Internal Use Only（非公開）

TR－SLT－0019

Weighted Sub-band MFCCs for speech recognition on the AURORA2 database

重み付きサブバンドMFCCを用いた音声認識

Benoit Verpeaux
ベノア ヴェルポー

August 30, 2002

In my work I try to develop a new technique for the calculation of the MFCC features. The basic idea is that, very often, the energy of "real-world" noises is not equally distributed along the frequency axis. The technique I present here is based on the division of the signal spectrum into sub-bands and on the weighting of these sub-bands before applying the IDCT. The evaluation on the AURORA2 database shows that the use of this technique, by enhancing the useful parts of the signal and by attenuating the noisy ones, results in an improvement of the recognition rate.

（株）国際電気通信基礎技術研究所
音声言語コミュニケーション研究所
〒619－0288「けいはんな学研都市」光台二丁目２番地２ TEL：0774－95－1301

Advanced Telecommunication Research Institute International
Spoken Language Translation Research Laboratories
2-2-2 Hikaridai "Keihanna Science City" 619-0288,Japan
Telephone:+81-774-95-1301
Fax :+81-774-95-1308

TABLE OF CONTENTS

TABLE OF FIGURES

TABLE OF TABLES

GLOSSARY

A/D conversion: **A**nalogical to **D**igital **conversion.**

ASR: **A**utomatic **S**peech **R**ecognition (**RAP** in French).

ATR: **A**dvanced **T**elecommunications **R**esearch Institute International.

AURORA: Name of a database designed to evaluate the performance of speech recognition algorithms in noisy conditions. This database was prepared as contribution to the ETSI STQ-AURORA DSR working group, which develops standards for distributed speech recognition.

CLIPS: **C**ommunication **L**angagière et **I**nteraction **P**ersonne-**S**ystème (communication through language and human-machine interaction): French laboratory based in Grenoble and working in collaboration with ATR for the "C-Star project phase III" on the research topic of automatic speech translation.

ENSERG: **E**cole **N**ationale **S**upèrieure d'**E**léctronique et de **R**adioéléctricité de **G**renoble (National Engineering School of Electronics and Radioelectricity of Grenoble): this is the school in which I followed studies for three years in order to become engineer.

HMM: **H**idden **M**arkov **M**odel: one of the main techniques used in ASR today.

HTK: **H**idden Markov model **T**ool**K**it: software developed by Cambridge University in order to build and manipulate HMMs for speech recognition procedures (see Appendix 1).

ICASSP: **I**nternational **C**onference on **A**coustics, **S**peech and **S**ignal **P**rocessing: this is a major conference on speech processing. It is held every year and groups researchers from all over the world.

ICSLP: **I**nternational **C**onference on **S**poken **L**anguage **P**rocessing: it is another major speech processing conference where many researchers present their state of the art achievements and work.

IDCT: **I**nverse **D**iscrete **C**osine **T**ransform: this is a mathematical transformation very useful in speech processing and particularly in ASR.

MFCC: **M**el-scale **F**requency **C**epstrum **C**oefficients: coefficients used in speech feature extraction (see Appendix 2).

SLT: **S**poken **L**anguage **T**ranslation Research Laboratories: one of the four main departments of ATR (the one in which I did my internship).

SNR: **S**ignal to **N**oise **R**atio: this ratio measures the importance of the noise compared to the interesting signal (RSB in French).

ACKNOWLEDGEMENT

I would like to thank **Dr. Satoshi Nakamura** (ATR-SLT dept1 head) and **Dr. Mitsunori Mizumachi** (ATR-SLT Researcher) my supervisors, for their scientific and technical support as well as for their warm welcome in ATR and in Japan. Thank you for helping me discovering Japan.

Thank you to **Yukiko Ishikawa** and **Makiko Tatsumi** (SHIEN group ATR) for their administrative help and everyday life support, their kindness and their friendliness.

Thank you to all the members of **ATR-SLT** for their warm welcome and especially to the members of the **Planning Section** and to **Megumi Kurita** who were always available to help.

Thank you to **Laurent Besacier** (CLIPS-France “Maître de conference”) for his support from France.

Thank you to **Marion, Yves, Hélèna, Olivier, Cyril and Thomas** for making my stay in Japan so pleasant.

I also would like to thank **Dr. Kaisheng Yao** (ex ATR-SLT Researcher) and **Dr. Tomoko Matsui** (ATR-SLT Senior Researcher) for their knowledge and their help.

This research work was supported in part by a contract with the Telecommunications Advancement Organization of Japan entitled, “A Study of speech dialogue translation technology based on a large corpus”.

INTRODUCTION

Lately, research in the field of speech recognition has greatly evolved. Indeed, systems achieving greater and greater recognition accuracies are being developed. Some applications are starting to pop up through great public products such as cellular phones, human-computer interfaces.... Nevertheless, there remains a great problem that still requires the help of research: robustness. Actually, it is true that many systems show good recognition accuracies, nevertheless these good performances are achieved in laboratory conditions: clean data, no surrounding noise and stable environment. The problem is that for everyday applications such conditions are never gathered. Indeed, there is always some background noise (public places, people speaking behind the main speaker, reverberation of the speaker's voice...). Moreover, in spontaneous speech conditions, the speakers hesitate and may start over their sentences and use spontaneous speech expressions such as: "umh", "euh".... Of course, the system must not consider this as useful speech information.

The main issue in speech recognition today is therefore robustness. A few main guidelines direct the researchers' work all over the world and research considers many possibilities to improve robustness. Two main directions are being followed: either through recognition itself by noise adaptation and compensation or as post-processing methods focusing on a way to (re)estimate the reliability of the results by using more sophisticated language models or confidence measures. Only the first aspect has been explored through the research work I have made. Indeed, my work focused on speech processing and noise compensation.

My work presents a new multi-band approach for robust feature extraction. Multi-band processing has been widely used for speech recognition [1, 2]. These techniques try to extract useful information from different frequency sub-band and then combine them in order to produce useful information for the recognizer. A problem with these techniques is the merging strategy used for collecting all the information coming from the different sub-bands. Indeed, this merging is not straightforward and various strategies can be applied to this step. The approach presented here tries to avoid this step. This is achieved by enhancing some of the sub-bands, which contain useful information, and by attenuating others, which mainly contain noise. The aim of this internship was to find a method to automatically determine how this process has to be done.

This document will consider both my methods and my achievements through the description of my work, the experiments I conducted and the analysis of the results I got.

