# Speaker Normalisation for Large Vocabulary Multiparty Conversational Speech Recognition

*Giulia Garau*
*Centre for Speech Technology Research*
*University of Edinburgh*
*Edinburgh EH8 9AB, UK*


Doctor of Philosophy
Centre for Speech Technology Research
School of Informatics
University of Edinburgh
2009

# Abstract

One of the main problems faced by automatic speech recognition is the variability of the testing conditions. This is due both to the acoustic conditions (different transmission channels, recording devices, noises etc.) and to the variability of speech across different speakers (i.e. due to different accents, coarticulation of phonemes and different vocal tract characteristics). Vocal tract length normalisation (VTLN) aims at normalising the acoustic signal, making it independent from the vocal tract length. This is done by a speaker specific warping of the frequency axis parameterised through a warping factor. In this thesis the application of VTLN to multiparty conversational speech was investigated focusing on the meeting domain. This is a challenging task showing a great variability of the speech acoustics both across different speakers and across time for a given speaker. VTL, the distance between the lips and the glottis, varies over time. We observed that the warping factors estimated using Maximum Likelihood seem to be context dependent: appearing to be influenced by the current conversational partner and being correlated with the behaviour of formant positions and the pitch. This is because VTL also influences the frequency of vibration of the vocal cords and thus the pitch. In this thesis we also investigated pitch-adaptive acoustic features with the goal of further improving the speaker normalisation provided by VTLN.

We explored the use of acoustic features obtained using a pitch-adaptive analysis in combination with conventional features such as Mel frequency cepstral coefficients. These spectral representations were combined both at the acoustic feature level using heteroscedastic linear discriminant analysis (HLDA), and at the system level using ROVER. We evaluated this approach on a challenging large vocabulary speech recognition task: multiparty meeting transcription. We found that VTLN benefits the most from pitch-adaptive features. Our experiments also suggested that combining conventional and pitch-adaptive acoustic features using HLDA results in a consistent, significant decrease in the word error rate across all the tasks. Combining at the system level using ROVER resulted in a further significant improvement. Further experiments compared the use of pitch adaptive spectral representation with the adoption of a smoothed spectrogram for the extraction of cepstral coefficients. It was found that pitch adaptive spectral analysis, providing a representation which is less affected by pitch artefacts (especially for high pitched speakers), delivers fea-

tures with an improved speaker independence. Furthermore this has also shown to be advantageous when HLDA is applied. The combination of a pitch adaptive spectral representation and VTLN based speaker normalisation in the context of LVCSR for multiparty conversational speech led to more speaker independent acoustic models improving the overall recognition performances.

# Acknowledgements

First and foremost I would like to express my gratitude to my supervisor Prof. Steve Renals for his expert guidance, for his unlimited patience, for always finding the time to meet me, and being always supportive and encouraging. I would also like to thank Dr. Simon King for being my second supervisor and for his advice.

I am also thankful to Dr. Jon Barker, Dr. Ralf Schlüter and Dr. Hiroshi Shimodaira for accepting to be in my examination panel.

A special thanks goes to the AMI–ASR team members, it was really a wonderful experience working with all of you! In particular thanks to: Dr. Thomas Hain for his guidance as a great team leader and for sharing his expertise with us, Ing. Martin Karafiat for his precious help with scripts and tools, Dr. Mike Lincoln, Dr. John Dines, Dr. Darren Moore, Dr. Iain McCowan, Dr. Phil Garner, Dr. Vincent Wan, Dr. Lukas Burget and everybody else who contributed to building the AMI-ASR system.

Thanks to Prof. Hideki Kawahara for providing the STRAIGHT code for spectral analysis.

I want to take the opportunity to recognise the contribution of many CSTR members as well with their precious comments during meetings and talks. Thanks to Melissa, Yolanda, Gregor, Sabrina, Partha, Mike, Ivan, Chao, Ingmar and Jochen for making my life in Edinburgh so enjoyable.

Thanks to Caroline Hastings and Avril Heron for being always so helpful with any question or problem I had (administrative and not) during my stay at CSTR. The members of computing support deserve a particular acknowledgment as well, especially Ross Armstrong, Ian Rae and all the members of the ECDF grid support team.

A big thanks goes also to my friends from Italy, which, despite the lack of telephone calls and emails, are always so present in everyday's life: Fulvia, Marzia, Pietro, Federica, Valeria, Nicola, Francesco, Daniela, Valentina, Annachiara and Ivana.

I am particularly thankful to my family: my mum Daniela and my dad Luigi, Alberto, Federica, Andrea and my beautiful little nieces Elena and Caterina and Cati.

And finally for my beloved husband Alfred, it is so wonderful sharing everything with you, without your love, company, support, understanding, patience, not to mention help, ..., this thesis would have never been completed!

# Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Giulia Garau*
*Centre for Speech Technology Research*
*University of Edinburgh*
*Edinburgh EH8 9AB, UK*)

# Table of Contents

# List of Figures

# List of Tables

# Publications list

- G. Garau and S. Renals, Pitch Adaptive Features for LVCSR, In Proc. Interspeech, September 2008 (chapter 7).
- G. Garau and S. Renals, Combining Spectral Representations for Large Vocabulary Continuous Speech Recognition, IEEE Transactions on Audio, Speech and Language Processing, March 2008 (chapter 6).
- T. Hain, L. Burget, J. Dines, G. Garau, M. Karafiat, D. van Leeuwen, M. Lincoln, V. Wan, The 2007 AMI(DA) System for Meeting Transcription, In Proc. of the Rich Transcription 2007 Spring Meeting Recognition Evaluation, May 2007.
- T. Hain, L. Burget, J. Dines, G. Garau, M. Karafiat, M. Lincoln, J. Vepa, and V. Wan, The AMI System for the Transcription of Speech in Meetings, In Proc. IEEE ICASSP, April 2007.
- T. Hain, L. Burget, L. Burget, J. Dines, G. Garau, M. Karafiat, M. Lincoln, J. Vepa, and V. Wan, The AMI meeting transcription system: Progress and performance, In Proc. of the Rich Transcription 2006 Spring Meeting Recognition Evaluation, 2006.
- G. Garau, S. Renals, and T. Hain., Applying Vocal Tract Length Normalization to Meeting Recordings, In Proc. Interspeech, September 2005 (chapter 5).
- T. Hain, J. Dines, G. Garau, M. Karafiat, D. Moore, V. Wan, R. Ordelman, and S. Renals, Transcription of Conference Room Meetings: an investigation", In Proc. Interspeech, September 2005.
- T. Hain, L. Burget, J. Dines, G. Garau, M. Karafiat, M. Lincoln, I. McCowan, D. Moore, V. Wan, R. Ordelman, and S. Renals, The 2005 AMI system for the Transcription of Speech in Meetings, In Proc. Workshop on Multimodal Interactions and Related Machine Learning Algorithms (MLMI), July 2005.
- T. Hain, L. Burget, J. Dines, G. Garau, M. Karafiat, M. Lincoln, I. McCowan, D. Moore, V. Wan, R. Ordelman, and S. Renals, The development of the AMI

system for the Transcription of Speech in Meetings, In Proc. Workshop on Multimodal Interactions and Related Machine Learning Algorithms (MLMI), July 2005.

# Abbreviations list

- **AMI** - Augmented Multiparty Interaction
- **AMIASR** - AMI Automatic Speech Recogniser
- **ASR** - Automatic Speech Recognition
- **CART** - Classification and Regression Tree
- **CMLLR** - Constrained Maximum Likelihood Linear Regression
- **CMN** - Cepstral Mean Normalisation
- **CMU** - Carnegie Mellon University
- **CTS** - Conversational Telephone Speech
- **CVN** - Cepstral Variance Normalisation
- **FFT** - Fast Fourier Transform
- **FW** - Frequency Warping
- **HLDA** - Heteroscedastic Linear Discriminant Analysis
- **HMM** - Hidden Markov Model
- **ICSI** - Interational Computer Science Institute
- **IDIAP** - Institut Dalle Molle d'Intelligence Artificelle Perceptive
- **IHM** - Indipendent Headset Microphones
- **LDA** Linear Discriminant Analysis
- **LVCSR** - Large Vocabulary Continuous Speech Recognition
- **MAP** - Maximum A Posteriori
- **MDM** - Multiple Distant Microphones
- **MFCC** - Mel Frequency Ceptral Coefficients

- **MF-PLP** - Mel Frequency Perceptual Linear Predicion
- **ML** - Maximum Likelihood
- **MLE** - Maximum Likelihood Estimation
- **MLLR** - Maximum Likelihood Linear Regression
- **MMI** - Maximum Mutual Information
- **MPE** - Minimum Phone Error
- **NB** - Narrowband
- **NIST** - National Institut of Standards and Technology
- **PLP** - Perceptual Linear Prediction
- **ROVER** - Recognition Output Voting Error Reduction
- **RT** - Rich Transcription
- **SAT** - Speaker Adaptive Training
- **SI** - Speaker Independent
- **SD** - Speaker Dependent
- **STFT** - Short Time Fourier Transform
- **STRAIGHT** - Speech Transformation and Representation using Adaptive Interpolation of weiGHTed spectrum
- **TEMPO** - Time-domain Excitation extractor using Minimum Perturbation Operator
- **VTLN** - Vocal Tract Length Normalisation
- **WB** - Wideband
- **WER** - Word Error Rate

# Chapter 1

# Introduction

Automatic speech recognition aims to mimic the human capabilities of perceiving speech using a machine. First attempts in this direction date back to the 1950's, when a speaker dependent system for isolated speech recognition was built at Bell Laboratories (Davis et al., 1952). In this system formant trajectories were used as a reference pattern to identify the best matching digit. At that time speech recognition systems were only able to recognise small vocabularies of 10–100 words in isolated mode (i.e. the speaker had to pause between words to make the system understand what he was saying).

In the 1960's some of the most important techniques such as filter bank analysis and dynamic programming were introduced. But it was not until the 1970's that the adoption of Linear Predictive Coding (LPC) and especially pattern recognition techniques enabled the development of medium vocabulary (100–1000 words) continuous speech recognition systems. For example the use of graph search was introduced by representing speech as a network of words (Lowerre and Reddy, 1976). The first language modeling techniques were also used for the development of the IBM speaker dependent dictation system VAT (Voice Activated Typewriter) (Jelinek et al., 1975). Furthermore at the same time AT&T developed a speaker independent voice dialing system using speaker clustering algorithms, where the number of different realisations for each word across a wide user population was determined.

The 1970's and 1980's saw the development of one of the more substantial breakthroughs in speech recognition: Hidden Markov Models (HMMs) (Poritz, 1988; Rabiner, 1989). This technology, pioneered by the IBM, IDA laboratoraties

and CMU (Baum, 1972; Ferguson, 1980; Bahl et al., 1983), saw one of its first applications even earlier in the Dragon system, developed in the 1970s by Baker (1975). HMMs are doubly stochastic models consisting of a set of hidden states. They include an underlying statistical model (a Markov chain), which is not observable, characterising the probabilistic relationship between the states, and a second process aimed at generating the sequence of observations associated to the hidden states. These statistical models enabled the integration of acoustic modeling and language modeling in a consistent framework, allowing to build the first large vocabulary continuous speech recognisers.

In the 1990's techniques to make the flexible HMM infrastructure more robust were investigated. Some of these technologies aimed at reducing the mismatch between training and testing conditions: such as the Maximum Likelihood Linear Regression (MLLR) family (Legetter and Woodland, 1994; Gales and Woodland, 1996; Digalakis et al., 1995) and Maximum a Posteriori (MAP) adaptation (Gauvain and Lee, 1994) techniques. These approaches were applied both to adapt the recogniser to specific speakers or to specific acoustic domains.

The use of speaker normalisation techniques such as Vocal Tract Length Normalisation (VTLN), aiming to normalise for the speaker's specific vocal tract length, was also wide spread in the 1990's (Cohen et al., 1995; Wegmann et al., 1996; Eide and Gish, 1996; Zhan and Waibel, 1989; Hain et al., 1999), although one of the first applications of VTLN dates back to 1977 when it was used in a vowel recognition system (Wakita, 1977).

Together with these technology evolutions, the application fields of speech recognition have also changed quite significantly over time. Several research advancements were driven by the speech recognition evaluations run by NIST. These evaluations aim at benchmarking the performances of the best ASR systems (Fiscus et al., 2007). In the early 1990s speech recognition systems were evaluated on constrained tasks such as the resource management (continuous military style speech) or read speech data such as the Wall Street Journal task. Automatic transcription of Broadcast News has been investigated since the late 1990s. Recently the speech recognition community has started investigating more challenging tasks such as conversational telephone speech, and multiparty meeting speech recognition. These tasks present an increasing number of challenges. Weintraub et al. (1996) com-

pared the recognition of spontaneous conversational speech to the recognition of read speech under similar testing conditions (same microphones, acoustics and transcription) finding that speaking style has a tremendous effect on the performances of an LVCSR system: the word error rate (WER) on read speech was half of that observed on spontaneous speech. Multiparty meeting speech recognition is therefore a rather challenging task, being at the moment the "most difficult actively researched domain for speech to text systems" (Fiscus et al., 2007). It is an extremely interesting task because it represents one of the most natural communication scenarios where humans freely interact without constraints.

Spontaneous speech is characterised by an increased speaker variability. Eskenazi (1993) compared the characteristics of read and spontaneous speech, pointing out that conversational speech typically shows more frequent deletion of consonants, wider formant space (F1/F2), smaller F0 range, shorter ungrammatical pauses, and higher variability from a phonologic point of view.

The main goal of this thesis is the investigation of speaker normalisation in the context of spontaneous speech recognition (particularly in multiparty meetings), in order to minimise the mismatch between acoustic models and training data.

## 1.1 Speaker normalisation

Automatic Speech Recognisers (ASR) are complex and composite systems consisting of a number of components which should all work in harmony in order to provide a good overall performance. Different components, which at a first sight look far apart in the processing chain, may have an influence on the behaviour of all the other building blocks. One first example is the influence of the segmentation of the acoustic signal on all the other blocks of the speech recogniser, influencing the way in which the language model will act, the normalisation and adaptation, and of course the decoding. Another example is the fact that normalisation and adaptation operate in a complementary way, trying on one hand to reduce the mismatch between acoustic features and acoustic models (normalisation), and on the other hand trying to make the acoustic models more suitable for a particular speaker (adaptation). Moreover the choice of the acoustic features has a primary effect on the entire ASR system, and in particular on the normalisation and adaptation behaviour.

Adaptation and normalisation techniques attempt to make the acoustic models and the features more suitable for the target testing conditions they have been applied to. These techniques allow the adaptation of the models from one acoustic domain to another or from one speaker to another. In particular, speaker variability is one of the main problems in current speaker independent ASR due to the presence of different speaking styles, accents and speaker characteristics (vocal tract length and shape). Vocal tract configuration has a substantial effect on the observed spectrum: for example, a typical female speaker exhibits formant frequencies around 20-25% higher than those of a male speaker. Vocal Tract Length Normalization is a state of the art technique which normalizes for inter-speaker variability. It is based on the speaker-specific warping of the frequency axis, parameterized by a scalar warp factor. This factor is typically estimated using Maximum Likelihood, that is maximizing the probability of a given speech recognition output given the acoustics (vocal tract length normalized features) and the acoustic models. This approach results in improved recognition accuracies, but also in incorporating in the optimisation variables other than the sole vocal tract length. Our most general question is to

> *investigate how VTLN may be applied to multiparty conversations and to discover what are the unique characteristics of this conversational domain from the speaker normalisation point of view.*

Initial experiments, on the use of ML VTLN in the meeting domain, reported a substantial improvement in accuracy (Garau et al., 2005). Investigating the behavior of the VTLN warping factors we have shown that unique stable estimates are not usually observed in dialogues. Instead warping factors appear to be influenced by the context of the meeting, in particular the current conversational partner. These results are consistent with predictions made by the psycholinguistic interactive alignment account of dialogue, when applied at the acoustic and phonological levels. Pickering and Garrod (2004) argued that, during a dialogue, production and comprehension are coupled so that two speakers can be seen to align at different levels: developing the same expressions to refer to particular objects, aligning in articulation, converging in accents and speech rates. The estimated warping factors of two interlocutors are typically non-aligned at the start of a meeting, but can be seen to align (or at least evolve through phases of alignment) as the meeting progresses. It

is therefore evident that VTLN, when applied with a maximum likelihood approach, is normalising for VTL variability during speech production (Dusan, 2005b).

Our second question, which arises from our preliminary investigations, is a dual one:

> *(1) to find acoustic features which are more suitable for normalisation and in particular for VTLN, (2) to isolate the primary function of VTLN (which is to normalise for the speaker specific vocal tract length) from the other normalisation effects, such as channel normalisation and the overall reduction of the mismatch between acoustic features and models.*

To answer these questions, we investigated novel approaches for speech signal processing which should be able to exploit a better time-frequency resolution, obtaining thus a more speaker independent feature representation. In the next section we will highlight our main efforts towards finding a more speaker independent speech representation by investigating the use of a pitch adaptive spectral representation based on STRAIGHT (Kawahara et al., 1999).

## 1.2 Contribution

In our preliminary experiments about the use of VTLN on multiparty meetings, we found that VTLN warping factors estimated using ML exhibited significant variability over time (Garau et al., 2005). This is consistent with the variation of pitch over time due to the variation of the larynx position. Therefore we investigated the use of a spectral representation which is less pitch-dependent in conjunction with VTLN. This study is based on the use of the pitch adaptive STRAIGHT spectral representation instead of the conventional short time Fourier transform for the computation of Mel Frequency Cepstral Coefficients (MFCCs) and MF-Perceptual Linear Prediction (PLP) coefficients. The spectral analysis of STRAIGHT uses a fundamental period adaptive window which gives equivalent resolution both in time and frequency domains; followed by an adaptive smoothing of the time-frequency representation. Therefore the resulting pitch adaptive spectral representation allows to extract pitch normalised features.

Experiments were performed on three large vocabulary tasks: WSJCAM0, Conversational Telephone Speech (CTS), and the multiparty meeting domain both for

the close talking (individual headsets) and the multiple distant microphone tasks. This set of experiments allowed us to benchmark the use of pitch adaptive features on a wide range of speaking styles, channel and acoustic conditions. WSJCAM0 is a fairly simple task consisting of read speech using a close-talking microphone in a quiet environment. CTS and the meeting domain are more challenging, involving spontaneous conversational speech. They are particularly useful in studying the effect of a pitch adaptive representation, because this domain is known to have richer prosodic variation. Moreover, CTS involves telephone speech which is subject to a bandpass filter that partly obscures the pitch; while the multiparty meetings were recorded in reverberant conditions with overlapping speech. On the meeting task the situation is further complicated when multiple distant microphones are used to record the conversation, and beamforming algorithms are applied to the recorded signals.

Experimental results showed that not only pitch adaptive features provide comparable results to conventional features and are particularly beneficial when VTLN is adopted, but their combination using HLDA and ROVER techniques provides consistent relative improvements across all different tasks (3–9% relative word error rate reduction).

The complementarity between conventional and STRAIGHT derived features was also further analysed by using separately the pitch adaptive and the smoothing part of STRAIGHT. In this experiments it was found that most of the complementarity is given by the pitch adaptive module. Pitch adaptive features also manifested increased speaker independence making them definitely more suitable features for speaker normalisation (one of our initial aims).

## 1.3 Thesis Structure

This thesis can be subdivided in two parts: a background one where the speech recognition problem and speaker normalisation are introduced, and an experimental part where we describe both the techniques and the experiments developed to answer the main research questions of this thesis. Both parts have a special focus on multiparty meetings. The experimental part can be subdivided in chapter 5 which deals with the study of the relationship between vocal tract length and the

fundamental frequency (and their changes due to the variation in larynx position), and chapter 6 and 7 which mainly study the use of a pitch adaptive spectral representation in conjunction with VTLN to deal with the effect of the fundamental frequency manifested as harmonic lines, therefore making the spectrum more speaker normalised (more speaker independent). More precisely:

- In chapter 2 we will present an overview of a HMM based speech recognition system showing feature extraction techniques.
- Chapter 3 will provide a background on speaker adaptation and normalisation, with particular attention to vocal tract length normalisation.
- Chapter 4 will describe the data and the main tools used in the experiments of this thesis focusing on meeting recognition (corpora, dictionary, language models, preprocessing, speaker adaptation and normalisation, and the NIST evaluations).
- In chapter 5 experiments on the use of VTLN for spontaneous speech recognition (conversational telephone speech and multiparty meetings) will be described and the behaviour of VTLN warping factors in the multiparty meeting domain will be analysed.
- In chapter 6 experiments on the use of STRAIGHT derived pitch adaptive features in conjunction with VTLN will be outlined on three large vocabulary continuous speech recognition tasks: WSJCAM0, CTS and meetings.
- A deeper experimental analysis on the use of STRAIGHT is presented in chapter 7.
- Finally chapter 8 summarises the main achievements of this research discussing the theoretical implications and the main findings.

# Chapter 2

# Automatic Speech Recognition Overview

## 2.1 Introduction

The problem of Automatic Speech Recognition (ASR) is finding the most probable sequence of words given the observed acoustics. The waveform is first processed by the feature extraction module to extract meaningful information in the form of the acoustic vectors $\mathbf{O} = \mathbf{o_1}, \mathbf{o_2}, ..., \mathbf{o_T}$. Then decoding is performed to find the sequence of words $\mathbf{W} = w_1, w_2, ..., w_N$ which most likely generates the observation sequence **O**. More precisely we want to solve the equation:

$$\mathbf{W}^* = \arg\max_{\mathbf{W}} P(\mathbf{W}|\mathbf{O}) = \arg\max_{\mathbf{W}} P(\mathbf{O}|\mathbf{W})P(\mathbf{W}) \tag{2.1}$$

where $P(\mathbf{O}|\mathbf{W})$ is the probability of the acoustic measurements of the observation **O** given the hypothesised sequence of words **W** and it is referred to as the acoustic model. The sequence of words **W** can be represented either by word units or by the concatenation of sub-word units such as phonemes. The choice of the speech units will be outlined in more detail in section 2.2.1. $P(\mathbf{W})$ is the a priori probability of the sequence of words **W** defined by language modeling.

#### 2.1.0.1 N-gram language modeling

The language model estimates the probability of the sequence of words $P(\mathbf{W})$. The most common language model form are N-grams. N-grams model the probability of the words $w_i$ by conditioning it to the n-1 preceding words: more specifically the probability $P(w_i|w_{i-1}, w_{i-2}, ..., w_{i-n+1})$ is modelled. For n=1 we have unigrams, for n=2 bigrams and for n=3 trigrams and so on. The N-gram probabilities are estimated by counting the sequences of n words in text corpora. Moreover the language model probability of a sequence of N words is computed by:

$$P(\mathbf{W}) = \prod_{i=1}^{N} P(w_i|w_{i-1}, w_{i-2}, ..., w_{i-n+1}). \tag{2.2}$$

where the joint distribution $P(w_1, w_2, ..., w_N)$ is factorised as a chain product of conditional probabilities in the form $p(w_i|w_{i-1}, ..., w_{i-n+1})$. Trigrams or at most fourgram language models are typically employed by large vocabulary continuous speech recognition (LVCSR) systems. This is because increasing the order of the language model results in requiring larger amounts of training data and a good trade off must be reached. Moreover the search space to find the most probable word sequence grows with the language model order.

## 2.2 Acoustic Modeling

### 2.2.1 The Speech Units Choice: context-dependent models

The choice of speech units plays a leading role in acoustic modeling. According to Lee (1990), in order to make an appropriate choice, two important characteristics should be fulfilled: consistency and trainability. The units should be consistent in the sense that multiple occurrences of the same unit should have a similar acoustic realisation. The trainability property requires a sufficient number of training examples.

Traditionally for small vocabulary systems, such as digit recognition, entire words have been adopted as base units. This is the most intuitive choice and it also presents several advantages, for example the capability of modeling context effects between adjacent phonemes within the same word, and the fact that there is no need for a pronunciation dictionary. Because of these advantages they are in fact

the best choice when sufficient data are available. The drawbacks of using words as speech units are: the linear increase of the necessary training data as well as the memory usage when the dictionary size increases, and the need of training new models when a new word is added to the recognition system.

For LVCSR systems, where many words need to be modelled and there are not enough examples to train separate models for each word, usually sub-word units are adopted, such as phonemes (monophones). Phonemes have the advantage of requiring few data for training (the number of phonemes in English is only around 45). However their main problem is that they assume complete context independence between phonemes and this is unfortunately not true.

Thus context-dependent models were introduced to model phones in context (Schwartz et al., 1985). Both the right and the left context should be considered: for left biphones we only consider the left context and for right biphones we only consider the right context, while we consider both at the same time using triphones. Triphones are the best choice from a consistency point of view and are often adopted in LVCSR systems. For continuous speech it is also crucial to model transitions between words, thus not only context-dependent phoneme models such as word-internal triphones (which only model the context inside words) but also cross-word triphones (modeling the context across adjacent words too) are adopted. An example of the pronunciation labels for monophones, word-internal triphones and cross-word triphones for the utterance “How are you doing” is shown in figure 2.1. `sil` represents silence and `sp` represents short pauses and the context dependent phonetic models have been represented following the Hidden Markov Model Tool Kit (HTK Young et al. (2006)) notation. For example `l-ph+r` is an occurrence of the phoneme `ph` with the left context represented by `l-` and the right context represented by `+r`.

The main problem of context-dependent models is trainability: if we consider for example triphones with a pronunciation dictionary of 45 phones there is a number of $45^3$ possible combinations and some of these may not even be seen in the training sets. Techniques which aim to address this trainability issue are described in section 2.2.3.

**Monophones:**

| | | | |
|---|---|---|---|
| **sil** | **hh** | **aw** | **sp** |
| **aa** | **r** | **sp** | **y** |
| **uw** | **sp** | **d** | **uw** |
| **ih** | **ng** | **sp** | **sil** |

**Triphones Word-Internal:**

| | | | |
|---|---|---|---|
| **sil** | **hh+aw** | **hh-aw** | **sp** |
| **aa+r** | **aa-r** | **sp** | **y+uw** |
| **y-uw** | **sp** | **d+uw** | **d-uw+ih** |
| **uw-ih+ng** | **ih-ng** | **sp** | **sil** |

**Triphones Cross-Word:**

| | | | |
|---|---|---|---|
| **sil** | **sil-hh+aw** | **hh-aw+aa** | **sp** |
| **aw-aa+r** | **aa-r+y** | **sp** | **r-y+uw** |
| **y-uw+d** | **sp** | **uw-d+uw** | **d-uw+ih** |
| **uw-ih+ng** | **ih-ng+sil** | **sp** | **sil** |

Figure 2.1: An example of the labels of monophones, word-internal and cross-word triphones for the utterance “How are you doing”

### 2.2.2 Hidden Markov Models

Acoustic modeling involves finding a way of estimating the likelihood of the observed sequence **O** given a certain word sequence **W**: $P(\mathbf{O}|\mathbf{W})$. In most of the state-of-the-art ASR systems $P(\mathbf{O}|\mathbf{W})$ is modelled using Hidden Markov Models (HMMs). These models are a natural choice for modeling speech which has a temporal structure represented by a sequence of acoustic observations. HMMs, introduced by Baum (1972), are defined as stochastic finite state automata consisting of a sequence of states **S** with transitions for each time $t$ from state $s_i$ to state $s_j$ with probability $a_{s_i,s_j}$ generating a sequence of observations **O**. In practice while the observation sequence **O** is known, the state sequence **S** is unknown or hidden, this is why they are called Hidden Markov Models. More explicitly the assumptions required by HMMs can be defined as follows:

- the Markovian assumption: a state $s_j$ is only conditioned on the previous state

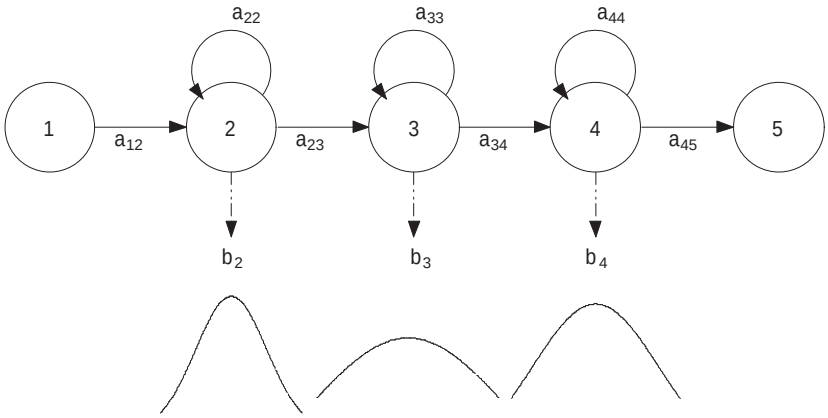



Figure 2.2: An example of a 3 state left-to-right HMM system

$s_i$ (and it is independent on all the other states);

- the observation at time $t$ is only dependent on the state which generated it (and it is independent of all the other observations);
- the stationarity property: the parameters of HMMs are fixed over time;
- discrete hidden states are considered;
- finally we assume continuous observations (parameterised by the acoustic feature vectors).

In our experiments we adopted cross-word triphone units with a 3 state left to right topology, shown in figure 2.2. In this topology only three states are emitting, while the entry and the exit states are simply used to connect models together such that phonemes are joined to form words, and words are joined to form utterances. The representation presented in the figure is that adopted by the HTK (Young et al., 2006), which was used for the experiments in this thesis.

Adopting an HMM with a state sequence $\{s_1, s_2, ..., s_T\}$, the acoustic modeling probability $P(\mathbf{O}|\mathbf{W})$ can be extended as follows:

$$\arg\max_{\mathbf{W}} P(\mathbf{O}|\mathbf{W}) = \arg\max_{\mathbf{w_1^N}} p(\mathbf{o}_1^T|w_1^N) = \max_{s_1^T : w_1^N} \prod_{t=1}^{T} p(\mathbf{o_t}|s_t; w_1^N) p(s_t|s_{t-1}; w_1^N) \quad (2.3)$$

$p(\mathbf{o_t}|s_t;w_1^N)$ are the emission probabilities and $p(s_t|s_{t-1};w_1^N)$ are the transition probabilities. A continuous observation HMM (Rabiner, 1989) can be defined through a parameter model set $\lambda$ which consists of a transition matrix $\mathbf{A}$, an initial state probability vector $\pi$, and an observation probability distribution for each state $b_{s_j}(\mathbf{o_t})$. The transition matrix $\mathbf{A}$ is defined as $\mathbf{A} = \{a_{ij} = P(q(t+1) = s_j|q(t) = s_i)\}$ and $\pi$ is given by $\pi_\mathbf{i} = P(q_1 = s_i)$ where $s_j$ are the individual states and $q(t)$ is the state at time $t$. The emission probabilities $b_{s_j}$ are continuous probability density functions usually approximated by a mixture of Gaussian distributions:

$$b_{s_j}(\mathbf{o_t}) = \sum_{m=1}^{M} \{c_{j,m}\mathcal{N}(\mu_{j,m}, \Sigma_{j,m}; \mathbf{o_t})\} = \tag{2.4}$$

$$= \sum_{m=1}^{M} \{c_{j,m} \frac{1}{\sqrt{(2\pi)^n \left|\Sigma_{j,m}\right|}} e^{-\frac{1}{2}(\mathbf{o_t}-\mu_{j,m})^T {\Sigma_{j,m}}^{-1}(\mathbf{o_t}-\mu_{j,m})}\}$$

where $c_{j,m}$ are the mixture weight coefficients of the $m$th mixture in state $j$, $\mu_{j,m}$ are the mean vectors and $\Sigma_{j,m}$ are the covariance matrices of the multivariate gaussian distribution $\mathcal{N}$ and $n$ is the dimensionality of the observation vector $\mathbf{o_t}$.

Following Ferguson (1980) Rabiner's tutorial on HMMs (Rabiner, 1989) defines the 3 fundamental problems of HMMs as follows:

- evaluation: finding a way to efficiently compute $P(\mathbf{O}|\lambda)$, the probability of the observation sequence $\mathbf{O} = \mathbf{o_1}, \mathbf{o_2}, ..., \mathbf{o_T}$ given the parameter model set $\lambda$
- decoding: finding the optimal state sequence $s_1, s_2, ..., s_T$ given the observation sequence $\mathbf{O}$ and the model $\lambda$
- learning: estimating the parameters of the model $\lambda$ which maximise the likelihood of the observation sequence of the training data $\mathbf{O}$ (i.e. maximising $P(\mathbf{O}|\lambda)$).

#### 2.2.2.1 The evaluation problem

The easiest solution to the so called "evaluation" problem is to account for every possible state sequence $\{s_1, s_2, ..., s_T\}$ given the $T$ observation vectors multiplying all the transition probabilities and emission probabilities:

$$P(\mathbf{O}|\lambda) = \sum_{s_1,s_2,...,s_T} \pi_{s_1} b_{s_1}(\mathbf{o_1}) a_{s_1,s_2} b_{s_2}(\mathbf{o_2}) ... a_{s_{T-1},s_T} b_{s_T}(\mathbf{o_T}). \tag{2.5}$$

This can be interpreted as follows: the system is initially in state $s_1$ with probability $\pi_{\mathbf{s_1}}$ and generates the observation $\mathbf{o_1}$ with probability $b_{s_1}(\mathbf{o_1})$; at time $t = 2$ the system goes in state $s_2$ with probability $a_{s_1,s_2}$ and generates the observation $\mathbf{o_2}$ with probability $b_{s_2}(\mathbf{o_2})$ and so on until the last state of the sequence $s_T$ is reached.

Unfortunately this approach is computationally unfeasible because it is $O(2 \cdot T \cdot N^T)$. Therefore a more efficient iterative solution has been proposed which is known as the forward-backward algorithm originally introduced by Baum et al. (1970). The forward part of the algorithm (which is the only part used to estimate the total likelihood $P(\mathbf{O}|\lambda)$), starts from the observation that the probability of being in state $s_j$ and having observed the sequence $\mathbf{o_1}, \mathbf{o_2}, ..., \mathbf{o_t}$ given the model $\lambda$ can be computed as the sum of the forward probabilities of all possible predecessor states $s_i$ weighted by the transition probability $a_{s_i,s_j}$ and the emission probability $b_{s_j}(\mathbf{o_t})$. The total likelihood is therefore given by:

$$P(\mathbf{O}|\lambda) = \sum_{i=1}^{N} \alpha_T(s_i) \tag{2.6}$$

where the forward probability $\alpha_t(s_j)$ is computed as in figure 2.4.

#### 2.2.2.2 The decoding problem

The decoding problem can be seen as finding the maximum likelihood state sequence given the observations and the acoustic model $\lambda$. This problem can be solved similarly to the forward algorithm, but here instead of the sum over all possible state sequences we aim to find the state sequence corresponding to the maximum probability (Viterbi, 1967; Forney, 1973).

The partial likelihood of the state sequence at time $t$ ending at state $s_j$ is given by:

$$\delta_t(j) = \begin{cases} \pi_{s_j} b_{s_j}(\mathbf{o_1}) & 1 \leq j \leq N \quad \text{if} \quad t = 1 \\ \max_{i=1,...,N} \left[\delta_{t-1}(i) a_{s_i,s_j}\right] b_{s_j}(\mathbf{o_t}) & 1 \leq j \leq N \quad \text{if} \quad 2 \leq t \leq T \end{cases} \tag{2.7}$$

and the optimal state sequence and correspondent probability is given by:

$$q_T^* = \arg\max_{1 \leq i \leq N} \delta_T(i), \quad P_T^* = \max_{1 \leq i \leq N} \delta_T(i). \tag{2.8}$$

The token passing algorithm (Young et al., 1989) can be used to trace back the most likely state sequence. This algorithm stores the information of each partial

path (the probability and the traceback information) in a structure called a token. For continuous speech the token is propagated through a network of multiple parallel hypotheses: this is built using both the information provided by the language model (which gives possible word sequences) and the pronunciation of the words contained in the dictionary. The search network can be built statically prior to decoding as in Mohri et al. (1998), or dynamically integrating the building process into the decoding as in Odell et al. (1994). Finding the most likely state sequence can be computationally expensive. In particular for large vocabulary speech recognition the hypothesis network can be large, especially when cross-word models and bigram or trigram language models are used. Therefore various approaches have been developed to reduce the computation effort. These are generally referred to as pruning. One of the most common pruning techniques is the beam pruning. For each frame the most likely partial path is found and its likelihood is used as the top of the beam of fixed width. Then the tokens having a likelihood falling outside the beam are pruned out. Unfortunately search errors may occur if the correct hypothesis is pruned out, thus it is important to choose carefully the beam width in order to achieve a good trade off between computational requirements and accuracy.

Although the main goal of recognition is to find the most likely word sequence (equation 2.1) it is possible to find the N best hypothesised word sequences just by storing the the N best tokens in each state, instead of only the best one. This is useful especially for large vocabulary speech recognition systems, since it allows to perform multiple rescoring passes. In this way higher order language models or different acoustic models can be used without having to solve equation 2.1 again from scratch. The N best hypotheses are usually stored in a compact form through word lattices (Richardson et al., 1995). A word lattice consists of a set of nodes, representing start and end points of words, and a set of arcs representing word hypotheses along with the acoustic and language model scores. A more compact representation of word lattices are the so called confusion networks (Mangu et al., 2000), where nodes do not represent points in time but only impose word sequence constraints. More in detail confusion networks represent all possible hypothesised word sequences, transforming the lattice space into slots each having a set of word hypotheses represented by arcs. In figure 2.3 we show a comparison of word lattices and confusion networks. Confusion networks are also useful for word error minimi-

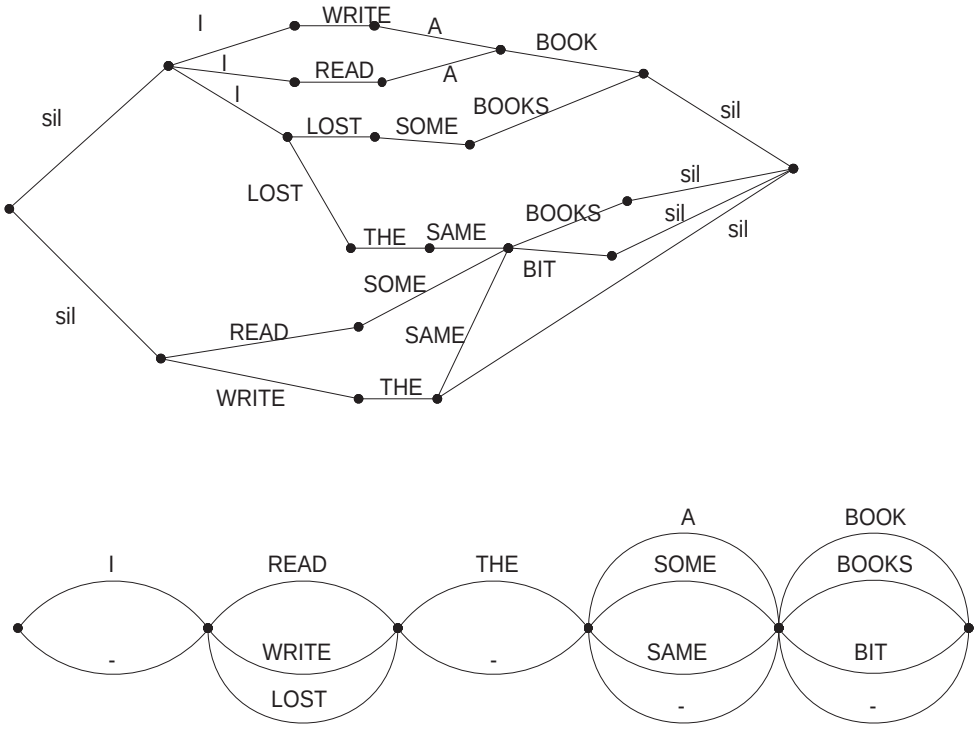



Figure 2.3: Word lattice and confusion network examples

sation when the output of multiple systems is combined (Evermann and Woodland, 2000; Fiscus, 1997).

#### 2.2.2.3 The learning problem: acoustic model parameter estimation

The problem of estimating all the model parameters consists in finding the $\lambda$ parameter model set which best represents the data observed in the training dataset. There are 2 main optimisation criteria: Maximum Likelihood (ML) and Maximum Mutual Information (MMI).

