNR estimation resulted in a much smaller reduction in most of the cases and even in an increase of the error rate in other cases.

The proposed one-step weighted algorithm showed almost the same performance as the two-step one with

Table 2: Recognition error rate (%) for the female speaker. LIN was trained with MLT and BLT (results in parentheses) algorithms

SNR	Cln	18dB	12dB	6dB	3dB	0dB
DP-C	0.1	3.5	31.9	67.0	70.6	75.6
DP-L	0.1 (0.1)	0.2 (0.1)	0.6 (1.2)	5.9 (11.5)	10.6 (31.9)	24.5 (53.5)
DPW-$\bar{D}$	0.1 (0.2)	0.0 (0.1)	0.0 (0.1)	0.7 (4.0)	6.1 (17.2)	17.9 (33.3)
DPW-n	0.1 (0.1)	0.1 (0.4)	0.3 (1.0)	3.0 (6.4)	9.5 (26.0)	30.1 (43.9)
DP2-$\bar{D}$	0.1 (0.1)	0.0 (0.0)	0.0 (0.1)	0.5 (3.5)	5.6 (15.9)	17.6 (32.1)
DP2-n	0.1 (0.1)	0.1 (0.2)	0.1 (0.4)	2.3 (4.0)	6.5 (20.6)	23.6 (38.2)

Table 3: Recognition error rate (%) for male speaker. LIN was trained with MLT and BLT (results in parentheses) algorithms

SNR	Cln	18dB	12dB	6dB	3dB	0dB
DP-C	0.0	16.8	49.9	65.1	69.4	74.6
DP-L	0.0 (0.3)	0.6 (0.4)	2.7 (1.6)	9.2 (9.8)	22.6 (21.9)	38.2 (41.8)
DPW-$\bar{D}$	0.1 (0.1)	0.0 (0.1)	0.0 (0.1)	2.2 (1.3)	7.8 (6.3)	24.1 (24.6)
DPW-n	0.5 (0.5)	0.8 (0.5)	3.3 (3.4)	11.5 (10.4)	22.0 (20.6)	36.1 (43.4)
DP2-$\bar{D}$	0.1 (0.1)	0.0 (0.1)	0.0 (0.2)	2.3 (1.2)	7.8 (6.6)	24.8 (25.2)
DP2-n	0.3 (0.3)	0.0 (0.1)	0.9 (1.7)	8.5 (8.2)	17.7 (17.2)	31.6 (38.9)

the reliability coefficient, but resulted in a poorer improvement when the SNR estimation was used as a weighting parameter. This must be due to the fact that in the one-step algorithm the influence of a frame on decisions must be proportional to its coefficient w, and the reliability coefficient includes not only the information concerning the segmental SNR, but also the LIN characteristic in the form of the mean-distortion curve (Fig. 5), and provides a more accurate estimation of the reliability of the information extracted from each frame.

7.3 Comparison of MLT and BLT algorithms

According to Tables 2 and 3, the reliability coefficient as a weighting parameter gave the best results, with the MLT algorithm for the female speaker and with the BLT algorithm for the male one. The error rate was kept below 1.5% at SNR = 6dB and below 10% at SNR = 3dB for both speakers.

Some preliminary experiments showed that the best results were achieved with the combination of MLT and reliability coefficient weighting. This could be the result of: first, the weakening of the learning constraints imposed by MLT, and secondly, the better matching between these constraints and the estimation of $d[LI(F_i^c), LI(F_i^n)]$ required by the reliability coefficient computation. In the MLT algorithm, the approximation between $LI(F_i^c)$ and $LI(F_i^n)$ (Fig. 6) seemed to be more natural than the approximation between F_i^c and $LI(F_i^n)$ in BLT (Fig. 2). However, further tests showed that the BLT algorithm could lead, depending on the LIN training conditions, to better results than the MLT one (male speaker).

8 Conclusions

The combination of LIN and weighted DP algorithms proved to be effective in reducing the influence of white Gaussian noise, and the error introduced for testing the clean signal was almost zero. The reliability coefficient gave better results than the SNR estimation as a weighting parameter and this must arise from the fact that this coefficient takes into account not only the local SNR estimation but also the characteristic response of LIN in the form of the mean-distortion curve (Fig. 5). The weighted DP algorithms helped to reduce the error rate, but its improvement decreased when the SNR became more severe. The proposed one-step DP matching was also shown to be effective in reducing the error rate, and led to approximately the same error rates as the two-step matching [4] when the reliability weighting was used.

The reliability coefficient as a weighting parameter seems to be a generic approach and could be employed with other noise cancelling techniques. Further studies are needed in order to develop a more accurate and generic estimation for this coefficient.

The MLT algorithm appears to be an interesting option to be used in combination with reliability weighting, although further tests are needed to delimit its efficacy. A drawback of LIN is the strong influence of training conditions (learning rate, initial weights and database) in the final results and several configurations had to be tested. In this sense, the inclusion of the reliability coefficient seems to be an important advance because it caused a reduction of the error rate in all the cases, independently of the training configurations. Future work includes the generalisation of the LIN structure to other types of noises, adaptation to new environments and a more precise delimitation of the influence of the training conditions.

9 Acknowledgment

N.B. Yoma was supported by a grant from CNPq-Brasilia/Brasil

10 References

1 RUMELHART, D.E., HINTON, G.E., and WILLIAMS, R.J.: 'Learning internal representations by error propagation', *in* RUMELHART, D.E., and McCLELLAND, J.L. (Eds.): 'Parallel distributed processing: explorations in the microstructures of cognition', Vol. 1, (MIT Press, Cambridge, MA, 1986), Chap. 8

2 HAYKIN, S.: 'Neural networks, a comprehensive foundation' (Macmillan College Publishing, 1994)

3 HUANG, X.D., ARIKI, Y., and JACK, M.A.: 'Hidden Markov models for speech recognition' (Edinburgh University Press, 1990)
4 KOBATAKE, H., and MATSUNOO, Y.: 'Degraded word recognition based on segmental signal-to-noise ratio weighting'. ICASSP 1994, Vol. 1, pp. 425–428
5 GHITZA, O.: 'Robustness against noise: the role of timing-synchrony measurement'. ICASSP 1987, pp. 2372–2375
6 VARGA, A., STEENEKEN, H.J.M., TOMLINSON, M., and JONES, D.: 'The noisex-92 study on the effect of additive noise in automatic speech recognition'. Technical report, DRA Speech Research Unit, UK, 1992
7 YOMA, N.B., McINNES, F., and JACK, M.: 'Improved algorithms for speech recognition in noise using lateral inhibition and SNR weighting'. Eurospeech'95, 1995, pp. 461–464

Robust speech pulse detection using adaptive noise modelling

N.B. Yoma, F. McInnes and M. Jack

Indexing terms: Speech recognition, Adaptive filters

The problem of speech pulse detection with additive noise at a signal-to-noise ratio (SNR) as low as 0 and –6dB is addressed. The noise is assumed to be reasonably stationary and correlated. Three techniques have been examined: the autoregressive analysis of noise: spectral density comparison: and the non-stationarity measure.

Introduction: The inaccurate detection of the endpoints is a major cause of errors in automatic speech recognition systems. Most of the endpoint detecting techniques are based on energy levels, pitch, zero- and/or level-crossing rates, and timing [1]. However, in many real environments the speech signal is corrnpted by additive noise and these parameters may be insufficient for the correct detection of a speech pulse if the signal-to-noise ratio (SNR) is low.

The contributions of this Letter concerns: (i) adaptive autoregressive modelling of noise in order to reduce the influence of the corrupting signal; and speech pulse detection aided by (ii) spectral density comparison between noise and noisy speech signals or (iii) non-stationarity measures.

The FIR filters used in the autoregressive analysis are trained with the LMS algorithm during non-speech intervals. The spectral density comparison is made between noisy speech frames and an estimation of noise in non-speech intervals. In contrast, non-stationarity measures are based on spectral distances between contiguous frames and do not require noise estimation. Preliminary experiments have shown that the AR analysis generally increases the discrimination between speech and noise, and that spectral density comparison and non-stationarity measures might be more effective than energy in indicating the presence of a speech pulse at low SNRs.

AR analysis of the noise signal: It is assumed that the noise $n(i)$ could be described by an AR process of order M, i.e. it would satisfy the following equation [2]:

$$H_A(z).N(z) = W(z) \tag{1}$$

where $N(z)$ and $W(z)$ are the z transform of the noise and a white noise process, respectively, and $H_A(z)$ is defined as

$$H_A(Z) = 1 + \sum_{k=1}^{M} a_k z^{-k} \tag{2}$$

If the noise is reasonably stationary, its autoregressive filter $H_A(Z)$ estimated in non-speech intervals may be used to increase the energy gap between the noise and the noisy speech signals. Since the speech signal is intrinsically non-stationary and has components in all the considered band (250–3200 Hz), its spectral density and that of the noise are likely to differ along time, even if the noise is correlated and mainly concentrated in low frequencies (below 1000 Hz). Consequently, it is expected that the attenuation caused by $H_A(Z)$ will be lower on average for the speech than for the corrupting signal. The filter $H_A(Z)$ is transversal or FIR, and its coefficients can be estimated using the classical LMS algorithm. If the coefficients a_i are replaced with c_i, where $c_i = -a_i$, the tap weights adaptation is given by

$$c_k(i+1) = c_k(i) + \eta n(i-k)e(i) \tag{3}$$

where η is the learning rate and $e(i)$ corresponds to the prediction error:

$$e(i) = n(i) - \sum_{k=1}^{M} c_k n(i-k) \tag{4}$$

Spectral density comparison: If the noise is assumed to be reasonably stationary, the noise spectral density could be considered valid between two consecutive silence periods and could be useful in detecting speech pulses. In the results presented in this Letter, the

spectral estimation was made with a 14 channel Mel-filter bank, the same used in recognition experiments [4], but neither logarithmic compression nor normalisation was applied. The spectral density comparison coefficient ($SD(i)$) for a frame i is defined, in the Euclidean metric context, as

$$SD(i) = 20 \times \log\left(\frac{\sqrt{\sum_{k=1}^{14}(E_k^n - E_{i,k})^2}}{\sqrt{\sum_{k=1}^{14}(E_k^n)^2}}\right) \quad (5)$$

where $S_i = (E_{i,1}, E_{i,2}, E_{i,3}, ..., E_{i,14})$ and $S^n = (E_1^n, E_2^n, E_3^n, ..., E_{14}^n)$ correspond, respectively, to the spectral estimation of frame i and that of the noise; E_k^n and $E_{i,k}$ represent the filter k output energies. The noise spectral estimation was computed as the average spectrum in 10 non-speech frames.