STARTING IDEA

When you look at the long-term spectra of the noises used in the AURORA2 database (see Appendix 3), you can see that some noises have properties very far from those of a white noise. For example, the restaurant noise shows energy in the 0-1kHz region and much less in the remaining spectral regions. Of course some noises of the database do not have such a spectral pattern (like the suburban train noise for example) but nevertheless, the majority of them does. Since this database is built in order to match real world noise conditions, the starting idea is to take advantage of this dis-symetry of the noise spectral pattern by enhancing spectral regions which are less affected by noise and by attenuating regions which are much more corrupted by noise (like the 0-1kHz region in the case of the restaurant noise).

I can already forecast that if this method is efficient, it will work correctly for noises that show strong differences of energy values along the frequency axis and that the results might not be as good for noises with "whiter" properties, that is to say, for noises with more equally distributed energy along the frequency axis.

METHOD

The approach used to develop the previous idea is described in *Figure 3.* The first part of the process is conventional. At first, the speech signal is pre-emphasized, then it is windowed with a Hamming window of 200 samples. After that a FFT algorithm is applied to these samples and only the magnitude of the spectrum is kept. Finally, the log of the 24 Mel-filterbank coefficients is computed.

From this point, my approach is different from the conventional one. Instead of applying the IDCT to the 24 coefficients in order to get the MFCCs (Mel Frequency Cepstral Coefficients, for more information on the meaning and calculation of these coefficients please refer to Appendix 2), the observation vector O is divided into two parts. The first one, O_L, consists of 24 coefficients. The first 12 are equal to the first 12 coefficient of O and the last twelve are equal to 0. The second one, $O_{H,}$ is composed of 12 nul coefficients (the first 12) and of the last 12 coefficients of O. As a consequence, the sum of these two new vectors is equal to O.

After that, the two vectors are weighted differently, the weights being chosen so that their sum is equal to 2. The reason for this choice is that the initial vector can be regarded as a result of this weighting process were the two weights are equal to 1 and so their sum equal to two.

After this weighting step, O_L and O_H are added to each other, the IDCT transfrom is applied to this sum in order to get coefficients kind of like MFCCs.

It is interesting to notice that this process is linear and so is the derivation. Therefore, the delta and acceleration of the MFCCs obtained with this method will also be affected by this weighting.

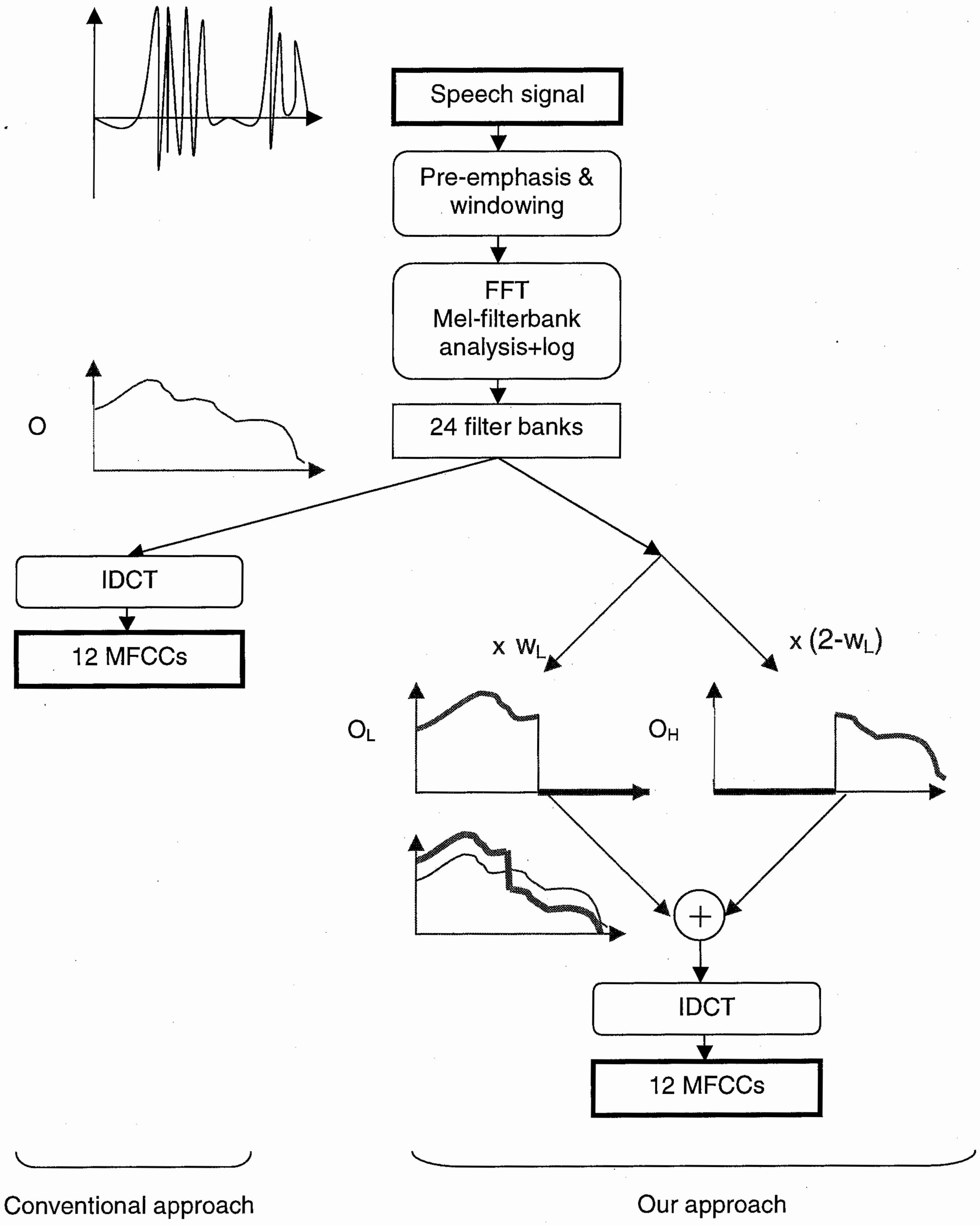

Figure 1: Two feature extraction approaches.

SUPERVISED EXPERIMENTS

The first experiments I conducted corresponded to the case where the sub-band coefficient, w_L, was set “manually” to certain values in order to test the efficiency of the method. The aim of such experiments was first, to check whether this method was useful or not and second, to get an idea of the limits of this technique.

Experimental setup

I used the AURORA2 database for both training and testing. The training process is described in Appendix 3. The acoustic features used for this training are conventional MFCCs. That is to say that no weighting process is applied in order to get those coefficients. The first 12 cepstral coefficients are used along with the 0th order coefficient. The delta and acceleration of those 13 coefficients are also extracted from the speech signal (see *Figure 4*).