Maximum likelihood criteria aim to maximise the probability of a given observation $\mathbf{O_W}$ belonging to a given word sequence $\mathbf{W}$ given a parameter model set $\lambda$:

$$L_{tot} = P(\mathbf{O}|\mathbf{W},\lambda). \tag{2.9}$$

To solve this maximisation problem there are no analytic solutions, instead iterative procedures such as Baum Welch or gradient techniques are used. To describe the Baum Welch algorithm we shall first introduce the backward part of the forward-

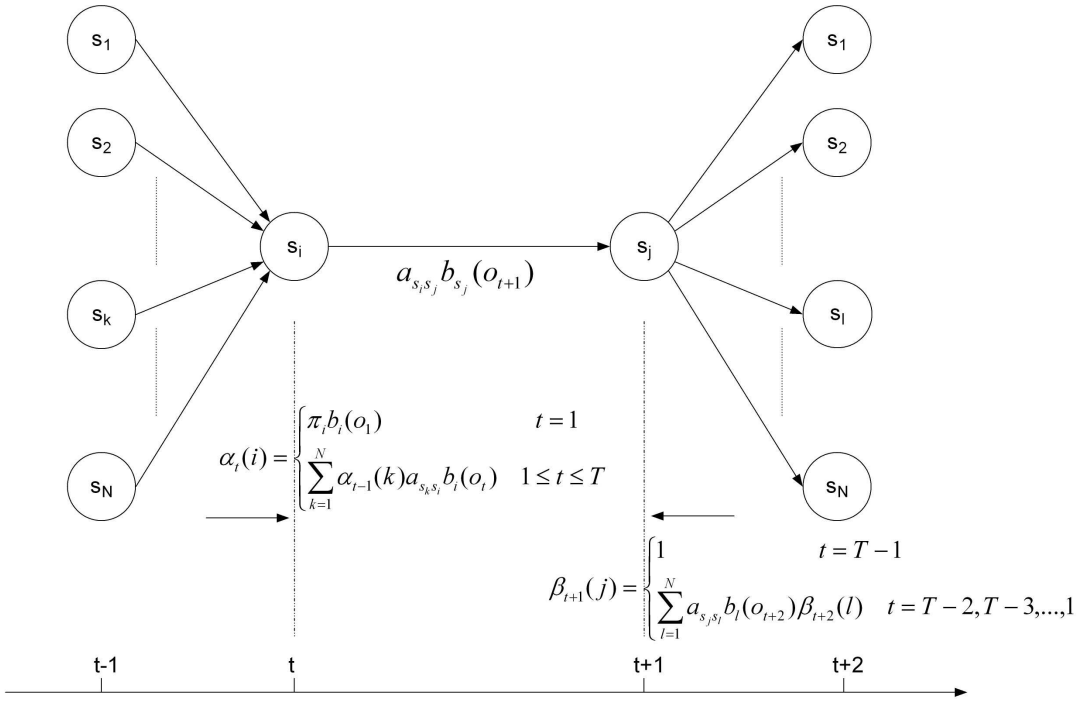



Figure 2.4: An illustration of the forward-backward algorithm

backward algorithm. The backward probability is defined as the probability of observing $\mathbf{o}_{t+1}, \mathbf{o}_{t+2}, \ldots, \mathbf{o}_T$ given that the system is in state $s_i$ at time $t$ and it is defined as in figure 2.4.

Then the Baum Welch algorithm defines a variable $\xi_t(i,j)$ as the probability of being in state $s_i$ at time $t$ and in state $s_j$ at time $t+1$ given the model and the observation sequence, and can be estimated as:

$$\xi_t(i,j) = P(q_t = s_i, q_{t+1} = s_j | \mathbf{O}, \lambda) = \frac{\alpha_t(i) a_{s_i,s_j} b_{s_j}(\mathbf{o_{t+1}}) \beta_{t+1}(j)}{\sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_t(i) a_{s_i,s_j} b_{s_j}(\mathbf{o_{t+1}}) \beta_{t+1}(j)} \quad (2.10)$$

as illustrated in figure 2.4.

A variable $\gamma_t(i)$ is also defined as the probability of being in state $s_i$ at time $t$ given the observation sequence and the model:

$$\gamma_t(i) = \sum_{j=1}^{N} \xi_t(i,j). \quad (2.11)$$

Then if we sum over time $\xi_t(i,j)$ we obtain the total expected number of transitions

from $s_i$ to $s_j$ and if we sum over time $\gamma_t(i)$ we obtain the total number of transitions to state $s_i$ so that the model parameters can be estimated as:

$$\hat{\pi}_i = \gamma_1(i); \quad \hat{\mathbf{a}}_{\mathbf{i},\mathbf{j}} = \frac{\sum_{t=1}^{T-1} \xi_t(i,j)}{\sum_{t=1}^{T-1} \gamma_t(i)}; \quad \hat{\mathbf{b}}_{\mathbf{j}}(\mathbf{o_t}) = \frac{\sum_{t=1}^{T} \mathbf{o}_t \gamma_t(j)}{\sum_{t=1}^{T} \gamma_t(j)} \ . \tag{2.12}$$

Now denoting the current model as $\lambda = \{\mathbf{A}, \mathbf{B}, \pi\}$ and the re estimated model as $\hat{\lambda} = \{\hat{\mathbf{A}}, \hat{\mathbf{B}}, \hat{\pi}\}$, we can iteratively replace $\lambda$ with $\hat{\lambda}$ and apply again the estimation formulas above increasing the probability of the observation sequence $\mathbf{O}$ until convergence is reached. The same solution could be obtained by maximising Baum's auxiliary function:

$$Q(\lambda, \hat{\lambda}) = \sum_{\mathbf{Q}} P(\mathbf{Q}|\mathbf{O}, \lambda) \log P(\mathbf{O}, \mathbf{Q}|\hat{\lambda}) \tag{2.13}$$

over $\lambda$ which was proved to increase the likelihood so that: $P(\mathbf{O}|\lambda) \leq P(\mathbf{O}|\hat{\lambda})$.

In speech recognition, since the observations are continuous signals, continuous observation densities are used in the HMMs and the emission probabilities are defined as in 2.4. Therefore we need to estimate the mixture weights $c_{j,m}$, the mean vector $\hat{\mu}_{\mathbf{j},\mathbf{m}}$ and the covariance matrix $\hat{\Sigma}_{j,m}$. It can be shown that the reestimation of the mixture densities coefficients can be expressed by the following formulas:

$$\hat{c}_{j,m} = \frac{\sum_{t=1}^{T} \gamma_t(j,m)}{\sum_{t=1}^{T} \sum_{m=1}^{M} \gamma_t(j,m)}, \tag{2.14}$$

$$\hat{\mu}_{j,m} = \frac{\sum_{t=1}^{T} \gamma_t(j,m) \cdot \mathbf{o_t}}{\sum_{t=1}^{T} \gamma_t(j,m)}, \tag{2.15}$$

$$\hat{\Sigma}_{j,m} = \frac{\sum_{t=1}^{T} \gamma_t(j,m) \cdot (\mathbf{o_t} - \hat{\mu}_{\mathbf{j},\mathbf{m}})(\mathbf{o_t} - \hat{\mu}_{\mathbf{j},\mathbf{m}})^T}{\sum_{t=1}^{T} \gamma_t(j,m)} \tag{2.16}$$

where $\gamma_t(j,m)$ is the probability of being in state $s_j$ at time $t$ with the $m$th mixture component. This can be estimated using the current set of parameters $\lambda$ and given by:

$$\gamma_t(j,m) = \frac{\alpha_t(j)\beta_t(j)}{\sum_{j=1}^{N} \alpha_t(j)\beta_t(j)} \frac{c_{j,m}\mathcal{N}(\hat{\mu}_{j,m}, \hat{\Sigma}_{j,m}; \mathbf{o_t})}{\sum_{m=1}^{M} \{c_{j,m}\mathcal{N}(\hat{\mu}_{j,m}, \hat{\Sigma}_{j,m}; \mathbf{o_t})\}}. \tag{2.17}$$

Equation 2.14 can be interpreted as the ratio between the number of times the system is in state $s_j$ and occupies the mixture $m$ and the total number of times the system is in state $s_j$, and in a similar way the estimated mean $\hat{\mu}_{j,m}$ is given by a

mean of the observations weighted by the occupancy probability of state $s_j$ with the $m$th mixture.

As mentioned above the model parameters can also be estimated using discriminative MMI training instead of ML. In this thesis only ML techniques have been adopted, therefore a extensive discussion of MMI theory will not be provided here. For an in-depth description of discriminative training techniques the reader may refer to Povey (2003). Maximum mutual information is a discriminative training criterion which maximises the ratio of the probability of the observation sequence given the acoustic model corresponding to the correct word sequence $\lambda_{\mathbf{W}_c}$ and the probability of the observation sequence given any acoustic model (corresponding both to correct and incorrect word sequences $\lambda_{\mathbf{W}_r}$). The MMI criterion is given by:

$$F = \arg\max_{\lambda} log\left(\frac{P(\mathbf{O}|\lambda_{\mathbf{W}_c})}{\sum_{r=1}^{R} P(\mathbf{O}|\lambda_{\mathbf{W}_r})}\right). \tag{2.18}$$

In practice in LVCSR systems lattices are generated recognising the training data and the MMI criterion is optimised on the alternative hypotheses contained in the lattices.

### 2.2.3 State Tying

As we mentioned in section 2.2.1 the most widely used speech units for LVCSR are cross-word triphones. Even if they have the advantage of being a consistent representation they show trainability problems due to the number of possible triphones occurring in speech. In the next paragraph we will provide an example of the amount of data needed to train a cross-word triphone HMM system.

In the English language we can consider 45 phonemes, therefore the total number of triphones would be $45^3$ (that is over 90000); of these only around 60000 can occur in practice due to the phonotactic constraints of the language. Typically 16 mixture components, 39 dimensional feature vectors and diagonal covariance matrices are used. For each state we would have $(39*2+1)*16 = 1262$ parameters. With a 3 state topology for each triphone we would have a total of 3876 parameters, for a total of 232 million parameters. Therefore modeling so many speech units would require a large amount of data and unfortunately this is not always possible. Moreover some of the triphones may be not well represented in the training data or they may not occur at all.

Various techniques were developed to address the data sparsity problem of triphones. In Schwartz et al. (1985) the use of a weighted combination of all the possible models (monophones, biphones with the left and the right context and triphones) was proposed. In this work the weights are determined according to several factors: the position of the phoneme, the type of model and the number of available training samples. However this technique does not exploit the fact that some triphones are similar and more specifically some phones have the same effect on neighbouring phones. In Lee (1990) the context effect is therefore automatically generalised: similar triphones are iteratively clustered together using a bottom-up procedure (where similar triphones are only merged when this results in an improvement).

For the experiments reported in this thesis a method called tree-based state tying or clustering was used (Young et al., 1994). This method ties together those states which are acoustically similar so that the data coming from similar states are pooled together and lead to more reliable parameter estimates. Phonetic decision trees are used to choose which states may be tied. These are basically binary trees where each node corresponds to a yes/no phonetic question: first of all a different tree is built for each monophone state, and all states for this given monophone are in the root node, then the states are recursively split according to the questions until the tree leaf nodes are reached and the states sharing the same leaf nodes are tied together and will share the same model parameters. The main advantage of this technique is that even unseen triphones can be modelled by simply finding the correspondent leaf nodes.

## 2.3 Feature Extraction

### 2.3.1 Mel Frequency Cepstral Coefficients

Mel Frequency Cepstral Coefficients (MFCCs) are one of the most widely used type of acoustic features (Davis and Mermelstein, 1980). A block diagram of their extraction is shown in figure 2.5.

The first block, the pre-emphasis filter, is a high pass filter which aims to emphasise high frequencies to which the human ear is more sensitive and it has the effect of a 6 dB/octave gain increase, making the average speech spectrum more

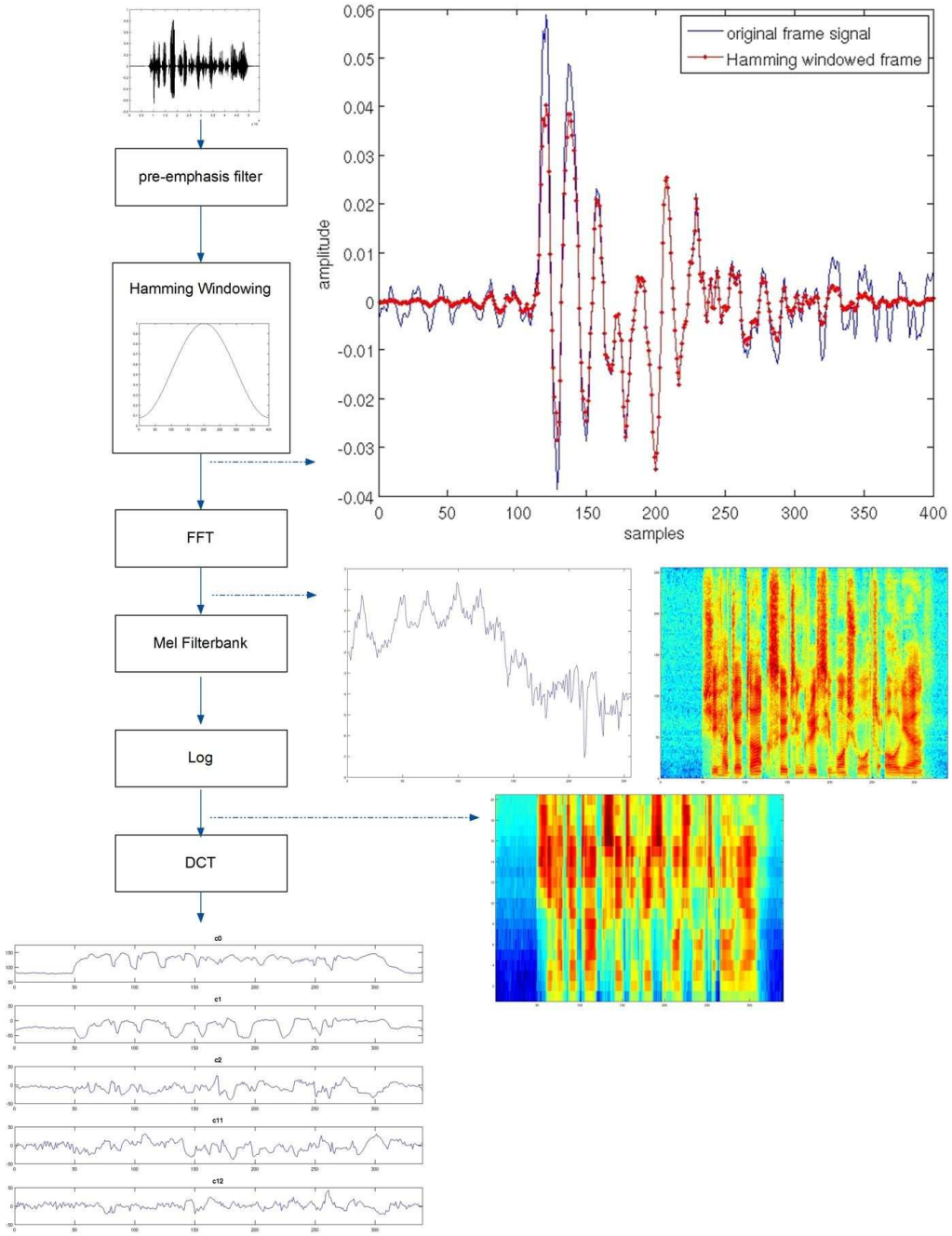



Figure 2.5: MFCC extraction block diagram

flat. The pre-emphasis filter is defined by:

$$H(z) = 1 - \frac{a}{z} \qquad 0 \leq a < 1 \tag{2.19}$$

In all the experiments of this thesis $a = 0.97$.

The pre-emphasis block is followed by the Hamming windowing block. This is a necessary step to compute the Short Time Fourier Transform (STFT) where the Fast Fourier Transform (FFT) of the speech signal is computed using a time sliding window (assuming that the signal is stationary). The duration of the window is typically set to 25 msecs and the shift is 10 msecs. A smoothing window is used to reduce the edge effect and it is usually a Hamming window, a particular type of Hanning window having the lowest possible peak to side lobe level in the frequency domain (approximately 43dB), and given by the formula:

$$w_H(n) = 0.54 - 0.46\cos\left(\frac{2\pi(n-1)}{N-1}\right) \tag{2.20}$$

where $N$ is the total number of the window samples and $n$ is the sample index. This window, having a bell shape in the time domain, has the desirable effect of favoring the speech samples towards the centre of the window. The correct trade-off between the window's duration and the frame length needs to be chosen: to capture rapid dynamics of the spectrum on one side we would need a short window in time, and at the same time a small frame length, in order to have a sufficient resolution in time; on the other side choosing a high overlap of windows would allow to reduce the noise generated by a particular placement of the window, but at the same time would give a too smooth speech representation, obscuring the true variations in the signal (Picone, 1993).

On each of the windowed signal frames the magnitude of the FFT, a computationally efficient version of the Discrete Fourier Transform (DFT), is computed. Psychoacoustic experiments have shown that the perception of sound frequency is not linear but approximately logarithmic. This was demonstrated by studying the auditory system capability of discriminating frequency components of a complex sound through auditory masking. This is also referred to as frequency resolution or selectivity and represents the ability of distinguishing overlapping tones at different frequencies. The cochlea may be viewed as a set of auditory filters placed on the basilare membrane each of them centred on a particular frequency. According

to Fletcher's studies (Fletcher, 1940) the human auditory frequency resolution is approximately in a logarithmic scale. Similarly the critical bandwidths, defined as the frequency range in which two sounds are not perceived independently, increase approximately logarithmically with frequency.

The most commonly used frequency warping functions defined in the literature are the Bark scale and the Mel scale. Davis and Mermelstein (1980) used Mel scaling for the implementation of MFCCs which is defined as:

$$f_M = 2595 \log_{10}\left(1 + \frac{f}{700}\right). \tag{2.21}$$

This is approximately linear up to 1000Hz and logarithmic beyond; the critical bandwidth is given by:

$$BW = 25 + 75\left[1 + 1.4\left(\frac{f_M}{1000}\right)^2\right]^{0.69}. \tag{2.22}$$

In practice the power spectrum is passed through a Mel Filterbank. Then the logarithm of each filter output is computed to take into account the fact that the human loudness perception (the perceived intensity) increases logarithmically with the sound intensity.

Finally the Discrete Cosine Transform (DCT), having the desirable effect of decorrelating and compressing the mel scale filter log energies, is performed:

$$c_{mel}(n,m) = \frac{1}{R}\sum_{l=0}^{R-1} \log\left(E_{mel}(n,l)\right) \cos\left(\frac{2\pi}{R} lm\right) \tag{2.23}$$

where $R$ is the total number of Mel filters, $l$ is the Mel filter index and $m$ is the cepstral coefficient index; the DCT as it is expressed in equation 2.23 is equivalent to computing the inverse Fourier transform of the logarithm of the Mel spectrum. Typically the first 12 cepstral coefficients are computed since higher order coefficients tend to be noisy and less informative.

### 2.3.2 Perceptual Linear Predictive Coefficients

Perceptual Linear Predictive analysis was introduced by Hermansky (1990) with the aim of making Linear Predictive (LP) analysis more consistent with the perceptual properties of the human auditory system. In this section we will introduce Hermansky's implementation of PLPs and at the same time we will outline the differences

of the latter with HTK's implementation (which was in fact used in the experiments presented in this thesis).

A block diagram of Hermansky's PLPs extraction can be seen in figure 2.6. Similarly to MFCCs the first step is the analysis through a Hamming window, described in the previous section, followed by the computation of the power spectrum using the $|FFT|^2$. To take into account the human ear frequency resolution the spectrum is warped along the frequency axis using a Bark scale filterbank with a frequency rescaling given by:

$$f_B = 13\arctan(0.76f) + 3.5\arctan(f/7500). \tag{2.24}$$

However in the HTK implementation a Mel scale filterbank is used, and the features are therefore often referred in the literature as MF-PLPs (Woodland et al., 1997).

While for the MFCCs the magnitude of the filters output was log compressed, for the PLPs, according to Robinson and Dadson's study of human perception of sound intensity, the equal-loudness curve is used (Robinson and Dadson, 1956). On top of this a cubic-root amplitude compression is performed, which emulates the non linear relation between the intensity and the perceived loudness of sound.

Spectral all-pole modeling is then performed and finally cepstral coefficients are extracted applying the DCT. The all-pole modeling theory (Quatieri, 2001) basically starts from the observation that the transfer function model from the glottis to the lips, consisting on the glottal flow $G(z)$, the vocal tract $V(z)$ and the radiation load $R(z)$ can be expressed as:

$$H(z) = AG(z)V(z)R(z) = \frac{A}{1-\sum_{k=1}^{p} a_k z^{-k}} \tag{2.25}$$

where $A$ represents the loudness of the sound and $G(z)$, $V(z)$ and $R(z)$ are all represented by all-pole functions. The basic assumption behind the methods for the estimation of the filter coefficients $a_k$ is the so called *autoregressive model* which states that each speech sample $s(n)$ can be represented as a linear combination of the past speech samples:

$$\tilde{s}(n) = \sum_{k=1}^{p} a_k s(n-k) \tag{2.26}$$

where the $a_k$ are the linear prediction coefficients (LPC). The LPCs are computed so that the prediction error given by the difference of the observed sample $s(n)$ and

the predicted value $\tilde{s}(n)$ is minimised. It can be demonstrated that this is equivalent to solving the equation:

$$\mathbf{R}_n \mathbf{a} = \mathbf{r}_n$$

$$\begin{bmatrix} r_n(0) & r_n(1) & r_n(2) & \dots & r_n(p-1) \\ r_n(1) & r_n(0) & r_n(1) & \dots & r_n(p-2) \\ r_n(2) & r_n(1) & r_n(0) & \dots & r_n(p-3) \\ \dots & \dots & \dots & \dots & \dots \\ r_n(p-1) & r_n(p-2) & r_n(p-3) & \dots & r_n(0) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ . \\ . \\ a_p \end{bmatrix} = \begin{bmatrix} r_n(1) \\ r_n(2) \\ . \\ . \\ r_n(p) \end{bmatrix} \tag{2.27}$$

where $\mathbf{R}_n$ is a Toeplitz matrix and $r_n(\tau)$ is the short time autocorrelation function of $s(n)$. The linear predictive coefficients could therefore be calculated by matrix inversion but a more computationally efficient method called Levinson recursion (Levinson, 1947) can be adopted.

In practice during PLPs computation the inverse DFT is applied to the Bark scaled spectrogram (Mel scaled spectrogram in the MF-PLPs case) and then the autocorrelation function is used for the LPC analysis.

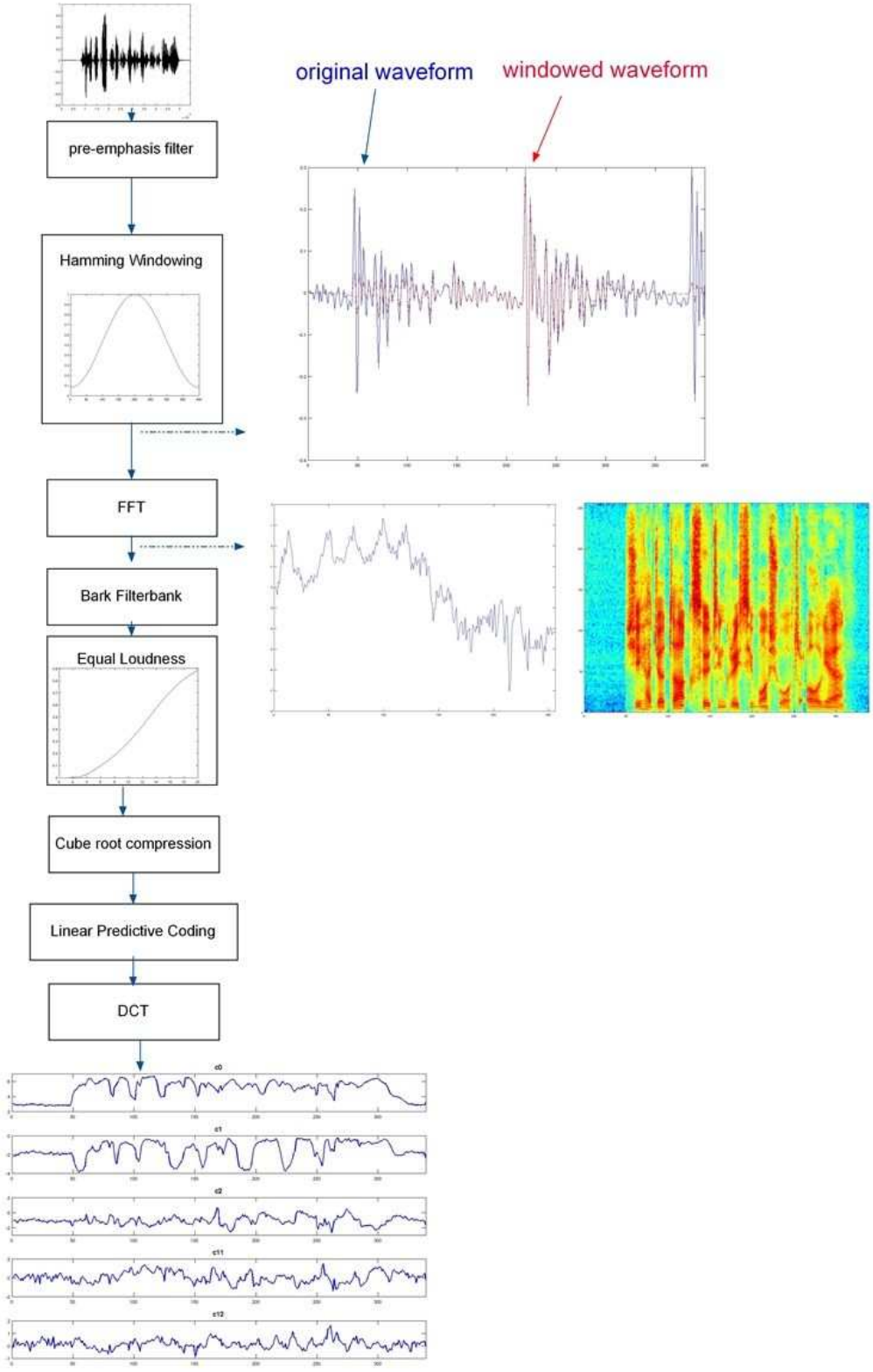



Figure 2.6: PLP extraction according to Hermansky (1990)

# Chapter 3

# Speaker adaptation and normalisation

## 3.1 Speaker Adaptation vs Speaker Normalisation

The accuracy of an ASR system can be significantly improved by matching the training and testing conditions. Unfortunately in practice this is hardly possible and more general speaker and domain independent systems, able to cope with different testing conditions, are built. These systems are subsequently adapted to the particular task on which they are tested. Possible sources of the acoustic mismatch include: speaker variability in terms of speaking styles, accents and physiological characteristics such as vocal tract length, different kinds of transmission channels, different type of microphones and the presence of different noises.

The mismatch between acoustic models and testing data could be simply avoided by training acoustic models on the same kind of data as the testing data: for example training speaker dependent or gender dependent models or if the mismatch is due to the acoustic channel, training a system on the same kind of data as the testing ones, i.e. on data affected by the same kind of noises. Of course the adoption of these kind of techniques may be impossible because, to build speaker dependent models, we would need a large amount of data for each speaker. Moreover this assumes to have the same speakers for training and testing, and this is unacceptable in the vast majority of real applications. Moreover in a real world application we would like to be able to deal with unseen testing conditions.

To deal with the mismatch between the acoustic model $\lambda$ and the testing data $X_{test}$ two main approaches were developed:

- normalisation: the acoustic data are normalised in order to adapt them to the model both during training and testing making them independent for example of speaker specific characteristics or of the transmission channel;

- adaptation: the model parameters are adapted to the acoustic data we want to recognise (to represent them more appropriately).

Figure 3.1 summarises the adaptation and normalisation processes and their interaction. In this figure the mismatch of the training data $X_{Train}$ and the testing data $X_{Test}$ is reduced in two ways: on the left side of the graph $X_{train}$ and $X_{Test}$ are normalised in the feature space; on the right side the acoustic model parameters of $\lambda_{Train}$ are modified to better match the acoustic data $X$ both for the training data $X_{Train}$ (performing adaptive training) and the test data $X_{Test}$ (adaptive recognition) to the model space. The two processes are also combined by applying adaptive training on the normalised acoustic features $\tilde{X}_{Train}$ obtaining $\tilde{\lambda}$ which is both adapted and normalised.

In normalised acoustic modeling we try to cope with speaker specific vocal tract length effects or channel specific effects for example, by normalising both training and testing data during signal analysis, in order to reduce the mismatch between training and testing. In fact normalising only one of them would leave some mismatch between acoustic models and testing data. Vocal Tract Length Normalisation (VTLN) is an example of speaker normalisation techniques and will be introduced in section 3.3.2. Other examples of normalisation techniques aiming mainly to normalise for the transmission channel are Cepstral Mean and Variance Normalisation (section 3.4.3) and techniques aiming more specifically at speaker normalisation such as the use of the Mellin transform and Wavelet based spectral representations (outlined respectively in section 3.4.1 and 3.4.2).

The adaptation approach modifies the acoustic model parameters in order to reduce the mismatch between $X_{Test}$ and $\lambda_{Train}$ transforming $\lambda_{Train}$ into $\lambda_{Test}$. There are two main techniques to do this: the Maximum A Posteriori (MAP) adaptation (Gauvain and Lee, 1994) (outlined in section 3.2.1) and the Maximum Likelihood

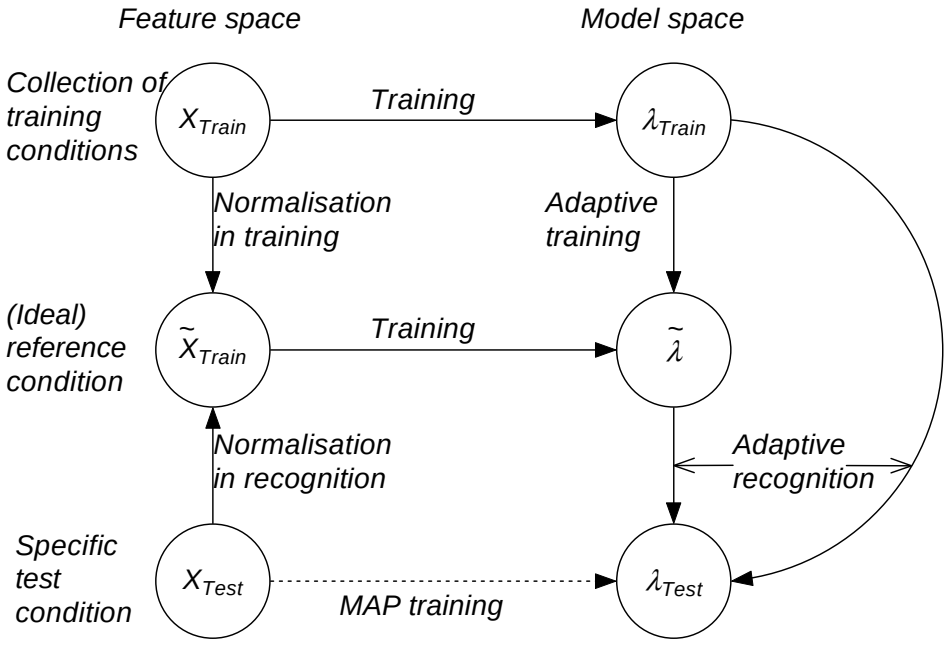



Figure 3.1: Normalisation and Adaptation (based on Pitz (2005))

Linear Regression family of techniques (Legetter and Woodland, 1994, 1995; Digalakis et al., 1995; Gales and Woodland, 1996) (outlined in section 3.2.2).

## 3.2 Adaptation Techniques

### 3.2.1 MAP Techniques

MAP estimation differs from ML estimation in the fact that for ML the parameter set $\lambda$ is assumed to be fixed but unknown while for MAP $\lambda$ is not fixed but is a random variable drawn by a prior distribution $p(\lambda)$; in practice for MAP given $T$ observation vectors $\mathbf{O} = \{\mathbf{o}_1, \mathbf{o}_2, ..., \mathbf{o}_T\}$:

$$\lambda_{MAP} = \arg\max_{\lambda} p(\lambda|\mathbf{O}) = \arg\max_{\lambda} p(\mathbf{O}|\lambda)p(\lambda) \tag{3.1}$$

where: $\lambda$ is assumed to be a random variable from space $\Lambda$ with a probability density function (*pdf*) $p(\lambda|\mathbf{O})$; $p(\lambda)$ is the prior *pdf* of $\lambda$ defined as informative if it is known what the parameters are likely to be. If the prior is not informative the MAP objective function reduces to the ML part only.

As an informative prior, in the case of speaker adaptation, we can choose the parameters of the speaker independent model $\lambda_{SI}$. Then the modification of the speaker independent means $\mu^{SI}_{j,m}$ can be performed by maximum likelihood methods using the adaptation data for each speaker, for each state $j$ and mixture $m$ as follows:

$$\hat{\mu}_{j,m} = \frac{N_{j,m}}{N_{j,m}+\tau}\mu^{ad}_{j,m} + \frac{\tau}{N_{j,m}+\tau}\mu^{SI}_{j,m} \tag{3.2}$$

where $N_{j,m}$ is the occupation likelihood of the adaptation data defined as:

$$N_{j,m} = \sum_{t=1}^{T} \gamma_t(j,m) \tag{3.3}$$

being $\gamma_t(j,m)$ the state occupancy at time t, and $\mu^{ad}_{j,m}$ the mean of the observed adaptation data computed as:

$$\mu^{ad}_{j,m} = \frac{\sum_{t=1}^{T}\gamma_t(j,m)\mathbf{o}_t}{\sum_{t=1}^{T}\gamma_t(j,m)} \tag{3.4}$$

and $\tau$ is a parameter used to weight the a priori knowledge of the adaptation data.

The advantage of MAP adaptation techniques is that increasing the amount of adaptation data the system converges to a speaker dependent one while the disadvantage is that the adaptation can be performed only on the parameters which correspond to the symbols observed in the adaptation data. Unfortunately for LVCSR systems adapting all the parameters would require enormous amounts of adaptation data. Nevertheless it is possible to use the MLLR adapted models as a prior yielding in this way a larger improvement.

### 3.2.2 MLLR Techniques

MLLR techniques compute a set of linear transformations of the means and the variances of a Gaussian mixture HMM system, maximising the likelihood on the adaptation data. Speech sounds are grouped into regression classes using a regression class tree so that they share the same transform. The mean and variance linear transformations can be expressed by:

$$\hat{\mu} = \mathbf{A}\mu + \mathbf{b} \tag{3.5}$$

$$\hat{\Sigma} = \mathbf{H}\Sigma\mathbf{H} \tag{3.6}$$

where $\mathbf{A}$ is an $n \times n$ transformation matrix, $\mathbf{b}$ is a bias vector and $\mathbf{H}$ is an $n \times n$ covariance transformation matrix.

In practice it is not feasible to estimate at the same time the mean transform $\mathbf{A}$ and the variance transform $\mathbf{H}$. Thus we first consider the mean transform estimation problem alone. If we define an auxiliary variable $\xi^{SI} = [1\ \mu^{SI^T}]$ we can estimate one transform $\mathbf{T} = [\mathbf{b}\ \mathbf{A}]$ so that:

$$\hat{\mu}^{ad} = \mathbf{T}\xi^{SI}. \tag{3.7}$$

Then the ML $\mathbf{T}$ transform can be estimated by maximising the likelihood of the observed acoustics $\mathbf{o}_{1:T}$ given the speaker independent acoustic model parameters $\mu^{SI}$ and $\Sigma^{SI}$, the transform $\mathbf{T}$ and the word sequence $w_{1:N}$:

$$\mathbf{T}^{\mathbf{ML}} = \arg\max_{\mathbf{T}} p(\mathbf{o}_{1:T}|\mu^{\mathbf{SI}}, \Sigma^{SI}, \mathbf{T}; w_{1:N}) \tag{3.8}$$

where the transcription $w_{1:N}$ can be either the output of a previous recognition pass (unsupervised adaptation) or a true manual transcription (supervised adaptation).

This optimisation is carried out by maximising an auxiliary function:

$$Q(\lambda, \hat{\lambda}) = \sum_{s_{1:T}\, w_{1:N}} p(\mathbf{o}_{1:T}|w_{1:N}, \lambda) \log p(\mathbf{o}_{1:T}|w_{1:N}, \hat{\lambda}) \tag{3.9}$$

where $s_{1:T}$ is the state sequence and $\hat{\lambda}$ is the updated model parameter set of the optimisation iteration. The variance transformation $\mathbf{H}$ is also estimated by expectation maximisation. It was found that the mean adaptation gives greater improvements than the variance adaptation (Gales and Woodland, 1996). The advantage of the MLLR technique is that it requires less adaptation data compared to MAP. However given a sufficient amount of training data, MAP performs better than a pooled Gaussian transformation approach since it works at the component level.

#### 3.2.2.1 Constrained MLLR

Constrained MLLR (CMLLR) is a special case of MLLR where the transform applied to the means is the same applied to the covariance matrices (Gales and Woodland, 1998):

$$\hat{\mu} = \mathbf{A}\mu + \mathbf{b}, \tag{3.10}$$

$$\hat{\Sigma} = \mathbf{A}\Sigma\mathbf{A}. \tag{3.11}$$

It can also be shown that this is equivalent to a linear transformation of the features so that:

$$\hat{\mathbf{o}}_t = \mathbf{A}\mathbf{o}_t + \mathbf{b}. \tag{3.12}$$

The fact that CMLLR can be directly applied to the feature space has obvious computational advantages (since transforming the acoustic model parameters could be in some cases rather computationally expensive). Furthermore CMLLR is particularly effective when adopted in a speaker adaptive training (SAT) fashion where the CMLLR transforms are estimated for each speaker in the training set and new acoustic models are trained on the adapted features.

While the combination of VTLN and MLLR seems to achieve additive improvements, it was observed that no further word error rate reductions are obtained when VTLN is applied in conjunction with CMLLR with respect to using CMLLR alone (Üebel and Woodland, 1999). In Pitz and Ney (2005) it was also shown that VTLN could be considered a restriction of CMLLR when cepstral coefficients are used (since for these features VTLN frequency warping is equivalent to a linear transformation of the cepstral coefficients).

## 3.3 Vocal Tract Length variability

Vocal tract length is defined as the distance between the lips and the glottis. The configuration of vocal tract has a substantial effect on the observed spectrum: for example, a typical female speaker exhibits formant frequencies around 20-25% higher than those of a male speaker. From infancy to adult age VTL grows both according to the body size and in a different measure according to the sex. Infants have the larynx in the standard, higher, mammalian position, but between the third month and the third year the larynx goes down to the throat, giving rise to a phenomenon known as the larynx descent. Then as children grow up there is a steady increase of VTL with body growth but there is no significant difference in VTL between boys and girls until puberty.

During puberty there is a second larynx descent only for males (Fitch and Giedd, 1999). It was argued that this is the main reason which leads to sex differences in vocal tract length. This confirmed Fant's theory that males have a disproportionately longer VTL than females (Fant, 1966).

VTL is highly correlated with body size but there is another component due to sex differences, although this is only true after puberty. Fitch (1994) also argued that the difference between boys and girls voices (before puberty) can only be due to behavioral and not anatomical differences. In fact boys seem to be able to protrude their lips in order to lower their speech formants to imitate adult males. This is because human listeners are able to use the acoustics of a speaker's voice as a cue for its body size estimation, and speakers are able, at least to a certain extent, to modify their speech acoustics by increasing their VTL using mechanisms such as that of protruding their lips by tensing the obicularis oris muscle on one end, and by lowering the larynx using the laryngeal strap muscles on the other end.

Another factor which influences the acoustics of speech is Glottal Pulse Rate (GPR), mainly determined by mass and length of the vocal folds (perceived as voice pitch). This cue too has proved to be quite important in the differentiation of speech acoustics according to sex because of the growth of human cartilages during puberty in males due to increased testosterone.

According to Smith and Patterson (2005) both VTL and GPR influence the perception of speaker's size and sex. In this study they scaled five English vowels pronounced by a male speaker by re-synthesizing them with different GPR and VTL values using the toolkit STRAIGHT (Kawahara et al., 1999). First of all a GPR independent spectral envelope of the signal was produced using a spectral representation based on the adoption of a pitch adaptive window. Then GPR scaling is realised through expansion and contraction of the time axis while VTL scaling is accomplished by compressing or expanding the speech envelope linearly along a linear frequency axis. These scaled versions with different combinations of GPR-VTL were presented to a group of listeners which had to judge for sex and age. They found that listeners make consistent judgements and both VTL and GPR changes influence them but VTL alone is strong enough to change speaker size judgements even with a steady GPR, while regarding sex and age there is a strong interaction between GPR and VTL. Irino and Patterson (2002) argue that human listeners are in fact able to segregate the information about VTL and VT shape using some kind of normalisation.

### 3.3.1 The effect of VTL on speech acoustics

The source/filter theory of speech explains the dependence of speech acoustics on VTL (Quatieri, 2001). The basic principle can be explained by approximating the vocal tract shape with a uniform lossless acoustic tube (which is a reasonable approximation at least for open vowels such as /aa/) with the closed end represented by the glottis and the open end represented by the lips. Such a wave-guide possesses uniformly spaced resonant frequencies, expressed by the following relationship:

$$F_k = \frac{c}{4L}(2k-1) \;\; k = 1,2,3,... \tag{3.13}$$

where $c$ is the sound speed and $L$ is the uniform tube length. So formant positions are (according to this approximation) inversely proportional to the length of the vocal tract so that a change of the scale by a factor of $\alpha^{-1}$ results in a scaling of the frequency axis by a factor $\alpha$.