Stationarity coefficient: If the noise is reasonably stationary its statistical properties are constant or change slowly, or might even present fast but small variations along time. To use these features of the corrupting signal in speech pulse detection, the non-stationarity coefficient ($NST(i)$) for a frame i is defined, in the Euclidean metric context, as

$$NST(i) = 20 \times \log\left(\frac{\sqrt{\sum_{k=1}^{14}(E_{i,k} - E_{i-1,k})^2}}{\sqrt{\sum_{k=1}^{14}(E_k^n)^2}}\right) \quad (6)$$

where $S_{i-1} = (E_{i-1,1}, E_{i-1,2}, E_{i-1,3}, ..., E_{i-1,14})$ and $S_i = (E_{i,1}, E_{i,2}, E_{i,3}, ..., E_{i,14})$ correspond, respectively, to the spectral estimations of two contiguous frames.

Results: The experiments were carried out using the Noisex-92 database [3]. The signals were lowpass filtered using a 10th order Tchebychev filter with a cutoff frequency of 3700 Hz, downsampled from 16000 to 8000 samples/s, and highpass filtered using a fourth order Tchebychev filter with a cutoff frequency of 120Hz and a minimum attenuation equal to 25dB. The data signal was divided in 25ms frames without overlapping. Each frame was processed with a Hamming window before the frame energy and spectral estimation being computed.

Three noises from Noisex-92 were considered (car, speech and Lynx) and for each case one AR FIR filter was trained using the noise-only samples files and the LMS algorithm. The learning rate was made equal to $0.1/(M \times noise\_power)$ and the LMS algorithm was active for 10 training frames (250 ms). The FIR taps were set to 0 at the beginning of the iterative procedure. To determine the optimum prediction order, several configurations were tested and the one that gave the lowest prediction error was chosen. The clean and noisy speech signals belonged to the male speaker from the Noisex-92 database. Table 1 shows the optimum number of taps for each filter and the ratio G between the attenuation gain on clean speech signals and the attenuation gain on the training noise signal after the AR FIR filter being estimated. This quotient G gives an idea of the energy gap increase between noise and speech due to the AR FIR filter. The clean signals corresponded to 10 utterances (one per digit) automatically end detected.

Table 1: Optimum AR FIR order and quotient (G) between the energy attenuation gains on clean speech signal and training noise

Noise	Car	Speech noise	Lynx
Optimum FIR order	2	2	4
G (dB)	13.1	6.6	5.3

Figs. 1 and 2 present the power envelope, spectral comparison and non-stationarity coefficients before and after processing the signal with the AR FIR filter. The power envelope corresponds to the difference between the mean frame energy (dB) and the mean noise energy estimation (dB) made in 10 non-speech frames. The utterance corresponds to the digit 'one' in the car noise, with SNR equal to 0 and –6dB, respectively. The word 'one' was chosen because it presents a signal mainly concentrated in low frequencies, and constituted a more challenging problem than, e.g. the digit 'six'.

Discussion: As can be seen in Table 1, the AR analysis led to a higher attenuation on average for the noises than for speech signals. According to Figs. 1 and 2, the AR FIR filters increased the discrimination between the speech signal and backgound noise in the power, spectral comparison and non-stationarity coefficient domains. When compared with the power envelope, spectral comparison and non-stationarity coefficients slightly increased the difference between speech and non-speech pulses before FIR processing, but gave similar results after FIR at SNR equal to 0dB (Fig. 1). According to Fig. 2 (SNR = –6dB), these coefficients increased the difference between speech and non-speech pulses after FIR processing, but the improvement achieved before AR analysis was not enough to highlight the speech signal from the backgound.

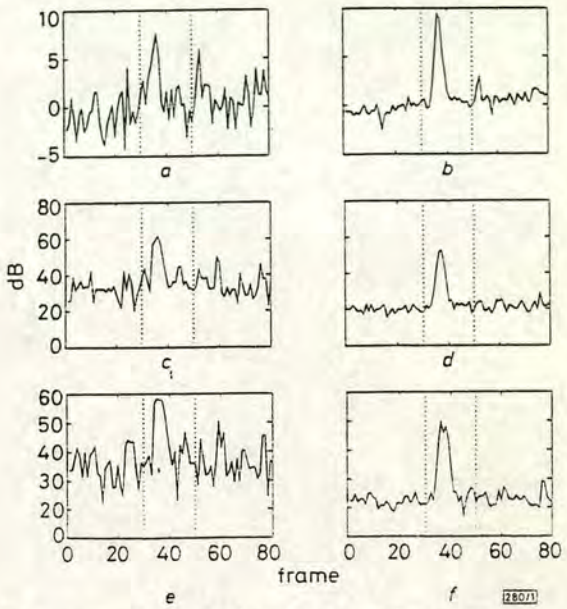


Fig. 1 *Power envelope spectral comparison and non-stationary coefficients before and after processing the signal with the AR FIR filter*

The utterance corresponds to the digit 'one' in car noise with SNR equal to 0dB
Dotted vertical lines: endpoints of speech signal
a Power envelope before FIR *b* Power envelope after FIR
c Spectral comp. before FIR *d* Spectral comp. after FIR
e Non-station. coeff. before FIR *f* Non-station. coeff. after FIR

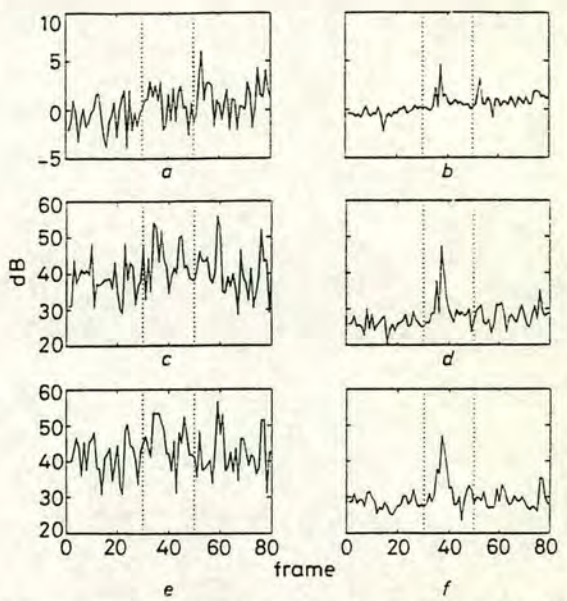


Fig. 2 *Power envelope spectral comparison and non-stationary coefficients before and after processing the signal with the AR FIR filter*

The utterance corresponds to the digit 'one' in car noise with SNR equal to –6dB
Dotted vertical lines: endpoints of speech signal
a–f As for Fig. 1

Concluding, the observed improvements were mainly due to the AR analysis, although spectral comparison and the non-stationarity coefficient might be useful at low SNRs. The AR FIR filters needed a low number of taps, the LMS algorithm seems to be fast enough to capture slow variations of the noise characteristics and only one microphone is necessary. Moreover, the AR analysis might be used by noise cancelling techniques in speech recognition. Future work includes some heuristics to develop an endpoint detector, automatic threshold estimation, and the study of AR adaptation techniques.

Acknowledgments: N.B. Yoma is supported by a grant from CNPq-Brasilia/Brasil.

7 May 1996
Electronics Letters Online No: 19960892

N.B. Yoma, F. McInnes and M. Jack (*Centre for Communication Interface Research, University of Edinburgh, 80 South Bridge, Edinburgh EH1 1HN, United Kingdom*)

References

1 JUNQUA, J.C., REAVES, B., and MAK, B.: 'A study of endpoint detection algorithms in adverse conditions: Incidence on a DTW and HMM recognizer'. Eurospeech'91, 1991, pp. 1371–1374
2 HAYKIN, S.: 'Adaptive filter theory' (Prentice Hall, Englewood Cliffs, NJ, 1991), 2nd edn.
3 VARGA, A., STEENEKEN, H.J.M., TOMLINSON, M., and JONES, D.: 'The Noisex-92 study on the effect of additive noise in automatic speech recognition', Technical report, DRA Speech Research Unit, U.K., 1992
4 YOMA, N.B., McINNES, F.R., and JACK, M.A.: 'Improved algorithms for speech recognition in noise using lateral inhibition and SNR weighting'. Eurospeech'95, 1995, pp. 461–464

WEIGHTED MATCHING ALGORITHMS AND RELIABILITY IN NOISE CANCELLING BY SPECTRAL SUBTRACTION

Nestor Becerra Yoma[*], *Fergus McInnes*, *Mervyn Jack*
Centre for Communication Interface Research
University of Edinburgh
80 South Bridge, Edinburgh EH1 1HN, U.K.
nestor@ccir.ed.ac.uk

ABSTRACT

This paper addresses the problem of speech recognition with signals corrupted by additive noise at moderate SNR. A technique based on spectral subtraction and noise cancellation reliability weighting in acoustic pattern matching algorithms is studied. A model for additive noise is proposed and used to compute the variance of the hidden clean signal information and the reliability of the spectral subtraction process. The results presented in this paper show that a proper weight on the information provided by static parameters can substantially reduce the error rate.