<table>
<tr><td>C_1</td><td>C_2</td><td>. . .</td><td>C_{12}</td><td>C_0</td><td>D_1</td><td>D_2</td><td>. . .</td><td>D_{12}</td><td>D_0</td><td>A_1</td><td>A_2</td><td>. . .</td><td>A_{12}</td><td>A_0</td></tr>
<tr><td colspan="5">MFCC</td><td colspan="5">DELTA</td><td colspan="5">ACCELERATION</td></tr>
</table>

Figure 2: 39-dimension acoustic vector.

The testing data used is the data refered to as “test A” and “test B” in the AURORA2 database. It includes 8 different noise conditions added to the speech signal with 7 different SNRs (going from CLEAN to –5dB, see Appendix 3). All together, this corresponds to 56 conditions. Acoustic vectors for the 1001 utterances for each of the 56 conditions are extracted using my weighting method. The sub-band coefficient used for this step is set “manually” to different values which are: 0, 0.4, 0.7, 0.8, 0.89, 0.9, 0.91, 0.92, 0.93, 0.94, 0.95, 0.96, 0.97, 0.98, 0.99, 1 (which is the conventional value), 1.01, 1.02, 1.03, 1.04, 1.05, 1.1, 1.2, 1.3, 1.6, and 2. This makes a total of 26 different values.

After that, the 26 sets of 56 conditions are combined with the HMM model previously trained in order to perform speech recognition. This step is accomplished using HTK tools.

Finaly, the sub-band coefficient that gives the best recognition rate for each of the 56 noise conditions is recorded. *Table 5* shows the relative improvement in word accuracy using this method compared to the conventional one. The number shown is the difference beetween the average recognition rate for all SNRs in the conventional case and in the case corresponding to the new method.

TEST	CONDITION		AVERAGE	OVERALL
A	SUBWAY	0.80	5.17	7.02
	BABBLE	11.23		
	CAR	5.21		
	EXHIBITION	3.43		
B	RESTAURANT	12.95	8.87	
	STREET	3.95		
	AIRPORT	12.14		
	STATION	6.42		

Table 1: Relative word accuracy improvement (in %) for the supervised mode.

Figure 5 shows two different choices for w_L as a function of the type of noise (suburban train or restaurant) and of SNR.

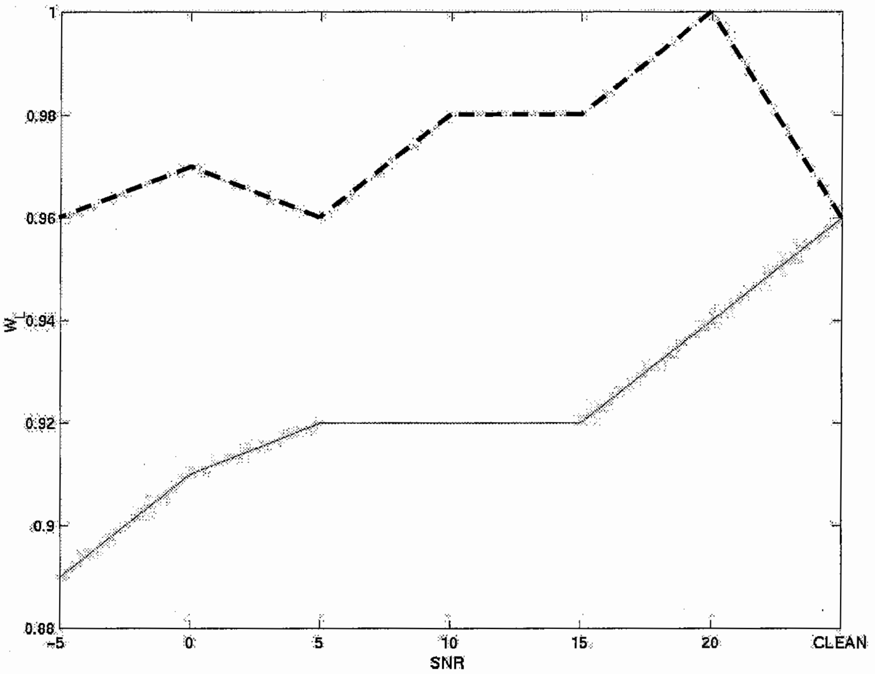

Figure 3: Choice of w_L for restaurant noise (solid line) and suburban train noise (dashed line).

Result analysis

The experiments give 3 types of information. The first one is that my method is useful since I get an overall improvement of roughly 7% in word accuracy. The second is that, as forecasted, the efficiency of this method really depends on the spectral pattern of the noise added to the signal. Indeed, a closer look at the results of *Table 5* shows that the improvement for the restaurant noise almost reaches 13% whereas the one for the suburban train noise is only of 0.8%. The third thing is that the experiments give a reference for the next step of my work, which is the w_L automatic estimation. Indeed, by knowing the value of w_L that gives the best results, I will now be able to get an idea of the efficiency of the estimation methods before conducting any speech recognition experiments.

AUTOMATIC SUB-BAND COEFFICIENT ESTIMATION

During this internship, I have mainly developed two methods concerning automatical estimation of the value of w_L. The first one is based on the use of noise estimation within each utterance. As a consequence, this one only uses noise information. This technique is called *equalization evaluation* because it has a link with usual equalization, as I will explain later. The second one uses a set of adaptation data selected from the testing data. w_L is then chosen to best fit a criterion relative to this set of data. This method will be referred to as *weight adaptation*.

Equalization evaluation

Starting idea

The starting point for this technique is very simple. The data used in the training process is never completely "clean". Even if the SNR is very important, there is always a bit of white noise in the signal, due to A/D conversion for example. The idea is that the weighting process might try to transform the noise added to the signal for it to become as close as possible to this white noise and thereby get closer to the performance achieved using clean data.

Implementation

I tried to use this idea to evaluate the sub-band coefficient. The implementation process is described in *Figure 6*. The first step is noise estimation. This is achieved by keeping only the 10 first frames of each utterance (i.e. about 100ms of speech signal) where it is assumed that there is no speech. Then the spectrum of the "signal" noise is compared to the one of a white noise and w_L is chosen in order to minimize the quadratic distance between the two spectra according to *Equation 1*. This is why this technique is called equalization.