### 3.3.2 Vocal Tract Length Normalisation in ASR

The first application of VTLN, dating back to the 1970's, reflects the computational power of the time. In an early vowel identification work Wakita (1977) proposed a method for vowel normalisation which consisted of reestimating formant positions for every vowel as $\hat{F}_i = \frac{l}{l_R}F_i = \alpha F_i$ where $l$ is the estimated VTL for that particular vowel and $l_R$ is the reference length. He found that, representing the vowel spaces in terms of the $F_1 - F_2$, $F_1 - F_3$ and $F_2 - F_3$ planes, the distributions of each vowel resulting from normalisation were more compact.

Cohen et al. (1995) introduced this technique in LVCSR systems reporting that a linear warping of the frequency axis could compensate for differences in VTL, resulting in a speech recognition system with a reduced word error rate (WER).

Over the past 10 years VTLN has become a standard normalisation technique in speaker independent speech recognition, proving particularly effective in the domain of conversational telephone speech (CTS) (Lee and Rose, 1996; Hain et al., 1999; Welling et al., 2002) since this task has long turn sections and the reliable estimation of VTL is not a problem.

In recent speech recognition systems the mismatch due to VTL variability was taken into account by scaling the frequency axis of the observed spectrum with a

warping function $g_\alpha$:

$$f \longmapsto \hat{f} = g_\alpha(f). \tag{3.14}$$

We can classify the various methods used for speaker normalisation in the literature in two ways:

- by the kind of frequency warping used
- by the method used to estimate the warping factor.

The various frequency warping functions $g_\alpha$ which can be adopted will be discussed in section 3.3.3 while the warping factors estimation methods will be described in section 3.3.4. There are 2 main methods for the estimation of warping factors: approaches based directly on speech features and the so-called maximum likelihood methods, outlined in section 3.3.4.1 and 3.3.4.2 respectively.

### 3.3.3 VTLN Frequency Warping functions

Several different warping functions were investigated, including:

- linear warping functions (Eide and Gish, 1996; Zhan and Waibel, 1989; Welling et al., 2002) in the form:

  $$\hat{f} = \alpha f \tag{3.15}$$

  or, as a generalisation, piecewise linear functions where different warping factors $\alpha$ are defined for different frequency bandwidths (shown in figure 3.2(a)).

- non linear warping functions or power functions for example Eide and Gish (1996) (figure 3.2(b)):

  $$\hat{f} = \alpha^{\frac{3f}{8000}} f \tag{3.16}$$

  or (figure 3.2(c)) (Molau et al., 2000):

  $$\hat{f} = \left(\frac{f}{f_N}\right)^{\alpha} f_N \tag{3.17}$$

  where $f_N$ is the Nyquist frequency. Alternatively a bilinear transform was used (Zhan and Waibel, 1989; McDonough, 1998; Dognin, 2004) (figure 3.2(d)):

  $$\hat{f} = f + 2\arctan\frac{(1-\alpha)\sin f}{1-(1-\alpha)\cos f} \tag{3.18}$$

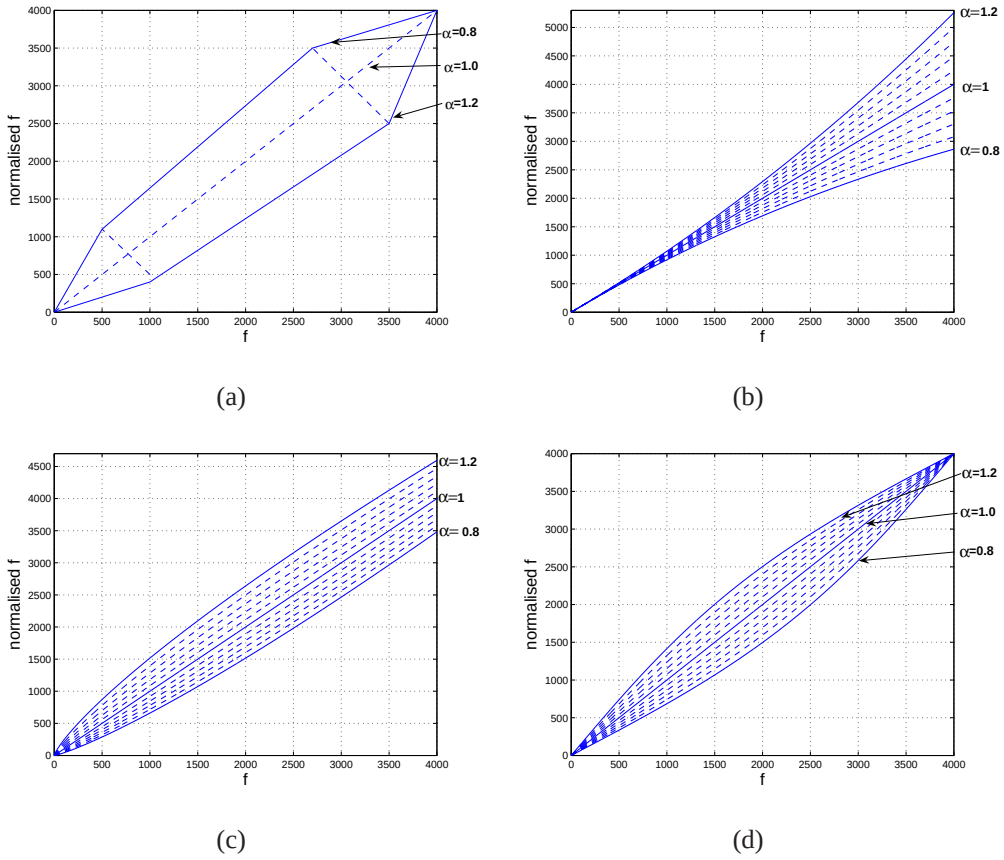



Figure 3.2: Frequency warping functions: (a) piecewise linear, (b) non linear used by Eide and Gish (1996) (eq. 3.16), (c) power function (eq. 3.17), and (d) bilinear function (eq. 3.18)

Piecewise and bilinear frequency warping ensure that $g_\alpha(f_N/2) = f_N/2$ while this is not true for warping functions 3.16 and 3.17.

Very few comparisons between ASR performances due to the use of different warping functions have been reported in the literature. Molau et al. (2000) compared the use of a piece-wise linear with a power function and a combination of both. They found that in their system the piecewise linear function performed slightly better than the other two. In fact a piece-wise linear warping function is a common choice for many systems. Zhan and Westphal (1997) compared piecewise linear and nonlinear warping reporting no significant difference in performances. Eide and Gish (1996) compared linear warping ($f = \alpha f$) with nonlinear observing slightly better performances using nonlinear.

An example of how Mel Frequency Cepstral Coefficients (MFCC) can be computed with a piecewise linear frequency warping function can be seen in Fig. 3.3. In practice frequency warping can even be incorporated in Mel-scaling by varying the spacing and width of the Mel-spaced filters (Lee and Rose, 1996).

### 3.3.4 Warping Factors Estimation Methods

#### 3.3.4.1 Signal Based Techniques

Signal based approaches attempt to estimate the warping factor directly from the acoustic signal, usually from formant positions (Eide and Gish, 1996; Claes et al., 1997).

For example Eide and Gish (1996) estimated warping factors as the ratio of the median third formant value for a particular speaker and the median of $F_3$ for all the speakers in the training set. Here the $F_3$ values to be included in the median computation were filtered using a criterion based on the voicing probability, the $F_1$ value, and of course the $F_3$ range. The improvement reported in speech recognition for the Switchboard task was about 10% relative WER reduction in the case of non-linear warping when both the test set and the train set were normalised.

Wegmann et al. (1996) used a piecewise linear frequency warping and the warping factors were selected using a generic voiced speech model. This model is a single probability distribution and is obtained with an iterative procedure alternating the estimation of the best warping factor for each training speaker and the use of the warped data to train a new model until the average score per speaker against the generic speech model was minimised. This method has the advantage of not requiring a first pass decoding as ML does (as will be seen in the next section) but at the same time it does not use formant positions directly as in Eide and Gish (1996).

#### 3.3.4.2 Maximum Likelihood Methods

In ML approaches (Lee and Rose, 1996; Hain et al., 1999; Welling et al., 2002) the speaker-specific warp factor $\alpha$ is usually obtained by maximising the likelihood of the normalised acoustic observation $X^{\alpha}$, given a transcription $W$ and an acoustic

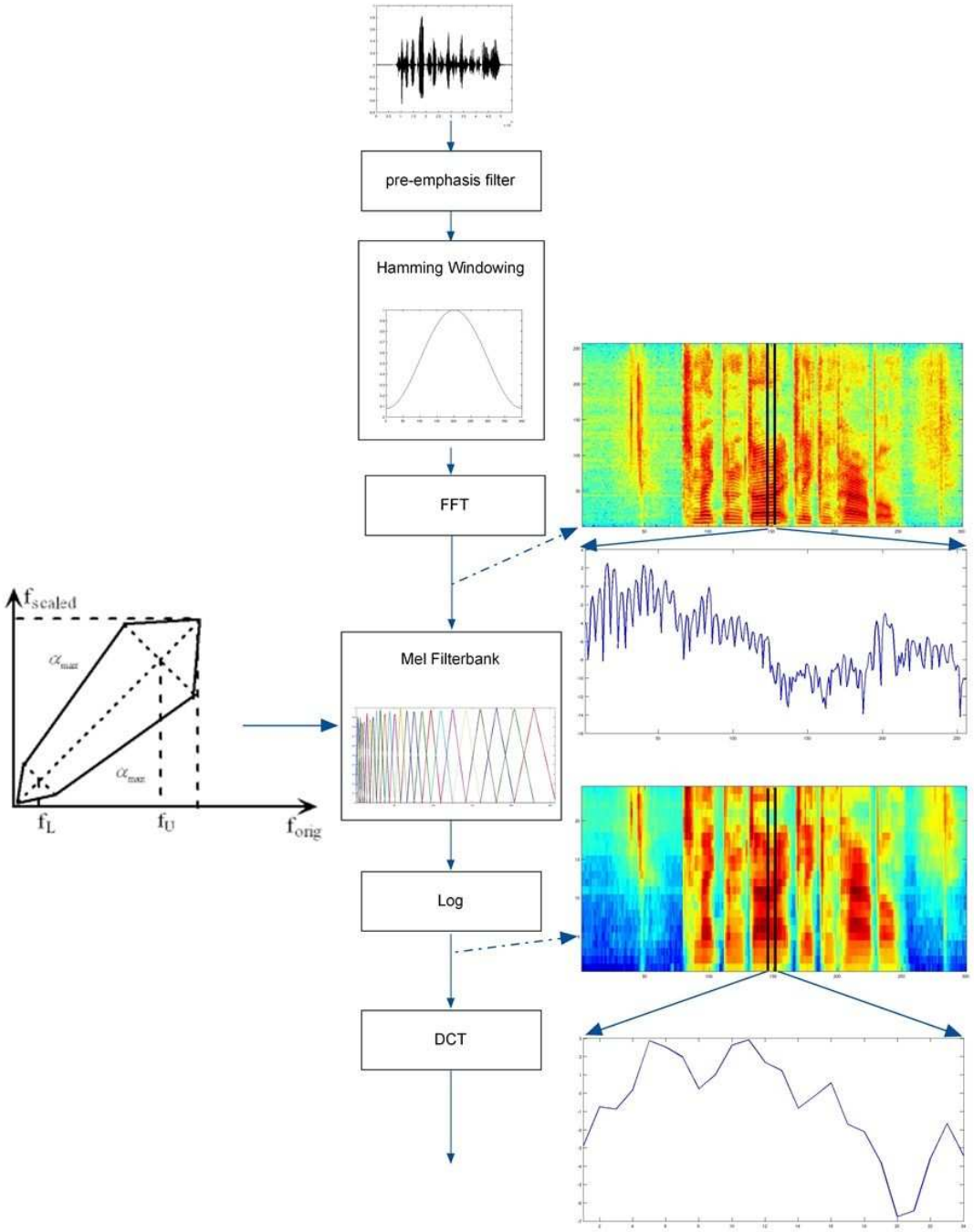



Figure 3.3: Front-end for VTLN for MFCC computation where the piece-wise linear warping is just an example of one of the possible frequency warping

model $\lambda$, so that the following equation has to be solved:

$$\alpha = \arg\max_{\alpha}(P(\mathbf{X}^{\alpha}|\mathbf{W},\lambda)). \tag{3.19}$$

Since:

$$P(\mathbf{X}^{\alpha}|\mathbf{W},\lambda) = \int p(\mathbf{X}^{\alpha}|\mathbf{W},\lambda)d\mathbf{X}^{\alpha} = \int p(f_{\alpha}(\mathbf{X})|\mathbf{W},\lambda)\left|\frac{\partial f_{\alpha}(\mathbf{X})}{\partial \mathbf{X}}\right| d\mathbf{X} \tag{3.20}$$

(where $f_{\alpha}(\mathbf{X})$ is the transformation applied to the acoustic observation and in particular to the features) it follows that the probability density function of $\mathbf{X}$ given a warping factor $\alpha$, a model $\lambda$ and a transcription $\mathbf{W}$ can be expressed by the following relationship:

$$p(\mathbf{X}^{\alpha}|\mathbf{W},\lambda) = p(f_{\alpha}(\mathbf{X})|\mathbf{W},\lambda)\left|\frac{\partial f_{\alpha}(\mathbf{X})}{\partial \mathbf{X}}\right| \frac{d\mathbf{X}}{d\mathbf{X}^{\alpha}} \tag{3.21}$$

where the middle term is the Jacobian determinant of the transformation. This factor has to be taken into account when the probabilities for different values of the warping factors need to be compared such as in the case of ML approaches. The importance of the Jacobian is connected to its dependence on $\alpha$. Many works have neglected it (i.e. Lee and Rose (1996); Zhan and Waibel (1989); Welling et al. (2002)), mainly because its effect was believed to be small. Furthermore it is quite difficult to estimate the Jacobian if the frequency warping is performed directly during signal analysis because the transform function $f_{\alpha}$ between non normalised and normalised features needs to be estimated. Pitz (2005) analysed the effect of the Jacobian determinant for VTLN. Pitz proved that VTLN is equivalent to a linear transformation of the MFCC feature vectors so that $\mathbf{X}_{\alpha}$ can be expressed as:

$$\mathbf{X}^{\alpha} = f_{\alpha}(\mathbf{X}) = \mathbf{A}\mathbf{X} \tag{3.22}$$

where $\mathbf{A}$ is a transformation matrix and the determinant of the Jacobian is just the determinant of $\mathbf{A}$. Given this assumption he found that although taking into account the Jacobian has a substantial effect on the distribution of the computed warping factors, in the case of piecewise linear frequency warping the Word Error Rate of the resulting ASR system does not seem to be particularly influenced.

According to equation 3.19 a transcription is needed in order to estimate warping factors. During testing, in order to obtain a preliminary transcription, a simple two pass approach can be adopted (Welling et al., 2002; Hain et al., 1999). A first

pass decoding with non-normalised models and features is performed in order to use the transcription for warping factor estimation, and finally a second pass decoding with normalised models and features is done.

Hain et al. (1999) presented a technique to train normalised models where an iterative procedure which alternates frequency warping factors estimation and training passes is used. The use of normalised models helps to reduce even more the mismatch between testing data and acoustic models.

Lee and Rose (1996) addressed the problem of ML efficiency using a maximum likelihood approach during training and a mixed approach during testing. For training they basically subdivided the training set in two subsets, an alignment set $A$ and a training set $T$; then they first train a model using the set $T$ and then find the optimal warping factor for each speaker in $A$ using that model; subsequently the sets are swapped and the process is repeated until the warping factors do not change significantly from one iteration to another. Finally a new normalised model is trained using all the normalised training data.

Lee and Rose pointed out that a two pass approach for testing is not ideal from an efficiency point of view because it requires two decoding passes. In fact they used a different procedure to estimate the warping factors in the testing phase. After the warping factor estimation on the training data they pooled all the data with the same $\alpha$ and trained for each warping factor a GMM $\theta$ using the unnormalised acoustic vectors. Then during testing the unnormalised acoustic vectors were scored against each GMM model to find the best $\alpha$ according to the following equation:

$$\hat{\alpha} = \arg\max_{\alpha} P(\mathbf{X}|\theta_{\alpha}). \tag{3.23}$$

In this way no first pass decoding is needed and at the same time the estimation is independent from formant tracking.

Molau et al. (2000) compared performances obtained estimating warping factors with a two pass procedure and Lee and Rose's more efficient approach reporting that there is no significant degradation in WER using the latter. In other words Lee and Rose's approach seems to have performances comparable to the two pass procedures without requiring any transcription at all for the estimation of the warping factors.

Although the ML approach is computationally expensive, it is robust and consistent with the overall optimisation of the speech recogniser, since it maximises the

likelihood—something not guaranteed by signal based approaches. Furthermore, the estimation of formant positions relies on voiced segments only and this can be challenging with conversational natural speech (Zhan and Waibel, 1989) because it requires an accurate alignment or a good probability of voicing estimation, whereas ML does not have the same requirement.

Only few works report comparisons between ML and signal based approaches performances. For example Zhan and Westphal (1997) investigated the estimation of warping factors computing them simply as:

$$\alpha_s = \frac{\bar{F}_{k,s}}{\bar{F}_k} \quad k = 1,2,3 \tag{3.24}$$

where $\bar{F}_{k,s}$ is the mean formant $F_k$ for a speaker $s$ and $\bar{F}_k$ is the mean formant for the whole training corpus. This was compared to ML estimation as well. Although ML seems to give the best performances in average none of the two methods seem to be consistently better for all speakers.

Estimating $\alpha$ by ML increases the matching score with the acoustic models, thus making the warping factor very model dependent. Moreover, the estimated warping factor is stable only when a considerable amount of data is available. This is well matched to tasks such as CTS where homogeneous speaker sides are available for every speaker, but it is an issue to be addressed for domains such as meetings or broadcast news (Kim et al., 2004a; Garau et al., 2005), where the amount of data per speaker varies consistently.

ML estimation of VTLN warping factors only indirectly normalises the spectrum to account for VTL: there are other factors (such as systematic pronunciation variation) which may also be normalised by spectral warping.

Furthermore Miguel et al. (2005, 2008) pointed out that using a unique warping function for every utterance (which is the minimum entity for which a warping factor can be estimated using ML techniques) is not appropriate because not all phonetic events have the same spectral variation as a consequence of vocal tract shape differences. Therefore they propose to expand the bi-dimensional trellis (HMM state space and observation space), adopted by Viterbi decoders, including a third dimension with all the N possible frequency warping factors. This augMented stAte space acousTic dEcoder (MATE) allows to have a different frequency warping for every frame with the added constraint of a smoothed transition between adjacent

frames. A second constraint is set on the HMM non-speech models so that the observation vectors associated to them cannot be "warped". Using this technique they get an improvement compared to "classic" VTLN. One of the biggest advantages of this approach is that it works in a single decoding pass, differently from ML VTLN which requires two passes. It has to be mentioned that the experiments they performed were on small vocabulary tasks (digits) and extending a LVCSR decoder would be challenging from a computational point of view.

## 3.4 Other Speaker Normalisation methods

Most of the works on VTLN simply applied frequency warping to the magnitude spectrum right before Mel frequency scaling is performed in the classical MFCC computation as depicted in figure 3.3. However finding a representation of speech independent of the VTL effect could be more appropriate than post-hoc frequency warping. In fact, both in the case of ML and parametric estimate, the warping factor values are context dependent and influenced by noise, making it difficult to obtain a reliable estimate.

In this context experiments were performed both using speech representations based on the Scale transform of the spectrogram (Umesh et al., 1999), making, in theory, the spectrogram independent on the VTL, and adopting a wavelet analysis (Mertins and Rademacher, 2005) instead of a time-frequency representation, such as the short time Fourier transform. We shall also outline other widely used speaker/channel normalisation techniques such as Cepstral Mean and Variance Normalisation (which aim to normalise for the transmission channel as well as for the speaker).

### 3.4.1 Mellin transform derived spectral representations

The Mellin transform applied to a spectrum has the property to make it insensitive to the scaling of the frequency (Irino and Patterson, 2002). This transform was mainly used in pattern recognition for image processing and radar and sonar signal processing because of its scale invariance property.

The Mellin transform of a function $f(t)$ is expressed by the relation:

$$S(p) = \int_0^\infty f(t)t^{p-1}dt \tag{3.25}$$

and it can be proved that if two functions $f(t)$ and $g(t)$ exist such that $f(t) = g(kt)$ where $t >= 0$ and $k$ is a non zero constant, the Mellin transforms of the two functions have the following relationship:

$$S_g(p) = k^{-p}S_f(p) \tag{3.26}$$

$S_g$ and $S_f$ have the same magnitude apart from a scale factor $|k^{-p}|$. It is hypothesised that the Mellin transform is similar to human processing of vowels segregating scale information from the actual structure information (Irino and Patterson, 1999).

In particular if $p = -jc + \frac{1}{2}$, then the Mellin transform is termed the Scale transform (Umesh et al., 1996), while if $p = -jc$ we have:

$$D(f(t)) = S(-jc) = \int_0^\infty f(t)t^{-jc-1}dt = \tag{3.27}$$
$$= \int_0^\infty f(t)\frac{e^{-jc\ln t}}{t}dt = \int_{-\infty}^\infty f(t)e^{-jc\ln t}d(\ln(t)) \quad .$$

First from this formulation it can be noticed that a scaling transformation does not change the magnitude but just brings a phase transformation. Second it can be seen that the Mellin transform is just a Fourier transform of the exponentially resampled continuous time signal (Irino and Patterson, 2002; Sena and Rocchesso, 2004). The actual resampling can be obtained by interpolation or, if signals sampled at a higher frequency rate are available, it would be possible to downsample according with the exponential axis.

The use of the Mellin transform for speech recognition feature extraction was investigated by Chen et al. (1998). In this work Mellin derived features were obtained applying a modified version of the Mellin transform to the log spectrum and then using the Discrete Cosine Transform (DCT) to decorrelate the Mellin spectrum. The modified Mellin transform is expressed by:

$$S^M(p) = p \cdot S(p). \tag{3.28}$$

With these features they obtained a relative error reduction of 26% with respect to MFCC. Moreover they found a significant reduction in the standard deviation

of the WER due to the fact that the features are not any more speaker dependent. Unfortunately this approach was not compared with standard VTLN techniques.

The use of the scale transform was investigated by Umesh et al. (1999) who derived scale-cepstrum features as the Scale transform of the logarithm of the magnitude of the spectrum. These features were compared with mel-cepstrum features, resulting in a better separability for vowels, but no ASR results were reported.

Irino and Patterson (2002) have suggested that VTL information can be extracted directly, and have proposed an auditory-inspired transform which separates VTL size from shape information. This account was supported by some recent perceptual experiments (Smith et al., 2005), which provide evidence for the hypothesis that the auditory system automatically normalises for VTL when processing speech or other vocalised sounds. They applied the Mellin transform to the so called Stabilised Auditory Image (SAI), a particular kind of spectral analysis based on the use of a "gammachirp" auditory filterbank, resulting in the so called Mellin Images (MI) which allowed to extract the shape information associated with a given vowel class across different VTLs (Irino and Patterson, 2002).

### 3.4.2 Wavelet based methods

The use of the wavelet transform to obtain vocal tract length invariant features was investigated by Mertins and Rademacher (2005) where these features were also compared and combined both with conventional MFCCs and scale transform derived features. The combination, performed using linear discriminant analysis applied on the concatenated feature vectors, gave improved accuracy on a phoneme classification task. The wavelet transform of a continuous time signal $x(t)$ is defined by:

$$W_x(t,s) = \frac{1}{\sqrt{s}} \int_{-\infty}^{\infty} x(t) \psi^* \left( \frac{\tau - t}{s} d\tau \right) \tag{3.29}$$

where $\psi(t)$ is the mother wavelet and $s$ is the scaling parameter. It can be demonstrated that the wavelet transform of a linearly frequency warped signal $x_\alpha(t) = \frac{1}{\sqrt{(\alpha)}} x(\frac{t}{\alpha})$ is related to the wavelet transform of $x(t)$ by:

$$W_{x_\alpha}(t,s) = W_x \left( \frac{t}{\alpha}, \frac{s}{\alpha} \right) \tag{3.30}$$

which in the $\log(s)$ domain is basically a translation by $\log(\alpha)$ and when a Fourier transform is applied translates in a phase factor which has no effect on the magnitude but only a time scaling effect:

$$F_\alpha(t,\mu) = e^{-j\mu\log(\alpha)} F\left(\frac{t}{\alpha},\mu\right) \tag{3.31}$$

A wavelet transform has also been investigated for ASR cepstral coefficient extraction by Wassner and Chollet (1996) because of its property of yielding an optimal time-scale resolution, since on one side it provides good time and poor frequency resolution at high frequencies and on the other side good frequency resolution and poor time resolution at low frequencies.

### 3.4.3 Cepstral Mean and Cepstral Variance Normalisation

Cepstral Mean Normalisation (CMN) and Cepstral Variance Normalisation (CVN) (Molau et al., 2003) are two normalisation techniques which aim to reduce the distortions due to the transfer channel through which the speech signal is transmitted. The effect of the channel (assuming a linear time-invariant one) can be seen as a filter $h(t)$ in the time domain which is convoluted to the input signal $s_i(t)$. In the frequency domain this translates in a multiplication such that:

$$s_o(t) = h(t) * s_i(t) \tag{3.32}$$

$$S_o(\omega) = H(\omega) \cdot S_i(\omega) \tag{3.33}$$

During feature extraction (i.e. of MFCCs) the logarithm is usually performed on the Mel spectrum before the DCT is applied. Thus the multiplication of the channel transfer function in the frequency domain is transformed into a summation. Therefore channel compensation is performed by subtracting the mean over time of the cepstral coefficients (which in fact represents the channel effect) from the cepstral coefficients. This is what is called CMN. It is also useful to normalise the variance of the cepstral coefficients by CVN especially in noisy conditions.

Moreover the channel effect can be actually subdivided in one part due to the transmission channel $C(\omega)$ and another one due to the current speaker who uttered the speech $V(\omega)$, thus the transfer function can be expressed as $H(\omega) = C(\omega) \cdot V(\omega)$ and CMN and CVN normalise both for the speaker and the channel effects. It

was also found that best results are obtained when CMN is performed at a speaker granularity rather than on a per utterance basis (especially when short sentences were uttered) (Westphal, 1997).

Another widely used approach is histogram normalisation (Molau et al., 2001, 2003; Hilger and Ney, 2006; Haverinen and Kiss, 2003; Droppo et al., 2005). This techniques aims to reduce the mismatch between the distribution of the test data and that of the training data. It is assumed that enough data for the estimation of the distribution are available. In this technique the test data are linearly transformed as in the following equation:

$$\tilde{Y} = P_{train}^{-1}(P_{test}(Y)) \tag{3.34}$$

where $P_{train}$ is the cumulative distribution function (CDF) of the training data and $P_{test}$ is the CDF of the test data while $Y$ and $\tilde{Y}$ are the test data respectively before and after the histogram normalisation. When the amount of available data is not sufficient to estimate $P_{test}$, quantile based histogram normalisation can be adopted (Hilger and Ney, 2006), which basically uses an approximated CDF independent from the amount of data available. In Molau et al. (2003) it was shown that CMN, histogram normalisation and VTLN are complementary and they can be used together to reach the best performances especially in noisy conditions.

## 3.5 Conclusions

In this chapter the main techniques of adaptation (MAP and MLLR) and normalisation (VTLN, CMN and CVN, histogram normalisation, and Mellin and Wavelet transform based methods) were outlined with a particular focus on VTLN. Normalisation and adaptation techniques aim both at reducing the mismatch between the training and the testing conditions. Normalisation techniques mainly act on the acoustic data by normalising it with respect to the acoustic channel (CMN and CVN) or speaker specific characteristics such as vocal tract length (VTLN). Adaptation techniques on the other hand act on the acoustic model parameters, modifying them with the goal of reducing the mismatch between the acoustic models and the specific testing data. In the particular case of CMLLR the adaptation can be performed either on the acoustic model parameters or equivalently on the acoustic data themselves, making this technique an hybrid between normalisation and adaptation.

In this thesis we focused on the investigation of speaker normalisation techniques such as VTLN performing our experiments in conjunction with CMN and CVN in the context of large vocabulary conversational speech. While experiments using MAP and MLLR were not considered in this thesis (being these adaptation techniques), experiments on the use of CMLLR were not performed because, as mentioned in section 3.2.2.1, it was found by Üebel and Woodland (1999) that no further improvements are found when CMLLR is performed in conjunction with VTLN.

# Chapter 4

# Automatic Speech Recognition of multiparty meetings

## 4.1 Introduction

Meetings are a rather unconstrained domain for automatic speech recognition, due to the high variability in terms of: acoustic conditions; speaking style; overlapping between speakers; speaker accent, age, and gender; and topics. The meeting type may also vary quite consistently: they can be both in the form of a lecture where a single speaker presents a particular topic to a small/large audience and a little discussion may follow, or a conference where people meet around a table to discuss several topics. This variety makes meetings an interesting domain for each step of the speech recognition process.

Recording conditions are the main challenge for the preprocessing step. During meetings users are confined within a meeting room but acoustic conditions may vary from one room to another (or even in the same room) because of the position of speakers, microphones and even the furniture arrangement. Besides reverberation and noise, both vocal sounds (e.g. cough, breath, and cross-talk) and non-vocal sounds (such as noise from laptops or from the street) constitute a serious problem. The number, the positions and the quality of the microphones may differ.

Most of the available meeting corpora were recorded both using Independent Headset Microphones (IHM) and Multiple Distant Microphones (MDM). While ASR on IHM is relatively more constrained, the presence of vocal and ambient

noise, reverberation and crosstalk[1] poses challenging problems. These are even more accentuated for the MDM condition which offers other challenges as well. On the MDM condition the exact location and configuration of the microphones is unknown. Moreover in this domain the speech segments need to be attributed to a given speaker in order to apply speaker normalisation and adaptation techniques. All these issues are addressed by the preprocessing part of the ASR usually referred to as the front-end. The U.S. National Institute of Standards and Technology (NIST) set MDM as the main testing condition for its Rich Transcription of meetings evaluations (NIST, 2004). The use of unobstrusive distant microphones for meeting recording and transcription is in fact a challenging but interesting domain because users prefer not to wear headset microphones.

Another important issue for meeting speech recognition is the relatively limited availability of data for this specific domain. Extensive corpora of conversational telephone speech (CTS) are available and they have proven to be rather useful to reach low word error rates for Large vocabulary Continuous Speech Recognition (LVCSR) systems (Evermann et al., 2005). For this reason many systems for the meeting domain are adapted from models trained using CTS data (Stolcke et al., 2004; Hain et al., 2005c, 2007a). In fact CTS recordings have similarities to meeting data being natural conversational speech.

Meetings can feature a large variety of topics and rather rich vocabularies thus the acquisition of suitable data for Language Modeling (LM) is also a relevant task. Moreover conversational speech is rather rich in hesitations, backchannel and fillers and fully unconstrained in terms of style and lexical register. Therefore sourcing large amounts of natural conversational speech for the LM training can be a challenging task.

Meetings offer an interesting domain from a speaker adaptation and normalisation point of view too. Speaker variability includes:

- speaking style, mostly affecting the language model;
- a wide range of accents, including native and non native speakers (with a sparse distribution of native languages) and various dialectal inflections affecting the pronunciation and therefore the acoustics as well;

[1] Sometimes evident in low quality microphone recordings such as for example when lapel microphones are used.

- wide demographics (age and gender) which heavily affect speech acoustics;
- large variation in the amount of speech available for each individual speaker.

Adaptation techniques such as MLLR and MAP can be used to tackle speaker's variation in pronunciation and articulation, while speaker normalisation techniques such as VTLN can be applied to normalise for the speaker specific vocal tract length. These techniques have proven to be particularly effective in the meeting domain especially when applied during training and testing.

Building a state-of-the-art large vocabulary speech recognition system targeted on meeting data is a rather challenging task which requires a significant effort both in terms of human and computational resources. This chapter will outline the various components of such a system taking as example the infrastructure I have contributed to develop as a member of the AMI ASR team. My key contributions to this system were in speaker normalisation and adaptation, however the whole system was the result of an extensive multi-site team-working effort involving a close collaboration between 8-10 researchers specialised on different sub-fields over the course of several years (Hain et al., 2005c,a,b, 2006, 2007b). During this time we also participated to the NIST meeting recognition evaluations in 2005, 2006 and 2007.

The overall structure of this chapter is as follows: in section 4.2 a description of the corpora used for training and testing meeting speech recognition systems will be provided, while in section 4.5 the NIST meeting evaluations will be briefly described; section 4.3 will outline the structure of an LVCSR system for meeting recognition and the various blocks will be briefly introduced in the following subsections; the approaches used for the development of the AMI ASR dictionary will be introduced in section 4.3.1; the data and the methodologies used for language modeling in the AMI ASR system will be outlined in section 4.3.2; automatic segmentation and MDM preprocessing will be outlined in section 4.3.3 and 4.3.4 respectively; acoustic modeling techniques adopted for the AMI ASR system development, with particular attention to the approaches used in this thesis, will be reported in section 4.3.5 with a brief overview of speaker normalisation and adaptation techniques; section 4.4 will introduce the ASR system combination techniques used in the experiments of chapter 6 and 7.

## 4.2 Data Resources

The experiments presented in this thesis were performed on three domains using the Wall Street Journal corpus WSJCAM0, conversational telephone speech and meeting data. Although the kind of speech provided by the WSJCAM0 corpus, being read clean speech, is far from meetings from a language modeling and acoustic modeling point of view, it provides a large vocabulary domain with a good trade off to perform experiments on new acoustic features in a reasonable time. As mentioned in section 4.1 conversational telephone speech, providing a large amount of conversational speech is an interesting domain both on its own and in order to exploit CTS models to adapt them to the meeting domain, which is our main interest.

### 4.2.1 The WSJCAM0 corpus

The WSJCAM0 corpus, recorded at Cambridge University in 1993, consists of native British English read speech (Robinson et al., 1995). Sentences were selected from the Wall Street Journal (WSJ0) text corpus and recorded in an acoustically isolated room with head-mounted microphones (sampled at 16 kHz). The training part of this corpus (si_tr) consists of 7861 utterances, corresponding to around 15 hours of speech, spoken by 39 female and 53 male speakers. We tested on the 20 000 words "open vocabulary" task development set (si_dt20a) which has 10 female and 10 male speakers.

### 4.2.2 Conversational Telephone Speech data

CTS is one of the richest domains for large vocabulary speech recognition providing rather large amounts of training data, including: the Switchboard–1 (Godfrey et al., 1992) and Switchboard–2 corpora which were originally recorded by Texas Instruments and LDC respectively and consist of two-sided telephone conversations by speakers from around the U.S. on various topics; Switchboard Cellular, mainly focused on GSM cellular phone calls; in the Call Home corpus speakers called family members or close friends (this corpus has been collected for various native languages such as: American-English, Egyptian-Arabic, Spanish, German, Mandarin and Japanese) ; and finally Fisher (Cieri et al., 2004), the largest one, including

| Dataset name | All - Tot. (F/M) | Switchboard–1 | Switchboard–2 | Call Home |
|---|---|---|---|---|
| ctstrain04 | 277h(145h/132h) | 248h(126h/122h) | 15h(8h/7h) | 14h(11h/3h) |
| ctstrain04sub | 71h(37h/34h) | 56h(29h/27h) | 8h(4h/4h) | 7h(4h/3h) |
| Dataset name | All - Tot. (F/M) | Switchboard–1 | Switchboard–2 | Cellular |
| NIST hub5 eval01 | 6h | 2h | 2h | 2h |

Table 4.1: CTS dataset statistics

2000 hours of speech from a variety of accents and English proficiency and a large variability in topics and speakers (shorter conversations were preferred compared to Switchboard and Call Home).

In this thesis experiments on the Conversational Telephone Speech task were performed training on two different sets. The larger set, defined as *ctstrain04*, consists of a total of 277 hours of speech, and the smaller one, a subset of the former, consists of a total of 71 hours of speech (all sampled at 8 kHz). Both sets have a good balance between female and male speakers (as can be observed in table 4.1 where the amount of speech for female and male speakers has been indicated in red and blue respectively). Moreover both sets comprise data from 3 different subsets: Switchboard–1, Switchboard–2 and Call Home English, all consisting of two-sided telephone conversations from different areas of the United States. While the experiments on CTS described in chapter 5 were performed training on *ctstrain04*, those described in chapter 6, were based on the *ctstrain04sub*.

Our test set for the CTS task is the NIST Hub5 Eval01 evaluation set[2] consisting of approximately 6 hours of speech in total, equally distributed between Switchboard–1 (SW1), Switchboard–2 (S23) and Switchboard-cellular (Cell), comprising 60 male and 60 female speakers.

### 4.2.3 Multiparty meeting data

In the following sections the most relevant corpora for the automatic speech recognition of meetings will be described.

[2] `http://www.nist.gov/speech/tests/ctr/h5_2001/index.htm`

#### 4.2.3.1 The two phases of the NIST meeting room corpus

The NIST meeting corpus was collected in the NIST Meeting Data Collection Laboratory in two phases (Garofolo et al., 2004; Mitchel et al., 2006). While the first phase consists of 15 hours of meetings all recorded in a conference configuration (using a single conference table), the second phase consists of 20 hours of speech recorded in two other configurations as well: classroom in the form of lectures where the student tables are placed opposite to the teacher's table, and discussion configuration where four tables are configured in a U shape. Moreover for both phases they recorded both video and audio with 5 cameras and 200 microphones respectively. Speakers wore both a headset and a lapel microphone, 4 microphones were placed on the table and 3 microphone arrays consisting of 59 microphones were positioned on the walls.

Speakers were chosen with a reasonable balance both between native and non-native English speakers and between male and female speakers. They recorded both real meetings (those which would have happened anyway) and scenario ones (where an artificial task was assigned to the participants) and the nature of the meetings varied quite significantly ranging from formal and structured meetings such as staff meetings to very interactive and collaborative meetings such as interactive game playing meetings.

#### 4.2.3.2 The ICSI meeting corpus

This collection of 75 meetings (72 hours of speech), was recorded at the International Computer Science Institute in Berkeley (Janin et al., 2003) with an average of 6 participants per meeting (maximum 10). The recording settings, audio only, consist of an individual headset microphone for each participant and six tabletop distant microphones of various quality (from omni-directional to a PDA), four of which were arranged in a staggered line on the table.

These meetings are weekly group meetings which would have occurred anyway on technical topics such as natural language processing (Even Deeper Understanding meetings), the ICSI meeting corpus (the Meeting Recorder meetings), robust speech recognition (the Robustness meetings), and internet architectures and networking (the Network Services group meetings). They also offer a variety of native

and non-native speakers with various proficiency levels. This meeting collection has become one of the most studied data resources for meeting speech recognition.

#### 4.2.3.3 The ISL meeting recordings

This is a collection of 100 meetings (approximately 100 hours of which about 50% of the data has been transcribed) collected at the Interactive System Labs of CMU, Pittsburgh. They had 3–8 participants with an average of 6 per meeting (Burger et al., 2002). Each speaker wore a lavalier microphone and they also used table microphones. Acoustic conditions were not particularly good in these meetings because the room was subdivided by two carpeted walls from the rest of a large room (which was in fact a lab). Three video cameras were placed in the room as well.

The most interesting feature of this corpus is the variety of meeting scenarios: project/work planning, work meetings where a specific project is discussed; military block parties where military personnel performs strategic exercises pretending to be in combat; sessions where the meeting group was given a particular game-like task; chatting where people where left free to chat, gossip and discuss common interests; and discussion where a particular topic was assigned to the group in the form of journal articles, video documentaries etc. These meetings consist of native and non-native speakers covering a wide age range.

#### 4.2.3.4 The AMI meeting corpus

The AMI meeting corpus (Carletta et al., 2005)[3] consists of a multimodal collection of 100 hours of meetings recorded in three instrumented meeting rooms at Edinburgh, IDIAP and TNO. The recording settings were similar across all these rooms which were instrumented with a set of synchronised devices, including lapel and headset microphones for each participant, an 8-element circular microphone array placed at the table centre, 6 video cameras (4 close-up, 1 for each participant, and 2 room-view), and capture devices at the data projector, the white board, and the handwritten notes of each participant (using digital pens).

This corpus is subdivided into scenario (about two thirds) and non scenario

[3]The annotated corpus is freely available from `http://corpus.amiproject.org`

meetings: the scenario ones are elicited meetings where a product development project has to be brought from kick-off to completion and consist of series of 4 meetings (project kick-off, functional design, conceptual meetings and detailed design meetings) where each of the 4 participants plays a prescribed role (project manager, marketing manager etc); the non scenario meetings are real meetings with 3–5 participants. Furthermore the annotation of the corpus includes several levels: orthographic transcriptions, dialogue acts, summarisation, head and hand movements, and focus of attention.

#### 4.2.3.5 Other resources

Other meeting corpora (which were not used in the experiments of this thesis) include the M4 meeting data, the VACE multimodal meeting corpus and the CHIL seminar data. The M4 meeting data (McCowan et al., 2003), recorded in the IDIAP smart meeting room, consists of 5 hours of multichannel audio-visual meeting data. These meetings were recorded in a similar setup to the AMI meetings: each participant wore a lapel microphone and an eight-element circular microphone array was also placed in the center of a rectangular table. Moreover 3 closed circuit television cameras were placed on the walls. These 4 people meetings were scripted in the sense that the sequence of meeting actions (such as for example monologue, discussion, presentation, and note-taking) has been pre-generated using an ergodic Markov model.