1. INTRODUCTION

Due to the fact that the intervals with highest energies are less corrupted by additive noise, it is reasonable to suppose that these intervals provide more reliable information for speech recognition than those intervals with lower energies. In [1] and [2] were proposed two weighted matching algorithms to take into account the local SNR. Both algorithms were tested with poorly correlated and white Gaussian noises but with different noise cancellation techniques. In [2] these two algorithms were tested in combination with a noise cancellation neural net and it was shown they could reduce the error rate. However, further experiments showed that the improvements due to the weighted Dynamic Programming algorithms depended on the neural net training conditions, and suggested that the weighting coefficient $w(t)$ should take into account not only the segmental SNR but the characteristics of the noise reduction method. Following this idea, in [3] was proposed the use of a weighting parameter based on reliability in noise cancelling. This parameter takes into account not only the local SNR but also the characteristic response of the noise cancellation method in the form of a mean distortion curve [3].

The contributions of this paper concern: a)combination of weighted matching algorithms with spectral subtraction (SS) technique; and b) analysis of SS in terms of reliability in noise cancelling. The approach covered by this paper has not been found in the literature and seems to be generic and interesting from the practical applications point of view. In this exploratory research, the techniques were tested with DTW recognition algorithms on an isolated word recognition task. DTW was used because it is a simple and generic algorithm which allows many noise cancelling techniques to be compared without the need for extensive tuning of the modelling. However, the authors believe the tecniques explored here could also be employed by a weighted Viterbi (HMM) algorithm previously proposed in [2].

[*]Supported by a grant from CNPq-Brasilia/Brasil

2. SINUSOIDAL MODEL FOR ADDITIVE NOISE

Given that $s(i)$, $n(i)$ and $x(i)$ are the clean speech, the noise and the resulting noisy signal, respectively, the additiveness condition may be set as:

$$x(i) = s(i) + n(i) \quad (1)$$

In the results presented in this paper, the signal was processed by 14 Mel filters. At the output of filter j the noisy signal is given by:

$$x_j(i) = s_j(i) + n_j(i) \quad (2)$$

and its mean energy in a frame by:

$$\overline{x_j^2(i)} = \overline{s_j^2(i)} + \overline{n_j^2(i)} + \overline{2s_j(i)n_j(i)} \quad (3)$$

where $\overline{x_j^2(i)} = \frac{1}{N}\sum_{i=1}^{N} x_j^2(i)$, $\overline{s_j^2(i)} = \frac{1}{N}\sum_{i=1}^{N} s_j^2(i)$, $\overline{n_j^2(i)} = \frac{1}{N}\sum_{i=1}^{N} n_j^2(i)$, $\overline{2s_j(i)n_j(i)} = \frac{1}{N}\sum_{i=1}^{N} 2s_j(i)n_j(i)$ and N is the length of the frames in number of samples.

If the speech signal and the noise are uncorrelated, $E(\overline{2s_j(i)n_j(i)}) = 0$ in a long term analysis, where $E()$ corresponds to the expected value. However, the condition $\overline{2s_j(i)n_j(i)} = 0$ may not be satisfied in a short term analysis (i.e. a 25 ms frame) and the noise is certainly not perfectly stationary. Consequently, once the noise is added the clean signal energy, $\overline{s_j^2(i)}$, becomes a hidden information and cannot be recovered with a 100% accuracy. As a result, $\overline{s_j^2(i)}$ should be treated as a stochastic variable and could be associated to a variance that indicates how accurate is the estimation of the clean signal energy.

Initially, the signals $s_j(i)$ and $n_j(i)$ are considered sinusoidal components with frequency f_j, the central frequency of filter j, with a phase difference ϕ. Under these assumptions,

$$\overline{x_j^2(i)} = \frac{a_{s_j}^2}{2} + \overline{n_j^2(i)} + a_{s_j}a_{n_j}cos(\phi) \quad (4)$$

where a_{s_j} and a_{n_j} are the amplitudes of the speech signal and noise components respectively: $\overline{s_j^2(i)} = a_{s_j}^2/2$ and $\overline{n_j^2(i)} = a_{n_j}^2/2$.

0-8186-7919-0/97 $10.00

3. CORRECTION OF THE SINUSOIDAL MODEL

The sinusoidal model for additive noise represented by equation (4) assumes that the components $s_j(i)$ and $n_j(i)$ at the output of filter j have frequency f_j and a phase difference ϕ in a given frame. These assumptions are not perfectly accurate in practice. Firstly, the 14 mel filters are not highly selective, which reduces the validity of the assumption of coherence between both components. Secondly, the phase ϕ between $s_j(i)$ and $n_j(i)$ is not necessarily constant and a few discontinuities in the phase difference may occur, although many of them are unlikely in a short term analysis (i.e. a 25 ms frame). However, the sinusoidal model represents the fact that there is a variance in the short term analysis and specifies the relation between this variance and the clean and noise signal levels. Due to the lack of coherence between $s_j(i)$ and $n_j(i)$ and to the discontinuity in the phase difference, the variance predicted by the model is higher than the the real one for the same frame length, and a correction should be included in (4). According to (4) and considering that the random variable ϕ was uniformly distributed between $-\pi$ and π:

$$Var[\overline{x_j^2(i)}|\overline{s_j^2(i)}, \overline{n_j^2(i)}] = 0.5a_{s_j}^2 a_{n_j}^2$$

In order to estimate the correction of the sinusoidal model, the coefficient r_j defined as

$$r_j = \frac{\overline{2s_j(i)n_j(i)}}{a_{s_j}a_{n_j}} \quad (5)$$

was computed with clean speech and only-noise frames. According to (5), $Var[r_j|\overline{s_j^2(i)}, \overline{n_j^2(i)}]$ should be equal to 0.5 but due to the lack of coherence between $s_j(i)$ and $n_j(i)$ and to the discontinuity in the phase difference,

$$Var[r_j|\overline{s_j^2(i)}, \overline{n_j^2(i)}] < 0.5$$

and a correction factor k_j needs to be included in (4):

$$\overline{x_j^2(i)} = \frac{a_{s_j}^2}{2} + \overline{n_j^2(i)} + a_{s_j}a_{n_j}\sqrt{k_j}cos(\phi) \quad (6)$$

where k_j is defined as

$$k_j = 2Var[r_j|\overline{s_j^2(i)}, \overline{n_j^2(i)}]$$

4. CHANNEL VARIANCE

With the sinusoidal model for additive noise represented by (6), the variance (or uncertainty) of the hidden information $\overline{s_j^2(i)}$ given the observed information $\overline{x_j^2(i)}$ is estimated. Solving (6) for a_{s_j} and using $\overline{s_j^2(i)} = a_{s_j}^2/2$:

$$\overline{s_j^2(i)} = a_{n_j}^2 k_j cos^2(\phi) + \overline{x_j^2(i)} - \overline{n_j^2(i)} - a_{n_j}\sqrt{k_j}cos(\phi)\sqrt{a_{n_j}^2 k_j cos^2(\phi) + 2(\overline{x_j^2(i)} - \overline{n_j^2(i)})} \quad (7)$$

The equation above sets $\overline{s_j^2(i)}$ as a function of ϕ, $\overline{n_j^2(i)}$ and $\overline{x_j^2(i)}$:

$$\overline{s_j^2(i)} = g(\phi, \overline{n_j^2(i)}, \overline{x_j^2(i)}) \quad (8)$$

The function $g(\phi, \overline{n_j^2(i)}, \overline{x_j^2(i)})$ was used to estimate $Var[\log(\overline{s_j^2(i)})|\overline{x_j^2(i)}]$ considering that the random variable ϕ was uniformly distributed between $-\pi$ and π and that $\overline{n_j^2(i)}$ is concentrated near its mean $E[\overline{n_j^2(i)}]$. The variance $Var[\log(\overline{s_j^2(i)})|\overline{x_j^2(i)}]$ is given by:

$$Var[\log(\overline{s_j^2(i)})|\overline{x_j^2(i)}] = E[\log^2(\overline{s_j^2(i)})|\overline{x_j^2(i)}] - E^2[\log(\overline{s_j^2(i)})|\overline{x_j^2(i)}]$$

where

$$E[\log^2(\overline{s_j^2(i)})|\overline{x_j^2(i)}] \simeq \frac{1}{2\pi}\int_{-\pi}^{\pi} \log^2[g(\phi, E[\overline{n_j^2(i)}], \overline{x_j^2(i)})]d\phi$$

and

$$E[\log(\overline{s_j^2(i)})|\overline{x_j^2(i)}] \simeq \frac{1}{2\pi}\int_{-\pi}^{\pi} \log[g(\phi, E[\overline{n_j^2(i)}], \overline{x_j^2(i)})]d\phi$$

The integrals for estimating $E[\log^2(\overline{s_j^2(i)})|\overline{x_j^2(i)}]$ and $E[\log(\overline{s_j^2(i)})|\overline{x_j^2(i)}]$ were computed by means of Simpson's rule with the interval $(-\pi, \pi)$ divided in 100 regular partitions. The difference $\overline{x_j^2(i)} - \overline{n_j^2(i)}$ in (7) was replaced with $\sigma(Est(\overline{s_j^2(i)}))$ (see section 5) when evaluating g().