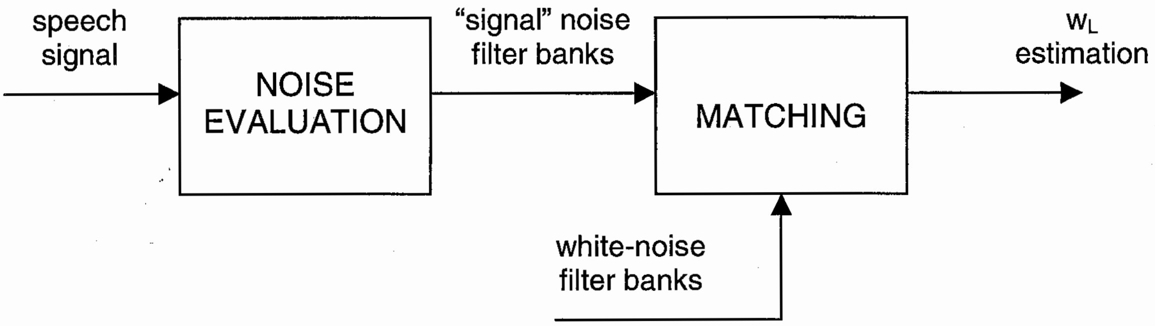

Figure 4: The equalization evaluation process.

$$w_L = \frac{\sum_{i=1}^{24} f_i n_i + 2 f_{i+12}^2 - f_{i+12} b_{i+12}}{\sum_{i=1}^{24} f_i^2} \qquad \textit{Equation 1}$$

Where f_i are the filter banks of the noise included in the speech signal and n_i, the filter banks of a white noise.

Experimental results

The experimental setup is almost the same as the one used for the supervised experiments. The only difference is that for the supervised experiments, the chosen sub-band coefficient was the same for all the 1001 utterances of a noise condition (i.e. one noise type and one SNR). In the following experiments, the sub-band is calculated for each utterance. The results are shown in *Table 6.*

TEST	CONDITION	SUPERVISED	EQUALIZATION	OVERALL
A	SUBWAY	0.8	-0.61	6.51
	BABBLE	11.23	8.48	
	CAR	5.21	7.03	
	EXHIBITION	3.43	2.49	
B	RESTAURANT	12.95	10.1	
	STREET	3.95	5.01	
	AIRPORT	12.14	11.88	
	STATION	6.42	7.73	

Table 2: Relative improvement (in %) in word accuracy: supervised mode and equalization evaluation.

These experiments show that, even if the overall improvement is not as important as the one obtained for the supervised mode, it is just 0.5% under which is not so bad. This method even achieves better results for three noise conditions (car, street and station). This is due to the fact that the evaluation method calculates a specific w_L for each utterance, which is, of course, more efficient than to use the same w_L for a given set of utterances.

Training noise equalization

In the previous experiments the noise pattern to be matched was the pattern of a white noise. But if a closer look is taken at the noise included in the training data, we can see that the noise of the signal is not exactly a white noise. Therefore, trying to match the real noise pattern in the previous processing might be a good idea. Unfortunately, the results I get by using it are really bad. Another idea might be to use a mixing between this noise and a white noise. This is what I tried to do and the results from these experiments are shown in *Table 7.*

TEST	CONDITION	SUPERVISED	EQUALIZATION(2)	OVERALL
A	SUBWAY	0.8	0.38	6.34
	BABBLE	11.23	10.46	
	CAR	5.21	4.5	
	EXHIBITION	3.43	2.92	
B	RESTAURANT	12.95	11.93	
	STREET	3.95	3.8	
	AIRPORT	12.14	11.01	
	STATION	6.42	5.71	

Table 3: Relative improvement (in %) in word accuracy: supervised mode and equalization evaluation(2).

The results obtained by using this noise pattern are quite close to the previous ones, 6,34% overall improvement (compared to 6.51%). Nevertheless, with this method, the results are always under those for the supervised mode results.

Limits of equalization

The results presented above show that this method achieves better performances than the baseline system. But they also show that the overall performance is under that for the supervised mode. So, even if w_L is calculated for each utterance.

This might represent a limit of this method. The problem is that I have only presented two ways of using equalization for the sub-band coefficient evaluation. Actually, I have tried many other noise patterns that achieved roughly the same performances. This shows that the method might not be the best one for evaluating w_L. This is due to two linked reasons, which are:

- it seems that, for the experiments, there is no “best” noise pattern to be matched.
- there might be a serious drawback to this method. Indeed, it only uses the noise information included in the speech data. It seems quite reasonable to think that w_L depends on both the speech signal and the noise information.

These reasons have encouraged me to try to develop another method for the sub-band coefficient evaluation in order to combine both speech and noise information. This leads to the second method I will introduce: weight adaptation.

Weight adaptation

This method uses a set of adaptation data. This data is composed of 50 utterances randomly chosen among the 1001 utterances composing the testing conditions in the AURORA2 database.

The different steps of this method are shown in *Figure 7*. I applied the same process for all adaptation utterances. The first step is to determine the state sequence corresponding to the utterance. This is done using a tool from HTK and I used forced alignment to get this state sequence. That means that, for every utterance, the words pronounced are already known. In a “real world” application this would imply that the system asked the user to say some precise sentences in order for it to adapt itself to the new environment. I also need O^L and O^H, which have been defined before. In fact, the O^L and O^H used here are not exactly the same has the ones that were previously defined. Indeed, I apply the IDCT to the previously defined O^L and O^H in order to get the “new” O^L and O^H. As a consequence, O now denotes an observation vector in the cepstral domain instead of an observation vector in the Mel-frequency domain.

The next step is to calculate the likelihood of the frame given the corresponding state of the trained model. This likelihood is then summed over all the frames of all the adaptation utterances. This finally gives a global likelihood that is a function of w_L. This function is given in *Equation 2*. The i index refers to the number of utterances in the adaptation utterances altogether. So, if there are only two adaptation utterances, and if they have 100 and 120 frames each, the i index would go from 1 to 220. The order in which the frames are used is not important. The j index refers to the mixture number used to model a given state (each state is modelled by a mixture of gaussians). In the AURORA2 database, a mixture of 3 gaussians models the words. There are a bit more for the “silence” model. m_{ij} denotes the different mixture coefficients of a given state.

$$f(w_L)=\sum_{i=1}^{N}\sum_{j=1}^{M}\log\left(\frac{m_{ij}}{\sqrt{(2\pi)^{39}\left|\Sigma_{ij}\right|}}\right)-\frac{(w_L o_i^L+(2-w_L)o_i^H-\mu_{ijk})\Sigma_{ij}^{-1}(w_L o_i^L+(2-w_L)o_i^H-\mu_{ijk})^T}{2}$$