The VACE corpus (Chen et al., 2006), recorded at the Air Force Institute of Technology, is a collection of wargames and military meetings with 6 participants. Sensors included headset microphones for each speaker, a set of tabletop microphones and a stereo calibrated camera pair for each participant.

Finally the CHIL data were collected in the context of the Computers in the Human Interaction Loop European project (Chu et al., 2005; Mostefa et al., 2007) in the Smart Room at the University of Karlsruhe, Germany. This corpus consists of 5 hours of technical seminars (12 in total) given by students with a variety of English fluency. They were recorded using both close talking and far field microphones (2 linear 8 channel microphone arrays and one 64-channel Mark III microphone array) and they were also provided to NIST for the Rich Transcription evaluations.

| Dataset name | All - Tot. (F/M) | ICSI | NIST | ISL | AMI |
|---|---|---|---|---|---|
| icsinistislami05 | 106h(28h/78h) | 67h(14h/53h) | 13h(5h/8h) | 9h(5h/4h) | 16h(3h/13h) |
| rt04seval | 99 min | 25 min | 24 min | 26 min | 24 min |

Table 4.2: Meeting data statistics

#### 4.2.3.6 Meeting training data

For meetings the training set adopted in this thesis, which was the same used for the AMI-ASR systems in the NIST RT05 and RT06 evaluations (Hain et al., 2007b; Fiscus et al., 2006), consisted of a total of over 100 hours of conversational meeting speech (sampled at 16 kHz) from four corpora of multiparty meeting recordings: 67 hours from the ICSI corpus, 13 hours from the NIST corpus, 9 hours from the CMU-ISL corpus and 16 hours from the AMI corpus, with 115 male and 49 female speakers. More detailed statistics about this data can be seen in table 4.2 where it can be also noticed that unfortunately there is an unbalanced distribution of female and male speakers.

## 4.3 ASR/LVCSR infrastructure

Figure 4.1 shows the overall training process of an LVCSR system. The first block in the acoustic modeling part is preprocessing. This step (described in more detail in section 4.3.3 and particularly for MDM preprocessing in section 4.3.4) has different functionalities during training and testing. During training it consists essentially in the use of speech enhancement techniques such as echo cancellation, noise cancellation and beamforming to improve the quality of the speech acoustic signal (manual segmentation is used in this phase). During testing preprocessing also involves automatic segmentation.

Acoustic features are extracted from the enhanced signal and normalised using cepstral mean and variance normalisation. The type of features used in the AMI ASR system is discussed in section 4.3.5. Acoustic features are used together with the manual segmentation (the utterance boundaries), the normalised transcription and the dictionary to train acoustic models using standard procedures. First monophone models are trained, then tied-state cross-word models are bootstraped by

initialising them from monophones, and then more accurate tied cross word models can be trained by initialising from cross-words. This procedure can be iterated until convergence of the WER on the development set. More sophisticated acoustic modeling approaches such as speaker adaptive training were used in the AMI ASR system and are described in section 4.3.5.

During text normalisation the transcription is transformed in a consistent form reducing lexical variability. This is achieved by: removing the eventual punctuation, converting everything to the same spelling , and unifying words such as numbers and acronyms. Text normalisation is also the first step to generate the word list of the AMI ASR pronunciation dictionary, described in more detail in section 4.3.1. Separate dictionaries are generated for testing and training. While for testing the word list of the data used for language model training is adopted, for training the word list from the reference transcription is used. Language model training is performed separately using a large amount of data coming from various sources, by training separate language models for each data resource and then using linear interpolation to combine them as outlined in section 4.3.2.

On top of the baseline training process shown in figure 4.1 speaker normalisation and adaptation techniques are also a rather important part of the AMI ASR system: for example VTLN is performed both during training and testing as is speaker adaptation (further details will be provided in 4.3.5).

Figure 4.2 shows the first baseline steps of the testing process for a meeting speech recognition system. Similarly to the training process, during testing the preprocessing is performed to enhance the speech signal. Since for testing, in a fully automatic system, the segmentation of the waveform in utterances is unknown, the segment boundaries are estimated applying speech activity detection techniques (see section 4.3.3) on the enhanced speech signals. Feature extraction follows the same procedures used during training. Finally the decoding step uses the acoustic and language models and the dictionary to produce an automatic transcription.

### 4.3.1 Dictionary

The design of a pronunciation lexicon is an important and critical aspect of a large vocabulary speech recognition system. An extensive overview of pronunciation modeling for LVCSR can be found in Fossler-Lussier (2003). Two main meth-

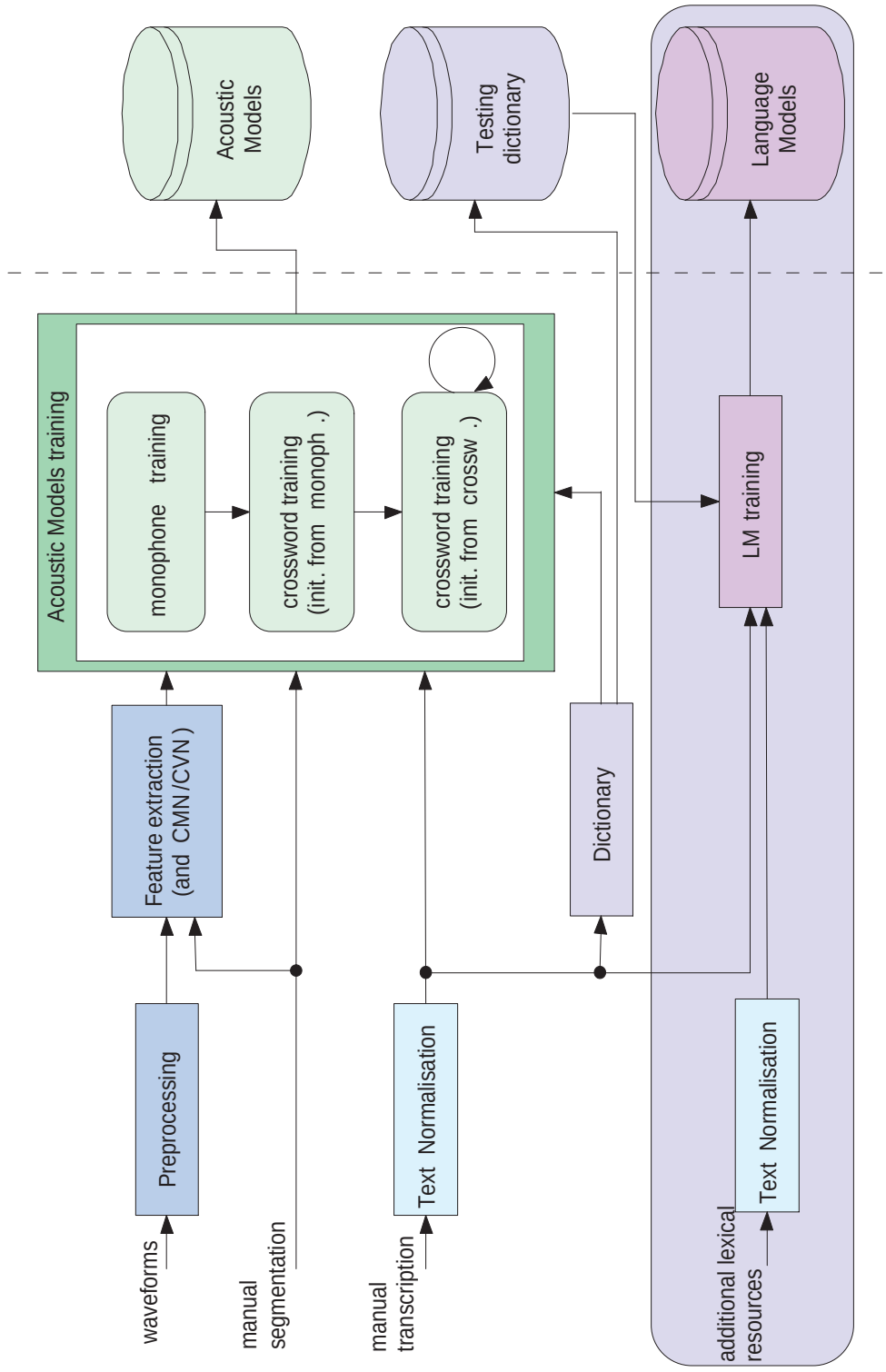



Figure 4.1: General flowchart of the AMI ASR system training including both acoustic and language modeling

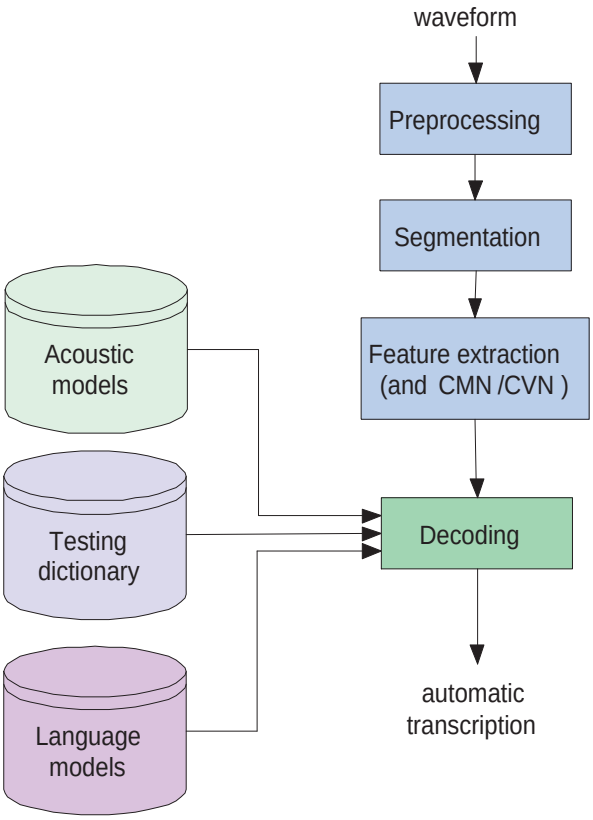



Figure 4.2: Baseline decoding flowchart

ods can be adopted: knowledge based approaches where linguistic observations are included in the model, and data-driven approaches where phonetic patterns are discovered in the corpora. Knowledge based methods can include both the compilation of the pronunciation dictionary by hand or the use of Letter To Sound (LTS) rules. Data-driven approaches, on the other hand, are automatic and may use phone recognisers to produce the most likely phonetic sequence.

To take advantage of both knowledge based and data-driven techniques, hybrid approaches have been proposed which automatically learn pronunciation rules from a training dictionary (which is usually hand made). One possibility is to use Classification and Regression Trees (CART), a particular kind of decision trees, but machine learning techniques such as neural networks may also be used. For CART the goal is to find some features which best describe the contexts which mostly influence the phone realisation. These features are defined by a set of questions which recursively subdivide the training data in two groups.

The AMI ASR dictionary (Hain et al., 2005c) starting point is based on the Unysin accent-independent keyword lexicon (Fitt, 2000). This relies on the use of "keysymbols" which enable the generation of pronunciations for a number of accents. In the AMI ASR dictionary, pronunciations were mapped to the General American Accent since this was the most present in our training data. The use of Unisyn leaves a number of out of vocabulary (OOV) pronunciations. To facilitate human intervention to produce these missing pronunciations, hypotheses pronunciations have been generated using a CART based LTS system (trained on the base Unisyn dictionary). The automatically generated pronunciations where then checked and corrected manually by the members of the AMI ASR team. The AMI ASR dictionary generation process is therefore a mixed approach which uses knowledge based techniques (the Unisyn lexicon), a hybrid approach such as CART to automatically create OOV pronunciations, and finally manual correction of the automatically generated pronunciations.

### 4.3.2 Language Modeling

Language modeling resources for conversational speech are sparse, since the transcription of natural conversations is an expensive process. On the other hand training an N-gram language model requires a large amount of text which should be as similar as possible to the target recognition task. To deal with sparse training data a baseline general purpose language model can be adapted with a small amount of domain specific data, or the domain specific training corpus can be augmented with out–of–domain data. For ASR of conversational speech the second approach has proved to be the most effective: Hain et al. (1999) used Broadcast News data to obtain a CTS language model while Bulyko et al. (2003, 2007) and Wan and Hain (2006) investigated the use of data collected from the web to build language models for meeting speech recognition. In both cases language models trained on in–domain data were interpolated with those trained on out–of–domain data so that, for a trigram case, the language model probabilities are computed as a weighted sum of the probabilities of the individual language models $l$:

$$P(w_i|w_{i-1},w_{i-2}) = \sum_l \lambda_l P_l(w_i|w_{i-1},w_{i-2}) \tag{4.1}$$

where the interpolation weights are usually estimated maximising the likelihood on a small held-out set.

As mentioned in section 4.2.3 meeting data covers a wide range of topics and conversational registers (from reporting to problem solving to more informal dialogues). Therefore the choice of the out of domain data is rather delicate: web data is usually collected by performing web search queries with the most frequent n-grams in the in–domain language model and then selecting a restricted number of pages by perplexity filtering, that is retaining only the pages having a perplexity (measured with an in domain language model) which is lower than a certain threshold.

In this thesis we performed ASR experiments on the WSJCAM0, on CTS and on meeting recordings. For the first task the standard MIT Lincoln Labs 20k Wall Street Journal trigram language models were used (Paul and Baker, 1992). For the CTS experiments, language models were trained on Switchboard, Call Home, Fisher, ICSI meetings and web data resources, while for the meeting language model training AMI, NIST and ISL meetings were also used (Hain et al., 2005c).

### 4.3.3 Preprocessing and automatic segmentation

Automatic segmentation is a crucial step in the preprocessing of an ASR system. It consists of automatically finding the time boundaries of the sentences which have to be recognised and it is also referred to as Speech Activity Detection (SAD). Even in the independent headset microphone task, the presence of cross-talk and vocal noises makes it infeasible to adopt threshold based techniques. Therefore most SAD systems simply consist of GMM/HMM based classifiers trained on various kind of features: typically MFCCs, PLPs, kurtosis etc. The AMI ASR system (Hain et al., 2005b) automatic segmentation consists in a Multi Layer Perceptron (MLP) classifier trained on PLPs as well as kurtosis. Another important preprocessing step is echo cancellation performed in the form of adaptive Least Mean Square (LMS) echo cancellation (Hain et al., 2005a).

### 4.3.4 Multiple distant microphone preprocessing

In the multiple distant microphone condition speech has to be recognised using multiple audio signals captured by a set of microphones with an unknown geometry. Directly applying ASR to each of these signals would be problematic[4] because there would be too much overlap between different sounds, including speech coming from different speakers and ambient noises. Instead the set of available microphones constitutes a microphone array and techniques to increase the sensitivity in the direction of the desired signal and decrease it in the noise signal direction can be used. The sensitivity in a specific direction is defined by the microphone array directivity pattern also known as the array response. Beamforming techniques (McCowan, 2001) aim to achieve a particular shaping and steering of the directivity pattern and they can be seen as a spatial filter. The directivity pattern for a linear equally spaced array of identical microphones depends on the number of the array elements $N$, on the distance between the elements $d$ and on the frequency $f$ and has the form of a sinc function of the angle $\phi$ of arrival to the array:

$$Dir(f,\phi) = \sum_{n=-\frac{N-1}{2}}^{n=\frac{N-1}{2}} w_n(f) e^{j\frac{2\pi f}{c} nd\cos(\phi)} \tag{4.2}$$

where $w_n$ is the weight of the n*th* element. As $N$ increases the amplitude of the side lobes decreases and as $d$ increases the width of the main lobe (called beam width) decreases making the array more directive for a certain angle $\phi$.

Furthermore, equivalently to the Nyquist theorem in the frequency domain, the spacial sampling theorem states that: in order to prevent spatial aliasing in the directivity pattern of the array, the distance $d$ between the microphones should be

$$d < \frac{\lambda_{min}}{2} \tag{4.3}$$

where $\lambda_{min}$ is the minimum wave length and is equal to $c/f_{max}$ (where $c$ is the speed of propagation for acoustic waves, approximately 330 m/sec in air). Therefore if we have a fixed $d$ then:

$$f_{max} < \frac{c}{2d}. \tag{4.4}$$

[4]Under the unrealistic assumption that each microphone can be strictly assigned to the closest speaker

So when the array geometry is fixed we can expect spatial aliasing for frequencies over a threshold $f_{max}$. In practice for those frequencies the side lobes become larger and therefore the array shows a high sensitivity even for unwanted directions.

Moreover the directivity or "spatial selectivity" varies according to the frequency. For a linear array the beam width is wider at lower frequencies and narrower at higher frequencies so that we can say that the array is less directive at lower frequencies.

One of the most used beamforming techniques is delay and sum. This technique basically combines the output of $N$ microphones. First of all the time delay of arrival $D_i$ of the signal $y_i$ recorded by the microphone $i$ with respect to a reference microphone $y_1$ is estimated. Then the delayed signals $y_i(t+D_i)$ are summed as can be seen in figure 4.3. When the delay varies over time, as it is in meeting recordings where speakers are free to move around, the beamformed signal is:

$$z(t) = \frac{1}{N}\sum_{i=1}^{N} y_i(t+D_i(t)) \tag{4.5}$$

and it can be shown that the directivity pattern is:

$$Dir(f,\phi,t) = \sum_{i=1}^{N} e^{-j2\pi f D_i(t)} = \sum_{i=1}^{N} e^{-j\frac{2\pi f(i-1)dcos(\phi(t))}{c}} \tag{4.6}$$

where $D_i(t) = (i-1)dcos(\phi(t))/c$. The time delay $D_i(t)$ can be estimated by cross correlation techniques. In particular in the presence of uncorrelated noises the most commonly used technique is the Generalised Cross Correlation method (Knapp and Carter, 1976). The estimation of the time delay allows in practice to do sound source localisation because the angle of arrival $\phi$ is directly related to the time delay.

A more general class of beamforming is filter and sum (of which delay and sum is a sub-class) where the received signals are first filtered and then summed, and the filters are frequency dependent:

$$y(f) = \sum_{n=1}^{N} w_n(f)x_n(f). \tag{4.7}$$

The AMI meetings were recorded using 8 element circular microphone arrays. This configuration is particularly interesting for meeting recordings because it provides a uniform ($360^o$) distribution of all possible locations of the speakers (Moore and McCowan, 2003). Moreover the directivity pattern of circular arrays shows

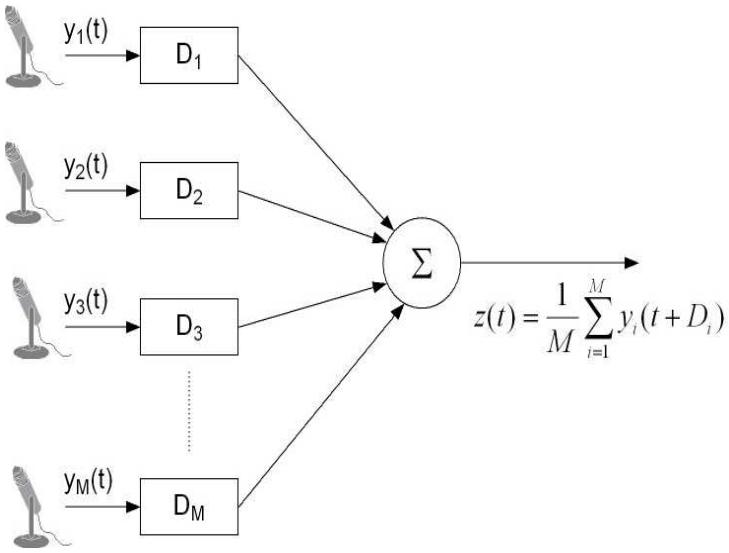



Figure 4.3: Delay and sum beamforming

good discrimination between speakers separated by at least $45^o$, making this configuration suitable for up to 8 participant meetings (McCowan et al., 2005).

The AMI ASR system uses the so called superdirective implementation of delay and sum beamforming with some additional preprocessing steps. First a gain calibration by the normalisation of each channel with the maximum amplitude level is performed, because it would not be possible to apply delay and sum on acoustic signals with different dynamic ranges; second Wiener filtering, a technique to reduce (additive) stationary noise, is applied to each distant channel (where the noise estimation is performed on the $M$ lowest energy frames); then the energy scaling factor and the delay of each channel is estimated by generalised cross correlation with respect to a given reference channel; finally the beamformer filters for each frame, used to perform delay and sum beamforming (Hain et al., 2005a), are estimated by means of the delay and scaling factor parameters from the previous step.

### 4.3.5 Acoustic Modeling

Both in the AMI ASR system and in the experiments of this thesis acoustic models are phonetic decision tree clustered Hidden Markov Models with left–to–right three–state topology and Gaussian mixture model (GMM) output distributions, trained using the Hidden Markov Model ToolKit (HTK) software (Young et al., 2006).

The overall training and decoding structure used in this thesis experiments was that developed for the AMI-ASR system (Hain et al., 2005a).

In the AMI ASR system the baseline acoustic models were trained using standard Maximum Likelihood Estimation (MLE) techniques, on the first 12 HTK Mel Frequency Perceptual Linear Prediction (MF-PLP) coefficients with the inclusion of the zeroth cepstral coefficient $c_0$ and first and second derivatives (therefore yielding 39–dimensional feature vectors). Moreover Cepstral Mean and Cepstral Variance Normalisation have been applied on a per channel basis, being therefore not only speaker specific but also channel specific.

CTS models for the AMI ASR system are trained on the full 270 hours training set *ctstrain04*. Since the amount of meeting data is rather limited compared to the CTS domain, and it was found that adapting from CTS to the meeting domain is beneficial (Stolcke et al., 2004), in the AMI ASR system we experimented with adapting from the CTS domain as well. The CTS models are Narrow Band (NB) in the sense that they have been trained on a limited bandwidth between 125 Hz and 3800 Hz, because of the telephone channel band pass effect.

In the AMI ASR system a procedure was developed to adapt the NB CTS models to the Wide Band (WB) meeting data[5]. Maximum Likelihood Linear Regression transforms from NB to WB are used as input transforms to adapt the NB CTS models to the WB meeting domain using Maximum a Posteriori (MAP) adaptation. Using this procedure it was found that the models adapted from CTS performed better than those trained on meeting data only (Hain et al., 2005b). Given the mismatch between the IHM and the MDM condition, in the AMI ASR system two different sets of models were trained for the two tasks, using acoustic data from the two domains. In particular the MDM acoustic models were trained on the beamformed acoustic signals.

The AMI ASR system is a multi-pass system consisting of several recognition steps with increasing degree of complexity. In the following steps the system makes use of more complex techniques, both on the feature extraction part, where Smoothed Heteroscedastic Linear Discriminant (SHLDA) and posterior features are used, and in the adaptation part, making use of MLLR, Constrained MLLR (CM-

[5]Meeting audio files are sampled at 16kHz and it has been shown that using the full bandwidth is beneficial

LLR) in a Speaker Adaptive Training (SAT) fashion, and embedded training VTLN.

The VTLN embedded training procedure, described in more detail in chapter 5, involves the alternation of warping factors estimation and training passes until the WER in the development set stabilises (which should correspond to the convergence of the warping factor values of the training set as well).

On top of VTLN embedded training, SAT techniques are also used by estimating CMLLR transforms in the training set and using then the CMLLR transformed speaker adapted features to train a new acoustic model.

## 4.4 ASR system combination

Different acoustic representations have different strengths and weaknesses for ASR. Approaches to combine multiple representations, at the feature, model and system level, have proven to be effective to reduce the word error rate. Feature combination may be carried out directly at the feature vector level by concatenating feature vectors, followed by a dimension reducing transform such as linear discriminant analysis (LDA) or heteroscedastic LDA (HLDA) (Burget, 2004a), indirectly at the model level (Kirchhoff et al., 2000; Zolnay et al., 2007), or as a postprocessing procedure applied to the outputs of multiple recognizers (Fiscus, 1997).

The simplest form of direct feature combination involves the concatenation of the acoustic feature vectors. This approach has a number of drawbacks including a substantial increase in the dimensionality of the feature space to be modelled, and the presence of strong correlations between components in the concatenated vector, which can cause problems for acoustic models based on diagonal covariance Gaussians. Both these problems are addressed through the use of dimension reducing, decorrelating transforms such as LDA, HLDA and principal components analysis (PCA). PCA estimates a global transform, and was found to be much less well-suited to the task compared with LDA and HLDA which allow the decorrelating transforms to be estimated on a per-class (or per-state) basis (Burget, 2004a). Schlüter et al. (2006) have observed that numerical problems can arise when estimating LDA transforms from a concatenation of strongly correlated feature vectors, and that model-based transforms are less susceptible to this problem.

Zolnay et al. (2007) have demonstrated that discriminant feature-level combina-

tion may be nested successfully inside a model-based combination approach, and this has resulted in reduced word error rates for two LVCSR tasks, VerbMobil-II and the European Parliamentary Plenary Sessions corpus. More recent work by this group, involving the investigation of auditory-inspired features from a gammatome filterbank, have indicated that a system level combination using ROVER (Fiscus, 1997) results in a significant reduction in word error rate (Schlüter et al., 2007).

### 4.4.1 LDA/HLDA

In our experiments and in the AMI ASR framework, feature-level combination was performed using HLDA (a generalisation of LDA), a procedure that enables the derivation of a linear projection that decorrelates concatenated feature vectors, and performs a dimensionality reduction. In both HLDA and LDA, each feature vector that is used to derive the transformation is assigned to a class. Since one of the goals of these techniques is to improve the discrimination between the classes used during decoding, HLDA/LDA classes are typically HMM states or mixture components. The class assignment is usually performed using Viterbi alignment. We have chosen to use HLDA in our experiments because this technique has proven to yield better performances than LDA, this being motivated by the HLDA ability to handle heteroscedasticity (Kumar and Andreou, 1998) (the property of having a different covariance matrix per class).

Hunt (1979) proposed the use of LDA to improve discrimination between syllables. Given a $n$ dimensional feature vector $x$ the goal of LDA is to find a linear transformation $\theta_p^T : \Re_n \rightarrow \Re_p$ with $p \leq n$ such as to project $\mathbf{x}$ in a $p$ dimensional space according to $\mathbf{y}_p = \theta_p^T \mathbf{x}$. The transform is chosen in order to maximise the between class covariance $\Sigma_{bc}$ and to minimise the within class covariance $\Sigma_{wc}$ and it is computed as the eigenvectors corresponding to the larger eigenvalues of $\Sigma_{bc} \times \Sigma_{wc}^{-1}$. The $n$ dimensions are therefore those corresponding to the best separation of individual classes. In one of its first applications on ASR, LDA was used in a small vocabulary continuous speech recognition system (Bahl et al., 1988) to introduce time information in the feature vectors by appending consecutive feature frames and using LDA to reduce to a smaller dimension.

The LDA method makes two assumptions: all the classes obey to a multivariate Gaussian distribution and share the same within class covariance matrix. HLDA

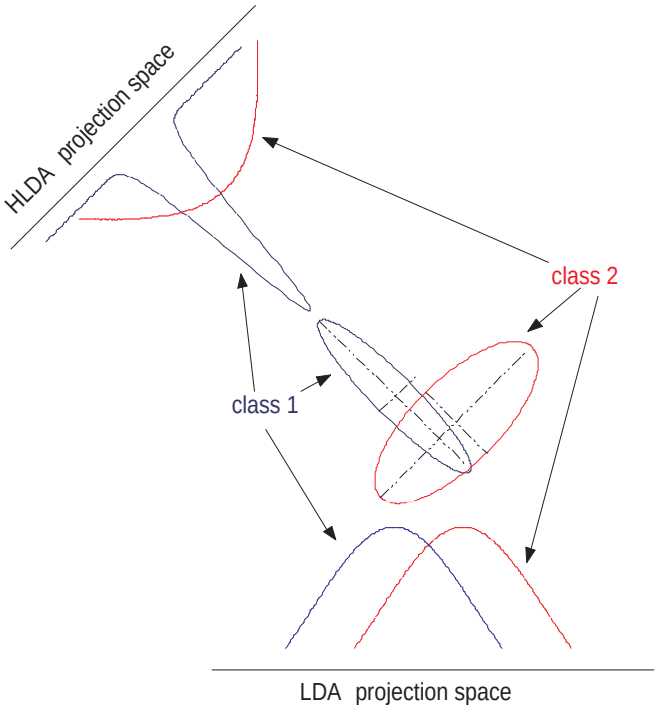



Figure 4.4: Comparison of LDA and HLDA projection for a 2 to 1 dimensional reduction case with 2 classes

relaxes the second assumption and it is therefore a generalisation of LDA. An illustration of this principle can be seen in figure 4.4 for a bidimensional case. In this example the LDA assumption of having the same covariance matrix for each class is not valid and LDA would project the features in a nuisance dimension, while HLDA projects the features in the useful dimensions where the classes are better separated.

The HLDA transform computation was derived by Kumar and Andreou (1998) although the idea of maximum likelihood estimation of the linear transform was introduced for the first time by Schukat-Talamazzini et al. (1995) where an ML optimisation of the transform was performed with respect to the likelihood function of the overall model. Similarly in Kumar's work the likelihood of the original data

$x_i$ is maximised finding the optimal transformation matrix $\mathbf{A}$:

$$\log L(\mathbf{x},\mathbf{A}) = -\frac{nN}{2} + \sum_{j=1}^{J} \frac{N_j}{2} \log \left( \frac{(\det \mathbf{A})^2}{(2\pi)^n \prod_{k=1}^{p} a_k \hat{\Sigma}^{(j)} a_k^T \prod_{k=p+1}^{n} a_k \hat{\Sigma} a_k^T} \right), \quad (4.8)$$

where $\hat{\Sigma}$ and $\hat{\Sigma}^{(j)}$ are the global and per class covariance matrix estimates respectively; and $N$ and $N_j$ are the total and per class number of training vectors. Moreover in equation 4.8 we assumed to handle with diagonal covariance matrices. Since the maximisation of equation 4.8 has no closed-form solution, an efficient iterative algorithm was proposed by Gales (1999). The use of this algorithm on ASR of TI connected Digits was investigated by Burget (2004a,b) and in our experiments we used the same implementation. The proposed iterative procedure computes the transform matrix $\mathbf{A}$, by periodically reestimating individual rows as follows:

$$\hat{\mathbf{a}}_k = \mathbf{c}_k \mathbf{G}^{(k)-1} \sqrt{\frac{N}{\mathbf{c}_k \mathbf{G}^{(k)-1} \mathbf{c}_k^T}} \quad (4.9)$$

where $\mathbf{c}_i$ is the $i$th row vector of co-factor matrix $C = |\mathbf{A}|\mathbf{A}^{-1}$ for the current estimate of $\mathbf{A}$ and

$$\mathbf{G}^{(k)} = \begin{cases} \sum_{j=1}^{J} \dfrac{\gamma_j}{\mathbf{a}_k \hat{\Sigma}^{(j)} \mathbf{a}_k^T} \hat{\Sigma}^{(j)} & k \leq p \\ \dfrac{N}{\mathbf{a}_k \hat{\Sigma} \mathbf{a}_k^T} \hat{\Sigma} & k > p. \end{cases} \quad (4.10)$$

$\gamma_j$ is the number of training feature vectors belonging to the $j^{th}$ class.

A restriction of HLDA when $p = n$ was investigated by Gopinath (1998), and it is referred to as Maximum Likelihood Linear Regression Transform (MLLT) or diagonalisation transform because it has the effect of transforming the features in a space where the assumption of diagonal covariance matrices is more valid. Moreover when MLLT is applied on top of LDA or HLDA consistent improvement can be seen (Saon et al., 2000b).

The main characteristic which sets apart HLDA from LDA is the assumption of a different covariance matrix for each class. In LDA the within class covariance matrix is approximately the weighted sum of the individual HLDA class covariance matrices. A minimum amount of in–class data is necessary to find reliable estimates for the individual HLDA covariance matrices. Therefore, in order to avoid

data sparsity, the type of classes used to estimate the HLDA transformation matrices should be carefully considered. In the experiments reported in this thesis we experimented with two possible classes choices (see section 6.4): in the first case we used classes corresponding to the HMM triphone states of our models and in the second case we used Gaussian mixture componenents of monophone models[6]. The class assignment has been achieved by performing Viterbi alignment.

To exploit the advantages of both LDA and HLDA, Smoothed HLDA (SHLDA), a technique which estimates the per class covariances $\Sigma^{(j)}$ as a weighted sum of the estimated per class covariance and the within class covariance, was introduced in Burget (2004b). For SHLDA the estimate of the class covariance matrix is given by:

$$\check{\Sigma}^{(j)} = \alpha\hat{\Sigma}^{(j)} + (1-\alpha)\Sigma_{wc} \tag{4.11}$$

where $\check{\Sigma}^{(j)}$ is the smoothed estimate of the covariance matrix of class j, $\Sigma_{wc}$ is the within class covariance matrix used in the LDA transform estimation and $\alpha$ is the smoothing factor and it is between 0 (pure LDA) and 1 (pure HLDA).

### 4.4.2 System-level combination

In addition to feature-level combination in the experiments of this thesis we also explored the use of system-level combination using ROVER (Fiscus, 1997), a technique to combine the output of multiple speech recognition systems. In ROVER, the transcriptions are first compared by aligning them using dynamic programming to minimise the number of substitutions, deletions and insertions. This alignment depends on the word sequence chosen as the reference.

The multiple alignments are then combined using a voting approach, performed either by choosing the most frequently recognised hypothesis (majority voting) or by selecting the hypothesis with the highest confidence score (maximum confidence score voting). The choice of the voting criteria is not limited to these two techniques and any approach able to disambiguate between multiple transcriptions can be adapted (Hillard et al., 2007). It is also possible to obtain a lower bound on the word error rate achievable by ROVER, by using an oracle combination in which the closest available word sequence to the correct transcription is selected. A disadvan-

[6]Monophone models are estimated as part of the triphones training process

tage of ROVER is the need to train and use for decoding each system separately, in contrast to HLDA which requires a single decoding pass.

A generalisation of the ROVER algorithms aligns confusion networks (a particular lattice representation outlined in section 2.2.2.2) instead of the 1-best hypotheses (Evermann and Woodland, 2000), hence taking into account multiple hypotheses from the same system at the same time and yielding therefore better results.

## 4.5 Testing Conditions: the NIST Rich Transcription Meeting Evaluations

The NIST meeting recognition evaluations, which have been run since 2002, give the opportunity to the participants to evaluate and compare the performances of their speech recognition systems in a competitive environment. Moreover their main goal is to improve automatic transcriptions making them more useful both for humans and machines. Although they comprise several tasks such as speaker activity detection and diarisation ("who spoke when") our main interest in this thesis is focused in the Speech To Text (STT) task.

NIST RT evaluations have a number of different acoustic conditions as well, the main ones are:

- independent headset microphone (IHM): requiring that, using a separate headset microphone signal for each meeting participant, the systems provide a separate transcription for each speaker;

- multiple distant microphones (MDM): multiple distant microphone signals are provided and the systems should output a single transcription stream comprising all the words said during the meeting.

We conducted experiments using both conditions, training separate acoustic models for each condition. For the MDM task, the speech has to be recognised from the output of a certain number of microphones, of unknown position, placed in the meeting room. The geometry of the microphone position varies depending on the site where the data was collected.

In this thesis we used as a testing set for the meeting systems, the NIST Rich Transcription Spring 2004 evaluation set[7], which is composed of about 90 minutes excerpted from 8 meetings (11 minutes each) recorded in four different data collection sites (CMU, ICSI, LDC and NIST). While all speakers had a noise canceling head mounted microphone, the number of multiple distant microphones varies according to the meeting room; in particular the CMU data had one distant microphone only. Moreover these meetings contain a total of 31 unique speakers (some of the speakers participated in more than 1 meeting) of which 18 were male and 13 were female speakers.

In this work the performances for the MDM condition are reported for the non-overlapping segments only while for the IHM condition all segments are recognised. Moreover the manual segmentation is used unless otherwise stated.

[7] http://www.nist.gov/speech/tests/rt/rt2004/spring/

# Chapter 5

# VTLN in meetings

## 5.1 Introduction

In this chapter we will describe a set of baseline experiments concerning the application of VTLN to multiparty conversational speech. The aim of these experiments is to assess the effectiveness of VTLN with respect to the multiparty meeting domain and its particular characteristics from a speaker normalisation point of view. The experimental setup for the application of VTLN is described in section 5.2.

First we report on experiments performed on the conversational telephone speech domain, a task where several successful VTLN applications have been reported (Hain et al., 1999, 2005d). Here the presence of distinct speaker sides[1] and the availability of several minutes of speech for each speaker, enabled stable estimations of the warping factors. These experiments are described in section 5.3.

A larger set of experiments was performed in the multiparty meeting domain. The experimental setup and baseline VTLN results on meeting data are outlined in section 5.4. The stability of the estimated speaker-specific warping factors was investigated, both for the same speaker across different meetings, and across time for the same speaker within a single meeting. The length of a speaker's vocal tract depends on the lips and the larynx positions, therefore, since this varies across time during speech production (Dusan, 2005b), we did not find stable estimates for the warping factors. We have investigated the relationship of the frequency warping

[1]For CTS data the two speakers share the same communication channel but their speech is recorded separately providing distinct speaker sides or speaker turns

factor to the addressee of the current speaker (in section 5.4.1 and 5.4.2), the formant positions (in section 5.4.3) and the quality of the transcriptions used for the warping factor estimation (see section 5.4.4).

A final set of analysis experiments on the AMI corpus investigated the relationship between warping factor values and recognition improvements (see section 5.5).

## 5.2 VTLN experimental setup

In this section we describe our application of VTLN. A maximum likelihood approach was adopted, using a piecewise linear frequency warping similar to those illustrated in Fig. 5.1 (Hain et al., 1999; Young et al., 2006). Given the warping factor $\alpha$ and the lower and upper cutoff frequencies $f_L$ and $f_U$, the warping function is in general defined in three regions (as shown in the left of Fig. 5.1), with the constraints that the minimum $f_{min}$ and the maximum $f_{max}$ frequencies (i.e. the lower and upper frequencies of the speech signal bandwidth) should be kept unvaried in the frequency warped space, as follows:

$$f_{warped} = \begin{cases} a_U \cdot (f_{orig} - c_U) + \frac{c_U}{\alpha} & f_{orig} > c_U \\ \frac{f_{orig}}{\alpha} & c_L \leq f_{orig} \geq c_U \\ a_L \cdot (f_{orig} - f_{min}) + f_{min} & f_{orig} < c_L \end{cases} \tag{5.1}$$

where the frequencies $c_L$ and $c_U$ shown in figure 5.1 are defined as:

$$c_L = \frac{2 \cdot f_L}{1 + \frac{1}{\alpha}}, \tag{5.2}$$

$$c_U = \frac{2 \cdot f_U}{1 + \frac{1}{\alpha}}. \tag{5.3}$$

The angular coefficients of the first and the third regions $a_L$ and $a_U$ are computed as:

$$a_L = \frac{\frac{c_L}{\alpha} - f_{min}}{c_L - f_{min}}, \tag{5.4}$$

$$a_U = \frac{f_{max} - \frac{c_U}{\alpha}}{f_{max} - c_U}. \tag{5.5}$$

In particular in the experiments presented in this thesis $f_L = f_U$ so that the warping function is defined by two regions as shown in the right part of figure 5.1.

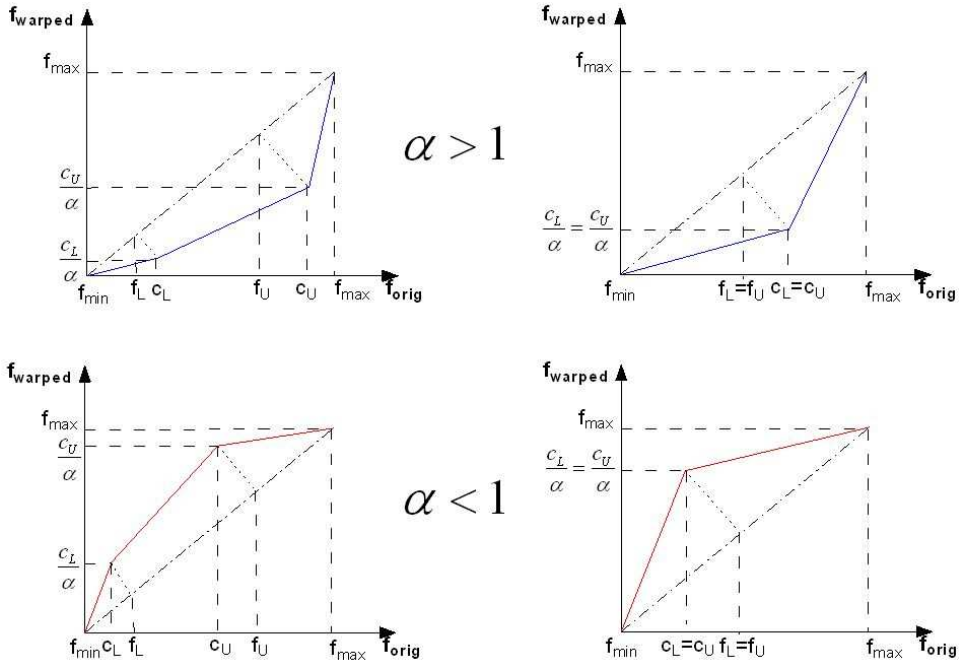



Figure 5.1: Piecewise linear frequency warping functions: on the left the general case and on the right the particular case adopted in the experiments of this thesis where the lower cutoff frequency $f_L$ is equal to the upper cutoff frequency $f_U$

The warping factor α is estimated using a Brent search technique based on quadratic interpolation [2], since the log-likelihood's trend for a given transcription tends to have a parabolic shape in function of the warping factor value.

