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

$$LI(E_i) = E_i + f(E_1, E_2, E_3, \ldots, E_N) \quad (1)$$

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.

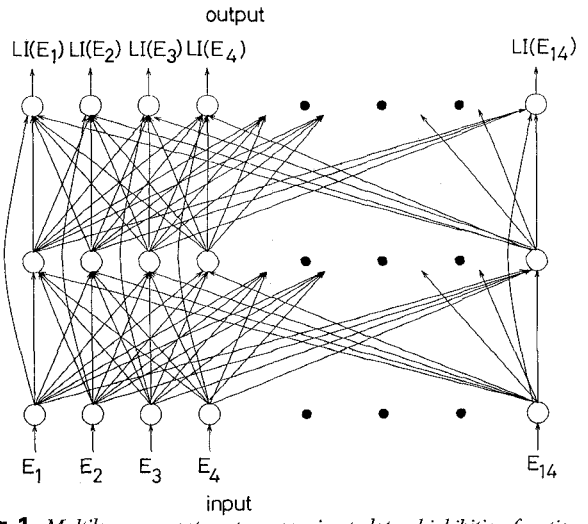

Fig. 1 *Multilayer perceptron to approximate lateral inhibition function set by eqn. 1*

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.

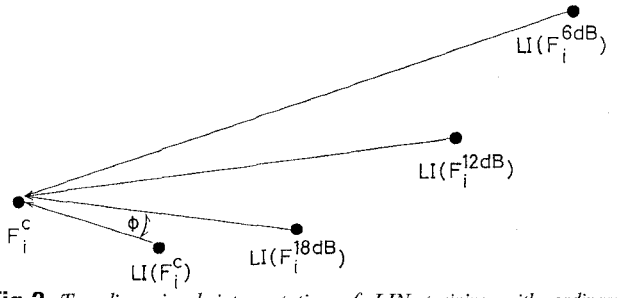

Fig. 2 *Two-dimensional interpretation of LIN training with ordinary back-propagation algorithm*
Reference is constant and equal to the clean frame

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.

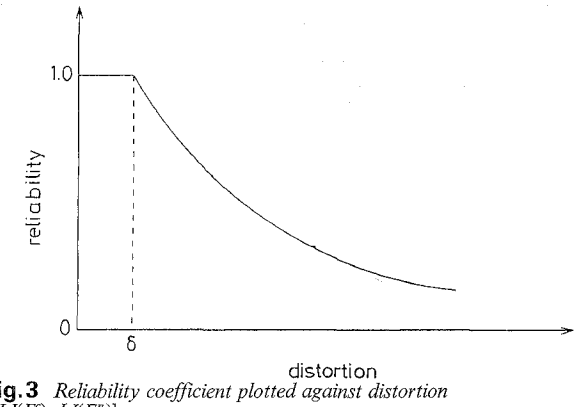

Fig. 3 *Reliability coefficient plotted against distortion* $d[LI(F_i^c), LI(F_i^n)]$

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.

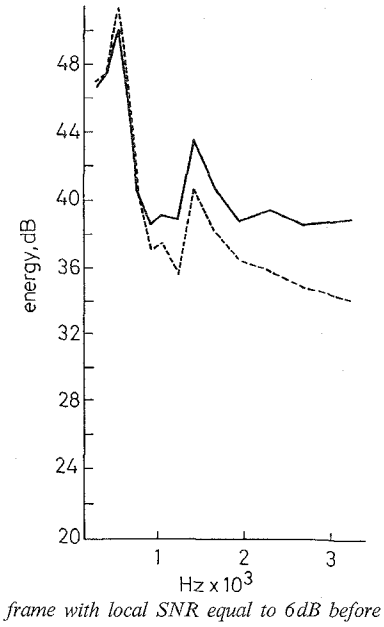

Fig. 4 *Noisy frame with local SNR equal to 6dB before and after LIN processing*
Frame corresponds to vowel ε
Highest component tends to be preserved and position of second format does not change
—— LIN input spectrum (SNR = 6dB)
..... LIN output spectrum

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.

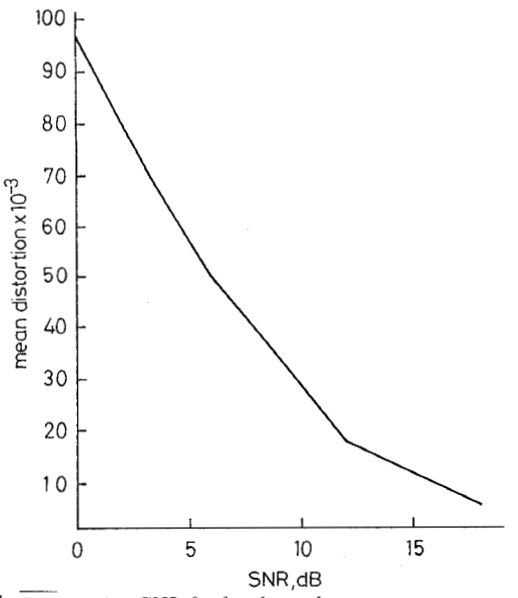

Fig.5 $\overline{D_{snr}}$ *against SNR for female speaker*

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.

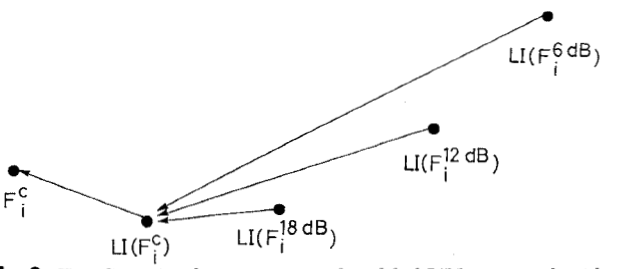

Fig.6 *Two-dimensional interpretation of modified LIN training algorithm (MLT)*

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 $\mathrm{Max}_i[\delta_{t-1}(i) \times a_{ij}] = \mathrm{Max}_i[\mathrm{Max}_h[\delta_{t-2}(h) \times a_{hi}] \times [b_i(O_{t-1})]^{w(t-1)} \times a_{ij}]$ 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