5. SPECTRAL SUBTRACTION

Spectral subtraction (SS) may be defined as

$$Est(\overline{s_j^2(i)}) = \overline{x_j^2(i)} - E(\overline{n_j^2(i)}) \quad (9)$$

where $Est(\overline{s_j^2(i)})$ is the estimation of the clean signal energy and $E(\overline{n_j^2(i)})$ is the mean noise energy estimation made in non-speech intervals. Due to the fact that $\overline{2s_j(i)n_j(i)} = 0$ may not be true in a short term analysis and that the noise energy presents fluctuations, $Est(\overline{s_j^2(i)})$ may be negative in those channels with low SNR. In order to avoid negative magnitude estimates a rectifying function $\sigma()$is applied:

$$\sigma(Est(\overline{s_j^2(i)})) = \begin{cases} Est(\overline{s_j^2(i)}) & \text{if } Est(\overline{s_j^2(i)}) \geq \varepsilon \\ \varepsilon & \text{if } Est(\overline{s_j^2(i)}) < \varepsilon \end{cases} \quad (10)$$

where ε is an arbitrary low constant.

6. WEIGHTED MATCHING ALGORITHMS

Some modifications were included in matching algorithms in order to weight the reliability of the information extracted from testing frames. A weighting coefficient $w(t)$ ($w(t) = 1$, maximum reliability; $w(t) = 0$, minimum reliability) is associated to each testing frame in order to be employed in the modified vesions of the DTW and Viterbi (HMM) algorithms [2]. The main idea behind the modifications made on Viterbi (HMM) and DTW algorithms is that the influence of a frame on decisions must be proportional to its coefficient $w(t)$. The proposed one-step weighted DP algorithm was compared with the two-step DP algorithm proposed in [1].

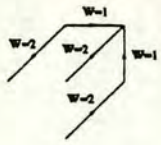


Figure 1. Local condition.

The proposed one-step DP equation that corresponds to the local condition shown in Fig.1 is given as follows :

$$G(t,r) = \min \left(\begin{array}{c} \frac{G(t-2,r-1)W(t-2,r-1)+2w(t-1)d(t-1,r)+w(t)d(t,r)}{W(t-2,r-1)+2w(t-1)+w(t)} \\ \frac{G(t-1,r-1)W(t-1,r-1)+2w(t)d(t,r)}{W(t-1,r-1)+2w(t)} \\ \frac{G(t-1,r-2)W(t-1,r-2)+2w(t)d(t,r-1)+w(t)d(t,r)}{W(t-1,r-2)+3w(t)} \end{array} \right)$$

and

$$W(t,r) = \begin{cases} W(t-2,r-1)+2w(t-1)+w(t) \\ W(t-1,r-1)+2w(t) \\ W(t-1,r-2)+3w(t) \end{cases}$$

This DP equation takes into account the weight $w(t)$ frame by frame, and the calculation of the overall distance, $G(t,r)$, is affected by $d(t,r)$ according to $w(t)$: if $w(t)=1$ (high *reliability* or local SNR), the weight of $d(t,r)$ is maximum; if $w(t)=0$ (very low *reliability* or local SNR), the importance of $d(t,r)$ is zero.

The algorithm proposed in [1] consists of the following two-step processing. Firstly, the optimal alignment path $c_k=(t_k,r_k), k=1,2,...,K$ is obtained using the ordinary DP matching algorithm, where t_k and r_k are the frame numbers of the testing and reference patterns respectively. The second step is the calculation of the global distance between the utterances weighted by $w(t_k)$ along the optimal path obtained at the first step.

7. RELIABILITY IN NOISE CANCELLING

It is reasonable to suppose that the uncertainty related to SS in a channel would be proportional to $Var[\log(\overline{s_j^2(i)})|\overline{x_j^2(i)}]$: the higher $Var[\log(\overline{s_j^2(i)})|\overline{x_j^2(i)}]$ is, the less reliable is the information provided by $Est(\overline{s_j^2(i)})$; and the lower this variance is, the higher is the probability of $Est(\overline{s_j^2(i)})$ being close to the clean signal information $\overline{s_j^2(i)}$. The weighting coefficient $w(t)$ [2] [3], to be used by the weighted algorithms (section 6) and that attempts to measure how reliable is the result of the noise cancelling method in a frame, could be related to the mean $Var[\log(\overline{s_j^2(i)})|\overline{x_j^2(i)}]$ in all the channels by means of the following function (Fig. 2) [3]:

$$w(t) = \begin{cases} 1 & \text{if } MeanVar \le \delta \\ \frac{\delta}{MeanVar} & \text{if } MeanVar > \delta \end{cases} \quad (11)$$

where

$$MeanVar = \frac{1}{14}\sum_{j=1}^{14} Var[\log(\overline{s_j^2(i)})|\overline{x_j^2(i)}] \quad (12)$$

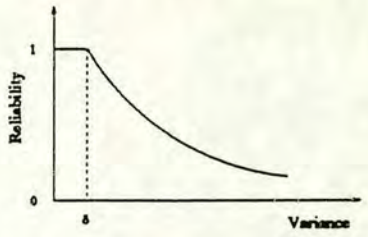


Figure 2. Reliability coefficient vs variance.

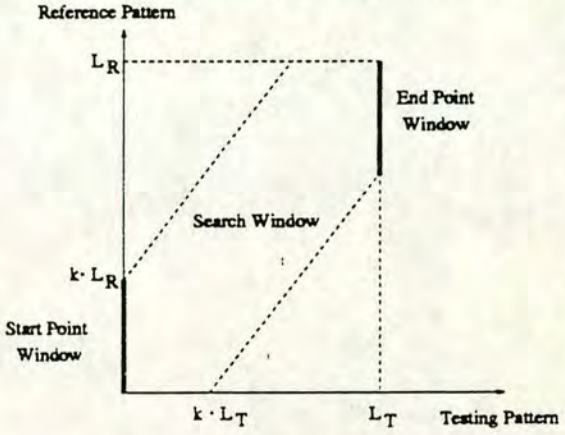


Figure 3. End-point constraints relaxation.

8. END POINT RELAXATION

The reliability in noise cancelling weighting was tested by means of isolated word Dynamic Time Warping algorithms. The isolated words were automatically end detected using an algorithm based on autoregressive analysis of noise [4] and the average length of the testing utterances decreases as the SNR gets more severe. Consequently, the endpoint constraints on the DP algorithms were relaxed by means of opening up the ends of the search region allowing the alignment path to start by comparing the first frame of the testing pattern with any of the first reference frames inside the search window, and to end by comparing the last test frame with any of the last reference frames inside the search window (see Fig. 3). Due to the fact that the length of the testing utterances presented a high variation, the sides of the search window were made proportional to the utterance length.

9. EXPERIMENTS

The proposed methods were tested with speaker-dependent isolated word (English digits from 0 to 9) recognition experiments. The tests were carried out employing the two speakers (one female and one male) from the Noisex database [5].

The signals were low pass filtered by using a filter with cut off frequency 3700 Hz, down sampled from 16000 to 8000 samples/sec, and high-pass filtered by employing a filter with cut off frequency 120 Hz. The data signal was divided in 25ms frames with 12.5ms overlapping. Each frame was processed with a Hamming window before the spectral estimation. The band from 300 to 3400 Hz was covered with 14 Mel 2nd order IIR digital filters. At the output of each channel the energy was computed and SS was applied. Finally, 10 cepstral coefficients were computed.

The results presented in this paper were achieved with

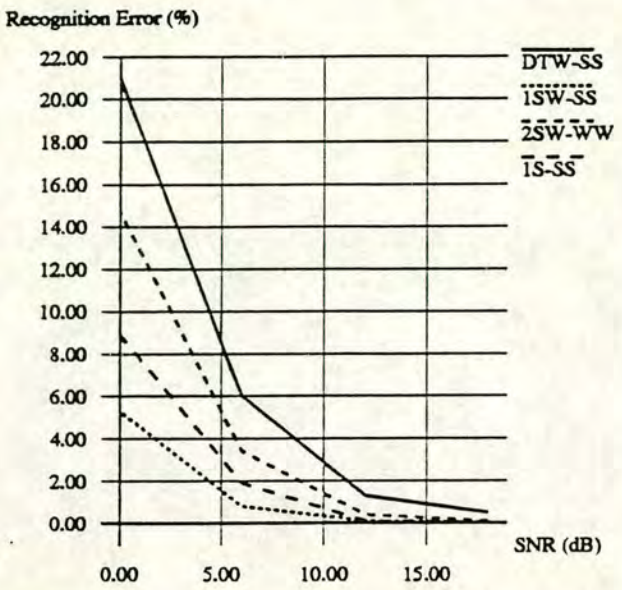


Figure 4. Results for the car noise (Noisex database).

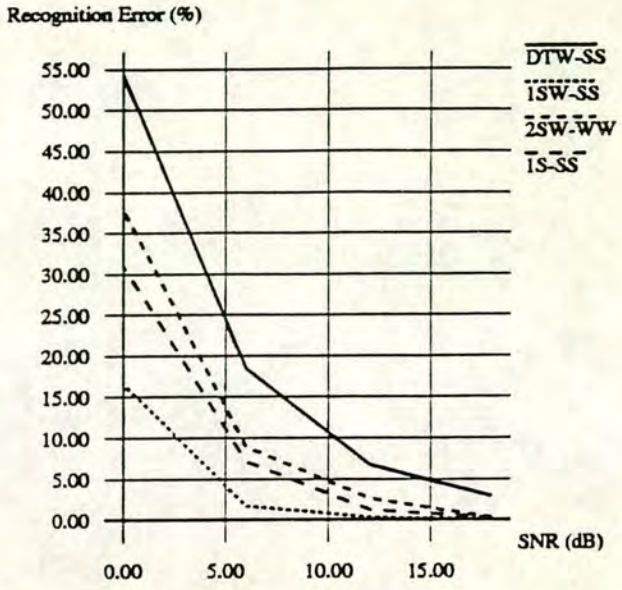


Figure 5. Results for the speech noise (Noisex database).