VTLN was applied both during training and testing. For training we used an iterative procedure with the following steps, figure 5.2 shows a block diagram of this method, (Hain et al., 1999):

1. warping factors α are estimated using a non-normalised model and normalised features are computed using the estimated warping factors;

2. a training pass is performed (adopting the single pass retraining technique (Young et al., 2006) starting from non-normalised models followed by a few Baum Welch iterations, typically four are sufficient);

3. the warping factors are estimated again using the acoustic models trained in

[2]Brent's method is an algorithm combining the bisection method, the secant method and inverse quadratic interpolation, aiming to find the minimum of a parabolic curve.

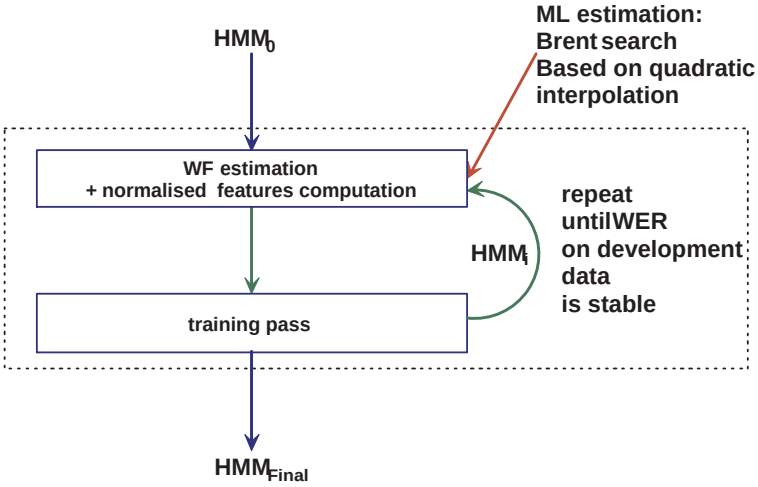



Figure 5.2: Block diagram of the iterative VTLN training procedure

the previous pass and normalised features are computed another time using the estimated warping factors;

4. another training pass is performed: similar to step 2 but starting from the normalised models of the previous pass;

5. steps 3 and 4 are repeated until the WER on the development data set stabilises.

This iterative procedure allows warping factors to converge, resulting in $\alpha$s in the range between 0.8 and 1.2, with the distribution of warping factors for female speakers decreasing to less than 1, and the distribution for males increasing to greater than 1. This behaviour is due to the fact that as the iterative VTLN training proceeds the acoustic models, being trained on features which are better speaker normalised, can better match the normalised acoustic data providing therefore an improved estimate of the warping factors (usually smaller values for female speakers and higher values for male speakers). The iterative VTLN training approach aims at improving the reliability of the estimated warping factor, iteration after iteration. Thus the distributions of the warping factors for male and female speakers tend to be increasingly separated until convergence is reached.

For testing a two pass decoding procedure was adopted as follows, in figure 5.3 a diagram representation of this method can be seen, (Welling et al., 2002; Hain

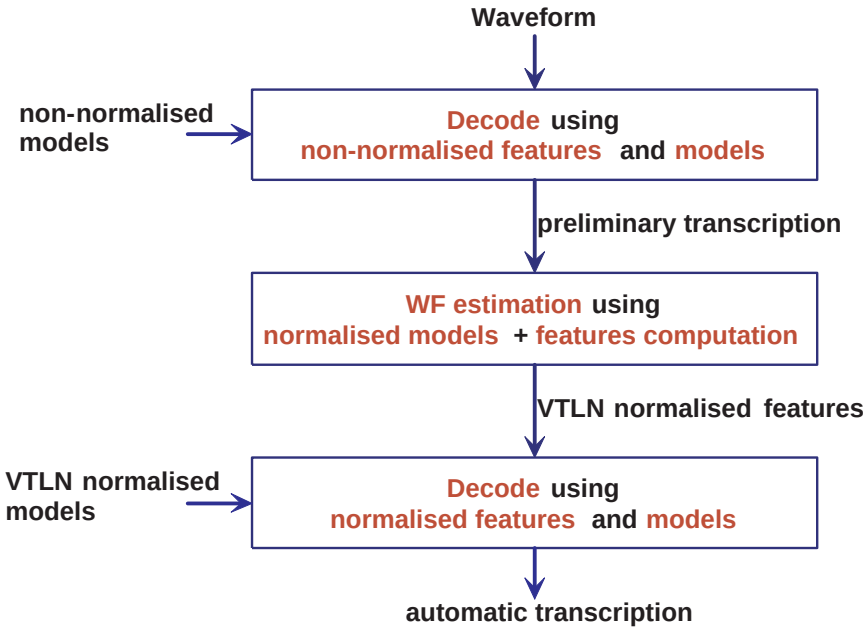



Figure 5.3: Block diagram of the two pass VTLN decoding procedure

et al., 1999):

1. decoding is performed using non-normalised features and models;

2. warping factors are estimated using normalised models and the preliminary transcription of the previous pass;

3. normalised acoustic features are computed and used for decoding with the normalised acoustic models trained using the iterative procedure.

It is possible to perform a VTLN test only procedure where the warping factors are estimated using non-normalised models in step 2. However the resulting WERs would be higher. Moreover the same models were used for warping factor estimation and for decoding having 16 mixtures per state of the HMMs. Welling et al. (2002) suggested that using low complexity acoustic models for warping factor estimation yields better performances. However this research direction was considered out of the scope of this thesis.

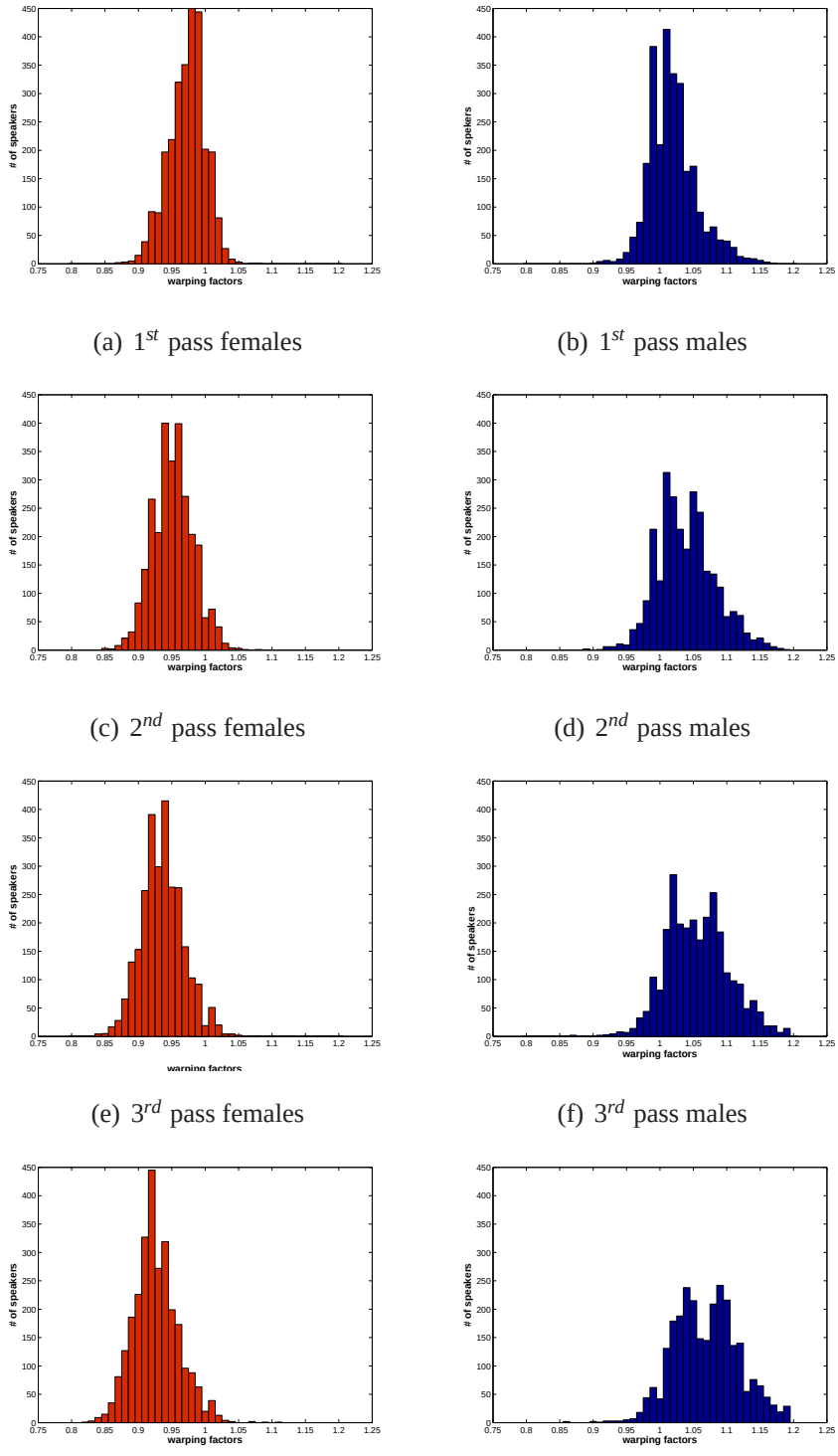
## 5.3 CTS Experiments

Experiments on the Conversational Telephone Speech task were conducted training on the *ctstrain04* set consisting of around 270 hours of speech from Switchboard and Callhome and testing on the NIST eval01 set (both described in section 4.2.2). As features we used CMN and CVN normalised MF-PLPs and we trained cross-word clustered acoustic models. The trigram language models used were those trained by the AMI ASR team as described in section 4.3.2.

For VTLN we used a piecewise frequency warping function where the values of the lower cutoff frequency $f_L$ and the upper cutoff frequency $f_U$ (see figure 3.3) were both set to 3400Hz. The iterative VTLN training technique described in the previous section was used yielding the results shown in table 5.1 which shows the WER for the baseline system without adaptation, the WER using VTLN during testing only, and the results for each of the 4 iterations and after training from scratch using the normalised features of the $4^{th}$ pass. From the baseline to the $4^{th}$ iteration after training from scratch a relative improvement of around 9% was obtained. The improvements in terms of WER are consistent with the stabilisation of the warping factor distributions after 4 passes as can be observed in figure 5.4. The warping factors distribution tends to shift towards values smaller than 1 for females and towards values larger than 1 for males. This incremental separation between the warping factor distributions for male and female speakers is due to the fact that the acoustic models improve after each iteration (being trained on features which are better speaker normalised), therefore providing more accurate estimates of the warping factors.

## 5.4 Meetings Experiments

We also applied VTLN to multiparty conversations in a meeting environment. Successful applications of VTLN have been reported on conversational telephone speech tasks, where there are distinct speaker sides and usually several minutes of speech per speaker (Hain et al., 1999). However in the case of meetings the amount of speech data per speaker can vary significantly, making it difficult to obtain stable estimates of the VTLN warping factor.




(a) $1^{st}$ pass females

(b) $1^{st}$ pass males

(c) $2^{nd}$ pass females

(d) $2^{nd}$ pass males

(e) $3^{rd}$ pass females

(f) $3^{rd}$ pass males

(g) $4^{th}$ pass females

(h) $4^{th}$ pass males

Figure 5.4: Warping factor distributions of the training set for each VTLN iteration for females and males in the CTS domain

| | Tot | Sub | Del | Ins | SW1 | S23 | Cell | F | M |
|---|---|---|---|---|---|---|---|---|---|
| No Adapt | 37.2 | 24.2 | 8.8 | 4.2 | 30.1 | 38.0 | 43.0 | 36.7 | 37.6 |
| Test only | 36.4 | 23.6 | 8.5 | 4.3 | 29.5 | 36.5 | 42.6 | 36.1 | 36.7 |
| $1^{st}$ pass | 35.7 | 22.9 | 8.9 | 3.8 | 29.1 | 35.4 | 42.2 | 35.0 | 36.4 |
| $2^{nd}$ pass | 35.0 | 22.5 | 8.8 | 3.7 | 28.5 | 34.6 | 41.4 | 34.2 | 35.8 |
| $3^{rd}$ pass | 34.5 | 22.0 | 8.7 | 3.7 | 27.7 | 34.2 | 40.9 | 33.6 | 35.3 |
| $4^{th}$ pass | 34.2 | 22.0 | 8.6 | 3.6 | 27.5 | 34.2 | 40.5 | 33.3 | 35.1 |
| $4^{th}$ TFS | 34.1 | 22.1 | 7.9 | 4.2 | 27.6 | 34.6 | 39.8 | 33.8 | 34.5 |

Table 5.1: VTLN CTS results on *eval01* training on the full *ctstrain04* set, from top to bottom: WER without any adaptation or normalisation, test only VTLN, $1^{st}$ pass VTLN, $2^{nd}$ pass VTLN, $3^{rd}$ pass VTLN, $4^{th}$ pass VTLN and $4^{th}$ pass VTLN Trained From Scratch (TFS). The testing set consists of approximately 6 hours of speech in total, equally distributed between Switchboard–1 (SW1), Switchboard–2 (S23) and Switchboard-cellular (Cell).

We performed a first set of experiments using the ICSI meetings corpus described in section 4.2.3.2 (Janin et al., 2003). We used 70 of the 75 ICSI meetings as training data. For testing we used the ICSI portions of the NIST Spring 2004 Meetings Evaluation development and evaluation sets, referred to as RT04sdev and RT04seval, respectively (NIST, 2004). Each of these test sets contains 10 minutes of two different meetings, with 12 different speakers in RT04sdev and 15 in RT04seval (described in more detail in section 4.5).

As mentioned in section 4.3.5 in the meeting domain best results are obtained by adapting the acoustic models from the conversational telephone domain where more data are available. Therefore as a starting point we used the acoustic models described in the previous section trained on 270 hours of CTS data using the VTLN iterative procedure. The resultant models were then MAP adapted to the meeting domain using 70 of the 75 ICSI meetings. VTLN training was performed, starting from the MAP adapted models, using the iterative procedure described in section 5.2. Each intermediate model was evaluated on both test sets (using a bigram language model and a vocabulary of 50k words), and the results are shown in

| | RT04sdev (ICSI) | RT04seval (ICSI) |
|---|---|---|
| noVTLN | 27.0 | 34.2 |
| VTLN 1 | 24.6 | 31.6 |
| VTLN 2 | 24.5 | 31.2 |
| VTLN 3 | 24.9 | 32.1 |
| VTLN 4 | 24.4 | 31.3 |
| VTLN 5 | 24.3 | 31.0 |

Table 5.2: Speech recognition results of VTLN experiments (% WER) on meetings, training on 70 ICSI meetings and testing on the ICSI part of the RT04sdev and RT04seval sets for five successive training passes of the iterative procedure.

table 5.2[3]. Moreover Cepstral Mean Normalisation (CMN) and Cepstral Variance Normalisation (CVN) were performed both during training and testing where the mean and variance was calculated over a complete channel for every speaker per meeting (Hain et al., 1999). Only two VTLN training passes were required for the convergence of the distribution of the warping factors, although after convergence some small ripples in the WER could be observed.

Another set of acoustic models for meetings was also trained starting both from baseline acoustic models which were CTS MAP adapted to the meeting domain using the full meeting training set described in section 4.2.3.6, and models trained on meeting data only. We refer to these models respectively as CTS-INIA and INIA, since they were trained using ICSI, NIST, ISL and AMI data (Hain et al., 2005c). Recall that in the CTS domain the warping function was chosen such that both the lower and the upper cutoff frequencies ($f_L$ and $f_U$ respectively) are set to 3400Hz. Since the meeting data was sampled at 16kHz, we experimented with two warping functions: one using $f_L = f_U = 3400Hz$ and the other using $f_L = f_U = 7200Hz$. For both configurations we trained both CTS-INIA and INIA models, resulting in four model sets.

These models were tested on the whole eval set of the full NIST 2004 meeting evaluation data and results are reported in table 5.3. Three passes of the VTLN

[3]Different warping factors were estimated for those speakers that occurred in both sets.

| | TOT | F | M | CMU | ICSI | LDC | NIST |
|---|---|---|---|---|---|---|---|
| noVTLN INIA | 40.6 | 39.6 | 41.1 | 45.2 | 26.0 | 54.9 | 33.5 |
| noVTLN CTS-INIA | 40.0 | 39.4 | 40.4 | 44.5 | 25.6 | 53.4 | 34.4 |
| INIA 3400 | 38.4 | 37.1 | 39.0 | 43.6 | 23.3 | 52.1 | 31.8 |
| INIA 7200 | 38.6 | 37.4 | 39.2 | 43.6 | 23.2 | 53.0 | 32.0 |
| CTS-INIA 3400 | 37.7 | 36.5 | 38.3 | 42.7 | 22.6 | 50.7 | 32.4 |
| CTS-INIA 7200 | 38.3 | 37.9 | 38.6 | 43.0 | 23.1 | 51.7 | 33.5 |

Table 5.3: Results of CTS-INIA and INIA baseline and VTLN models (on the NIST 2004 meeting transcription evaluation set) where 3400 and 7200 indicate the $f_L = f_U$ values in Hertz in the piece-wise linear frequency warping functions

iterative training procedure were performed and trigram language models were used for decoding.

It can be noticed that choosing a value for $f_L = f_U = 3400Hz$ gives the best improvement together with the use of CTS adapted models which remain consistently the best models even after the application of VTLN.

### 5.4.1 Warping Factors Behavior Analysis

The amount of data per speaker in each meeting varies considerably with a minimum of 3 seconds to a maximum of more than 1 hour of speech per speaker per meeting with an average utterance duration of about 2.4 seconds in the training set. This aspect of the meeting data affects the reliability of the VTLN warping factor estimates. Figure 5.5 shows the distribution of the number of utterances per speaker. It can be seen that about a third of the speakers have less than a hundred utterances per meeting.

Figure 5.7 illustrates (for a few selected speakers) how the estimated warping factor depends on the number of utterances from which it is estimated. More precisely the set of utterances was extended from 2 to 5, 10, 20, ... , 600; i.e. each time the previous subset was augmented by selecting at random some additional utterances (from the same speaker). This behaviour is seen for most speakers. Here CMN and CVN were also performed using different amounts of data. The ML esti-

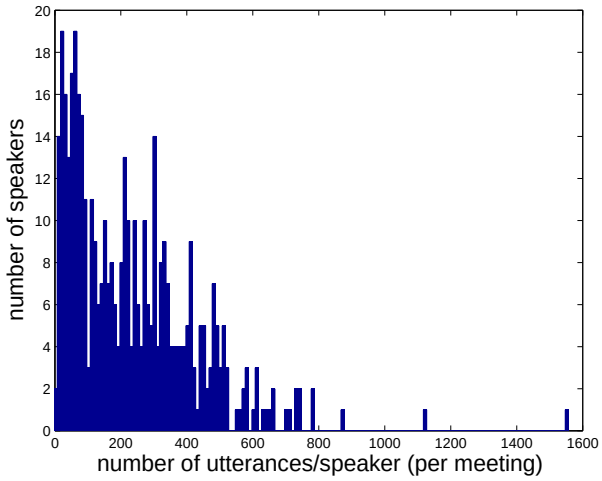



Figure 5.5: Distribution of the number of utterances per speaker (per meeting) for the ICSI training dataset

mate for the VTLN warping factor needs about twenty utterances before it begins to stabilise. Moreover in our experiments we computed a different warping factor for every speaker for every meeting and we observed that the estimated warping factors vary across different meetings. This is also shown in figure 5.6 where every vertical line goes from the minimum estimated warping factor value to the maximum and the point in the middle corresponds to the mean.

If the estimated warping factors do indeed correspond to normalising for variability in VTL between speakers, then we would expect their estimates to be more stable. This variability is highlighted if we compute the warping factor as a moving average across ten utterances (figure 5.8).

Multiparty meetings are characterised by a rich speaker turn structure, and we have investigated the influence of this on the warping factor estimates. In particular, we have investigated the dependence of the warping factor estimated for a speaker given the speaker that they are addressing. Accurate labelling of which participant(s) each utterance is addressed to is rather labour intensive—and can be difficult from an audio-only recording of a meeting (such as the ICSI meetings used in this experiment). We have made the approximation that a speaker speaking at a given time is addressing the most recent speaker (not including backchannel-type utterances).

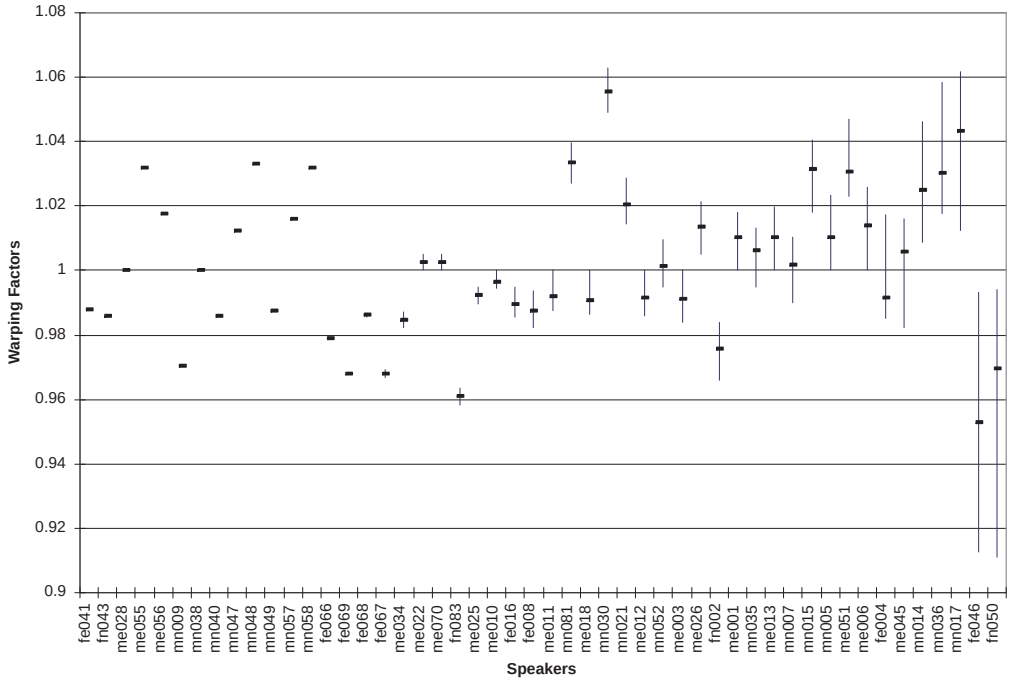



Figure 5.6: Warping factor variation across different meetings

For each utterance of each speaker we estimated a local warping factor using that utterance and the previous nine utterances. Our first question was whether the distribution of the warping factor for speaker A ($wf(A)$) has a dependence on the previous speaker. We used a hypothesis testing procedure to do this, where the null hypothesis $H_0$ is that the mean value of the warping factor of speaker $A$ given that s/he spoke after speaker $B$ is equal to the global warping factor value for A computed using all the data for that meeting. The probability to accept $H_0$ has been computed as $P(t)$ with:

$$t = \frac{wf(A) - \mu(wf(A|B))}{\frac{\sigma(wf(A|B))}{\sqrt{n}}} \tag{5.6}$$

where $\mu(wf(A|B))$ is the mean warping factor of A after B, $\sigma$ is the standard deviation and $n$ is the number of data (utterances) considered.

We studied eight meetings from the ICSI training dataset taken from different meeting types (Janin et al., 2003) and in a way that some of the speakers were present in more than one meeting. Using the Student t-test ($p = 0.05$) we found that for 84% of the speaker pairs the mean warping factor $\mu(wf(A|B))$ was significantly

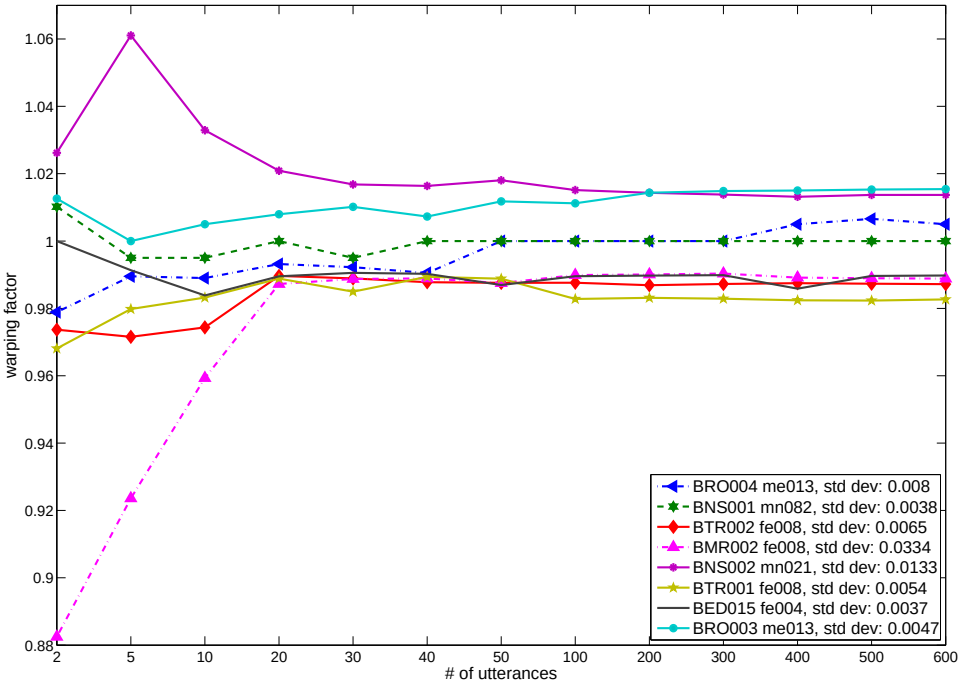



Figure 5.7: Trend of the warping factor values using different amount of utterances for the estimation

different from the global warping factor for $A$. Thus it appears that the turn taking process has some influence over warping factors. We also performed an unpaired t-test on the distributions of the warping factors of $A|i$ and $A|j$ for every speaker $i \neq A$ and $j \neq A$ with $i \neq j$. Here the null hypothesis $H_0$ is that the mean warping factor of $A|i$ and $A|j$ is the same. At 5% significance we found that in 78% of the cases the means of the two distributions were significantly different and we could reject the null hypothesis. Therefore we could hypothesise a possible reason could be that a given speaker $A$ will speak differently according to whom they are addressing and that the ML estimate of the warping factor could take this into account. This would be inline with the psycholinguistic theories on dialogue which will be described in section 5.4.2.

We performed a speech recognition experiment computing for every speaker a different warping factor for every possible speaker turn. We tested on a set of

5 complete meetings from the ICSI corpus, referred to as *amieval* (Hain et al., 2005c), which were excluded from the training. We compared normalising with a global warping factor per speaker with normalising with warping factors conditioned on the previous speaker. These experiments indicated that the WER obtained without VTLN (32.6%) was improved by both global speaker warping (27.1%) and speaker-conditioned warping (28.0%), but no improvement was found using speaker-conditioned warp factors compared to the use of the global warping factors.

### 5.4.2 A possible interpretation of the Warping Factors trend

Figure 5.8 (bottom) plots $wf(i|j)$ and $wf(j|i)$ against time. It shows the local warping factor estimated for speaker *me003* for utterances following utterances by speaker *me012* and vice versa (*me012* after *me003*) for the *BED003* ICSI meeting. This figure may be segmented in a sequence of intervals: segments where the two warping factor sequences show a similar behaviour (aligned) and segments where the warping factor dynamics are nonaligned. A similar structure can be also observed for the fundamental frequency F0 (figure 5.8, top) which plots the mean F0 value for each utterance.

A possible explanation of this structure could be a psycholinguistic account of dialogue, referred to as the *interactive alignment model* (Pickering and Garrod, 2004). In this account of dialogue it is argued that linguistic comprehension and production representations are shared between interlocutors in a dialogue *"making use of each others choices of words, sounds, grammatical forms, and meanings"* (Garrod and Pickering, 2004). This is referred to as *alignment* and it is argued that it occurs at many levels: phonetic, phonological, lexical, syntactic and semantic. Interactive alignment is manifested at these different levels within a dialogue, for example the use of similar syntactic structures, lexical repetitions, and common pronunciations. Krauss and Pardo (2006) have suggested that alignment in dialogue may be clearly observed at the phonological level and have presented preliminary evidence based on the vowel space (in terms of the first two formants) of interlocutors in two party dialogues. Their results suggest that the parties in a dialogue align at the phonological level as initially divergent pronunciations converge as the dialogue progresses. Kakita (1996) has presented evidence of the convergence of F0

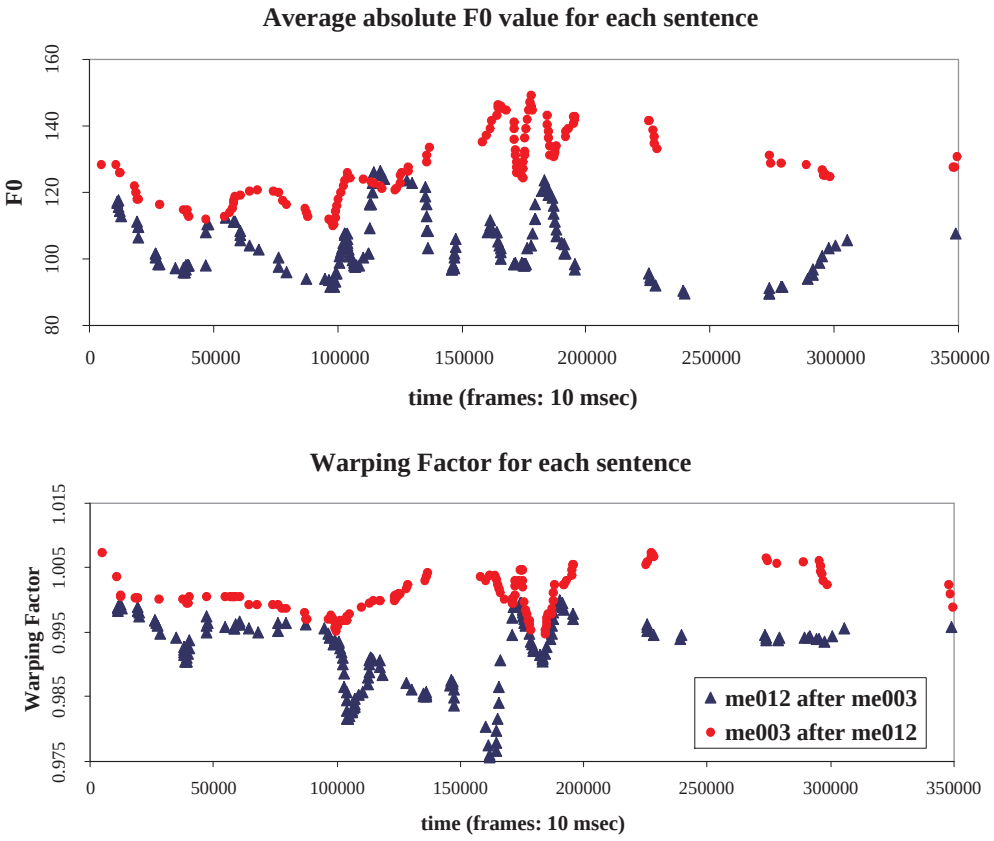



Figure 5.8: Trend of the warping factor of two speakers: me012 after me003 and me003 after me012

between parties in a dialogue.

The behaviour of the warping factor estimates could be explained with the interactive alignment account of dialogue. The estimated warping factors of two interlocutors are typically non-aligned at the start of a meeting, but can be seen to align (or at least go through phases of alignment) as the meeting progresses. In addition to the length of the vocal tract, there is a well known relationship between the VTLN warping factor and F0 (Eide and Gish, 1996; Wegmann et al., 1996) (since these two measures are both influenced by the vocal tract length). It is therefore natural to think that the same phenomenon observed by Krauss and Pardo (2006) could also be observed for the VTLN warping factor. Moreover our experiments considering only 8 of the ICSI meetings provide an initial analysis of the influence of the interactive alignment on the behaviour of warping factors; a wider set of experiments

(on constrained conversations collected ad hoc) would be helpful to further validate this theory.

Experiments on the use of a different warping factor per utterance (where warping factors are computed using a moving window of a given number of utterances) have been run. The results of these experiments can be seen in table 5.4. We tested both with a window of 10 and with a window of 5 utterances and we performed CMN and CVN using the sentences used to compute the warping factors for that particular utterance. The models used were the best VTLN CTS-INIA models. Unfortunately adopting a moving window for warping factor computation does not seem to help but it has to be observed that acoustic models where trained using global warping factors and it may be the case that using the same technique for warping factor estimation for the training set would make the models match the testing data better.

| | TOT | F | M | CMU | ICSI | LDC | NIST |
|---|---|---|---|---|---|---|---|
| VTLN (global) | 37.7 | 36.5 | 38.3 | 42.7 | 22.6 | 50.7 | 32.4 |
| VTLN (moving window: 10 ) | 38.6 | 37.6 | 39.1 | 42.7 | 25.2 | 51.2 | 33.0 |
| VTLN (moving window: 5 ) | 38.0 | 37.2 | 38.4 | 42.5 | 23.3 | 50.6 | 33.8 |

Table 5.4: Results (WER) of CTS-INIA VTLN models using a global warping factor (first row) compared to using a per utterance based warping factor computed with a moving window of 10 and 5 utterances (second and third rows respectively)

### 5.4.3 ML estimated warping factor values and formant positions

In order to better understand how warping factors estimated by ML are influenced by formant positions some experiments have been performed to study their relationship. To do so the entire ICSI training data set was taken under consideration. Using forced alignment to find vowel positions and the Snack toolkit[4], a mean value for $F0$, $F1$, $F2$, $F3$ and $F4$ for each occurrence of each vowel was computed. Then the estimated correspondent "global" warping factor has been associated to each of these occurrences. The analysis was based on MATLAB multiple linear regression

[4]Available from: www.speech.kth.se/snack/download.html

| vowel | $R(F_0)$ | $R(F_1)$ | $R(F_2)$ | $R(F_3)$ | $R(F_4)$ | $R(F_{0-4})$ |
|---|---|---|---|---|---|---|
| aa | 0.4848 | 0.5347 | 0.4223 | 0.3676 | 0.3036 | 0.6854 |
| ae | 0.4934 | 0.5350 | 0.5713 | 0.6632 | 0.4838 | 0.8003 |
| ah | 0.4909 | 0.3859 | 0.5685 | 0.5521 | 0.4111 | 0.7287 |
| ao | 0.4830 | 0.3854 | 0.3333 | 0.2162 | 0.3281 | 0.5962 |
| aw | 0.5177 | 0.6305 | 0.6076 | 0.5190 | 0.3607 | 0.7873 |
| ax | 0.4201 | 0.1523 | 0.3438 | 0.4918 | 0.3402 | 0.6088 |
| axr | 0.5204 | 0.1907 | 0.5039 | 0.1028 | 0.2225 | 0.6677 |
| ay | 0.5218 | 0.3763 | 0.5867 | 0.5670 | 0.4347 | 0.7809 |
| eh | 0.4908 | 0.5076 | 0.4807 | 0.5774 | 0.4496 | 0.7467 |
| er | 0.5202 | 0.2398 | 0.7077 | 0.1812 | 0.3474 | 0.7880 |
| ey | 0.5088 | 0.3455 | 0.5760 | 0.5334 | 0.5139 | 0.7478 |
| ih | 0.4477 | 0.2890 | 0.4612 | 0.6148 | 0.4920 | 0.7173 |
| iy | 0.5660 | 0.1513 | 0.4104 | 0.4540 | 0.4859 | 0.6866 |
| ow | 0.5003 | 0.3277 | 0.3694 | 0.4551 | 0.3689 | 0.6517 |
| oy | 0.6282 | 0.3343 | 0.4744 | 0.5347 | 0.3629 | 0.7740 |
| uh | 0.5065 | 0.2549 | 0.6262 | 0.4211 | 0.4633 | 0.7305 |
| uw | 0.5027 | 0.1202 | 0.3096 | 0.4917 | 0.4609 | 0.6334 |

Table 5.5: Correlation results based on phones between ML estimated warping factors and formant positions

function *regress* in a way similar to Dusan (2005a) where the correlation between speaker's height and formant positions in the TIMIT corpus was studied. The values of the correlation *R* for each vowel for every formant and for the combination of all formants can be seen in the table 5.5. Warping factors are highly correlated with formant positions altogether for most of the vowels, while correlation with each formant is smaller.

### 5.4.4 Experiments on making VTLN faster

As described in section 5.2, the estimation of VTLN warping factors using ML requires a preliminary transcription. Thus VTLN decoding is performed in two passes

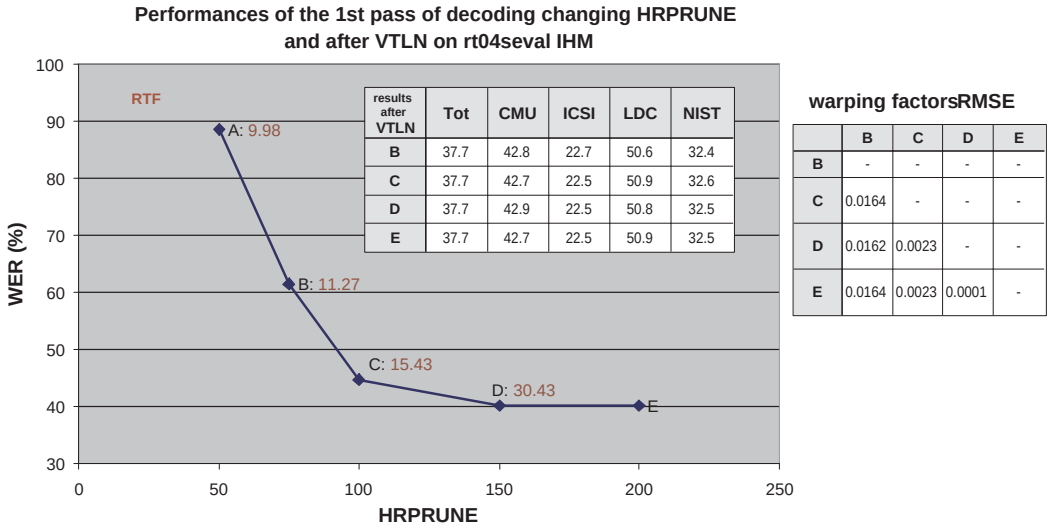
with the first one obtaining the preliminary transcription. Therefore we performed some experiments to evaluate if it is feasible to use a heavier pruning during the first decoding pass, in order to speed-up the process.

We reported the results of these experiments in figure 5.9: the main graph reports the WER of the first decoding in function of the beam searching log probability threshold for pruning where we also reported the real time factor (RTF) in red for each point; in the table on the right we report the root mean square error (RMSE) between the warping factors estimated using transcriptions of various quality (obtained with various pruning thresholds) [5]; in the table of results (in the middle of figure 5.9) the second pass decoding WER using features normalised with the warping factors estimated with the various quality transcriptions was reported. We measured a difference in the warping factor value estimated using various transcription qualities, observing RMSEs ranging from 0.016 between the most pruned and the less pruned system (B and E) and 0.0001 between the less pruned systems (E and D). Even so the WER after decoding was basically the same on all experiments, meaning that the quality of the first pass transcription does not exert a substantial influence on the second pass decoding result.