Equation 2

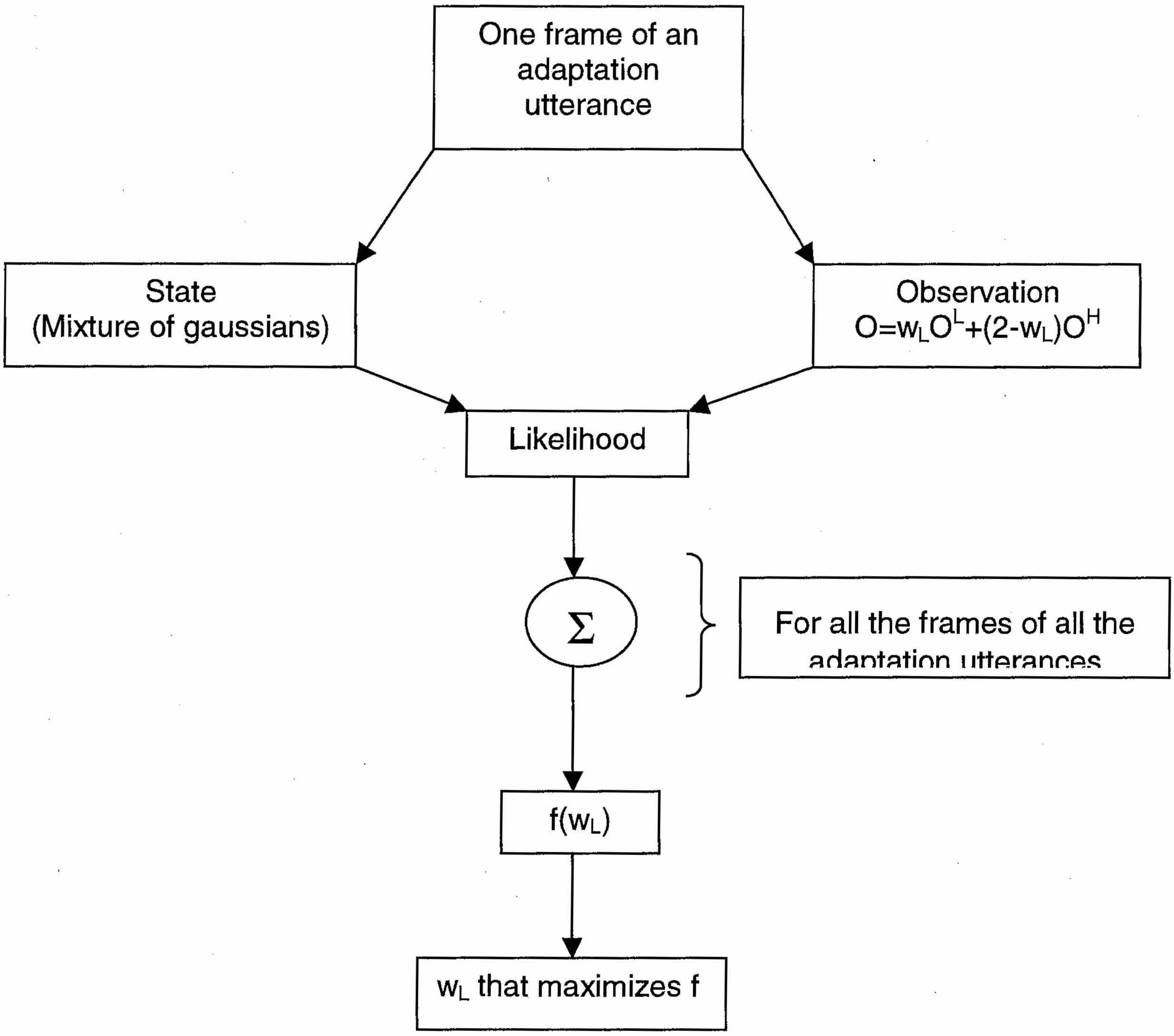
One frame of an adaptation utterance
State (Mixture of gaussians)
Observation $O=w_L O^L+(2-w_L)O^H$
Likelihood
Σ
For all the frames of all the adaptation utterances
$f(w_L)$
w_L that maximizes f

Figure 5: The weight adaptation process.

Experimental results

I then tried to estimate the performance of this evaluation technique. The experiments I conducted were only simulated experiments. First, w_L was calculated for the 56 noise conditions (8 types of noise at 7 different SNRs). Then, since this technique implies that w_L should be the same for each utterance of a given noise condition (which is not the same as for the previous evaluation technique), they are the same as the experiments conducted in the supervised mode. The only difference is that w_L is automatically calculated. As a consequence, I can use the results obtained in the supervised mode. Indeed, those previous experiments enable me to have the relationship between w_L and the word accuracy for all the noise conditions. The problem is that in the supervised mode, I didn't try all the possible values of w_L (of course I didn't!). So the "simulated" results obtained using a w_L equal to 0.96 will be slightly different from what they would really be if a w_L equal to 0.957 was used nevertheless they will be accurate enough to give a good idea of the performance of this technique. The results from these experiments are shown in *Table 8.*

TEST	CONDITION	SUPERVISED	ADAPTATION	OVERALL
A	SUBWAY	0.8	0.16	5.71
	BABBLE	11.23	8.63	
	CAR	5.21	4.88	
	EXHIBITION	3.43	1.3	
B	RESTAURANT	12.95	11.19	
	STREET	3.95	2.58	
	AIRPORT	12.14	11.39	
	STATION	6.42	5.52	

Table 4: Relative improvement (in %) in word accuracy: supervised mode and weight adaptation.

The results are quite good since the improvement is not so far from the one obtained for the supervised mode. Nevertheless, they are not as good as the results of the estimation method. This allows us to stress a drawback of this method: w_L is the same for a given noise condition. Therefore, it is a kind of averaged value for this condition no matter what it should be for a specific utterance with slightly different noise conditions.

FURTHER IMPROVEMENTS

There are at least three possible improvements for my method:

- concerning the w_L evaluation method, it might be a good idea to try to combine the two approaches I have described here. The weight adaptation could be applied in order to get a first estimation of w_L and then equalization could be used in order to take into account noise information that are more specific of to the utterance and to get the final estimation of w_L
- more generally, my technique only uses two sub-bands. Some noises like the suburban train noise of the AURORA2 database have almost the same energy in low and in high frequencies. Therefore, my technique is not so useful for that kind of noise. So, it could be interesting to use more sub-bands in order to deal with more noises and also to improve the performance.
- finally, the two sub-bands I use have the same number of filter banks (12). But it might be interesting to use a non-symmetric division of the whole band.

These are three points that would require more time to be investigated, which unfortunately I hadn't.

GLOBAL CONCLUSION: OVERALL REVIEW

This internship was my first real research project. Moreover, it was also my first experience having a concrete work to conduct during quite a long period (five months). Those two aspects were very interesting since they really enabled me to make use of all I have been learning for the past three years.

This internship also enabled me to work in ATR, which was a very good experience for me. I learned how people from all over the world could gather and work on great research projects putting together not only their capacities and ideas but also the ways of working that they had been taught in their home countries. I learned how to work with Japanese, Chinese, American, Australians, Europeans ....