1000 recognition tests for each SNR. The following configurations were tested: the ordinary DTW algorithm [6] with SS (DTW-SS); the proposed one-step weighted DP algorithm [2] with SS (1SW-SS); the two-step DP matching [1] (2SW-SS) also with SS; and finally, the proposed one-step DP algorithm with SS but without reliability in noise cancelling weighting, $w(t) = 1$ (1S-SS). The constant δ was made equal to 43, a value that was shown to be suitable according to some tests. For each configuration several search window widths, k (Fig. 3), were tested and the one that gave minimum error rate was chosen to plot the graphs shown in Figs. 4 and 5.

10. DISCUSSION AND CONCLUSION

As can be seen in Figs. 4 and 5, the one step algorithm in combination with the noise cancellation reliability weighting gave the lowest error rate. This reduction in the error rate was due to a) the ability of the one step algorithm in normalising the overall distance to the length of the alignment path, and b) the information provided by the noise cancellation reliability coefficient. The ordinary DTW does not take into consideration which point of the start window the optimal alignment path begins and was very sensitive to the search window width. Consequently, when k was increased (Fig. 3), DTW-SS and 2SW-SS increased the error rate after reaching an optimum search window. On the other hand, the DP equation shown in section 6 computes the overall weight $W(i, j)$ step-by-step and was almost independent to the alignment path length. As a result, 1SW-SS should be compared with 1S-SS in order to separate the improvement due to the alignment path normalisation and the one due to the noise cancelling reliability weighting.

When compared with 1S-SS, 1SW-SS showed reductions of 58% and 40% in the error rate at SNR=6dB and 0dB for the car noise. At SNR=18dB and 12dB both configurations gave error rate equal to 0 and 0.1%, repectively. For the speech noise, 1SW-SS presented reductions of 75%, 76% and 46% at SNR=12, 6 and 0dB. At SNR=18dB the error rate went from 0.3% to 0. As can be seen, the improvement due to the reliability weighting was higher for the speech noise than for the car one. This must result from the facts that the speech noise is less stationary than the car noise so the estimation of noise energy is less accurate, and that the reliability coefficient is also a function of the local SNR so low energy intervals have low weight in the pattern matching process. Therefore, noise cancellation reliability weighting made the SS process more robust to variations in the noise stationarity.

REFERENCES

[1] Hidefumi Kobatake, Yousuke Matsunoo. *Degraded Word Recognition Based on Segmental Signal-to-Noise Ratio Weighting.* ICASSP 1994, Vol. I, pp.425-428.

[2] N.B.Yoma, F.R.McInnes, M.A.Jack. *Improved Algorithms for Speech Recognition in Noise Using Lateral Inhibition and SNR Weighting.* Eurospeech'95, pp.461-464.

[3] N.B.Yoma, F.R.McInnes, M.A.Jack. *Use of a Reliability coefficient in noise cancelling by Neural Net and Weighted Matching Algorithms.* ICSLP'96, pp. 2297-2300.

[4] N.B.Yoma, F.R.McInnes, M.A.Jack. *Robust speech pulse detection using adaptive noise modelling.* IEE Electronics Letters, Vol. 32, No. 15, July 1996, pp. 1350-1352.

[5] A. Varga, H.J.M. Steeneken, M. Tomlinson and D. Jones. *The Noisex-92 study on the effect of additive noise in automatic speech recognition.* Technical report, DRA Speech Research Unit, U.K., 1992.

[6] H. Sakoe, S. Chiba. *Dynamic Programming Algorithm Optimization for Spoken Word Recognition.* IEEE Trans. on ASSP, vol. ASSP-26, No 1, Feb. 1978, pp. 43-49

SPECTRAL SUBTRACTION AND MEAN NORMALIZATION IN THE CONTEXT OF WEIGHTED MATCHING ALGORITHMS

Nestor Becerra Yoma, Fergus R. McInnes, Mervyn A. Jack*
Centre for Communication Interface Research, University of Edinburgh
80 South Bridge, Edinburgh EH1 1HN, U.K.
E-Mail:nestor@ccir.ed.ac.uk

ABSTRACT

Additive and convolutional noises are the main problems to be solved in order to make speech recognition successful in real applications. A model for additive noise is used to deduce a spectral subtraction (SS) estimation and to show that the channel transfer function could be effectively removed after the additive noise being cancelled by SS. Then, SS and mean normalization are tested in combination with a weighting procedure to reduce the influence of the rectifying function. All the experiments were done in the context of weighted matching algorithms and the approaches proved effective in cancelling both additive noise and the transmission channel function.

1. INTRODUCTION

In [1], a model for additive noise using IIR filters was proposed and used to compute the reliability related to the spectral subtraction (SS) process. The reliability in noise cancelling was used to weight DP algorithms and shown to be useful in reducing the error rate. Nevertheless, the low selectivity of the IIR filters made the system more vulnerable to convolutional distortions and the use of a DFT bank filter is desirable because it provides an infinite rejection outside the filter band.

If there is only convolutional distortion, a widely used technique is Cepstral Mean Normalizatoin (CMN). CMN is effective and efficient but its behaviour is hard to predict when additive noise is also present [2].

The contributions of this paper concern: a)the generalisation of the model for additive noise for the case of DFT filters; b)the proof that under some conditions, the log of the SS estimation is equal to the expected value of the log of the clean signal energy; and c) the proof that the effect of an unmatched transmission channel can effectively be removed by means of the classic mean normalization technique after SS. The approach covered by this paper has not been found in the literature and seems to be generic and interesting from the practical applications point of view. The results presented in this paper provide a theoretical justification for the use of mean normalization after SS, and show that both techniques in combination with weighted matching algorithms can effectively remove both additive and convolutional distortions.

*Supported by a grant from CNPq-Brasilia/Brasil

2. MODEL FOR ADDITIVE NOISE USING DFT FILTERS

Given that $s(i)$, $n(i)$ and $x(i)$ are the clean speech, the noise and the resulting noisy signal, respectively, the additiveness condition in the temporal domain may be set as:

$$x(i) = s(i) + n(i) \tag{1}$$

In the results presented in this paper, the signal was processed by 14 DFT Mel filters. If $S(k)$, $N(k)$ and $X(k)$ correspond to the FFT transform of $s(i)$, $n(i)$ and $x(i)$ at the point k, and ϕ_k is the phase difference between $S(k)$ and $N(k)$, the additiveness condition is then set by:

$$X(k) = S(k) + N(k) \tag{2}$$

According to the cosine rule,

$$|X(k)|^2 = |S(k)|^2 + |N(k)|^2 + 2 \cdot |S(k)| \cdot |N(k)| \cdot cos(\varphi_k) \tag{3}$$

The energy at the output of the filter m, $\overline{x_m^2}$, is computed by means of:

$$\overline{x_m^2} = \sum_{k \in filter\ m} G(m,k) \cdot |X(k)|^2 \tag{4}$$

where $G(m,k)$ is the set of weights that define the filter m. If $|X(k)|^2$ in (4) is replaced with the expression given in (3), $\overline{x_m^2}$ can be set as

$$\overline{x_m^2} = \overline{s_m^2} + \overline{n_m^2} + \sum_{k \in filter\ m} 2 \cdot G(m,k) \cdot |S(k)| \cdot |N(k)| \cdot cos(\phi_k) \tag{5}$$

where: $\overline{s_m^2}$ and $\overline{n_m^2}$ are the filter m mean frame energy of the clean speech and noise signal, respectively.

Assuming that the phase difference $\phi(k) = \phi$, $N(k)$ and $X(k)$ are considered constant inside each one of the 14 DFT Mel filters indexed by m:

$$\overline{x_m^2} = \overline{s_m^2} + \overline{n_m^2} + 2 \cdot \sqrt{\overline{s_m^2}} \cdot \sqrt{\overline{n_m^2}} \cdot cos(\phi) \quad (6)$$

The model for additive noise represented by (6) assumes that the components $|S(k)|$ and $|N(k)|$ and the phase difference ϕ are constant inside every filter in a given frame. These assumptions are not perfectly accurate in practice. Firstly, the 14 DFT mel filters are not highly selective, which reduces the validity of the assumption of low variation of these parameters inside the filters. Secondly, the phase ϕ between $|S(k)|$ and $|N(k)|$ is not necessarily constant and a few discontinuities in the phase difference may occur, although many of them are unlikely in a short term analysis (i.e. a 25 ms frame). However, this model represents the fact that there is a variance in the short term analysis and specifies the relation between this variance and the clean and noise signal levels. Due to these approximations the variance predicted by the model is higher than the the real one for the same frame length, and a correction should be included. Using (6) and considering that ϕ was uniformly distributed between $-\pi$ and π:

$$Var[\frac{\overline{x_m^2}}{2} | \overline{s_m^2}, \overline{n_m^2}] = 0.5 \cdot \overline{s_m^2} \cdot \overline{n_m^2}$$

In order to estimate the correction of the model, the coefficient k_m defined as

$$k_m = \frac{\overline{x_m^2} - \overline{s_m^2} - \overline{n_m^2}}{2 \cdot \sqrt{\overline{s_m^2}} \sqrt{\overline{n_m^2}}} \quad (7)$$

was computed with clean speech and only-noise frames. According to (6), $Var[k_m | \overline{s_m^2}, \overline{n_m^2}]$ should be equal to 0.5 but due to the approximations this variance is lower than 0.5 and a correction factor c_m needs to be included in (6):

$$\overline{x_m^2} = \overline{s_m^2} + \overline{n_m^2} + 2 \cdot \sqrt{c_m} \cdot \sqrt{\overline{s_m^2}} \cdot \sqrt{\overline{n_m^2}} \cdot cos(\phi) \quad (8)$$

where c_m is defined as

$$c_m = 2Var[k_m | \overline{s_m^2}, \overline{n_m^2}]$$

Solving (8) for $\overline{s_m^2}$

$$\overline{s_m^2}(\phi, \overline{n_m^2}, \overline{x_m^2}) = 2 \cdot A^2 \cdot cos^2(\phi) + B - 2 \cdot A \cdot cos(\phi) \cdot \sqrt{A^2 \cdot cos^2(\phi) + B} \quad (9)$$

where $A = \sqrt{\overline{n_m^2} c_m}$ and $B = \overline{x_m^2} - \overline{n_m^2}$.