## 5.5 AMI meeting experiments

We also performed some VTLN experiments within a joint effort of the AMI ASR team for the automatic transcription of the entire AMI corpus. For these experiments the corpus was transcribed using a five–fold cross-validation technique (it was subdivided in five parts and acoustic models were trained on four parts and tested on the fifth part iteratively). An initial decoding was performed using non normalised acoustic models and MF-PLP features, then warping factors were estimated for the entire corpus and a system was trained and tested on VTLN HLDA MF-PLP features (that is 13 VTLN MF-PLP cepstral coefficients with $\Delta$s, $\Delta\Delta$s and $\Delta\Delta\Delta$s dimensionality projected from 52 to 39 dimensions using HLDA). Furthermore the same experiment was performed on close talking microphones using both manual and automatic segmentation. On the manual segmentation task we obtained

[5] System B provided a WER of around 60% with an RTF of 11.27, while system A provided a WER of almost 90% with an RTF of 10, therefore given the negligeable decrease in RTF compared to the large increase in WER, the ouput of system A was not considered for warping factor estimation.




| results after VTLN | Tot | CMU | ICSI | LDC | NIST |
|---|---|---|---|---|---|
| B | 37.7 | 42.8 | 22.7 | 50.6 | 32.4 |
| C | 37.7 | 42.7 | 22.5 | 50.9 | 32.6 |
| D | 37.7 | 42.9 | 22.5 | 50.8 | 32.5 |
| E | 37.7 | 42.7 | 22.5 | 50.9 | 32.5 |

warping factorsRMSE

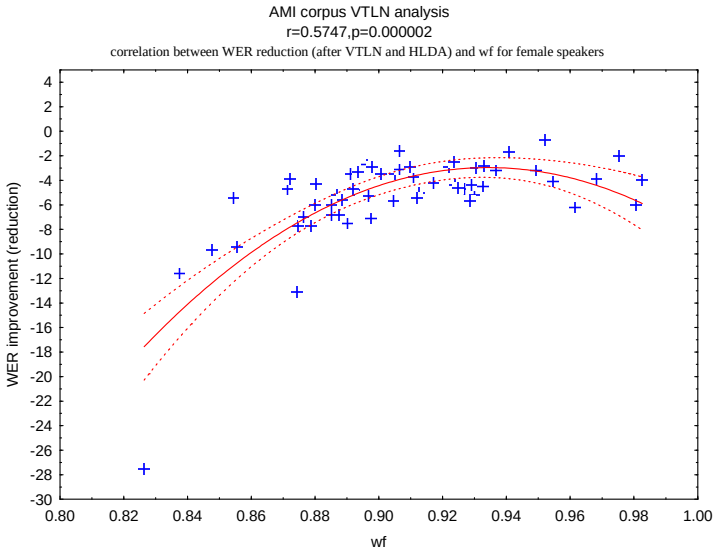
| | B | C | D | E |
|---|---|---|---|---|
| B | - | - | - | - |
| C | 0.0164 | - | - | - |
| D | 0.0162 | 0.0023 | - | - |
| E | 0.0164 | 0.0023 | 0.0001 | - |

Figure 5.9: Speeding up VTLN: on the left graph the WER of the first pass in function of the pruning decoding setting HRPRUNE (beam searching log probability threshold) with the corresponding real time factors (RTF) in red; in the table on top of the graph WERs after VTLN using the correspondent transcriptions obtained from the first pass decoding; on the right root mean square errors between the warping factors estimated using various transcription qualities

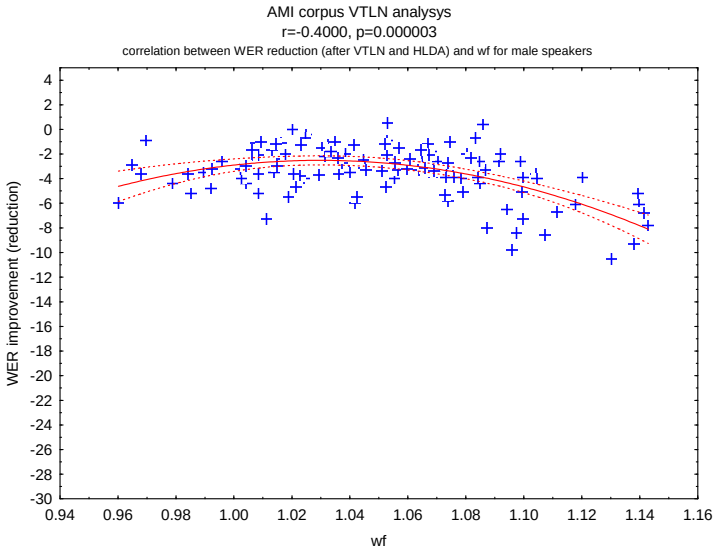
an overall reduction in WER from 43.2% to 39.4%.

This experiment was an excellent opportunity to look at some statistics of the warping factor behaviour since it was performed on a large amount of data. The correlation between the WER improvement from the non-normalised to the VTLN HLDA system was investigated. The change in WER is plotted against the warping factor value in figures 5.10(a) and 5.10(b) for female and male speakers respectively. Not surprisingly we can notice that the more the warping factor is smaller than 1 (in the case of female speakers) or larger than 1 (in the case of male speakers) the larger reduction in WER is obtained.

Finally we looked at the comparison of warping factors computed using the manual transcription and those computed using the first pass automatic transcription using manual segmentation which were also compared to the warping factor values estimated using the automatic segmentation. In table 5.6 we report the root mean square error between the warping factors estimated in these three configurations.




(a) Female speakers




(b) Male speakers

Figure 5.10: WER improvement vs warping factor values for the AMI corpus from a non-normalised system to a VTLN system with HLDA where $r$ indicates the correlation coefficient, $p$ is the statistical level of significance and quadratic regression lines along with the 95% confidence intervals were plotted using a statistics toolkit.

| RMSE | manual segm. manual transcr. | manual segm. automat. transcr. | automat. segm. automat. transcr. |
|---|---|---|---|
| man. segm. man. transcr. | ——- | 0.0115 | 0.0116 |
| man. segm. automat. transcr. | 0.0115 | ——- | 0.0026 |
| automat. segm. automat. transcr. | 0.0116 | 0.0026 | ——- |

Table 5.6: RMSE between warping factors computed using the manual transcription, the automatic transcription using manual segmentation and the automatic transcription using the automatic segmentation

It can be noticed that while using the automatic transcription (instead of the true manual transcription) there is a small difference in the estimated warping factors, there is not such a difference between the use of the automatic segmentation and the manual segmentation. However in section 5.4.4 we noticed that such small differences in the estimated warping factors do not affect the performances when they are used for the second pass of decoding (also observed by (Welling et al., 2002).

## 5.6 Conclusions

In this chapter we have studied the application of ML VTLN to multiparty conversations. We have found consistent improvements both in the conversational telephone speech and in the meeting domain (observing a relative WER reduction for both tasks of around 8%). Moreover we have studied the behaviour of the warping factors during multiparty conversations, finding that:

- The warping factor estimated for the current speaker is influenced by the conversational situation.
- Given the same speaker multiple ML VTLN frequency warping factors are found for different conversations (meetings) and within the same meeting

across time. We can hypothesise that this could be related to the phonological alignment observed by Krauss and Pardo (2006).

- The warping factor is also highly correlated with the fundamental frequency F0 and the higher order formants.

The correlation of warping factors with formant positions and with F0 motivated the experiments described in the next chapter about the use of a pitch adaptive spectral representation in conjunction with VTLN.

# Chapter 6

# Pitch adaptive spectral representations

## 6.1 Introduction

Frequency warping factors are known to be correlated with the fundamental frequency (Wegmann et al., 1996; Eide and Gish, 1996; Faria and Gelbart, 2005) being both influenced by the vocal tract length. It is therefore of interest to explore the use of a pitch-adaptive analysis. As it will be discussed in section 6.2, pitch-synchronous and pitch-adaptive representations were investigated in the context of speaker recognition (Ezzaidi and Rouat, 2000; Kim et al., 2004b; Zilca et al., 2003) and for small vocabulary ASR in the presence of noise (Ghulam et al., 2004; Bozkurt and Couvreur, 2005). However, investigation of pitch-adaptive representations for LVCSR has been rather limited.

In this chapter the use of spectral representations derived from STRAIGHT, a pitch-adaptive analysis developed by Kawahara et al. (1999), reviewed in section 6.3, is explored. This analysis results in a smoothed time-frequency representation from which it is possible to extract MFCCs and MF-PLP cepstral coefficients. These pitch-adaptive acoustic representations are combined with conventional representations both at the feature level using heteroscedastic linear discriminant analysis (HLDA, section 4.4.1) and at the decoding level using the ROVER technique to combine the outputs of multiple decodings (see section 4.4.2).

The combination of multiple acoustic feature streams has the potential to im-

prove the accuracy of automatic speech recognition (ASR) (Kirchhoff et al., 2000; Zhu et al., 2004; Zolnay et al., 2007; Schlüter et al., 2007; Hillard et al., 2007). Different acoustic representations have different strengths, and thus will tend to result in ASR systems that make different errors. The combination of acoustic feature representations is a way to exploit complementary information and to take advantage of the strengths of particular representations.

In section 6.4 a set of experiments using the combination of conventional and pitch adaptive spectral representations on three LVCSR tasks is outlined: transcription of dictated newspaper text (WSJCAM0); conversational telephone speech (CTS) recognition; and transcription of multiparty meetings using both close-talking and distant microphones. This set of experiments allowed to test the approach in a range of speaking styles and channel conditions. Although, the WSJCAM0 task consists of read speech using a close-talking microphone in a quiet environment, the other two tasks are more challenging. Both are concerned with spontaneous conversational speech. Moreover, CTS involves telephone speech which is subject to a bandpass filter that partly obscures the pitch, while the multiparty meetings were recorded in reverberant conditions with overlapping speakers. The situation is further complicated for the meeting task when multiple distant microphones are used to record the speech, and beamforming algorithms are applied to the recorded signals.

The results of the experiments reported in this chapter suggest that combining conventional and STRAIGHT-based acoustic features using HLDA results in a consistent decrease in the word error rates.

## 6.2 Pitch Adaptive Analysis

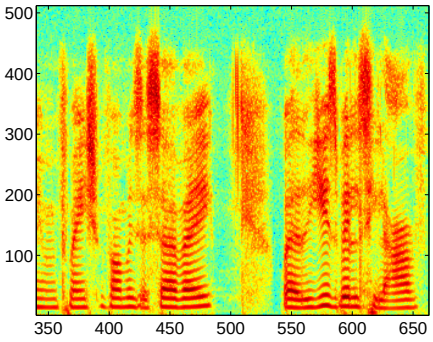
The short time Fourier transform (STFT) involves the computation of a separate Fourier transform for each frame of the signal waveform under a sliding window. This process is affected by the uncertainty principle, which states that it is impossible to have an arbitrary resolution both in time and frequency (Quatieri, 2001). The effect of this physical law is that the use of a long window in time (longer than 2 fundamental periods of the signal) leads to a good resolution in frequency and poorer time resolution, whereas a short window in time leads to the converse, good

time resolution at the cost of frequency resolution. For speech, in particular, the fundamental frequency of the signal varies over time, and if a fixed size window is applied, then its effect will be evident on the spectrum, particularly for high pitch speakers. This effect will be apparent even after the application of a Mel-scaled filterbank, in which the standard filter bandwidth in the lower frequency region is usually around 200–300 Hz. This is not broad enough to remove the harmonic structures for high pitched speakers, usually females, although it is able to provide a smooth representation for low pitched speakers (males) (Gu and Rose, 2001). This phenomenon can be observed in the left part of figure 6.1 which shows the conventional STFT spectrogram computed using a fixed 25 msec length Hamming window and the Mel scaling spectrograms derived from it using 48 (figure 6.1(c)) and 24 filters (figure 6.1(e)) for a high pitched female speaker (chosen for her small warping factor which in our meeting experiments was 0.837). It can be noticed that the pitch interference [1] which is particularly evident in the narrow-band SFTF spectrogram can still be seen in the Mel scaling spectrogram with 48 frequency bands and even with 24 bands (which is the number of filters used commonly in speech recognition front-ends). On the other hand this effect is filtered out for low pitched speakers such as in the example of figure 6.2 (male speaker with a warping factor of 1.16) where the harmonic lines due to the pitch artefacts are more narrowly spaced and are therefore smoothed out by the Mel scaling filterbank when 24 filters are used.

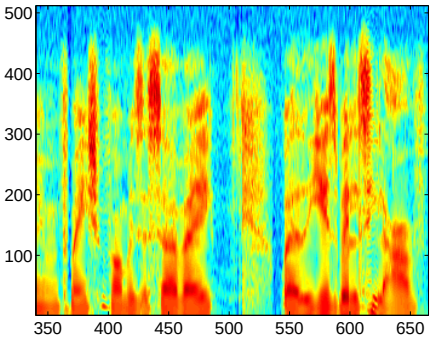
It is therefore of interest to investigate the use of a pitch-adaptive window that adapts according to the current estimate of the fundamental frequency for the extraction of conventional features such as MFCCs.

In speech synthesis and speech coding, where it is important to generate the correct fundamental frequency, pitch-synchronous analyses were well studied (Rao et al., 2003). The use of pitch-synchronous features has also been investigated for speaker recognition. Voice source information, as manifest in the pitch, is a speaker-specific characteristic, and source features derived from a pitch-synchronous analysis were proposed as features for speaker recognition by Ezzaidi and Rouat (2000). In this work the use of pitch synchronous features derived from the Instantaneous Frequency (IF) and the short term envelope (AM) for speaker identification was

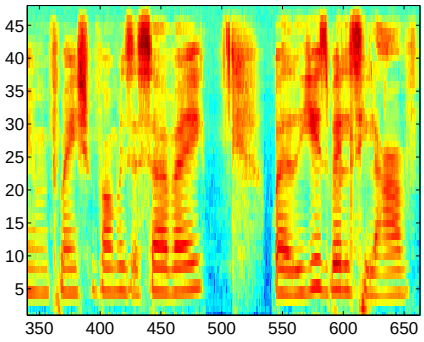
[1] The term “pitch interference” was used in Kawahara et al. (1999) to describe the influence of the pitch on the whole spectrogram, although it would be probably more precise to speak about pitch artefacts and this term will be employed in the rest of this thesis.



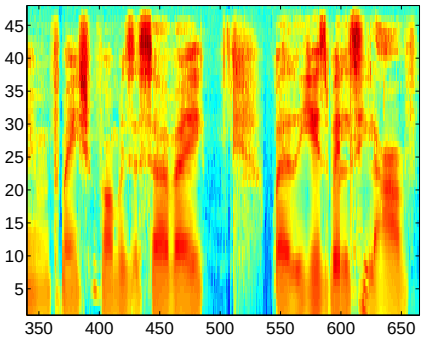

(a) STFT spectr.



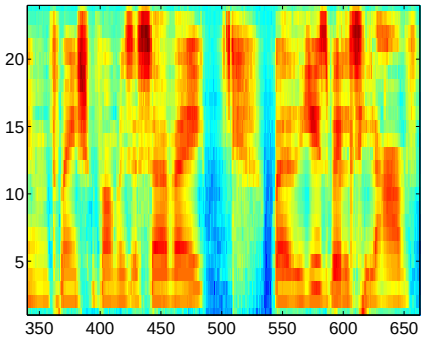

(b) STRAIGHT spectr.



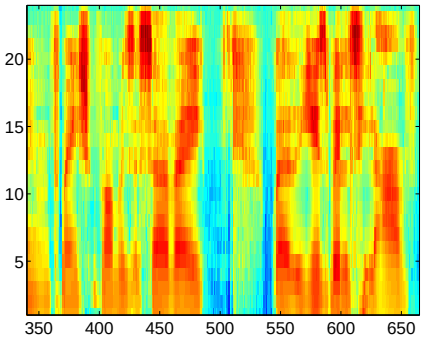

(c) Mel scale spectr.: 48 filters from STFT




(d) Mel scale spectr.: 48 filters from STRAIGHT

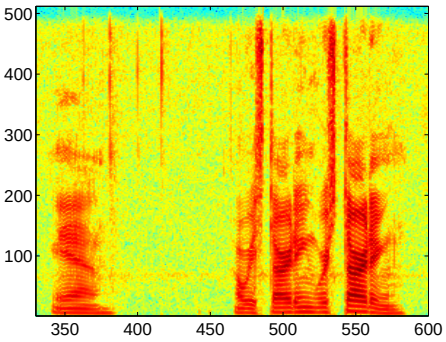



(e) Mel scale spectr.: 24 filters

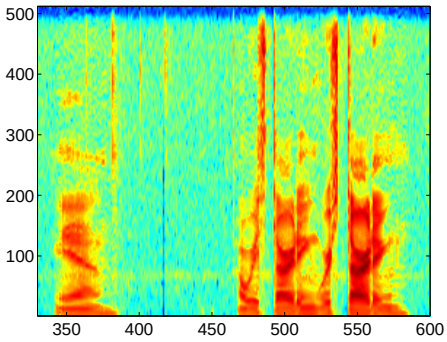



(f) Mel scale spectr.: 24 filters from STRAIGHT

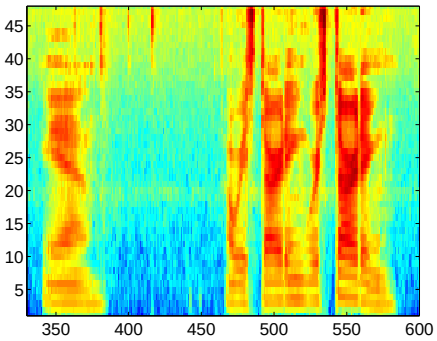
Figure 6.1: On the left: Short Time Fourier Transform and Mel scaling spectrograms using 24 and 48 filters for a rather high pitched female speaker; on the right: STRAIGHT and Mel scale spectrograms for the same speaker



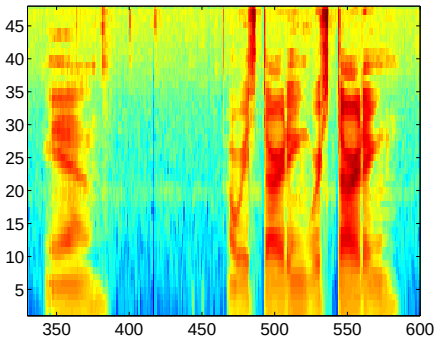

(a) STFT



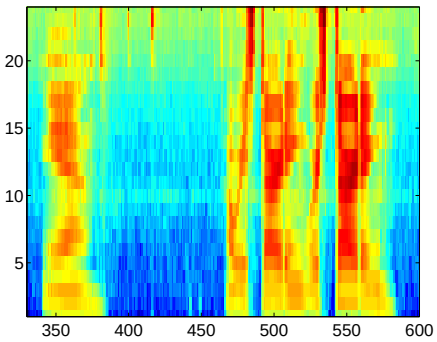

(b) STRAIGHT spectr.



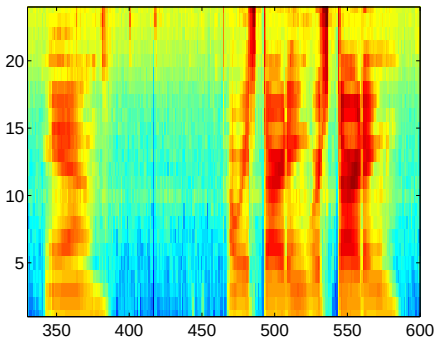

(c) Mel scale spectrogram: 48 filters




(d) Mel scale spectr.: 48 filters from STRAIGHT




(e) Mel scale spectrogram: 24 filters




(f) Mel scale spectr.: 24 filters from STRAIGHT

Figure 6.2: Short Time Fourier Transform and Mel scaling spectrograms using 24 and 48 filters for a low pitched male speaker

investigated. The signal is first filtered in subbands through a cochlear filterbank. From the output of each filter the IF and AM are then computed and averaged over time adopting intervals synchronised to the position of the Glottal Peak and a length equal to the pitch period $T_0$ ($[0\ T_0]$) and averaged over intervals $[T_0/2\ 3T_0/4]$ and $[3T_0/4\ T_0]$. In this way multiple measures of the spectral envelope and the instantaneous frequency are computed over a single pitch cycle. The performances of the newly obtained features are compared and combined with those of a system trained on MFCCs finding that AM, IF and MFCC features are complementary on the telephone speech corpus SPIDRE (a subset of the Switchboard corpus).

Zilca et al. (2003) proposed a pitch-adaptive analysis, referred to "depitching", which attempts to filter out pitch information from the speech signal, by having an integer number of pitch periods in every frame. The "depitch" procedure consists in 3 steps: the signal is windowed and LPC (Linear Predictive Coding) analysis is performed, then a single pitch cycle is extracted from the centre of the residual frame and it is interpolated to cover the duration of the entire frame. Then the interpolated residual is LPC filtered to get a speech signal. MFCCs are then extracted from the depitched waveform. Although depitched features alone resulted in lower accuracy for speaker recognition, combining systems using conventional and depitched MFCCs resulted in a significant improvement, with a more uniform error distribution across speakers.

The fundamental frequency provides prosodic information and information about the speaker but, for non-tonal languages, pitch is not used to encode words and phonemes. Therefore, factoring out the pitch information in speech recognition should result in a system with a greater speaker independence. Two basic approaches were reported in the literature: the use of pitch-synchronous or pitch-adaptive acoustic features, and acoustic models in which the pitch is explicitly modelled as a variable. An example of the latter approach (Stephenson et al., 2002) uses dynamic Bayesian networks (DBNs) in which the variables corresponding to the MFCCs are conditioned on the pitch, although this did not result in a significant improvement in accuracy. Some improvement on the use of pitch as an auxiliary feature in conjunction with tandem features was found by Magimai-Doss et al. (2004) especially in noisy conditions for the OGI numbers database.

Bozkurt and Couvreur (2005) investigated a pitch-synchronous analysis based

on group delay features (the negative of the differential phase spectrum) extracted using a window centered at the glottal closure instant, from which a phase spectrum was computed. Applying these features to ASR, in combination with MFCCs, resulted in a significant increase in accuracy over a baseline MFCC system on the AURORA–2 corpus. Holmes (2000) proposed the use of fixed length excitation synchronous windows for the Mel frequency cepstral coefficients extraction. These features were tested and compared with "fixed" analysis windows based features for various window lengths on a digit recognition task, finding a significant improvement using a 10 ms excitation synchronous window. An alternative pitch-synchronous representation, pitch synchronous zero crossing peak-amplitude (PS-ZCPA), has also shown some promise in reducing errors on noisy speech (the AURORA–2J corpus) (Ghulam et al., 2004).

Irino et al. (2002) employed the pitch-adaptive STRAIGHT representation, discussed in the next section, using it as the underlying spectral representation for the extraction of MFCCs. STRAIGHT-based MFCCs were compared with conventional MFCCs in HMM-based speech recognition on a small Japanese database, but no significant improvement in accuracy was observed. In this chapter, the use of STRAIGHT-based acoustic features is explored, in conjunction with speaker normalisation using VTLN, and in combination with conventional MFCC and MF-PLP features.

## 6.3 STRAIGHT based features

STRAIGHT (Kawahara et al., 1999) is a vocoder consisting of analysis and synthesis parts. The spectral analysis of STRAIGHT uses a pitch-adaptive window which gives equivalent resolution in both time and frequency domains. An interpolation is then performed on the partial information given by the adaptive windowing. This is achieved by using a second order B-spline as a smoothing function for surface reconstruction, constrained on the preservation of quantities such as the energy and the perceived loudness of the signal. This results in a smoothed time-frequency representation which is not affected by the artefacts arising from signal periodicity.

In this work STRAIGHT-based MFCCs were derived by replacing the classic STFT, which typically uses a Hamming window, with the STRAIGHT spectral anal-

ysis using the following window:

$$w(t) = \frac{1}{\tau_0} \exp\left(-\pi(t/\tau_0)^2\right) \tag{6.1}$$

$$W(\omega) = \frac{\tau_0}{\sqrt{2\pi}} \exp\left(-\pi(\omega/\omega_0)^2\right) . \tag{6.2}$$

This window is ideally Gaussian both in time and frequency and it was chosen by Kawahara et al. (1999) because of its isometric properties (it is the only smooth non-zero function which transforms to itself) and its unique property of minimum time-bandwidth product. The shape of the window depends on the estimated fundamental frequency $f_0 = 1/\tau_0 = 2\pi/\omega_0$. If we compare it with a 25 msecs Hamming window: for $f_0 \cong 80Hz$ they are almost equivalent; while for $f_0 < 80Hz$ the pitch adaptive window gives a better frequency resolution and lower temporal resolution; and for $f_0 > 80Hz$ it provides a better temporal resolution and lower frequency resolution.

The pitch used for the window computation can be estimated using various algorithms: TEMPO, the algorithm for pitch tracking provided in the STRAIGHT framework (Kawahara et al., 1999), is based on the use of the so-called *fundamentalness* measure, obtained using a wavelet Gabor filter designed to highlight the fundamental frequency (maximal filter output) and to reject harmonic replicas. However, other pitch trackers may be used and most of the experiments reported here employed the RAPT pitch tracking algorithm (Talkin, 1995), implemented as ESPS get_f0[2], which is based on cross-correlation in the time domain. As discussed further in section 6.4, although no significant difference between the use of the two pitch trackers was found when working on clean read speech, get_f0 proved to be more reliable for conversational telephone speech, as well as being more computationally efficient.

The STRAIGHT pitch spectrogram of a telephone speech signal is compared with a conventional STFT spectrogram in figure 6.3. The harmonic structure, visible in the STFT, is not present in the smoother STRAIGHT spectrogram. The lowest part of the figure shows the pitch value plotted along with the width of the analysis window in the time domain (measured at $1/3$ of the height of the window in number of samples), illustrating how the spectrogram resolution follows the value of the fundamental frequency of the signal. A reliable pitch estimate is important, since

[2]Available from: `http://www.speech.kth.se/snack/`

pitch tracking errors such as pitch doubling can lead to a very wide window in the frequency domain and poor spectral resolution. For unvoiced speech a default value of about 10ms was used for the window width (measured at $1/3$ of the maximum window amplitude), corresponding to a fundamental frequency of 160 Hz.

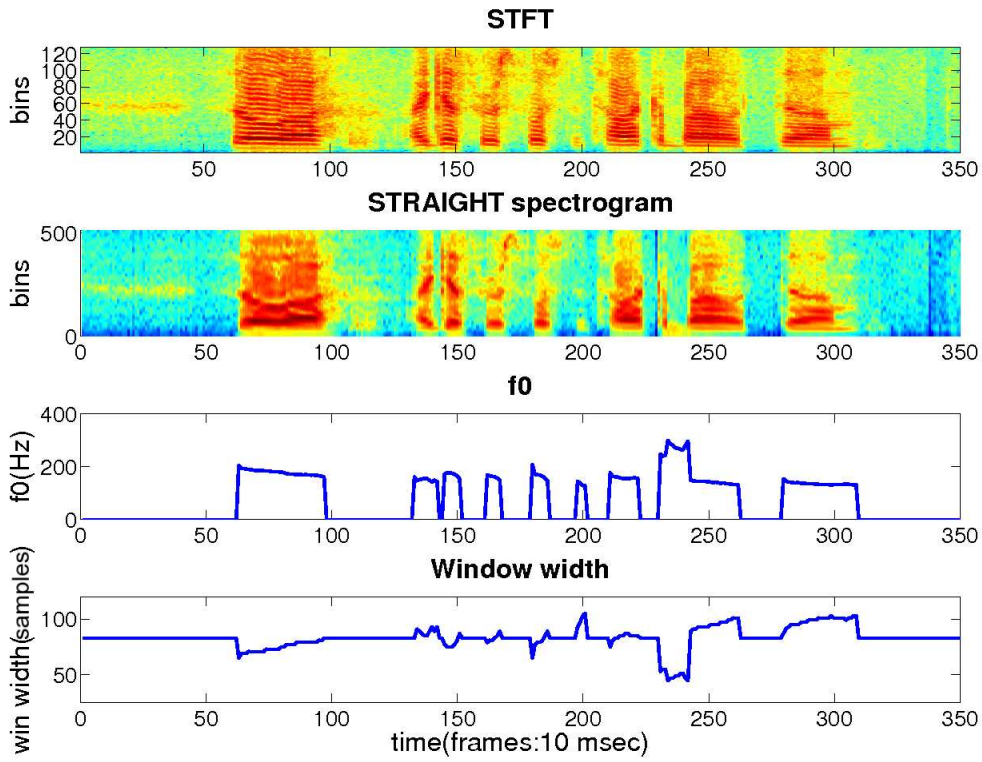



Figure 6.3: Example of STFT spectrogram, STRAIGHT spectrogram, f0 and spectral analysis window width in the time domain for a telephone speech signal, with a sample rate of 8 kHz.

Figure 6.4 shows a block diagram of the extraction procedure for STRAIGHT derived MFCCs. The log STRAIGHT (power) spectrogram is processed through a Mel scaled filterbank and decorrelated using the discrete cosine transform. A comparison of the output of the Mel-scaled filterbank for conventional MFCCs and STRAIGHT derived MFCCs can be observed in figures 6.1 and 6.2 for a high pitched and a low pitched speaker respectively. It can be noticed that the artefacts of the pitch, still present in the Mel scaled spectrogram of the conventional features for the high pitched speaker, is not present in the case of the STRAIGHT derived Mel

spectrogram which is smoother.

Our STRAIGHT derived MFCCs computation is similar to the feature extraction process presented in Irino et al. (2002) but here we perform a normal DCT instead of a warped DCT because we do not require feature inversion. MF-PLPs were also extracted from the log STRAIGHT spectrogram, by Mel scaling, followed by equal loudness pre-emphasis, cube root compression and linear predictive cepstral analysis. Figure 6.5 shows a block diagram of STRAIGHT PLP extraction. In addition, we employed a VTLN frequency warping procedure, shown in the figures and described below.

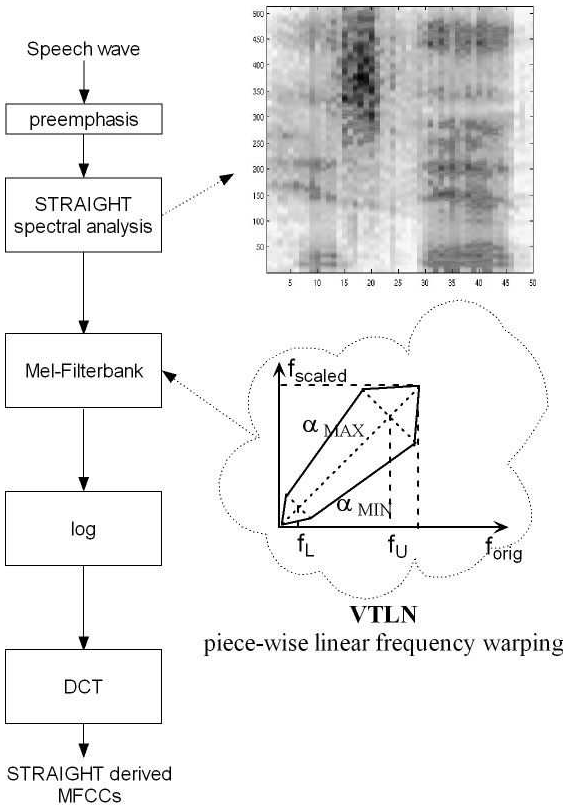



Figure 6.4: A block diagram of STRAIGHT MFCCs extraction with VTLN frequency warping

The centres of the filters of the Mel scaled filterbank are moved according to a piecewise linear frequency warping function where different warping factors $\alpha$ are defined for different frequency bandwidths (depicted in the VTLN box in figures 6.4 and 6.5) and described in more details in section 3.3.2.

As described in section 5.2 the warping factors are estimated using maximum

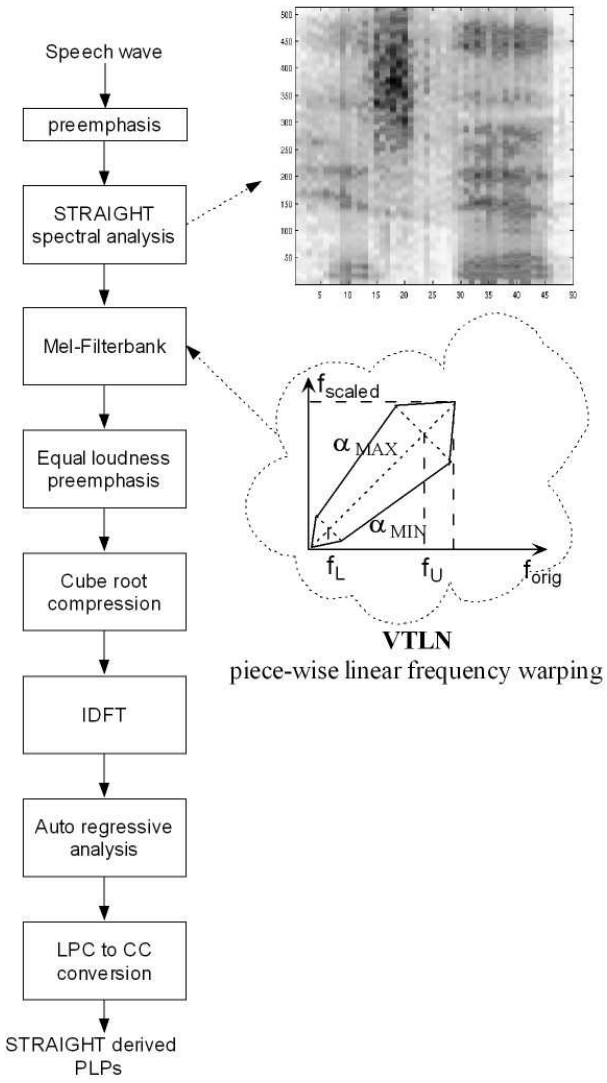



Figure 6.5: A block diagram of STRAIGHT PLPs extraction with VTLN frequency warping

likelihood in the acoustic model training process (Hain et al., 1999), such that the speaker-specific warp factor $\alpha$ is set to maximise the likelihood of the normalised acoustic observation feature vectors $X^{\alpha}$, given a transcription $W$ and an acoustic model $\lambda$ (Welling et al., 2002; Hain et al., 1999).

## 6.4 Experiments

VTLN warping factors attempt to normalise for the variation of the vocal tract length across different speakers. In our previous experiments about the use of VTLN on multiparty meetings, described in chapter 5, it was found that VTLN warping factors estimated using the ML method are not really constant over time. This variation was partly explained with the fact that warping factors are correlated with pitch. It is therefore of interest to investigate the use of a spectral representation which is less dependent on pitch, such as the pitch adaptive representation of STRAIGHT, in conjunction with VTLN.

STRAIGHT provides a smoother spectral representation conceived for speech modification and we expect VTLN, which performs frequency warping, to benefit from this smoother pitch independent spectral representation. The main goal of the experiments described in the next sections is to investigate ways of applying and benefiting the most from this representation: focusing on the VTLN speaker normalisation context and evaluating the proposed approach on a wide range of tasks corresponding to different challenging acoustical domains. In particular we expected that high pitched female speakers would benefit the most from a pitch adaptive representation; in fact for these speakers the Mel filters bandwidths are not sufficiently wide to smooth the harmonic lines due to pitch artefacts. Conventional MFCCs are affected by pitch artefacts, while STRAIGHT provides a smoother pitch adaptive spectral representation, sensitive to pitch tracking errors and sometimes too smooth and therefore less informative than the conventional STFT. Even if an absolute improvement over conventional features is desirable overall, we are aware that the new features have both pros and cons, thus the idea of combining them with conventional features was envisaged from the beginning. Several works in the literature have shown that, while it is sometimes difficult to get consistent improvements when new features are introduced, it is often possible to build more robust systems

combining new and conventional features: this is for example true for gammatone features (Schlüter et al., 2007), Tandem features (Zhu et al., 2004), and in particular for pitch synchronous features (Bozkurt and Couvreur, 2005; Ezzaidi and Rouat, 2000; Zilca et al., 2003).

In our work in order to exploit the advantages of both conventional and STRAIGHT representations, we combined them using HLDA. As mentioned in section 4.4, Schlüter et al. (2006) argued that numerical problems could arise when strongly correlated features are combined with LDA. Although it could be argued that STRAIGHT and conventional MFCCs were extracted in a similar way in our experiments, the correlation of these two feature streams is highly dependent on the window used in the particular instant of time considered, which on its turn depends on the pitch. Moreover the interpolation of the STRAIGHT spectrogram to compensate for pitch errors affects this representation differentiating it from the conventional STFT anyway. In fact the use of HLDA provides consistent improvements in all our experiments.

### 6.4.1 Experimental setup

Baseline acoustic models were trained using conventional MFCCs (computed with a 25ms window with a 10ms shift); for each domain we also trained models using STRAIGHT derived MFCCs. For each representation 12 cepstral coefficients plus the zeroth cepstral coefficient (C0) and first and second derivatives were also computed, resulting in a 39-element feature vector (13 coefficients + 13 $\Delta$ + 13 $\Delta\Delta$). The acoustic models were state clustered cross-word triphones with 16 mixture components per state. We also performed VTLN during both training and testing, using an iterative method which alternated the estimation of warping factors and the estimation of acoustic model parameters, described in detail in section 5.2. VTLN was applied both to the standard MFCC system and to the STRAIGHT derived MFCC system.

A number of experiments were carried out to determine the sensitivity of the STRAIGHT-based features to the pitch tracking algorithm that was used. An initial set of experiments employed the Keele pitch extraction reference corpus (Plante et al., 1995). This corpus features ten British English speakers reading a phonetically-balanced story, for which the fundamental frequency ground truth was obtained

from a laryngograph signal. The corpus is not large enough to re-estimate the acoustic models, and it is from a different domain to any of the domains studied here. Since it consists of British English read speech, we automatically transcribed the Keele corpus using WSJCAM0 models, described in detail in section 6.4.2, which are trained on British English read speech. Moreover we used the same language model adopted in the WSJCAM0 experiments, the MIT Lincoln Labs 20k Wall Street Journal trigram language model. The word error rates are rather high (over 40%) because we have used acoustic and language models from a different domain and no development data was available to adapt the models to this new domain. However it is possible to compare the performance of recognizers using STRAIGHT-based features. Therefore we extracted STRAIGHT derived MFCCs both using the reference pitch, and the TEMPO and the RAPT pitch trackers, observing less than 1% difference in word error rate between features using the ground truth pitch track (43.6%), versus features using the TEMPO or RAPT algorithms (both 44.7%). Although there is a small improvement in using the reference pitch tracks, we conclude that both of the automatic pitch tracking algorithms offer acceptable performances. It is likely that training with reference pitch tracks might result in further improvements, but a database suitable for large vocabulary speech recognition with laryngograph signals is not available.

For this data, and for WSJCAM0, the ASR performance for systems using TEMPO and get_f0 was almost identical. For the CTS domain we observed that get_f0 resulted in significantly lower word error rates compared with TEMPO (see table 6.3). Since get_f0 also has lower computational demands, we used this pitch tracker for all our experiments (except where stated).

Figure 6.6 shows a block diagram of the HLDA training process. VTLN features are extracted separately for conventional and STRAIGHT derived features and they are CMN and CVN normalised and concatenated. From these feature vectors (78 dimensional in this case) an HLDA transform is trained using LDA as an initialisation starting point. Then feature reduction is performed to the desired dimension (39 in this example) and triphone tied-clustered CMN/CVN HLDA models are trained.

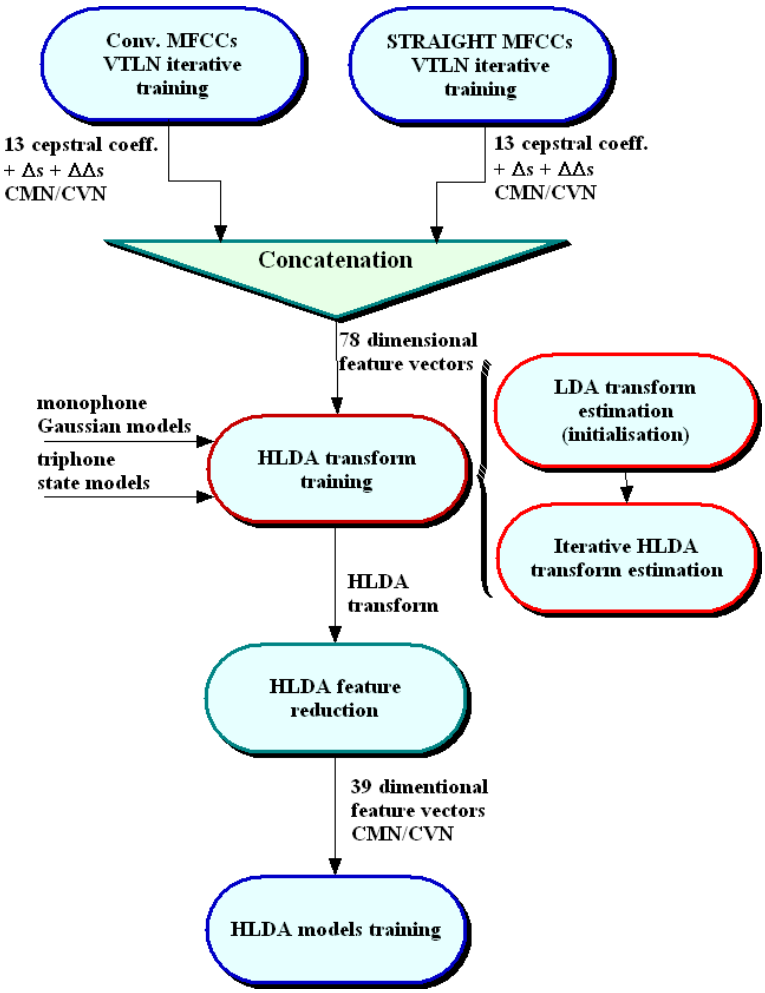



Figure 6.6: A block diagram of the HLDA training process

### 6.4.2 WSJCAM0

Our first set of experiments were performed on the WSJCAM0 corpus. As described in section 4.2.1 we trained on the official set denoted as si_tr and tested on the 20 000 words "open vocabulary" task development set (si_dt20a). We used the standard MIT Lincoln labs 20k Wall Street Journal trigram language model (Paul and Baker, 1992).