Working in Japan was also full of surprises. I saw how one of the top countries in state of the art technologies manages to combine its own knowledge with methods and ideas coming from various countries and how it took profit from everything that fell upon its hands.

So as a conclusion I would say that this internship was very self-rewarding both for personal and professional aspects.

REFERENCES

PUBLICATIONS

[1] Okawa (S.), Bocchieri (E.) and Potamianos (A.)
Multi-band speech recognition in noisy environments
Proc. of ICASSP 1998, pp. 641-644,Seattle (USA).

[2] Bourlard (H.) and Dupont (S.)
A New ASR approach based on independent processing and recombination of partial frequency bands
Proc. of ICSLP 1996, Philadelphia (USA).

[3] Hirsch (H. –G.) and Pearce (D.)
The Aurora experimental framework for the performance evaluation of speech recognition systems under noisy conditions
Proc. of ICSLP 2000, vol. 4 pp.29-32, Beijing (China).

BOOKS

[4] Rabiner (L.) and Juang (B. –H.)
Fundamentals of speech recognition
Prentice Hall International Editions, 1993
Chap. 6: Theory and implementation of hidden Markov models (pp. 321-389).

[5] Young (S.) and al.
The HTK Book
Revised First Edition, Dec. 2001.

APPENDIX

APPENDIX 1: HTK Hidden Markov model ToolKit

The information given here was extracted from the following web page: http://htk.eng.cam.ac.uk/.

This software was originally designed by the Speech vision and Robotics Group of the Cambridge University Enginneering department (CUED) in 1989 where it was used to build CUED's large vocabulary speech recognition systems. Then, the software was developed by Entropic Research Laboratory Inc. from 1993 and was licensed by Microsoft in 1999.

HTK is a portable toolkit for building and manipulating hidden Markov models. It consists of a set of library modules and tools available in C source form. The tools provide sophisticated facilities for speech analysis, HMM training, testing and result analysis. The software supports HMMs using both continuous density mixture gaussians and discrete distributions and can be used to build complex HMM systems. The HTK release contains extensive documentation and examples.

The HTK book [5] is very helpful to learn how to use HTK, it also gives information on how the software actually works and on the theoretical background concerning HMMs.

APPENDIX 2:MEL-Frequency Cepstral Coefficients (MFCC) [4, 5]

The MFCCs are one kind of parameters used in ASR to describe the speech signal. In order to calculate these coefficients, the procedure shown in *Figure 12* is applied to the speech signal.

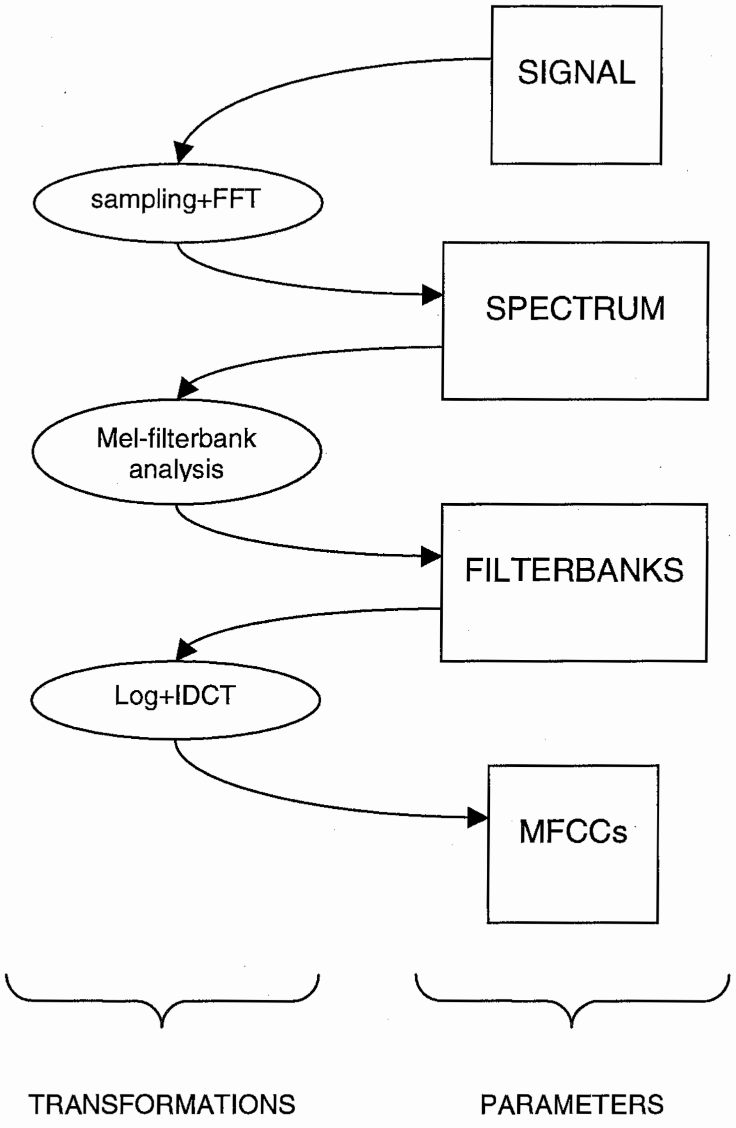

Figure 6: Procedure applied to the speech signal to calculate the MFCCs.

Some of the transformations, such as FFT and sampling, are quite traditional in signal processing but the Mel-filterbank analysis and the role of the log+inverse discrete cosine transform are more specific to speech processing. Therefore, their role will be detailed a little more in the next section.

Cepstrum analysis

A very widely spread model for human speech production is that the speech is the result of the filtering of a periodic signal, produced by the glottis, by the vocal tract (see *Figure 13*), the signals on *Figure 13* are shown in the frequency domain.