3. CHANNEL VARIANCE AND RELIABILITY IN NOISE CANCELLING BY SS

With the model for additive noise represented by (9), the variance (or uncertainty) of the hidden information $\overline{s_m^2}$ given the observed information $\overline{x_m^2}$ is estimated in the logarithmic domain assuming that the random variables ϕ and $\overline{n_m^2}$ are uncorrelated, ϕ is uniformly distributed between $-\pi$ and π and that $\overline{n_m^2}$ is concentrated near its mean $E[\overline{n_m^2}]$. The variance $Var[\log(\overline{s_m^2})|\overline{x_m^2}]$ is given by:

$$Var[\log(\overline{s_m^2})|\overline{x_m^2}] = E[\log^2(\overline{s_m^2})|\overline{x_m^2}] - E^2[\log(\overline{s_m^2})|\overline{x_m^2}] \quad (10)$$

where

$$E[\log^2(\overline{s_m^2})|\overline{x_m^2}] \simeq \frac{1}{2\pi} \int_{-\pi}^{\pi} \log^2[\overline{s_m^2}(\phi, E[\overline{n_m^2}], \overline{x_m^2})] d\phi$$

$$E[\log(\overline{s_m^2})|\overline{x_j^2}] \simeq \frac{1}{2\pi} \int_{-\pi}^{\pi} \log[\overline{s_m^2}(\phi, E[\overline{n_m^2}], \overline{x_j^2})] d\phi$$

Equation (16) below suggests that the expected value of the hidden information $\log(\overline{s_m^2})$ is approximately equal to the log of the spectral subtraction (SS) estimation ($Est(\overline{s_m^2})$) if $Est(\overline{s_m^2}) = \overline{x_m^2} - E[(\overline{n_m^2})]$, where $E[\overline{n_m^2}]$ is the mean noise energy estimation made in non-speech intervals. In order to avoid negative magnitude estimates a rectifying function $r()$ is applied:

$$r(Est(\overline{s_m^2}), \varepsilon) = \begin{cases} Est(\overline{s_m^2}) & \text{if } Est(\overline{s_m^2}) \geq \varepsilon \\ \varepsilon & \text{if } Est(\overline{s_m^2}) < \varepsilon \end{cases} \quad (11)$$

where ε is an arbitrary low constant. As in [1] the weighting coefficient w, to be used by the weighted algorithms [1] and that attempts to measure how reliable is the result of the noise cancelling method in a frame, was defined as:

$$w = \begin{cases} 1 & \text{if } TotalVar \leq \delta \\ \frac{\delta}{TotalVar} & \text{if } TotalVar > \delta \end{cases} \quad (12)$$

where

$$TotalVar = \sum_{m=1}^{14} Var[\log(\overline{s_m^2})|\overline{x_m^2}] \quad (13)$$

$Var[\log(\overline{s_m^2})|\overline{x_m^2}]$ was estimated by means of $E[\log(\overline{s_m^2})|\overline{x_j^2}] \simeq \log[\overline{x_m^2} - E[\overline{n_m^2}]]$ (see section 4) and the integral for estimating $E[\log^2(\overline{s_m^2})|\overline{x_m^2}]$ was computed by means of Simpson's rule with the interval $(-\pi, \pi)$ divided in 100 regular partitions and replacing the difference $B = \overline{x_m^2} - \overline{n_m^2}$ in (9) with $r(Est(\overline{s_m^2}), \varepsilon)$.

4. ADDITIVE AND CONVOLUTIONAL NOISE CANCELLING

Given the model for additive noise represented by (9), the expected value of $\log[\overline{s_m^2}(\phi, \overline{n_m^2}, \overline{x_m^2})]$ given the observed information $\overline{x_m^2}$ would be:

$$E[\log(\overline{s_m^2})|\overline{x_m^2}] = E[\log(2 \cdot \frac{A^2}{B} \cdot cos^2(\phi) + 1 - 2 \cdot \frac{A}{B} \cdot cos(\phi) \cdot \sqrt{A^2 \cdot cos^2(\phi) + B})|\overline{x_m^2}] + E[\log(B)|\overline{x_m^2}] \quad (14)$$

Assuming again that the random variables ϕ and $E[\overline{n_m^2}]$ are uncorrelated, ϕ is uniformly distributed between $-\pi$ and π and that $\overline{n_m^2}$ is concentrated near its mean $E[\overline{n_m^2}]$, $E[\log(\overline{s_m^2})|\overline{x_m^2}]$ can be written as:

$$E[\log(\overline{s_m^2})|\overline{x_j^2}] \simeq \frac{1}{\pi}\int_{-\pi}^{\pi}\log[\sqrt{\frac{(\bar{A})^2\cdot cos^2(\phi)}{\bar{B}}+1}-\frac{\bar{A}}{\sqrt{\bar{B}}}\cdot cos(\phi)]d\phi+\log(\bar{B}) \quad (15)$$

where $\bar{A}=\sqrt{E[\overline{n_m^2}]\cdot c_m}$ and $\bar{B}=\overline{x_m^2}-E[\overline{n_m^2}]$. Replacing the variable ϕ with $u=-\frac{\bar{A}}{\sqrt{B}}\cdot cos(\phi)$, the integral in (2) becomes

$$\frac{1}{\pi}\int_{-\pi}^{\pi}\log[\sqrt{\frac{(\bar{A})^2\cdot cos^2(\phi)}{\bar{B}}+1}-\frac{\bar{A}}{\sqrt{\bar{B}}}\cdot cos(\phi)]d\phi = \frac{2\cdot\sqrt{\bar{B}}}{\bar{A}\cdot ln(10)\cdot\pi}\int_{-\frac{\bar{A}}{\sqrt{B}}}^{\frac{\bar{A}}{\sqrt{B}}}\frac{sinh^{-1}(u)}{\sqrt{1-\frac{\bar{B}}{(\bar{A})^2}\cdot u^2}}du=0$$

because the functions $sinh^{-1}(u)$ and $\sqrt{1-\frac{\bar{B}}{(\bar{A})^2}\cdot u^2}$ are odd and even respectively. Consequently,

$$E[\log(\overline{s_m^2})|\overline{x_j^2}] \simeq \log(\overline{x_m^2}-E[\overline{n_m^2}]) \quad (16)$$

This result means, according to the model for additive noise, that the expected value of the hidden information $\log(\overline{s_m^2})$ is equal to the log of the SS estimation if SS is defined as being $\overline{x_m^2}-E[\overline{n_m^2}]$. If the gain introduced by the transmission channel is considered constant inside each one of the 14 DFT Mel filters the convolutional distortion can be represented by $H=[h_1,h_2,h_3,....,h_m,...h_{14}]$ and due to the fact that H is constant along time

$$E[\log(h_m\cdot\overline{s_m^2})|\overline{x_j^2}] \simeq E[\log(\overline{s_m^2})|\overline{x_j^2}]+h_m^l \quad (17)$$

where $h_m^l=\log(h_m)$. Therefore, the convolutional distortion could be effectively removed after the additive noise being cancelled by means of SS.

5. SS AND MEAN NORMALIZATION

If there is only convolutional distortion, a widely used technique is Cepstral Mean Normalization (CMN). However, when the speech signal is also corrupted by additive signals, CMN loses its effectiveness [2]. Nevertheless, as was shown in the last section, the effect of an unmatched transmission channel could effectively be removed after the additive noise being removed by means of SS given that the SS estimation, $Est(\overline{s_m^2})$, is defined as being equal to $\overline{x_m^2}-E[\overline{n_m^2}]$. Due to the fact that $Est(\overline{s_m^2})$ may be negative in those channels with low SNR a rectifying function $r()$is applied. In order to model the effect introduced by this rectifying function, the distribution of $\overline{n_m^2}$ needs to be known but this is difficult to achieve in real applications where the noise should be estimated in short non-speech intervals.