Table 6.1 shows our baseline results for this corpus. The top four lines show the word error rates for the conventional and STRAIGHT-based MFCC systems with and without VTLN. The conventional system has a lower word error rate than the

| | Dimension | # mixtures | Total | Female | Male |
|---|---|---|---|---|---|
| STD MFCCs | 39 | 30352 | 13.2 | 12.8 | 13.5 |
| STRAIGHT MFCCs | 39 | 29440 | 14.4 | 13.7 | 15.2 |
| STD MFCCs + VTLN | 39 | 31312 | 12.5 | 12.0 | 13.0 |
| STRAIGHT MFCCs + VTLN | 39 | 30720 | 13.0 | 12.5 | 13.5 |
| STRAIGHT + STD MFCCs + VTLN | 78 | 39152 | 15.4 | 15.2 | 15.7 |

Table 6.1: Word error rates on the WSJCAM0 si_dt20a dataset along with the model complexity (total number of mixture components), comparing conventional and Straight-based MFCCs, with and without VTLN. The combined system (bottom line) used concatenated feature vectors with no dimension reduction. d is the overall feature dimension.

STRAIGHT-based system, with the difference between the two reduced by half in the case of VTLN. The final line of the table shows the baseline feature combination experiment, in which the two feature vectors are simply concatenated at each frame, ending up with a 78-element feature vector. This resulted in a considerable increase to the word error rate, as might be expected due to the correlations in the resulting feature vectors. To reduce the correlations within the combined feature vector, and to reduce the overall dimensionality, we applied HLDA to the concatenated features. Table 6.2 summarises the main results of these experiments, in terms of the word error rates with respect to the reduced dimensionality and the choice of class in the HLDA.

The upper part of table 6.2 (*xwrd*) shows the obtained results when the HLDA statistics were estimated using the states of the cross-word triphone HMMs, a total of 1 927 classes. The lower part (*mono*) shows the results obtained using monophone mixture components as classes — 2 208 in total (46 phones, 3 states/phone, 16 gaussians/state). The *xwrd* condition is more focused on discriminating between triphone states, allowing a consistent definition between the HLDA classes and the acoustic triphone models (used during recognition). On the other hand the *mono* condition, using mixture components as classes, ensures that the distribution of the feature vectors corresponding to each class are more gaussian. For each HLDA type of class, we experimented with different dimension reductions, with the best results

| d | # mixtures | HLDA content/classes | Total | Female | Male |
|---|---|---|---|---|---|
| 52 | 38752 | xwrd/states | 12.3 | 11.9 | 12.8 |
| 39 | 37136 | xwrd/states | 12.4 | 12.1 | 12.7 |
| 52 | 37792 | mono/components | 12.3 | 11.9 | 12.8 |
| 39 | 35472 | mono/components | 12.1 | 11.4 | 12.8 |

Table 6.2: Error rates and model complexity (number of mixture components) after combining conventional and STRAIGHT derived MFCCs using HLDA, testing on WSJCAM0 si_dt20a. The xwrd/states condition indicates that the states of cross-word triphone models are used as HLDA classes; the mono/components condition indicates that Gaussian components of monophone models are used as HLDA classes.

being obtained with a reduction from 78 to 39 dimensions. For comparison we also show results for 52 dimensions. The best results were achieved using monophone state mixture components as classes, yielding 3.2% relative improvement compared to the baseline standard MFCC system.

### 6.4.3 Conversational Telephone Speech

The next set of experiments on CTS data, are based on the 72 hour training set described in section 4.2.2 (*ctstrain04sub*) and on the NIST hub5 *eval01* test dataset.

We used clustered cross-word triphone acoustic models with about 3 600 tied states. For this task we conducted several experiments in which we compared the accuracies of systems using conventional and STRAIGHT derived MFCCs, with and without cepstral mean and variance normalisation (CMN/CVN), and with and without VTLN. We also compared the use of the TEMPO and get_f0 pitch trackers for STRAIGHT, in this case on systems without normalisation (no CMN/CVN and no VTLN). We used the same trigram language model in all cases, with a vocabulary of 50 000 words, trained on various additional sources including web data, broadcast news transcripts and email text (more details can be found in section 4.3.2).

Word error rates for various configurations are shown in table 6.3. The first three rows show results in the case of no normalisation, including a comparison between TEMPO and get_f0 pitch trackers for STRAIGHT. Conventional MFCCs result in

the best performance, and get_f0 gives a significant decrease in word error rate of 4% relative compared with TEMPO. We note that telephone speech is significantly more challenging for pitch tracking owing to the bandpass filtering and other channel effects (Rabiner et al., 1976). Applying CMN/CVN and VTLN results in a decrease in word error rate by over 10% for both conventional and STRAIGHT-based systems. As in the WSJCAM0 task, the gap between conventional and STRAIGHT-based systems is considerably reduced when VTLN is applied: the difference in WER is reduced from 3% to 1.6%. This is evidence that the smoother spectral representation offered by STRAIGHT is well-matched to VTLN, which uses frequency warping to normalise speech to increase speaker independence.

We combined the two normalised systems using HLDA both using triphone states and monophone mixtures as classes. Each combination yielded an 8% relative improvement compared to the baseline, a conventional MFCC system with VTLN and CMN/CVN. The improvements are consistent for both female and male speakers and for all the testing subsets. This is a significant result, since the baseline system is strong, given the training set of 72 hours.

### 6.4.4 Multiparty Meetings

Our final, and most extensive, set of experiments is in the domain of multiparty meetings. For this task we trained separate systems for close talking microphones (IHM) and distant microphones (MDM) on a set of about 100 hours of meeting speech (described in section 4.2.3) and tested on the NIST Rich Transcription Spring 2004 evaluation data both in the IHM and MDM condition.

We used clustered cross-word triphone acoustic models with 16 mixture components per state and around 4 400 tied states in total, and trained a set of models for each condition using VTLN. We used a vocabulary of 50 000 words and a trigram language model trained on web collected data, meeting data and CTS data as described in section 4.3.2. As for the other tasks we constructed baseline systems using the conventional and STRAIGHT-based systems independently, then produced a combined feature stream by concatenation and dimension reduction using HLDA (using both monophone Gaussian components and cross-word triphone states as classes). The resulting systems corresponded to a sub-system (denoted as VTLN enhanced P1) of the state-of-the-art AMI-ASR meeting transcription system (Hain

| | # mixtures | TOT | F | M | SW1 | S23 | Cell |
|---|---|---|---|---|---|---|---|
| MFCC (no CMN/CVN) | 86288 | 42.7 | 41.8 | 43.6 | 36.5 | 43.3 | 47.9 |
| STRAIGHT (TEMPO no CMN/CVN) | 83018 | 47.6 | 46.0 | 49.1 | 40.7 | 49.0 | 52.8 |
| STRAIGHT ($get\_f0$ no CMN/CVN) | 83296 | 45.7 | 44.5 | 46.9 | 40.0 | 46.6 | 50.3 |
| MFCC+CMN/CVN+VTLN | 85836 | 37.6 | 37.0 | 38.3 | 31.8 | 37.1 | 43.5 |
| STRAIGHT ($get\_f0$) +CMN/CVN+VTLN | 84197 | 39.2 | 38.2 | 40.1 | 33.6 | 39.0 | 44.5 |
| MFCC + STRAIGHT ($get\_f0$) +CMN/CVN+VTLN+HLDA(xwrd) | 102560 | 34.6 | 33.6 | 35.6 | 28.3 | 34.5 | 40.5 |
| MFCC + STRAIGHT ($get\_f0$) +CMN/CVN+VTLN+HLDA(mono) | 98928 | 34.7 | 33.8 | 35.6 | 28.6 | 34.7 | 40.5 |

Table 6.3: Word error rates (and model complexity in terms of total number of mixture components) on the CTS NIST Hub5 eval01 data for conventional and STRAIGHT derived MFCCs, and their combination using HLDA. TEMPO and get_f0 pitch trackers are compared for Straight features (lines 2–3). Both triphones states and monophone mixture components are used as HLDA classes for a feature reduction from 78 to 39 dimensions (lines 6–7). CMN and CVN are cepstral mean and variance normalisations. Tot: total WER; M: WER for male speakers; F: WER for female speakers

et al., 2007b) which participated in the NIST RT evaluation 2006, with the difference that MFCC features were used rather than MF-PLP features. Moreover both the IHM and MDM models used in the experiments described in this chapter were trained on meeting data only (there is no MAP adaptation from the CTS domain, which was used for the NIST evaluations).

The results for the IHM condition are shown in table 6.4. The STRAIGHT derived MFCCs result in slightly higher word error rates than conventional MFCCs; we note that pitch extraction is also challenging in the meeting domain. Lower error rates are observed for female speakers using STRAIGHT, while for male speakers lower error rates are observed for conventional MFCCs. Combination of the two systems using HLDA with monophone Gaussian component classes results in an absolute reduction in word error rate of 1.8% (5% relative) compared with the base-

| | # mixtures | TOT | F | M | CMU | ICSI | LDC | NIST |
|---|---|---|---|---|---|---|---|---|
| MFCC+VTLN (A) | 70304 | 38.4 | 38.5 | 38.3 | 42.7 | 23.9 | 52.1 | 30.9 |
| STRAIGHT+VTLN (B) | 69264 | 39.3 | 38.3 | 39.7 | 44.7 | 24.8 | 53.1 | 31.2 |
| MFCC+STRAIGHT +VTLN | 88275 | 42.1 | 44.4 | 41.0 | 45.6 | 28.5 | 55.4 | 37.0 |
| MFCC+STRAIGHT VTLN+HLDA xwrd (E) | 88400 | 37.3 | 37.6 | 37.2 | 41.4 | 23.8 | 51.9 | 29.4 |
| MFCC+STRAIGHT VTLN+HLDA mono (F) | 83312 | 36.6 | 36.3 | 36.7 | 41.0 | 22.5 | 51.2 | 28.5 |

Table 6.4: Word error rates (and model complexity in terms of number of mixture components) for meeting transcription (IHM condition) using the RT04seval testing set. Results are given for baseline systems using conventional and Straight-derived MFCCs, and for combined feature vectors obtained using HLDA. Tot: total WER; M: WER for male speakers; F: WER for female speakers.

line conventional MFCCs.

Word error rates for the MDM condition are shown in table 6.5. In this case there is a 2% absolute difference between the baseline conventional and STRAIGHT systems, which is larger than for the IHM case. Beamformed signals from distant microphones have increased additive channel noise, compared with the IHM condition, leading to less reliable pitch tracking, and hence less reliable estimates of the pitch-adaptive window in STRAIGHT. However, the combination of the two systems by HLDA using monophone Gaussian classes results in a substantial decrease in word error rate of 3.6% absolute (7.3% relative), which is consistent over the different subsets.

There is also a large difference between word error rates for male and female speakers. Beamforming is known to have less directionality at lower frequencies, while it has some aliasing at higher frequencies. Since, in male voices, information content and the fundamental frequency is concentrated at lower frequencies, it is possible that the higher error rate observed results from this limited directionality at low frequencies and therefore less reliable pitch tracking.

| | TOTAL | Female | Male | CMU | ICSI | LDC | NIST |
|---|---|---|---|---|---|---|---|
| MFCC+VTLN | 49.5 | 46.8 | 50.8 | 55.7 | 26.2 | 60.1 | 33.1 |
| STRAIGHT+VTLN | 51.5 | 48.6 | 52.9 | 57.4 | 26.2 | 63.4 | 34.6 |
| MFCC+STRAIGHT VTLN+HLDA xwrd | 46.8 | 42.2 | 49.1 | 52.5 | 24.3 | 58.1 | 29.5 |
| MFCC+STRAIGHT VTLN+HLDA mono | 45.9 | 42.7 | 47.4 | 50.8 | 21.3 | 57.7 | 30.1 |

Table 6.5: Word error rates for meeting transcription (MDM condition) using the RT04seval testing set. Results are given for baseline systems using conventional and Straight-derived MFCCs, and for combined feature vectors obtained using HLDA.

### 6.4.5 Further experiments on meetings

Higher order cepstral coefficients are known to be the most affected by the spectral harmonic components due to the pitch (Irino et al., 2002), hence systems using conventional MFCCs typically limit their dimensionality to twelve coefficients plus C0 or the log energy. However, using the smoothed STRAIGHT spectral representation, which is not affected by spectral harmonics, we should be able to exploit the information in higher order coefficients. To assess this possibility, we carried out a set of experiments using both the first 17 and the first 21 cepstral coefficients (plus C0) and their first and second temporal derivatives, resulting respectively in 51- and 63-dimension acoustic feature vectors. Experiments were performed in the IHM meeting domain both for the STFT-based MFCCs and our pitch-adaptive MFCCs. In practice the extraction of higher order cepstral coefficients was carried out by simply taking the first 17 and the first 21 coefficients output of the DCT block respectively.

The results of these experiments are shown in table 6.6, where we repeat the results of the 39-dimension systems to facilitate comparison. It is interesting to observe that the systems based on 21 and 17 STRAIGHT derived MFCCs have a lower word error rate than both 13, 17 and 21 conventional MFCC based systems. In particular the higher order MFCC system yields a greater error rate for female

speakers ($5^{th}$ row of table 6.6) compared to the higher order STRAIGHT derived MFCC systems ($6^{th}$ row): this is due to the fact that for high pitched speakers the Mel filter bandwidths are not sufficiently broad to remove the harmonic structure which affects the higher order coefficients. On the other hand STRAIGHT derived features, which are not influenced by pitch harmonics, are able to exploit the information of higher order coefficients even for female speakers for which they perform significantly better than STFT based features.

As an analysis experiment to have an idea of the contribution of the higher order coefficients, both conventional and derived from STRAIGHT, we concatenated the first 13 cepstral coefficients derived from the STFT and those from the $14^{th}$ to the $21^{th}$ derived from STRAIGHT and viceversa. The results of these experiments are shown in the $7^{th}$ and $8^{th}$ row of table 6.6. While taking the first coefficients from the conventional feature stream and those of higher order from the STRAIGHT derived MFCCs yields even a small improvement compared to the 63 dimensional STRAIGHT MFCCs alone, the opposite results in a degradation compared to both 63 dimensional setups. This is what we would expect and confirms that STRAIGHT derived higher order cepstral coefficients are responsible for a significant improvement while those derived from the STFT (being affected by the pitch artefacts the most) have a negative effect on the performances of the system.

HLDA combination based on monophone gaussians as classes was performed to combine the best performing systems for STRAIGHT derived and conventional MFCCs (the 63 dimension systems) reducing from 126 to both 39 and 63 dimensions. As can be seen in the second-last and last row of table 6.6 the best result is obtained reducing to 63 dimensions (probably because reducing to 39 we throw away too much information). Moreover the combination of higher dimensional features yields a significant 9% relative WER reduction compared to the baseline 39 dimension conventional MFCCs system.

We also performed some experiments on the use of STRAIGHT for MF-PLP extraction. Here a PLP implementation based on that of HTK (Young et al., 2006) was used, where the Mel frequency scaling is performed on the STRAIGHT spectrogram. Similarly to MFCCs, twelve cepstral coefficients plus C0 were extracted along with their first and second derivatives. WERs of systems based on STRAIGHT derived MF-PLPs were compared with those of conventional MF-PLPs extracted by HTK

| | d | # mixt. | TOT | F | M | CMU | ICSI | LDC | NIST |
|---|---|---|---|---|---|---|---|---|---|
| MFCC+VTLN (A) | 39 | 70304 | 38.4 | 38.5 | 38.3 | 42.7 | 23.9 | 52.1 | 30.9 |
| STRAIGHT+VTLN (B) | 39 | 69264 | 39.3 | 38.3 | 39.7 | 44.7 | 24.8 | 53.1 | 31.2 |
| MFCC+VTLN | 51 | 78784 | 37.1 | 37.9 | 36.7 | 41.8 | 22.4 | 51.0 | 30.7 |
| STRAIGHT+VTLN | 51 | 77184 | 36.9 | 36.5 | 37.1 | 41.8 | 22.6 | 50.4 | 30.1 |
| MFCC+VTLN (C) | 63 | 82432 | 37.1 | 38.5 | 36.4 | 41.3 | 22.2 | 51.5 | 31.2 |
| STRAIGHT+VTLN (D) | 63 | 81564 | 36.7 | 36.4 | 36.8 | 41.0 | 22.3 | 50.8 | 30.0 |
| 13 conv. + 8 STRAIGHT MFCCs | 39+24 | 83024 | 36.4 | 37.1 | 36.0 | 40.4 | 22.7 | 50.1 | 29.6 |
| 13 STRAIGHT + 8 conv. MFCCs | 39+24 | 81456 | 37.7 | 39.0 | 37.1 | 42.9 | 23.3 | 51.4 | 30.6 |
| MFCC 63 +STRAIGHT 63 VTLN+HLDA mono | 39 | 84304 | 35.8 | 35.7 | 35.8 | 39.8 | 21.8 | 50.8 | 27.8 |
| MFCC 63 +STRAIGHT 63 VTLN+HLDA mono | 63 | 99184 | 34.9 | 35.8 | 34.5 | 38.7 | 21.2 | 48.9 | 28.5 |

Table 6.6: Extended dimensionality experiment on RT04seval testing set using VTLN features for the IHM condition. From top to bottom: conventional MFCCs 39 dimensions; STRAIGHT MFCCs 39 dimensions; conventional MFCCs 51 dimensions, STRAIGHT derived MFCCs 51 dimensions; conventional MFCCs 63 dimensions, STRAIGHT derived MFCCs 63 dimensions; concatenation of the first 13 conventional MFCCs and from the $14^{th}$ to the $21^{st}$ STRAIGHT MFCCs; concatenation of the first 13 STRAIGHT MFCCs and from the $14^{th}$ to the $21^{st}$ conventional MFCCs; combination of the 63 dimensional systems using HLDA with monophone mixtures as classes reducing from 126 to 39 and 63 dimensions. The model complexity in terms of total number of mixture components has also been reported.

| | TOT | F | M | CMU | ICSI | LDC | NIST |
|---|---|---|---|---|---|---|---|
| MF-PLP+VTLN (G) | 37.4 | 35.8 | 38.3 | 42.5 | 23.3 | 50.8 | 30.4 |
| STRAIGHT MF-PLP +VTLN (H) | 38.4 | 37.4 | 38.9 | 43.7 | 24.4 | 51.9 | 30.3 |
| MF-PLP+STRAIGHT MF-PLP VTLN+HLDA mono (I) | 36.2 | 36.0 | 36.3 | 40.0 | 22.4 | 51.0 | 28.5 |

Table 6.7: MF-PLP experiment on RT04seval testing set using VTLN features for the IHM condition. From top to bottom: conventional MF-PLPs 39 dimensions; STRAIGHT MF-PLPs 39 dimensions; HLDA combination from 78 to 39 dimensions using monophone mixtures as classes.

and these two feature streams were concatenated and reduced through HLDA from 78 to 39 dimensions using monophone mixture components as classes. Results are shown in table 6.7. Word error rates were somewhat lower both for the individual feature systems and for the combination through HLDA, compared with the MFCC experiments. The combination by HLDA yields a word error rate reduction of 1.2% absolute (3.2% relative) compared with conventional PLPs.

### 6.4.6 ROVER experiments on meetings

To fully exploit the complementarity of conventional and pitch adaptive representations, we performed combination experiments at the system level using majority voting ROVER for the IHM condition of the meeting domain. We considered all the different IHM systems discussed in the previous subsections denoted with an alphabetic letter. Results are reported in table 6.8, where we also present WERs for the ROVER oracle to provide a lower bound on the achievable word error rates for each combination. Results for each individual system are reported in tables 6.4, 6.6 and 6.7, and each of the nine systems is identified by a letter. *A* and *B* denote the conventional and STRAIGHT derived systems for lower order MFCCs, while *C* and *D* are the same but for higher order MFCCs; *E* and *F* are the HLDA combinations of *A* and *B* with monophone Gaussian classes and triphone state classes respectively; finally *G* and *H* are the MF-PLP systems from conventional and STRAIGHT derived spectral representations, while *I* is their combination using HLDA and monophone

| | ROVER voting | | | | | | |
|---|---|---|---|---|---|---|---|
| | TOT | F | M | CMU | ICSI | LDC | NIST |
| E | 37.3 | 37.6 | 37.2 | 41.4 | 23.8 | 51.9 | 29.4 |
| F | 36.6 | 36.3 | 36.7 | 41.0 | 22.5 | 51.2 | 28.5 |
| I | 36.2 | 36.0 | 36.3 | 40.0 | 22.4 | 51.0 | 28.5 |
| A C G | 36.0 | 35.8 | 36.1 | 40.6 | 22.0 | 49.8 | 29.0 |
| B D H | 36.4 | 35.2 | 37.0 | 41.7 | 22.2 | 49.8 | 28.8 |
| A B C D G H | 34.9 | 33.5 | 35.6 | 39.8 | 21.0 | 48.5 | 27.2 |
| A B C D E F | 34.1 | 33.3 | 34.5 | 38.7 | 20.5 | 47.6 | 26.8 |
| A B C D | 34.9 | 33.4 | 35.6 | 39.8 | 21.0 | 48.5 | 27.1 |
| G H I | 35.4 | 34.3 | 35.9 | 40.0 | 21.5 | 49.3 | 27.8 |
| A B E F | 35.1 | 34.4 | 35.5 | 39.8 | 21.3 | 49.2 | 27.2 |
| A B G H | 36.5 | 35.1 | 37.2 | 41.8 | 22.6 | 49.7 | 28.8 |
| A B E F G H I | 34.9 | 33.8 | 35.4 | 39.7 | 21.1 | 48.8 | 26.8 |
| A B C D E F G H I | 33.8 | 32.6 | 34.4 | 38.4 | 20.1 | 47.2 | 26.6 |

| | ROVER oracle | | | | | | |
|---|---|---|---|---|---|---|---|
| | TOT | F | M | CMU | ICSI | LDC | NIST |
| A C G | 27.9 | 26.8 | 28.4 | 31.8 | 15.9 | 40.1 | 21.0 |
| B D H | 29.6 | 28.4 | 30.2 | 34.6 | 17.0 | 41.4 | 22.6 |
| A B C D G H | 23.9 | 22.6 | 24.6 | 28.1 | 13.2 | 34.5 | 17.3 |
| A B C D E F | 22.4 | 21.3 | 23.0 | 26.3 | 12.3 | 33.0 | 15.6 |
| A B C D | 26.2 | 25.2 | 26.7 | 30.6 | 14.5 | 37.6 | 19.4 |
| G H I | 27.3 | 25.8 | 28.1 | 31.5 | 15.7 | 38.9 | 20.6 |
| A B E F | 25.9 | 24.5 | 26.6 | 30.0 | 14.6 | 37.7 | 18.5 |
| A B G H | 28.0 | 26.3 | 28.9 | 32.8 | 16.1 | 39.7 | 20.9 |
| A B E F G H I | 23.0 | 21.3 | 23.9 | 27.0 | 12.9 | 33.5 | 16.2 |
| A B C D E F G H I | 20.9 | 19.5 | 21.6 | 24.7 | 11.4 | 30.6 | 14.5 |

Table 6.8: System level combination in the meeting domain (IHM condition) on RT04seval IHM, using ROVER. The left hand tables show majority voting ROVER results and the right shows ROVER oracle results for comparison. Nine systems are combined, labelled A–I. $A$ and $B$ denote the conventional and STRAIGHT derived systems for 39 dimensional MFCCs, while $C$ and $D$ are the same but for 63 dimensions; $E$ and $F$ are the HLDA combinations of $A$ and $B$ with monophone Gaussian classes and triphone state classes respectively; finally $G$ and $H$ are the MF-PLP systems from conventional and STRAIGHT derived spectral representations (39 dimensions), while $I$ is their combination using HLDA and monophone Gaussian classes. Results for the individual systems are shown in tables 6.4, 6.6 and 6.7, while in this table we repeated the HLDA system combination results for direct compasison.

Gaussian classes.

First of all comparing the combinations *ACG* (STFT spectral representations) and *BDH* (STRAIGHT representations), we observe that while they have similar accuracies overall, STRAIGHT representations seem to favour female speakers while male speakers are recognised better by the conventional STFT based features. When they are merged together in *ABCDGH* the greatest improvement is still maintained for females.

ROVERing the HLDA system outputs with those of the original ones used for the combination gives a substantial improvement with respect to the HLDA feature combinations: *ABEF* gives a 1.5% improvement compared to *E* alone, while *ABCDEF* is 0.8% better than *ABCD*; similarly for PLPs, *GHI* improves the HLDA combination system *I* by 0.8% also. This is of interest because it indicates that ROVER acts in a complementary way to HLDA, being able to further improve the already combined systems.

Complementarity between MFCC- and PLP-based systems is more difficult to exploit than that between conventional and STRAIGHT-based systems. When we consider the combination of all the MFCC based systems *ABCD* with the PLP-based systems *GH*, we observe that *ABCDGH* has a similar error rate to *ABCD* for the majority voting experiment, although there was a substantial improvement in the oracle case. On the other hand, the contribution of the higher order representations (*CD*) is evident (around 1% absolute), and occurs consistently when comparing *ABCDEF* with *ABEF*, *ABCDGH* with *ABGH*, and *ABCDEFGHI* with *ABEFGHI*.

Finally the best result is obtained by combining all the available systems *ABCDEFGHI*, consistent with Schlüter et al. (2007). This yields a substantial decrease in word error rate of 2.4% absolute (6.6% relative) compared with the best HLDA system *I* (HLDA combination of PLPs), and 2.9% absolute (7.9% relative) compared with the best single stream system *D* (higher order STRAIGHT derived MFCCs). Overall, by combining HLDA and ROVER we were able to reduce the word error rate by 4.6% absolute (12% relative) compared with the baseline normalised lower order MFCC system. The oracle results indicate an upper limit of the exploitation of the complementarity between representations.

### 6.4.7 Experiments discussion

STRAIGHT derived features have proven to benefit the most from VTLN (as we expected). Unfortunately in most of the experiments they were not able to outperform conventional features. Even so there are several exceptions: in particular for high pitched female speakers, pitch harmonics are still evident after Mel scaling in conventional MFCCs, thus pitch adaptive features are able to outperform conventional features.

The combination of MFCCs and STRAIGHT features through HLDA was successful in all the tasks. MFCCs are affected by pitch artefacts but they are extracted from a sharper representation, while STRAIGHT features are affected by pitch tracking errors, but are smoother and devoid of pitch artefacts. The two spectral representations are thus complementary and their combination provides consistent improvements. The effect of the smoothing and the pitch adaptive modules will be separately studied in chapter 7. Pitch tracking errors are more frequent and have the most influence in telephone speech because of the band-pass filtering channel effect, in the meeting domain because of the presence of cross-talk, and in case of beamformed signals because of the decreased directionality at lower frequencies. The telephone line and beamforming effects particularly affect male speakers (having a lower pitch); this also accentuates the predilection of STRAIGHT towards high pitched female voices, but this is clearly evident in the WSJCAM0 and in the IHM meeting domain as well. Nevertheless the combination using HLDA is able to yield consistent improvements even in more challenging domains (CTS and MDM meetings), where the actual relative improvement is even larger.

In order to analyse our experiments and to better exploit the complementarity of the pitch adaptive spectral representation, ROVER system combination was also performed. This is a reasonable experiment when a large number of independent speech recognition outputs have been made available. These experiments confirmed the predilection of STRAIGHT systems for female speakers, the importance of the information contained in higher order coefficients (which can be exploited thanks to the pitch adaptation of STRAIGHT); and the complementarity of HLDA and ROVER techniques.

## 6.5 Conclusions

We have investigated a pitch adaptive acoustic parameterisation for speech recognition, derived from the STRAIGHT approach to time-frequency analysis, with a particular focus on speaker normalisation (VTLN) and combination with conventional features using HLDA. We performed experiments on three large vocabulary domains, using standard data sets and evaluation protocols: WSJCAM0, conversational telephone speech and multiparty meeting transcription, considering both close-talking and microphone array conditions in the latter domain.

In each domain we observed significant reductions in word error rate through the combination of conventional and STRAIGHT-based features using HLDA. The resulting systems based on these combined representations were able to achieve relative reductions in word error rate of 3.2% on WSJCAM0, 8% on conversational telephone speech, and for the meeting domain 4.7% for the IHM condition and 7.3% for the MDM condition. In both the WSJCAM0 and CTS domains, we found that STRAIGHT derived features benefit the most from VTLN (because of their smoother representation) particularly for male speakers. VTLN on male speakers lowers the centers of the Mel filters making the filters width narrower too, thus more able to capture the thin horizonal spectral lines due to pitch artefacts. For female speakers the pitch artefacts are even more evident in the conventional STFT representation, while the pitch adaptive spectral representation makes the formant positions more evident and therefore easier to catch by the VTLN warped filterbank. Therefore VTLN on male speech benefits more from a smoother spectrogram showing even better improvements than VTLN on female. Note that in general the speakers benefiting the most from the use of a pitch adaptive representation are high pitched speakers for which the pitch artefacts are more evident.

Moreover experiments on the CTS domain showed that the influence of the pitch tracker is of importance for STRAIGHT derived feature extraction.

Experiments on the use of pitch adaptive MF-PLPs for the meeting IHM task showed a 3.2% relative WER improvement when combined with conventional MF-PLPs using HLDA. On the same task the use of higher order coefficients (20 MFCCs plus C0) was evaluated both for standard and pitch-adaptive features, finding that STRAIGHT-based features performed better than conventional features, particularly

for female speakers. In fact, for STFT derived features, higher order coefficients are strongly affected by pitch artefacts which is more evident in high-pitched speakers. Finally ROVER system level combination was applied on top of HLDA feature level combination finding that further improvements can be achieved merging the output of the baseline systems with the correspondent HLDA combined system; therefore showing that ROVER is complementary to HLDA.

We have explored the use of pitch-adaptive spectral representations in ASR, as a complement to conventional STFT representations. Extensive experiments over three standard large vocabulary tasks allow us to conclude that the use of such complementary information, combined using HLDA, provides consistent, significant reductions in word error rate.

# Chapter 7

# Experimental analysis of the use of STRAIGHT in LVCSR

## 7.1 Introduction

In chapter 6 we used a pitch adaptive spectral representation, STRAIGHT (Kawahara et al., 1999), to perform experiments on three Large Vocabulary Continuous Speech Recognition (LVCSR) tasks: WSJCAM0, conversational telephone speech and multiparty meeting data.

STRAIGHT derived features provided substantial improvements in all the tasks when combined with conventional MFCCs, suggesting that they are complementary to the latter.

In this chapter we analyse the individual contribution of each representation in two ways. First in section 7.2, we decouple the pitch adaptive and smoothing aspects of STRAIGHT. Experiments performed on the meeting speech recognition task highlight the importance of using a pitch adaptive spectral analysis and the benefit of combining it with a conventional fixed window spectral analysis. Second in sections 7.3 and 7.4, a speaker independence metric was used to compare pitch adaptive features with conventional features: it was found that the pitch adaptive component of STRAIGHT provides improved speaker independence. Reduced inter-speaker variability is particularly beneficial when feature combination techniques such as Heteroscedastic Linear Discriminant Analysis (HLDA) are employed.

## 7.2 Decoupling the pitch adaptive and the smoothing effect of STRAIGHT

The STRAIGHT spectral analysis has two concurrent effects: on one side a pitch adaptive window is used for spectral analysis and on the other side smoothing is performed interpolating the partial information provided by the pitch adaptive spectral analysis itself. In the experiments described in the previous chapter we observed, both on the WSJCAM0, the CTS and the IHM and MDM meeting conditions, that for 39 dimensional systems conventional MFCCs outperformed the STRAIGHT derived MFCC systems. Therefore we conducted some experiments to decouple the two STRAIGHT effects both on the IHM and the MDM meeting task. To compute STRAIGHT derived features without the pitch adaptive analysis we just kept the window's width fixed (to 80 Hz), and to compute STRAIGHT derived features without smoothing this step is just skipped in the processing.

Figure 7.1 shows a plot of the spectral contour for one frame of voiced speech for the short time Fourier transform (STFT), and for STRAIGHT, while figure 7.2 compares the STRAIGHT spectral envelope with that of STRAIGHT using only the smoothing and STRAIGHT using only the pitch adaptive component. It can be noticed that when the pitch adaptive module of STRAIGHT is used with no smoothing some harmonics are still present, while using the smoothing part alone on the other hand seems to yield a very smooth spectral envelope.

We performed the experiments in the meeting domain in the IHM task training and testing using the same data used for the experiments described in the previous chapter (chapter 6) and described in section 4.2.3.6 and 4.5.

The results of these experiments are reported in table 7.1. First we observe that the pitch adaptive analysis without smoothing (S2) gives a small but not significant improvement over conventional MFCCs (M1) and an even larger improvement on S1 (STRAIGHT derived MFCCs). This is particularly evident for female speakers while for male speakers there is a substantial improvement especially when compared to purely STRAIGHT derived MFCCs (S1). Smoothing is particularly bad for male speakers and this is also confirmed by the experiment on the use of the smoothing part only of STRAIGHT without pitch adaptive analysis (S3). The MFCCs extracted using the smoothing component only of STRAIGHT performed consistently

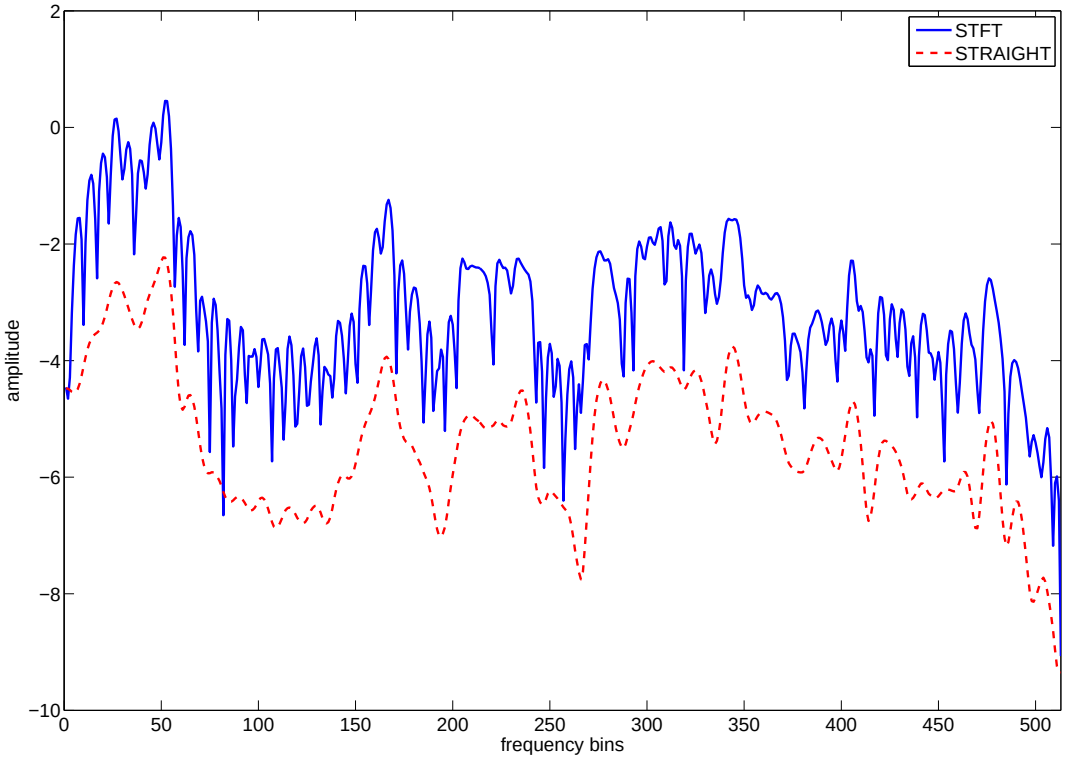



Figure 7.1: A comparison of the STFT and STRAIGHT spectral analysis

worse than conventional MFCCs.

We also combined conventional MFCCs with the pitch adaptive only (M1+S2) and smoothing only (M1+S3) STRAIGHT derived MFCCs using HLDA feature combination with monophone mixture components as classes reducing from 78 to 39 dimensions. While none of this combinations outperformed the combination of conventional and STRAIGHT derived MFCCs (M1+S1) overall, the combination with pitch adaptive only STRAIGHT derived MFCCs (M1+S2) gave better performances for female speakers (for which pitch adaptive analysis is more important). The combination with smoothing only STRAIGHT derived MFCCs (M1+S3) on the other hand gave a smaller improvement. This is further evidence that the complementarity between conventional and STRAIGHT derived MFCCs is arisen from the use of a pitch adaptive window by the latter.

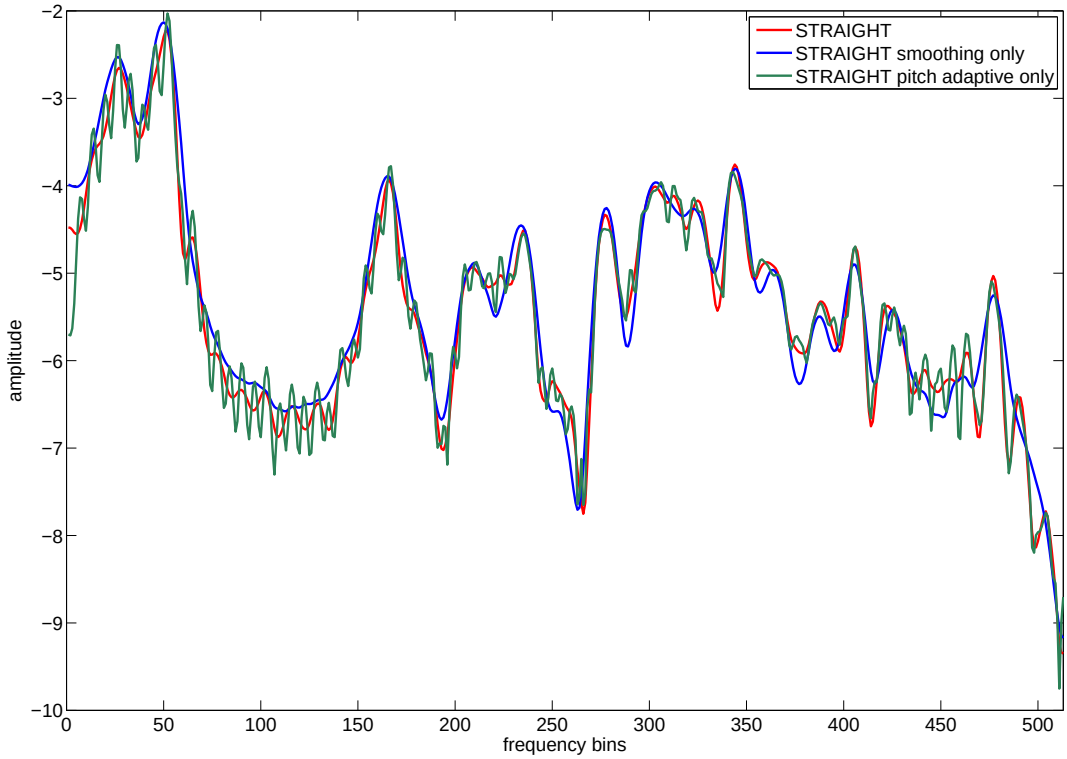



Figure 7.2: A comparison of the STRAIGHT spectral analysis with pitch adaptive only and smoothing only

## 7.3 Statistical measures of the acoustic features speaker independence

One of the aims of using a pitch adaptive spectral representation for feature extraction is to obtain features which have increased speaker independence. Ideally we would like to have features which only vary across different classes and which have as little as possible variation across different speakers within the same class used for speech recognition.

Haeb-Umbach (1999) investigated the effectiveness of speaker normalisation techniques such as CMN and CVN and VTLN proposing to use the LDA objective function as an effectiveness measure. A similar treatment of the effect of speaker normalisation techniques was also presented by Saon et al. (2002) where they relate VTLN and LDA techniques showing the importance of applying LDA on top

| | | TOT | F | M | CMU | ICSI | LDC | NIST |
|---|---|---|---|---|---|---|---|---|
| MFCC | M1 | 38.4 | 38.5 | 38.3 | 43.7 | 23.9 | 52.1 | 30.9 |
| STRAIGHT MFCC | S1 | 39.3 | 38.3 | 39.7 | 44.7 | 24.8 | 53.1 | 31.2 |
| STRAIGHT MFCC pitch adapt. only | S2 | 38.2 | 38.2 | 38.3 | 43.4 | 24.2 | 51.8 | 30.7 |
| STRAIGHT MFCC smoothing only | S3 | 40.1 | 39.9 | 40.1 | 45.1 | 25.5 | 55.2 | 31.3 |
| HLDA 78 to 39 mono | M1 + S1 | 36.6 | 36.3 | 36.7 | 41.0 | 22.5 | 51.2 | 28.5 |
| HLDA 78 to 39 mono | M1 + S2 | 36.9 | 36.1 | 37.3 | 41.1 | 22.0 | 51.8 | 30.0 |
| HLDA 78 to 39 mono | M1 + S3 | 37.3 | 36.6 | 37.6 | 42.1 | 23.3 | 50.7 | 30.2 |

Table 7.1: Experiment on RT04seval testing set using VTLN features for the IHM condition. From top to bottom: conventional MFCCs 39 dimensions (M1); STRAIGHT MFCCs 39 dimensions (S1); STRAIGHT MFCCs 39 dimensions with pitch adaptive analysis only (no smoothing) (S2); STRAIGHT MFCCs 39 dimensions with smoothing only (no pitch adaptive analysis) (S3); HLDA combination of M1 and S1, M1 and S2, M1 and S3 all reducing from 78 to 39 dimensions using monophone mixtures as classes.

of speaker normalised features (which ideally eliminate completely inter-speaker variability) in order to achieve better class separability using LDA. The aim of this section is to summarise the results of these two papers to introduce the analysis we conducted on STRAIGHT derived features.