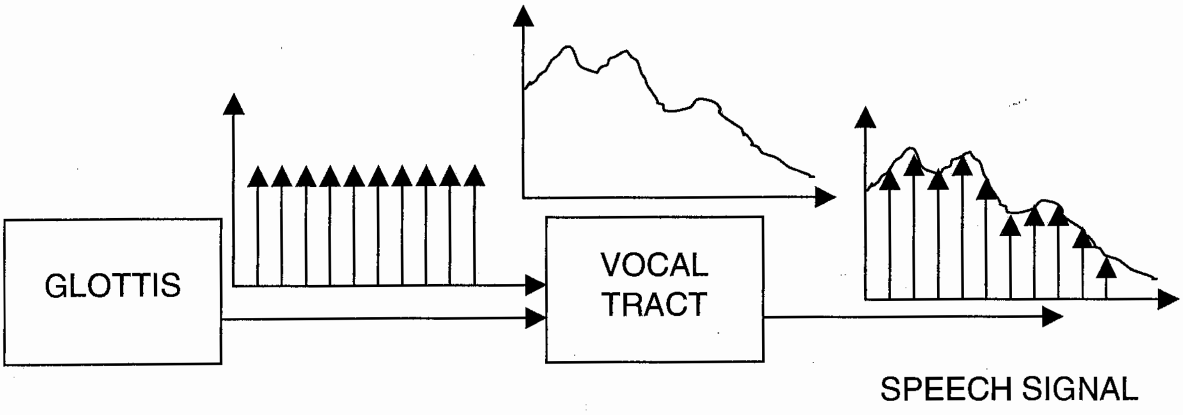

Figure 7: The speech production process.

In ASR, the information from the vocal tract is more useful than the information from the glottis. Therefore, it would be interesting to separate these two parts of the signal. This is done by using the log and the IDCT transform.

The first step is to take the log of the speech spectrum in order to transform the speech signal into a sum of two terms thanks to the following equation:

$$S^{l} = \log(S) = \log(S_{glottis}) + \log(S_{tract})$$

The second step is to apply the IDCT to S^{l} in order to go back to a kind of temporal domain, which is called cepstral domain (or quefrency domain). In this domain, $S^{l}_{glottis}$ and S^{l}_{tract} are located in two different regions on the quefrency axis. The information concerning the vocal tract can very easily be selected by keeping the low quefrency coefficients. These coefficients are called cepstrum coefficients. *Figure14* shows an example of this processing.

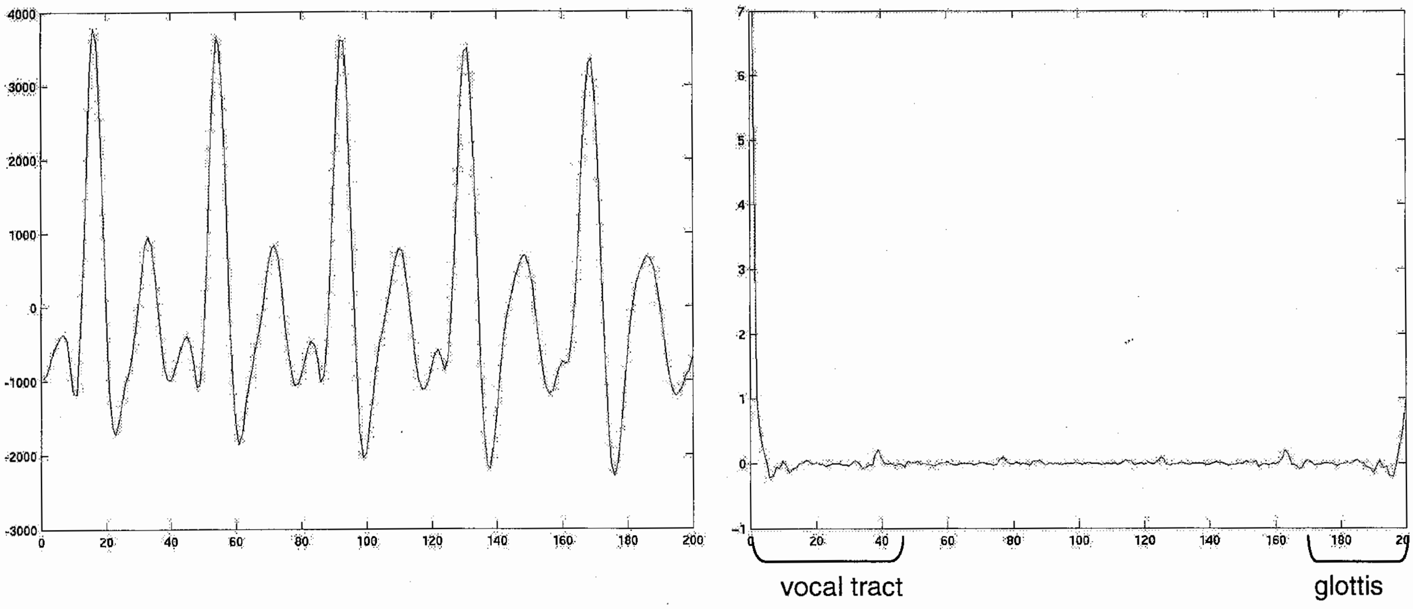

Figure 8: Extraction of the cepstrum coefficients.

Mel-filterbank analysis [3]

The human ear resolves frequencies non-linearly across the audio spectrum and empirical evidence suggests that designing a front-end to operate in a similar non-linear manner improves recognition performance. A good way to achieve this is through the use of a filter bank analysis. The problem is to choose the spacing of filters on the frequency axis. Perceptual studies showed that one possibility is to space the filters along the critical band in order to choose bands that give equal contribution to speech articulation. The MEL scale is a variant on the critical band scale and is defined by:

$$Mel(f) = 2595 \log_{10}(1 + \frac{f}{700})$$

In order to implement this filter bank, the speech is transformed using a Fourier transform and the magnitude is extracted. The magnitude coefficients are then binned by correlating them with triangular filters equally spaced on the Mel-scale. Here binning means that each FFT magnitude coefficient is multiplied by the corresponding filter gain and the results are accumulated (see *Figure 15*).

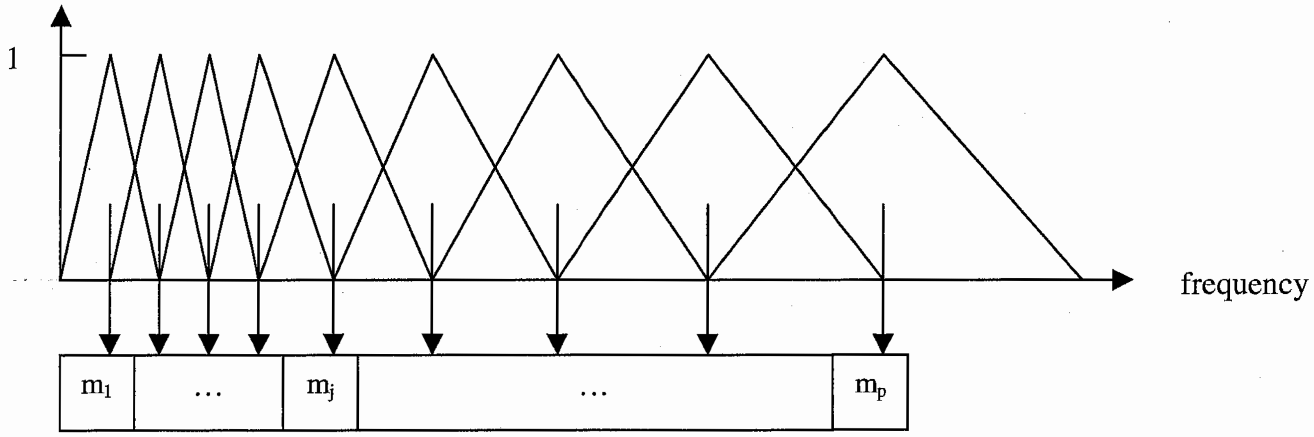

Figure 9: The Mel filter bank analysis.