The insertion of a transmission channel results in an additive constant in both the logarithmic and cepstral domain, and can be cancelled by subtracting the mean from all input vectors. In this paper the mean normalization technique was applied in the logarithmic domain, before the cepstral transform. The mean was computed by

$$\overline{\log[Est(\overline{s_m^2})]} = \frac{\sum_{k=1}^{K}w(k,m)\cdot\log[Est(\overline{s_m^2})]}{\sum_{k=1}^{K}w(k,m)} \quad (18)$$

where K is the number of frames of the utterance (or set of utterances) and $w(k,m)=1$ for the ordinary arithmetic mean. A weighted arithmetic mean was also tested where $w(k,m)$ was defined as:

$$w_{k,m} = \begin{cases} 1 & \text{if } Var[\log(\overline{s_{k,m}^2})|\overline{x_m^2}]\le\delta \\ \frac{\delta}{Var[\log(\overline{s_{k,m}^2})|\overline{x_m^2}]} & \text{if } Var[\log(\overline{s_{k,m}^2})|\overline{x_m^2}]>\delta \end{cases} \quad (19)$$

where $Var[\log(\overline{s_{k,m}^2})|\overline{x_m^2}]$ was estimated according to (10). The idea of (19) is to give a low weight to those bands with low SNR in the computation of the means in order to reduce the effect introduced by the rectifying function.

6. EXPERIMENTS

The proposed methods were tested with speaker-dependent isolated word (English digits from 0 to 9) recognition experiments. The tests were carried out employing the two speakers (one female and one male) and the car noise from the Noisex database [3]. The isolated words were manually rather than automatically end detected in order to eliminate any effect introduced by the discriminative selection of speech intervals with higher energies. The signal processing was as in [1]. At the output of each channel the energy was computed. SS was applied and the log of the energy was calculated. In every frame, the log energies were normalized to the highest component and 10 cepstral coefficients were computed. In these experiments the noise estimation was made only once using just 250ms of non-speech signal and was kept constant for all the experiments at the same global SNR. The results presented in this paper were achieved with 1000 recognition tests for each SNR. Unless the opposite is specified, a one-step weighted DP algorithm previously proposed in [1] was used in all the experiments. Where spectral tilt experiments were performed clean reference utterances are compared with noisy testing utterances corrupted by additive plus convolutional noise. The tilt applied was a flat frequency response up to a break point frequency of 250Hz followed by a +6dB/oct tilt above 250Hz. The +6dB/oct spectral tilt was chosen instead of +3dB/oct, usually used in many papers, to make the testing conditions more severe.

In all the experiments SS was applied in the linear domain utterance by utterance and the convolutional distortion was cancelled after SS using one, two, five or 10 additive-noise-free utterances (from different words of the

vocabulary) every time by means of mean normalization in the logarithmic domain (LMN). In all the tests where the weighted DP algorithm was used the parameter δ was made equal to 10 in (12) and (19), a value that was shown to be suitable according to some experiments.

In experiments with SS and mean normalization, the means were computed in the logarithmic domain, before cepstral transform. The folowing configurations were tested: *SS*, SS with ordinary DTW; *WSS*, SS with the one-step weighted algorithm [1]; $WSS - LMN$, SS and mean normalization with the ordinary arithmetic mean (18); and $WSS - WLMN$, SS and mean normalization with the weighted arithmetic mean (18)(19). The results are shown in Table 1 (without spectral tilt) and Table 2 (with spectral tilt). Figure 1 shows the recognition for $WSS - WLMN$ using different number of utterances to cancel the convolutional noise. In Tables 1 and 2, the means were estimated using 10 additive-noise-free utterances (one per word of the vocabulary.

Table 1: Recognition error rate (%) for speech signal corrupted only by additive noise (car).

SNR	*18dB*	*12dB*	*6dB*	*0dB*
SS	4.4	7.3	12.9	21.5
WSS	0.1	0.4	1.8	8.6
WSS-LMN	0.3	2.9	8.6	28.7
WSS-WLMN	0.3	0.7	2.7	8.2

Table 2: Recognition error rate (%) for speech signal corrupted by additive noise (car) and spectral tilt (6dB/oct).

SNR	*18dB*	*12dB*	*6dB*	*0dB*
SS	24.6	24.8	29.15	36.4
WSS	23.0	26.3	32.5	40.2
WSS-LMN	0.3	1.8	6.9	21.8
WSS-WLMN	0.4	0.7	3.6	10.7

7. DISCUSSION

As can be seen in table 1, *WSS* (weighted DP algorithm with SS) showed a substantial reduction in the error rate in all the SNR's when compared with *SS* (ordinary DTW with SS). When compared with *WSS*, $WSS - LMN$ increased the error rate. However, when the weighted arithmetic mean was used $WSS - WLMN$, the mean normalization almost did not affect the recognition accuracy. According to table 2, the spectral tilt dramatically decreased the recognition accuracy at all the SNR's for *SS* and *WSS*. The use of the ordinary mean normalization technique $WSS - LMN$substantially reduced the error rate, but the best results were achieved in $WSS-WLMN$ with the weighted mean. Comparing the results of the table 2 with the ones in table 1, $WSS - WLMN$ was almost completely robust to the convolutional distortion

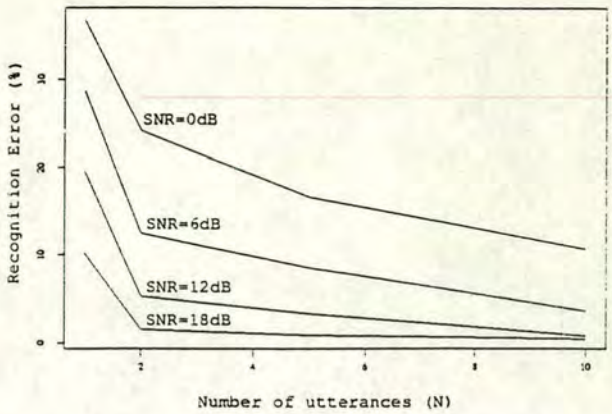


Figure 1: Recognition error rate (%) for speech signal corrupted by additive noise (SNR equal to 18, 12, 6 and 0 dB), and spectral tilt (6dB/oct) as a function of the number of utterances (N) used to estimate the weighted arithmetic mean in $WSS - WLMN$.

at all the SNR's. However, as can be seen in Figure 1, the mean normalization technique is strongly dependent on the length of the speech signal used to estimate the coefficient means and and the required number of utterances apparently increases for lower SNR's.

8. CONCLUSION

The results presented in this paper show that the channel response can effectively be removed after the additive noise being cancelled by means of SS. even when additive noise is estimated with just a few frames. In these experiments the noise estimation was made only once using just 250ms of non-speech signal and was kept constant for all the experiments at the same SNR. Moreover, weighting the information along the noisy speech signal helped to cancel both additive and convolutional noises and good results were achieved with techniques easily implemented such as SS and mean normalization.

References

[1] N.B.Yoma, F.R.McInnes, M.A.Jack. *Weighted Matching Algorithms and Reliability in Noise Cancelling by Spectral Subtraction.* Proceedings ICASSP 97, Vol.2, pp. 1171-1174.

[2] M.F Gales, S.J. Young. *Robust speech recognition in additive and convolutional noise using parallel model combination.* Computer Speech and Language (1995)9, pg. 289-307.

[3] A. Varga, H.J.M. Steeneken, M. Tomlinson and D. Jones. *The Noisex-92 study on the effect of additive noise in automatic speech recognition.* Technical report, DRA Speech Research Unit, U.K., 1992.

References

A.Acero, & R.Stern. (1990). Environmental robustness in automatic speech recognition. *Pages 849–852 of: Proceedings ICASSP'90.*

A.Papoulis. (1991). *Probability, Random Variables, and Stochastic Processes*. 3rd edn. McGraw-Hill.

A.P.Dempster, N.M.Laird, & D.B.Rubin. (1977). Maximum-likelihood from incomplete data via the EM algorithm. *J. Royal Statist. Soc.Ser.B (methodological)*, **39**, 1–38.

A.Varga, H.J.M.Steeneken, M.Tomlinson, & D.Jones. (1992). *The Noisex-92 study on the effect of additive noise in automatic speech recognition*. Technical Report. DRA Speech Research Unit, UK.

A.Wakao, K.Takeda, & F.Itakura. (1996). Variability of Lombard effects under different noise conditions. *Pages 2009–2012 of: Proceedings ICSLP'96.*

B.Mak, J.C.Junqua, & B.Reaves. (1992). A robust speech/non-speech detection algorithm using time and frequency-based features. *Pages I–269,272 of: Proceedings ICASSP'92.*

B.Raj, E.B.Gouvea, P.J.Moreno, & R.M.Stern. (1996). Cepstral Compensation by Polynomial approximation for environment-independent speech recognition. *Pages 2340–2344 of: Proceedings ICSLP'96.*

Compernolle, D.Van. (1989). Noise adaptation in a hidden Markov model speech recognition system. *Computer Speech and Language*, **3**, 151–167.

C.W.Seymour, & M.Niranjan. (1994). An HMM based Cepstral-domain speech enhancement scheme. *Pages 1595–1598 of: Proceedings ICSLP'94.*

D.A.Pierre. (1986). *Optimization Theory with Applications*. Dover Edition.

D.Burshtein. (1996). Robust Parametric Modeling of Durations in Hidden Markov Models. *IEEE Trans. ASSP*, **4**(3).

D.E.Rumelhart, G.E.Hinton, & R.J.Williams. (1987). Learning Internal Representations by Error Propagation. *Chap. 8 of:* D.E., & J.L.McClelland (eds), *Parallel Distributed Processing:Explorations in the Microstructures of Cognition, Vol.1*. Cambridge, MA: MIT Press.

D.Mansour, & B.H.Juang. (1989). The short-time modified coherence representation and noisy speech recognition. *IEEE Trans. ASSP*, **37**, 795–804.

F.H.Liu, A.Acero, & R.Stern. (1992). Efficient joint compensation of speech for the effects of additive noise and linear filtering. *Pages 257–260 of: Proceedings'92 ICASSP.*

Gales, M.F. (1995). *Model-Based Techniques for Noise Robust Speech Recognition*. PhD Thesis. Engineering Department, Cambridge University.

Gales, M.F, & S.Young. (1995). Robust speech recognition in additive and convolutional noise using parallel model combination. *Computer Speech and Language*, **9**, 289–307.

Gales, M.F, & S.Young. (1996). Robust Continuous Speech Recognition Using Parallel Model Combination. *IEEE Trans. on Speech and Audio Processing*, **4**(5), 352–359.

Ghitza, O. (1987). Robustness against noise: the role of timing-synchrony measurement. *Pages 2372–2375 of: Proceedings ICASSP'87.*

H.Hermansky, N.Morgan, A.Bayya, & P.Kohn. (1991). Compensation for the effect of the Communication Channel in Auditory-like Analysis of speech (RASTA-PLP). *Pages 1367–1370 of: Proceedings Eurospeech'91.*

H.Hermansky, N.Morgan, & H.Hirsch. (1993). Recognition of speech in additive and convolutional noise based on Rasta spectral processing. *Pages II–83/86 of: Proceedings ICASSP'93.*

H.Kobatake, & Y.Matsunoo. (1994). Degraded Word Recognition Based on Segmental Signal-to-Noise Ratio Weighting. *In: Proceedings ICASSP'94.*

H.Sakoe, & S.Chiba. (1978). Dynamic Programming Algorithm Optimization for Spoken Word Recognition. *IEEE Trans. on ASSP*, **vol. ASSP-26**(1), 43–49.

J.C.Junqua. (1989). The Lombard reflex and its role on human listeners and automatic speech recognizers. *J.Acoust. Soc. Am.*, **85-2**, 849–900.

J.D.Ferguson. (1980). Variable duration models for speech. *Pages 143–179 of:* J.D. Ferguson, Ed. Princeton, NJ (ed), *Proc. Symp.Applic. Hidden Markov Models Text Speech.*

J.Koehler, N.Morgan, H.Hermansky, H.G.Hirsch, & G.Tong. (1994). Integrating Rasta-PLP into speech recognition. *Pages I–421/424 of: Proceedings ICASSP'94.*

J.M.Romano. (1996). *LMS algorithm.* Personal Communications.

K.Laurila. (1997). Noise robust speech recognition with state duration constraints. *Pages 871–874 of: Proceedings ICASSP'97.*

L.E.Baum, & J.E.Eagon. (1967). An inequality with applications to statistical estimation for probabilistic functions of a Markov process and to models for ecology. *Bull.AMS*, **73**, 360–363.

L.E.Baum, T.Petrie, G.Soules, & N.Weiss. (1970). A maximization technique occuring in the statistical analysis of probabilistic functions of Markov chains. *Ann.Math.Stat.*, **41**, 164–171.

L.Lamel, L.R.Rabiner, A.E.Rosenberg, & J.G.Wilpon. (1981). An Improved Endpoint Detector for Isolated Word Recognition. *IEEE Transactions on Acoustic, Speech and Signal Processing*, **ASSP-29**(4), 777–785.

L.Rabiner, & B.H.Juang. (1993). *Fundamentals of speech signal processing.* Prentice Hall.

L.R.Bahl, & et al. (1986). Maximum mutual information estimation of hidden Markov model parameters for speech recognition. *Pages 49–52 of: Proceedings ICASSP'86.*

L.R.Rabiner, & S.Levinson. (1981). Isolated and connected word recognition-theory and selected applications. *IEEE Trans. Commun.*, **COM-29**.

L.R.Rabiner, B.H.Juang, S.E.Levinson, & M.M.Sondhi. (1985). Recognition of isolated digits using hidden Markov models with continuous mixture densities. *AT&T Technical Journal*, **64**, 1211–1234.

L.R.Rabiner, J.G.Wilpon, & F.K.Soong. (1989). High Performance connected digit recognition using Hidden Markov Models. *IEEE Trans. ASSP*, **37**, 1214–1225.

M.Berouti, R.Schwartz, & J.Makhoul. (1979). Enhancement of Speech Corrupted by Acoustic Noise. *Pages 208–211 of: Proceedings ICASSP'79.*

M.F.Gales. (1997). 'Nice' model-based compensation approach to robust speech recognition. *Pages 55–64 of: ESCA-NATO Tutorial and Research Workshop on Robust Speech Recognition for unknown communication channel.*

M.H.Savoji. (1989). A robust algorithm for accurate endpointing of speech signals. *Speech Communication*, **8**, 45–60.

M.J.Hunt, & C.Lefebre. (1989). A comparison of several acoustic representations for speech recognition with degraded and undegraded speech. *Pages 262–265 of: Proceedings ICASSP'89.*

M.J.Russell, & R.K.Moore. (1985). Explicit Modelling of State Occupancy in Hidden Markov Models for Automatic Speech Recognition. *Pages 5–8 of: Proceedings ICASSP'85.*

N.B.Yoma, F.McInnes, & M.Jack. (1995). Improved Algorithms for Speech Recognition in Noise Using Lateral Inhibition and SNR Weighting. *Pages 461–464 of: Proceedings Eurospeech'95.*

N.B.Yoma, F.McInnes, & M.Jack. (1996a). Lateral Inhibition Net and weighted matching algorithms for speech recognition in noise. *IEE Proceedings, Vision, Image and Signal Processing*, **143**(5), 324–330.

N.B.Yoma, F.McInnes, & M.Jack. (1996b). Robust speech pulse detection using adaptive noise modelling. *IEE Electronics Letters*, **32**(15), 1350–1352.

N.B.Yoma, F.McInnes, & M.Jack. (1996c). Robust speech pulse detection using adaptive noise modelling and non-stationary measure. *Pages 69–72 of: Proceedings IVTTA'96.*

N.B.Yoma, F.McInnes, & M.Jack. (1996d). Use of a reliability coefficient in noise cancelling by neural net and weighted matching algorithms. *Pages 2297–2300 of: Proceedings ICSLP'96.*

N.B.Yoma, F.McInnes, & M.Jack. (1997a). Spectral Subtraction and Mean Normalization in the context of Weighted Matching Algorithms. *Pages 1411–1414 of: Proceedings Eurospeech'97.*

N.B.Yoma, F.McInnes, & M.Jack. (1997b). Weighted Matching Algorithms and reliability in noise cancelling by spectral subtraction. *Pages 1171–1174 of: Proceedings ICASSP'97.*

N.B.Yoma, F.McInnes, & M.Jack. (1998a). Improving performance of spectral subtraction in speech recognition using a model for additive noise. *Accepted for publication in IEEE Trans. on Speech and Audio Processing.*

N.B.Yoma, F.McInnes, & M.Jack. (1998b). Weighted Viterbi algorithm and state duration modelling for speech recogniton in noise. *In: Proceedings ICASSP'98 (accepted for publication).*

O.Siohan. (1995). On the robustness of linear discrinant analysis as a preprocessing step for noisy speech recognition. *Pages 125–128 of: Proceedings ICASSP'95.*

P.E.Gill, W.Murray, & M.H.Wright. (1981). *Practical Optimization.* Academic Press.

P.Moreno. (1996). *Speech recognition in noisy environments.* PhD Thesis. Dept. of Electrical and Computer Engineering, CMU.

S.E.Levinson. (1986). Continuously Variable Duration Hidden Markov Models for Automatic Speech Recognition. *Computer Speech and Language*, **1**, 29–45.

S.F.Boll. (1979). Suppression of acoustic noise in speech using spectral subtraction. *IEEE Trans. ASSP*, **27**(2), 113–120.

S.Haykin. (1991). *Adaptive Filter Theory*. 2nd edn. Prentice Hall, Englewood Cliff, NJ.

S.Haykin. (1994). *Neural Networks, A Comprehensive Foundation*. Macmillan College Publishing Company.

S.J.Young, N.H.Russell, & J.H.S.Thornton. (1989 July). *Token Passing: a simple conceptual model for connected recognition systems*. Tech. rept. Cambridge University Engineering Department.

S.Kullback, & R.A.Leibler. (1951). On information and sufficiency. *Ann.Math.Stat.*, **22**, 79–86.

S.V.Vaseghi, B.P.Milner, & J.J.Humphries. (1994). Noisy speech recognition using cepstral-time features and spectral-time filters. *Pages II–65/68 of: Proceedings ICASSP'94*.

T.Claes, & Compernolle, D.Van. (1996). SNR-Normalization for robust speech recognition. *Pages 331–334 of: Proceedings ICASSP'96*.

T.Claes, F.Xie, & Compernolle, D.Van. (1996). Spectral estimation and normalization for robust speech recognition. *Pages 1997–2000 of: Proceedings ICSLP'96*.

T.K.Moon. (1996). The Expectation-Maximization algorithm. *IEEE Signal Processing Magazine*, **13**(6), 47–60.

Veth, J.de, & Boves, L. (1996). Comparison of channel normalisation techniques for automatic speech recognition over the phone. *Pages 2332–2336 of: Proceedings ICSLP'96*.

X.D.Huang, Y.Ariki, & M.A.Jack. (1990). *Hidden Markov Models for Speech Recognition*. Edinburgh University Press.

Y.Ephraim, & L.R.Rabiner. (1990). On the relations between modeling approaches for speech recognition. *IEEE Trans. Information Theory*, **36**(2), 372–380.

Y.Linde, A.Buzo, & R.M.Gray. (1980). An Algorithm for vector quantizer design. *IEEE Trans. Commun.*, **COM-28**(6), 84–95.