APPENDIX 3: the AURORA 2 database [8]

This database has been designed in order to evaluate the performances of speech recognition algorithms in noisy conditions. It consists of a set of utterances and of selections of “real-world” noises added to the speech over a range of signal to noise ratios.

Speech data

The “TIDigits” database is taken as a basis. This part contains the recordings of male and female US-American adults speaking isolated digits and sequences of up to 7 digits. The original 20kHz data has been down sampled to 8 kHz with an “ideal” 4 kHz low-pass filter.

Noise data

Eight different noise signals were used to simulate a “real world” environment. They were chosen in order to represent the most probable application scenarios for telecommunication terminals. They were recorded at different places:

-suburban train,
-crowd of people,
-car,
-exhibition hall,
-restaurant,
-street,
-airport,
-train station.

Figure 16 shows the long-term spectra of some of these signals.

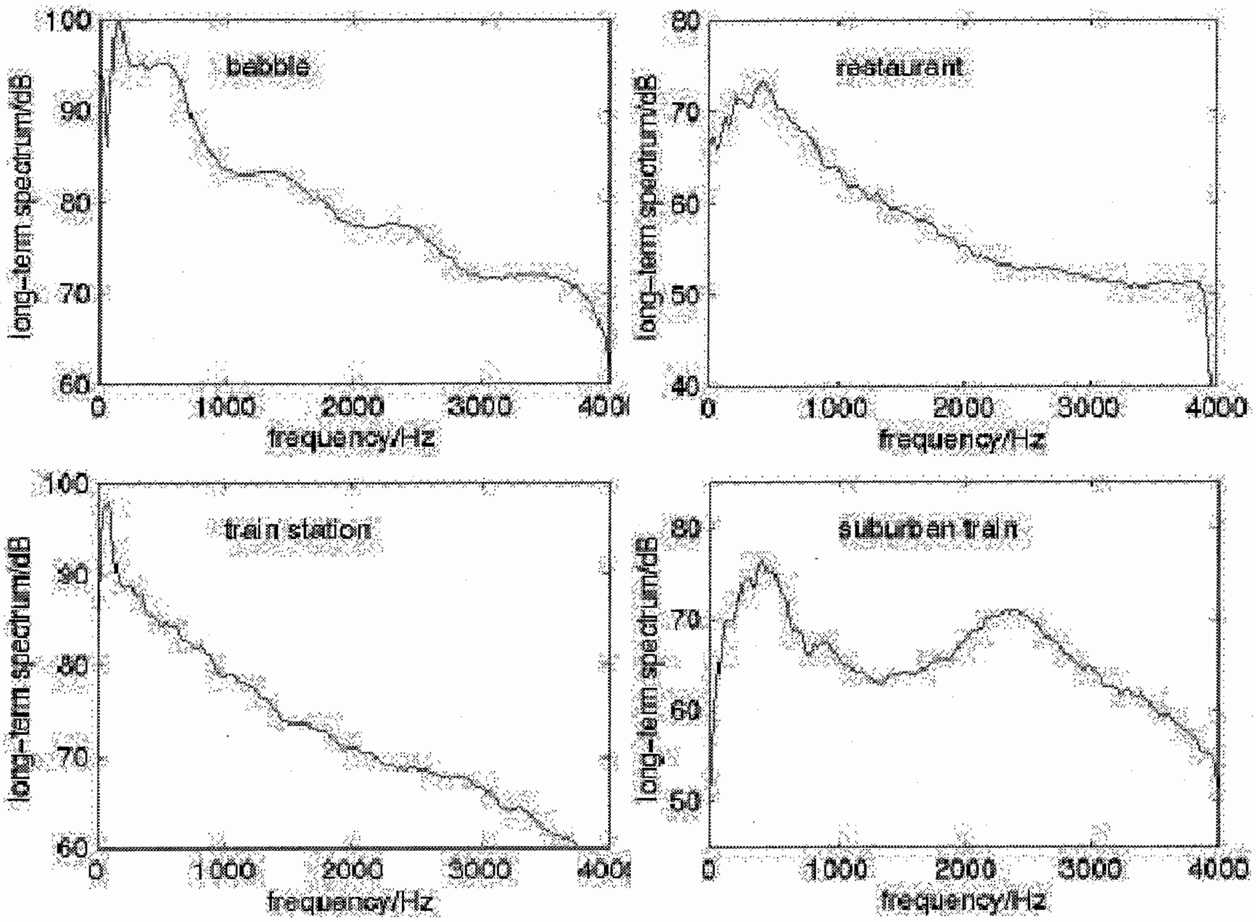

Figure 10: Long term spectra of some of the noises used in the AURORA2 database.

Finally, these noises are added to the "clean data" at six different SNRs: 20dB, 15dB, 10dB, 5dB, 0dB and –5dB.

TRAINING DATA

This database offers HMM model-training conditions but we have only used the "clean training" set. This set is composed of 8440 utterances selected from the training part of the TIDigits containing the recordings of 110 adults (55 male and 55 female). Those data are clean data used for training any kind of speech recognition algorithm.

TESTING DATA

For the testing data, this database offers 3 sets but we have only used the sets A and B for our experiments. Each set is build the same way. 4004 utterances from 52 male and 52 female speakers in the TIDigits test part are split into 4 subsets each containing 1001 utterances. Recordings of all speakers are present in each subset. A noise signal is added to each subset of 1001 utterances at six different SNRs: 20dB, 15dB, 10dB, 5dB, 0dB, -5dB. Furthermore the clean case obtained without adding noise is taken as a seventh condition. Set A and set B only differ by the noise which are added.

HMM MODEL AND MODEL TRAINING

The reference recognizer is based on the HTK software package version 2.2 from Entropic. The digits are modeled as whole word HMMs with the following parameters:

- 16 states per word
- simple left-to-right models without skip-state transitions
- mixture of 3 gaussians per state
- only mean and variance of acoustic coefficients are used

This model is then trained by applying the Baum-Welch re-estimation scheme contained in HTK tool HERest. The data used for this training are the training data set described above.