Suppose each acoustic feature vector $x_i$ is labelled according to the class $j$ and the speaker $s$ to which it belongs (the association of a particular frame to a class $j$ can be done automatically by forced alignment). We can define the corresponding total number of feature vectors $\mathbf{x_i} \in (j,s)$ as $N^{(j,s)}$; therefore the corresponding mean and variance will be respectively defined as:

$$\hat{\mu}^{(j,s)} = \frac{1}{N^{(j,s)}} \sum_{i=1}^{N^{(j,s)}} \mathbf{x_i}^{(j,s)}, \tag{7.1}$$

$$\hat{\Sigma}^{(j,s)} = \frac{1}{N^{(j,s)}} \sum_{i=1}^{N^{(j,s)}} (\mathbf{x_i}^{(j,s)} - \hat{\mu}^{(j,s)})(\mathbf{x_i}^{(j,s)} - \hat{\mu}^{(j,s)})^T. \tag{7.2}$$

And if $N^{(j)}$ is the total number of feature vectors belonging to class $j$, the corresponding class specific mean and variance are defined as:

$$\hat{\mu}^{(j)} = \frac{1}{N^{(j)}} \sum_{s \in S} N^{(j,s)} \hat{\mu}_{(j,s)}, \tag{7.3}$$

$$\hat{\Sigma}^{(j)} = \frac{1}{N^{(j)}} \sum_{s \in S} N^{(j,s)} \hat{\Sigma}^{(j,s)} + \hat{\Sigma}_{B^S_{(j)}}. \tag{7.4}$$

where $\hat{\Sigma}_{B^S_{(j)}}$ is the between speaker covariance for class $j$ computed as:

$$\hat{\Sigma}_{B^S_{(j)}} = \frac{1}{N^{(j)}} \sum_{s \in S} N^{(j,s)} (\hat{\mu}^{(j,s)} - \hat{\mu}^{(j)})(\hat{\mu}^{(j,s)} - \hat{\mu}^{(j)})^T. \tag{7.5}$$

The within-class covariance is therefore due to two distinct components: the variance due to the classes themselves and the between speaker covariance:

$$\hat{\Sigma}_{wc} = \frac{1}{N} \sum_{j \in J} N^{(j)} \hat{\Sigma}^{(j)} = \underbrace{\frac{1}{N} \sum_{j \in J} \sum_{s \in S} N^{(j,s)} \hat{\Sigma}^{(j,s)}}_{\hat{\Sigma}^N_{wc}} + \underbrace{\frac{1}{N} \sum_{j \in J} N^{(j)} \hat{\Sigma}_{B^S_{(j)}}}_{\hat{B}^S} \tag{7.6}$$

where $\hat{B}^S$ is the total between speaker covariance and $\hat{\Sigma}^N_{wc}$ is the within class covariance matrix we would have if the features were ideally speaker independent. The total covariance $\hat{\Sigma}$ is given by the sum of the within class covariance $\hat{\Sigma}_{wc}$ and the between class covariance $\hat{\Sigma}_{bc}$ which is given by:

$$\hat{\Sigma}_{bc} = \sum_{j \in J} \frac{N^{(j)}}{N} (\hat{\mu}^{(j)} - \hat{\mu})(\hat{\mu}^{(j)} - \hat{\mu})^T. \tag{7.7}$$

The goal of LDA is finding the projection $\theta$ which maximises the across class covariance and minimises the within class covariance in the projected space that is:

$$\theta = \arg\max_{\theta} \frac{|\theta \hat{\Sigma}_{bc} \theta^T|}{|\theta \hat{\Sigma}_{wc} \theta^T|} = \arg\max_{\theta} \frac{|\theta \hat{\Sigma}_{bc} \theta^T|}{|\theta (\hat{\Sigma}^N_{wc} + \hat{B}^S) \theta^T|}. \tag{7.8}$$

As already observed in section 4.4.1 the solution of equation 7.8 can be found by computing the eigenvectors corresponding to the $p$ largest eigenvalues of $\hat{\Sigma}^{-1}_{wc} \hat{\Sigma}_{bc}$, with the product of the $p$ largest eigenvalues corresponding to the LDA objective function. Saon et al. (2002) argued that, since ideally the between-speaker covariance should be zero for speaker normalised features, the LDA objective function for

normalised features should always be higher than that of non normalised features. Unfortunately even using speaker normalisation techniques, the between-speaker covariance is not completely zero but it makes sense to use the LDA objective function to measure inter-speaker independence of the features.

Similarly we can also demonstrate that the HLDA objective function measured on normalised features is larger than that measured on non speaker normalised features. In fact we recall that from equation 4.8 HLDA transforms are estimated by maximising the likelihood of the original data given the estimated statistics with an objective function inversely proportional both to the total covariance $\hat{\Sigma}$ and the per class covariance matrices $\hat{\Sigma}^{(j)}$. We have shown that the total covariance matrix $\hat{\Sigma} = \hat{\Sigma}_{bc} + \hat{\Sigma}_{wc}$ can be further decomposed into two parts: the covariance that would be obtained if the features were perfectly speaker normalised, and the between speaker covariance (equation 7.6). The per class covariance matrix $\hat{\Sigma}^{(j)}$ (equation 7.4) can be also split into a class-specific covariance and the between speaker covariance matrix for the class. Ideally, if the features were completely speaker normalised, the between speaker covariance would be null and therefore the likelihood of equation 4.8 would be greater for normalised features compared to features with some speaker dependence. Unfortunately even using speaker normalisation techniques, the between-speaker covariance is not completely zero (for example coarticulation differences are not normalised by VTLN) and the LDA objective function can be used as a measure of speaker independence of the features.

In Haeb-Umbach (1999) the trace (the sum of the eigenvalues) of the ratio between the between class covariance matrix $\hat{\Sigma}_{bc}$ and the between speakers covariance matrix is used as a measure of speaker normalisation effectiveness ($\hat{\Sigma}_{bc}/\hat{B}^S$). In our work we applied this measure to compare conventional MFCCs and our STRAIGHT derived MFCCs from an inter-speaker variability point of view.

## 7.4 Measuring the speaker independence of STRAIGHT derived features

We adopted the inter-speaker independence measure introduced in Haeb-Umbach (1999) using Gaussian components of monophone models as classes in order to

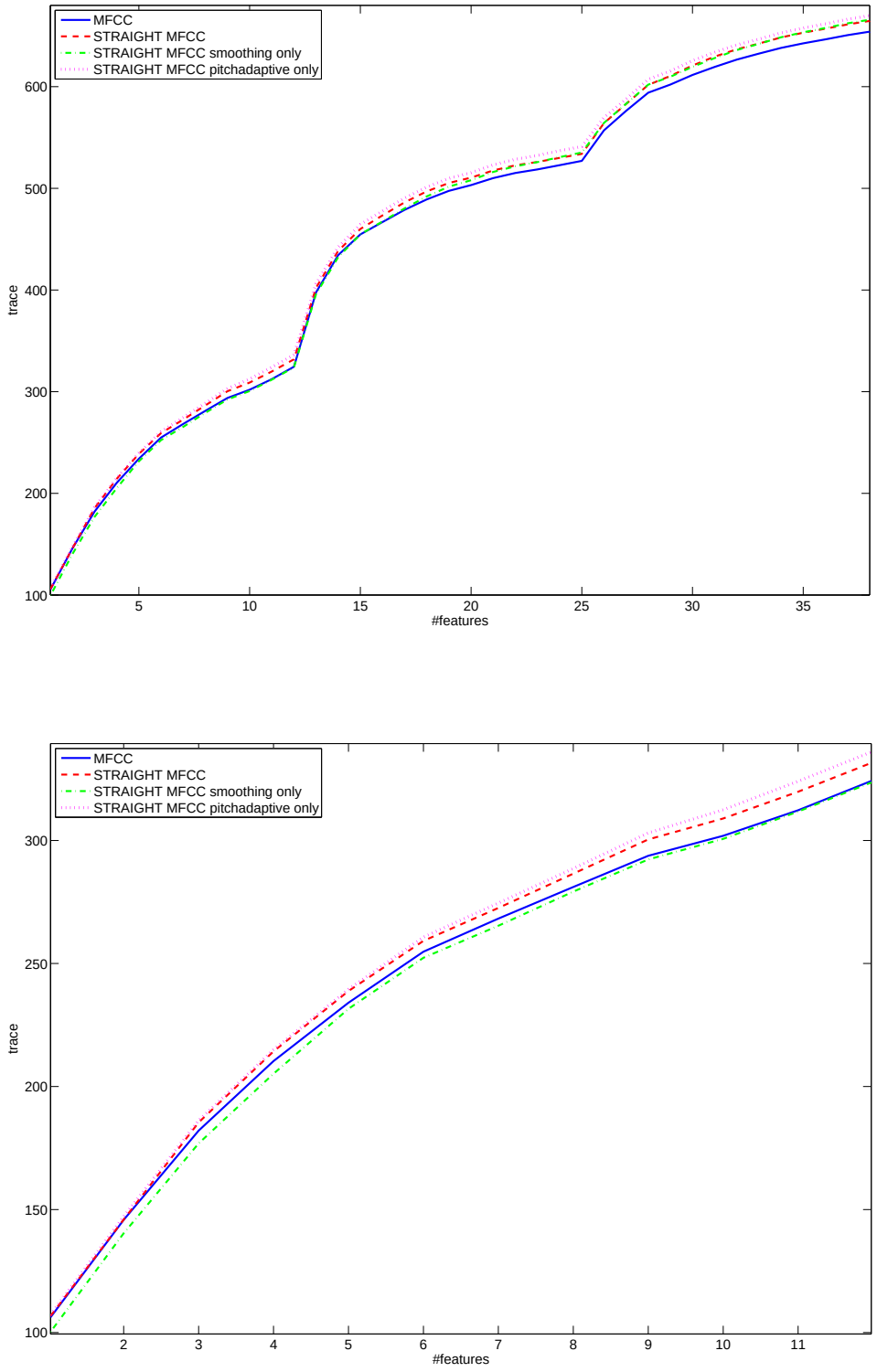



Figure 7.3: Trace measure as a function of the feature dimension measured using the whole meeting IHM training data

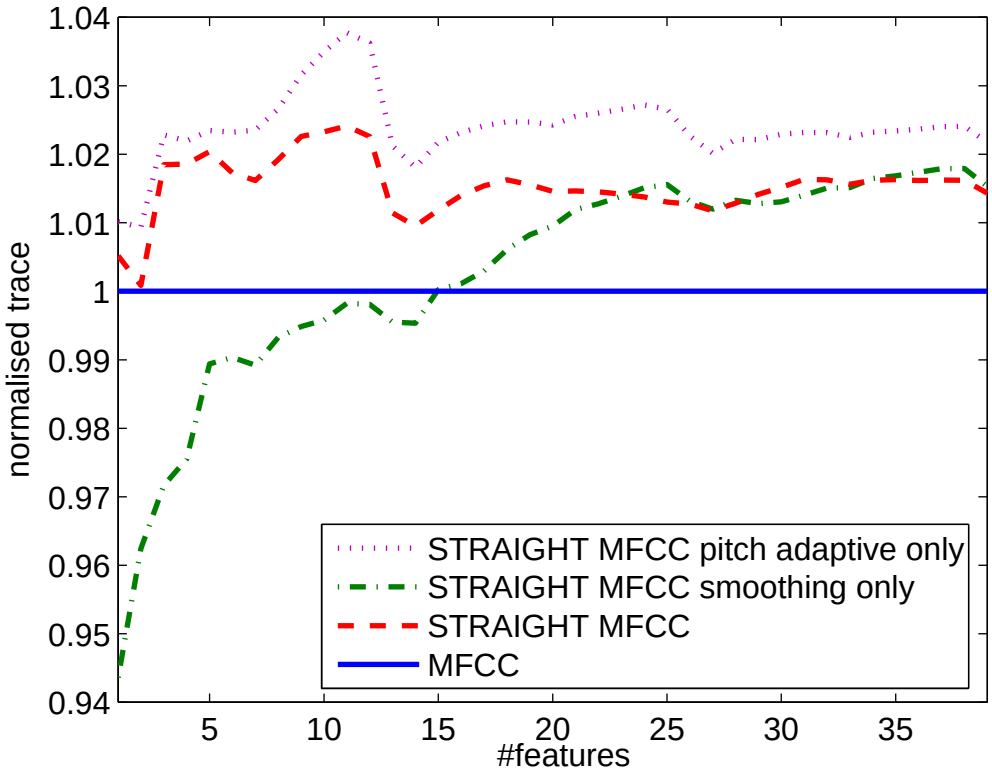



Figure 7.4: Trace measure as a function of the feature dimension measured using the whole meeting IHM training data normalised with the trace measure of the MFCC features

maintain the same type of classes used in our HLDA combination experiments. We compared conventional MFCCs, STRAIGHT derived MFCCs without the smoothing, STRAIGHT derived MFCCs without the use of the pitch adaptive window and STRAIGHT derived MFCCs with both smoothing and the pitch adaptive window usage. We used the entire meeting IHM corpus (described in section 4.2.3.6) which contains a total of 115 male and 49 female speakers. The results of this experiment, using 39 dimensional feature vectors (12 cepstral coefficients plus C0 plus $\Delta$s and $\Delta\Delta$s), are shown in figure 7.3 for the whole training set and in figure 7.5 and 7.7 for male and female speakers respectively. In figure 7.4, 7.6 and 7.8 the traces normalised using the trace of the MFCC features are also shown for better comparison. The trend of the trace measure shows 3 large humps due to the different nature on cepstral coefficients and their first and second derivatives and to the fact that obviously as lower order cepstral coefficients are more discriminative so are their correspondent first and second derivatives (the gradient is higher for lower order coefficients and their derivatives), while higher order cepstral coefficients are more noisy and therefore less discriminative; thus they have a correspondent eigenvalue which is smaller than that of lower order coefficients.

Looking at the magnified lower part of figure 7.3 (which shows the trace trend for the first 12 cepstral coefficients only) we can observe that STRAIGHT derived MFCCs using the pitch adaptive windowing but without smoothing show the higher inter-speaker independence. Pitch adaptive features are more speaker independent than both conventional MFCCs and smoothing only STRAIGHT derived MFCCs. STRAIGHT derived features using the pitch adaptive component only are the most speaker invariant.

Comparing the lower magnified part of figure 7.5 for male speakers and figure 7.7 for female speakers we can observe that they are slightly different: while for female speakers for lower order cepstral coefficients the trace is lower for conventional MFCCs and this curve is then crossed by the STRAIGHT derived features without the use of the pitch adaptive window, in the male picture this happens only for the 10th cepstral coefficient. Most importantly both for females and males the features derived from a pitch adaptive representation show a higher trace and therefore evidence of higher speaker independence for the cepstral coefficients themselves.

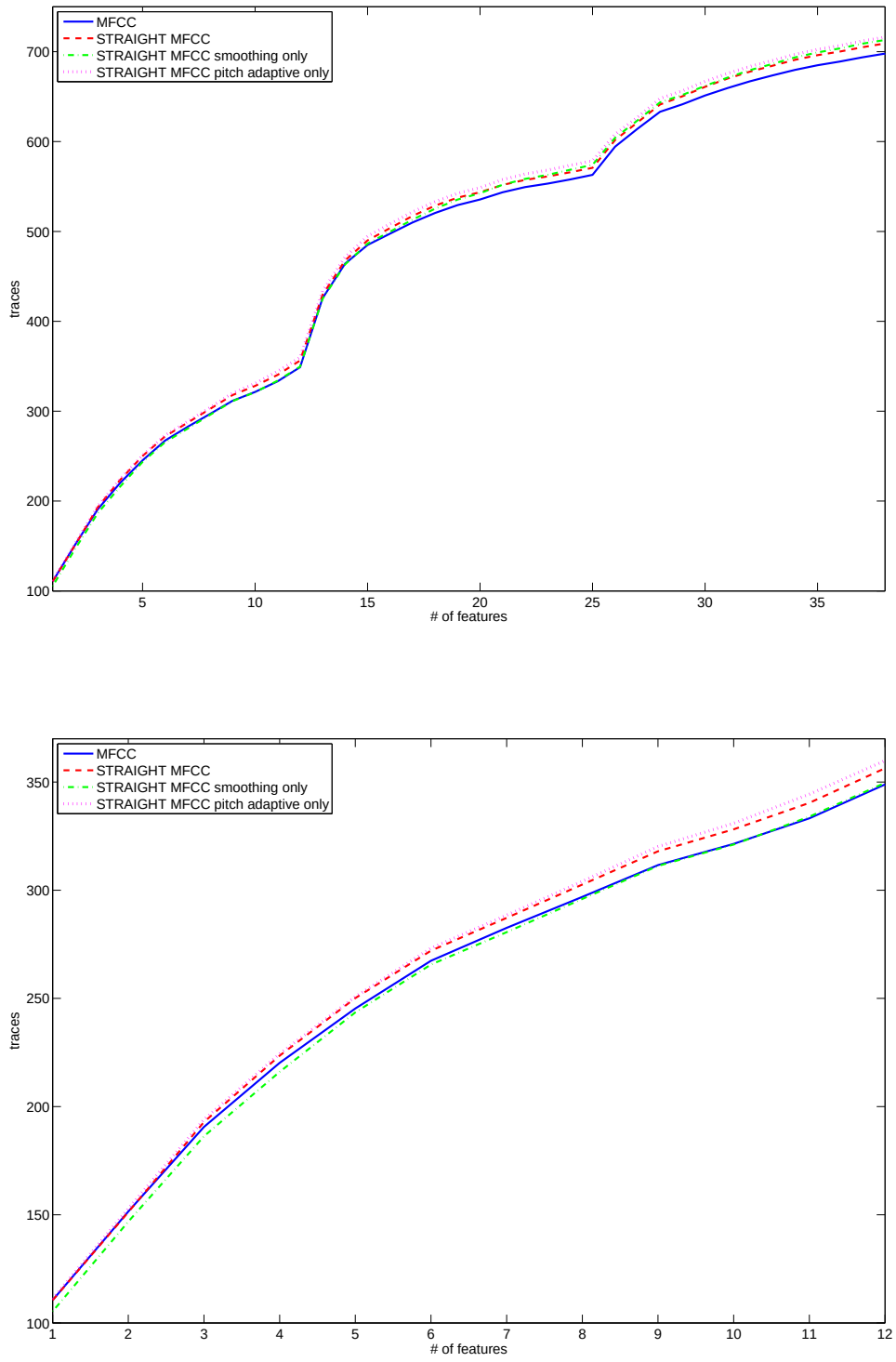



Figure 7.5: Trace measure as a function of the feature dimension measured using the male part of the meeting IHM training data

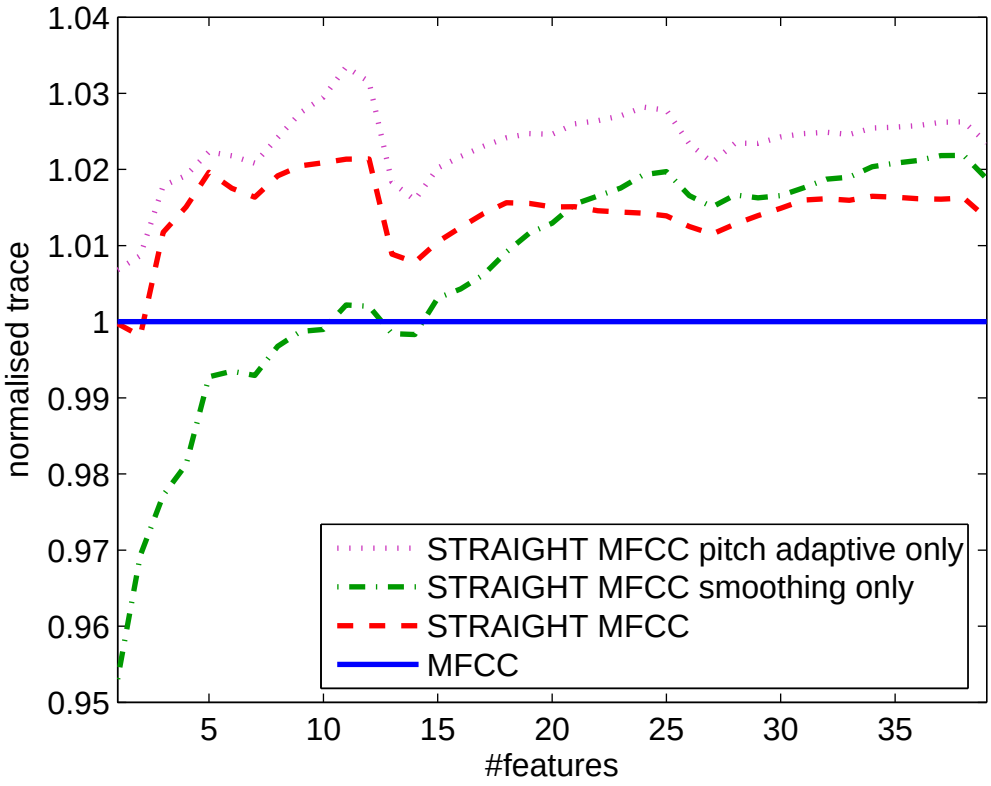



Figure 7.6: Trace measure as a function of the feature dimension measured using the male part of the meeting IHM training data normalised with the trace measure of the MFCC features

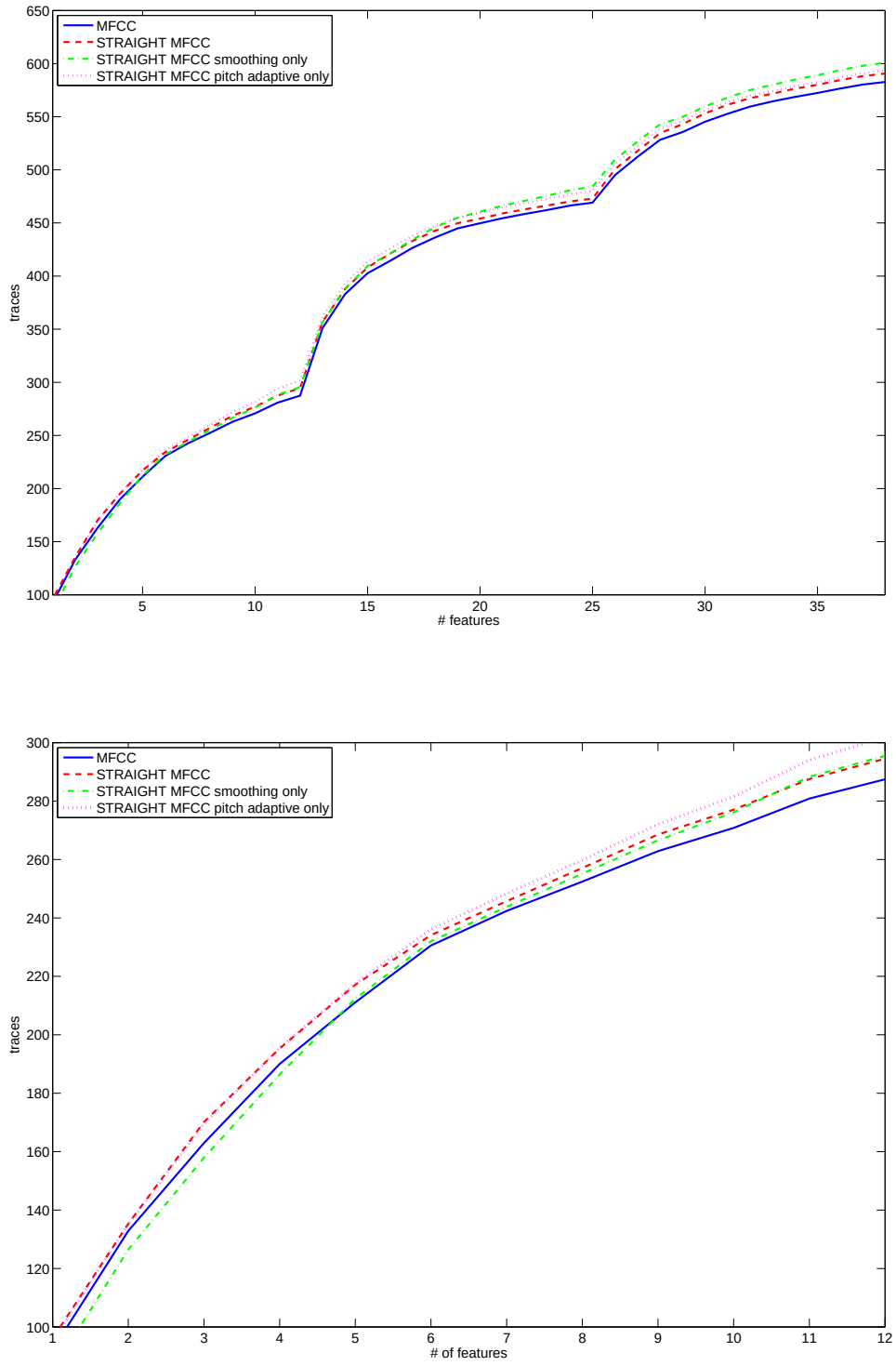



Figure 7.7: Trace measure as a function of the feature dimension measured using the female part of the meeting IHM training data

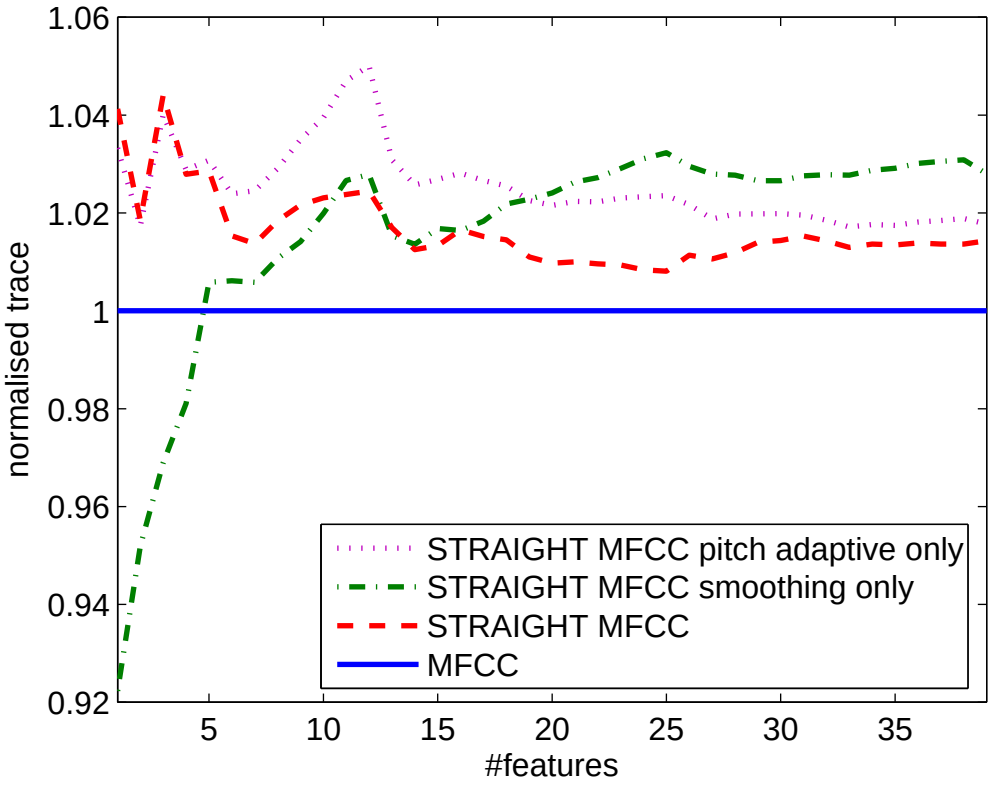



Figure 7.8: Trace measure as a function of the feature dimension measured using the female part of the meeting IHM training data normalised with the trace measure of the MFCC features

| | | TOT | F | M | CMU | ICSI | LDC | NIST |
|---|---|---|---|---|---|---|---|---|
| MFCC | M1 | 38.4 | 38.5 | 38.3 | 43.7 | 23.9 | 52.1 | 30.9 |
| STRAIGHT MFCC | S1 | 39.3 | 38.3 | 39.7 | 44.7 | 24.8 | 53.1 | 31.2 |
| STRAIGHT MFCC pitch adapt. only | S2 | 38.2 | 38.2 | 38.3 | 43.4 | 24.2 | 51.8 | 30.7 |
| STRAIGHT MFCC smoothing only | S3 | 40.1 | 39.9 | 40.1 | 45.1 | 25.5 | 55.2 | 31.3 |
| HLDA 39 to 39 mono | M1 | 37.6 | 37.7 | 37.5 | 41.7 | 23.1 | 53.1 | 29.5 |
| HLDA 39 to 39 mono | S1 | 37.4 | 37.2 | 37.5 | 42.3 | 21.8 | 53.7 | 28.6 |
| HLDA 39 to 39 mono | S2 | 37.1 | 36.0 | 37.7 | 41.8 | 22.3 | 52.2 | 29.3 |
| HLDA 39 to 39 mono | S3 | 39.6 | 38.8 | 40.1 | 44.2 | 25.0 | 55.4 | 30.7 |

Table 7.2: Experiment on RT04seval testing set using VTLN features for the IHM condition. From top to bottom: conventional MFCCs 39 dimensions (M1); STRAIGHT MFCCs 39 dimensions (S1); pitch adaptive only STRAIGHT MFCCs 39 dimensions (S2); smoothing only STRAIGHT MFCCs 39 dimensions (S3); HLDA 39 to 39 dimension projection of M1 (conventional MFCCs); HLDA 39 to 39 dimension projection of S1 (STRAIGHT derived MFCCs); HLDA 39 to 39 dimension projection of S2;HLDA 39 to 39 dimension projection of S3.

Saon et al. (2002) argued that LDA gives better performances on more speaker independent features and we have shown in the previous section (7.3) that this should be also true for HLDA. To assess this on our STRAIGHT derived features we applied HLDA directly on the 39 dimensional conventional MFCCs, STRAIGHT derived MFCCs and STRAIGHT derived MFCCs using the pitch adaptive module only and the smoothing module only projecting to 39 dimensions (performing in this way MLLT (Gopinath, 1998)).

The results of this experiment are shown in table 7.2. The improvement obtained by the use of HLDA is larger for pitch adaptive STRAIGHT derived MFCCs than for

conventional MFCCs and smoothing only STRAIGHT MFCCs. We hypothesise that this is due to the better speaker independence of pitch adaptive features as shown similarly by Saon et al. (2002) for LDA applied on VTLN features.

## 7.5 Conclusions

In the previous chapter a STRAIGHT based pitch adaptive spectral representation was successfully applied to extract acoustic features for a challenging LVCSR task, multiparty conversational speech in the meeting domain. The combination with conventional MFCCs using HLDA was particularly beneficial yielding consistent improvements over conventional features alone.

In this chapter the two key components of STRAIGHT, pitch adaptive analysis and smoothing through interpolation, were studied independently. Experimental results showed that adopting pitch adaptive features can improve speech recognition performances. Non smoothed pitch adaptive features outperformed smoothed non pitch adaptive features, when combined with conventional MFCCs. This improvement is principally due to the adoption of a pitch adaptive representation. The use of a pitch adaptive representation is particularly beneficial for female speakers, because for high pitched speakers the Mel filters are not broad enough to remove the horizontal spectral lines due to the pitch artifacts.

We have also measured the speaker independence of all the features adopted in this study. Using an LDA based metric we found evidence that the pitch adaptive features are more speaker independent than conventional MFCCs. We observed that the improved speaker independence has the desirable effect of making HLDA more effective and making STRAIGHT derived features more suitable for this technique than conventional features.

# Chapter 8

# Conclusions

The main goal of this thesis was studying the application of speaker normalisation techniques such as VTLN to multiparty conversational speech and in particular multiparty meetings. Therefore the principal research question I aimed to answer in this dissertation is:

> *How can we apply speaker normalisation and in particular VTLN to multiparty conversational speech?*

More specifically this problem has been subdivided in two sub-questions:

1. *Which are the most important features of multiparty conversational speech from a VTLN point of view?*
2. *Is it possible to improve the conventional feature extraction methods to obtain features which are better suited for speaker normalisation and thus more speaker independent?*

Two main research threads were therefore investigated to answer respectively the two proposed research questions above recalled. First the application of maximum likelihood VTLN to spontaneous conversational speech, with particular attention to multiparty meetings, was investigated finding consistent WER reductions (8% relative) both on CTS and meeting data. The stability of the warping factors, parameterising vocal tract length normalisation, was studied both for the same speaker across different meetings and across time for the same speaker within a single meeting, finding no stable estimates, even if vocal tract length should be constant at least to a certain extent. Thus we investigated the variability of the warping factors in connection with the rich speaker turn structure characterising meetings.

This study was conducted looking at the dependence of the warping factor estimated for a speaker given the current speaker's addressee. We found that ML estimated warping factors appear to be influenced by the context and particularly by the current conversational partner. It is thus likely that speakers address others according to whom they are speaking to and that this is reflected in the ML estimate of the warping factor. We also hypothesised that the behaviour of the warping factor estimates is in line with the interactive alignment account of dialogue: the estimated warping factors of two speakers are typically non aligned at the beginning of a meeting but can be seen to align as the meeting progresses. According to the interactive alignment account, during a dialogue two speakers could be seen to align at multiple levels: lexical, syntactic, phonological, phonetic and in terms of the formant space. In particular warping factors are known to be highly correlated with pitch (as we also found in the experiments of this thesis) and the variation of warping factors can be at least partly explained by a shift in formant frequencies caused by interactive alignment.

Therefore we investigated the use of pitch adaptive features in the context of multiparty spontaneous speech in conjunction with VTLN (this is the main second thread of this thesis). In particular we adopted the pitch adaptive spectral representation of STRAIGHT for the extraction of acoustic features such as MFCCs and PLPs. This spectral representation is computed in two steps: first a pitch adaptive spectral analysis is performed adopting a window which adapts to the F0 value, second a smoothing through interpolation of the partial information given by the pitch adaptive spectral analysis is performed. For the unvoiced segments the value of the pitch determining the analysis window width was fixed to a constant (160Hz). One possible alternative could have been to use an interpolation of the pitch contour for this regions. A recent modification to STRAIGHT (Kawahara, 2007) provides non-zero pitch trajectories even for unvoiced and silence segments and it would be interesting to investigate the effect of this on pitch adaptive features for speech recognition.

We also combined conventional and pitch adaptive features using both feature level combination in the form of HLDA and system level combination in the form of ROVER. As well as in the meeting domain, both for the close talking and the distant microphone tasks, we also performed our experiments on the WSJCAM0 corpus (read speech) and conversational telephone speech.

Results on the use of STRAIGHT derived features have shown that the pitch independent features achieve performances comparable to those of conventional features. In addition they benefit particularly from VTLN and yield especially good results for female speakers. Combining STRAIGHT derived and conventional features, both using feature and system combination techniques, we found that the information carried is complementary. HLDA feature combination was able to achieve a consistent relative decrease in the word error rate of 3–9% across all three domains, with the largest relative reductions being observed on the telephone speech and distant microphone tasks. A further 8% relative reduction in word error rate was observed when ROVER combination (using majority voting) was applied to the meeting transcription task. The success in applying the STRAIGHT spectral representation to three different challenging tasks, allowed us to make strong conclusions about the usefullness of a pitch adaptive representation in the LVCSR domain, particularly if used in conjunction with VTLN. Besides providing consistent word error rate reductions when combined to conventional features, the pitch adaptive features have proved to be able to benefit the most from VTLN. This demonstrates their better suitability in conjunction with this technique both on the WSJCAM0 and the CTS task.

The complementarity between conventional and STRAIGHT derived features could be explained as follows: STRAIGHT derived features, given their more accurate representation (independent from the pitch artefacts), provide information complementary to the conventional STFT derived features. Conventional STFT features on their turn, because of the sharper spectral envelope of STFT, contain important information which was smoothed out in the STRAIGHT representation. We have combined the conventional and STRAIGHT features at the cepstral coefficient level. It would be however possible to combine directly the STFT and the STRAIGHT spectrograms after the Mel scaling is performed. This could be done in several ways. One possibility is to apply directly HLDA to the concatenated Mel spectrograms, skipping the DCT step, since this is believed to be not necessary when discriminant linear transforms are used (Yu and Waibel, 2000; Saon et al., 2000a). In this way the two frontends could be integrated at a lower level reducing the computational overload and making the feature extraction process more consistent. Moreover recent works have shown the benefit of system combination through

cross-adaptation (Giuliani and Brugnara, 2006; Hoffmeister et al., 2007). It would therefore be possible to use the output of the STRAIGHT derived MFCC system to adapt through MLLR the conventional MFCC system and viceversa.

Further investigations on meetings, isolating the pitch adaptive from the smoothing component of the STRAIGHT spectral representation, have also shown that when the two main modules are used separately it is the pitch adaptive part to provide most of the complementarity with the conventional features. Moreover evidence of an improved speaker independence due to the pitch adaptive analysis was also observed.

It is well known that the pitch artefacts, which manifest themself through spectral harmonic lines in the spectrogram, particularly affects higher order coefficients (Irino et al., 2002). The pitch artefacts can be in fact still present after the Mel Filterbank is applied (especially for high pitched speakers). This is the main reason why in most of the speech recognition systems only the first 12 cepstral coefficients are used. In our experiments we found that the adoption of a smooth pitch adaptive spectral representation enables to use higher order cepstral coefficients even for high pitched (female) speakers, yielding a significant improvement compared to the conventional features. In fact in this case STRAIGHT derived MFCCs outperform conventional MFCCs.

The peculiarities of conversational speech from a speaker normalisation point of view (our main starting point question) were studied through investigations of the VTLN warping factor behaviour finding a dependence on the rich meeting speaker turn structure. It is therefore interesting to take into account the discourse structure by estimating VTLN parameters depending on the speaker turn. However the relationship between warping factors and speaker turns is not direct but it is filtered through the dependence of the formant space on the speaker turns themselves. In fact warping factors are directly dependent on the formant space which can shift because of the speaker turn structure (speakers speak differently according to their current addressee to facilitate the conversation flow adapting even the pitch of their voice). Therefore we investigated the use of pitch adaptive spectral analysis aiming to find features which are at the same time more suitable for VTLN and more speaker independent. Not only the adoption of these features provided consistent word error rate reductions, particularly in the multiparty conversational domain of

meetings, but we also found evidence of better speaker independence.

# Bibliography

Bahl, L. R., Brown, P. F., de Souza, P. V., and Mercer, R. L. (1988). Speech Recognition with Continuous-Parameter Hidden Markov Models. In *Proc. ICASSP*, volume 1, pages 40–43.

Bahl, L. R., Jelinek, F., and Mercer, R. L. (1983). A Maximum Likelihood Approach to Continuous Speech Recognition. *IEEE Journal of Pattern Analysis and Machine Intelligence*, 5:179–190.

Baker, J. K. (1975). The Dragon System – an Overview. *IEEE Transactions on Acoustics, Speech and Signal Processing*, ASSP-23(1):24–29.

Baum, L. (1972). An Inequality and Associated Maximization Technique in Statistical Estimation of Probabilistic Functions of Markov Processes. *Inequalities*, 3:1–8.

Baum, L. E., Petrie, T., Soules, G., and Weiss, N. (1970). A Maximisation Technique Occuring in the Statistical Analysis of Probabilistic Functions of Markov Chains. *Annals of Mathematical Statistics*, 41:164–171.

Bozkurt, B. and Couvreur, L. (2005). On the Use of Phase Information for Speech Recognition. In *Proc. EUSIPCO*.

Bulyko, I., Ostendorf, M., Siu, M., Ng, T., Stolcke, A., and Çetin, Ö. (2007). Web Resources for Language Modeling in Conversational Speech Recognition. *ACM Transactions on Speech and Language Processing*, 5.

Bulyko, I., Ostendorf, M., and Stolcke, A. (2003). Getting More Mileage from Web Text Sources for Conversational Speech Language Modeling using Class-Dependent Mixtures. In *Proc. HLT*, volume 2, pages 7–9.

Burger, S., MacLaren, V., and Yu, H. (2002). The ISL Meeting Corpus: the Impact of Meeting Type on Speech Style. In *Proc. ICSLP*, pages 301–304.

Burget, L. (2004a). Combination of Speech Features using Smoothed Heteroscedastic Linear Discriminant Analysis. In *Proc. ICSLP*, pages 2549–2552.

Burget, L. (2004b). *Complementarity of Speech Recognition Systems and System Combination*. PhD thesis, Brno University of Technology.

Carletta, J., Ashby, S., Bourban, S., Flynn, M., Guillemot, M., Hain, T., Kadlec, J., Karaiskos, V., Kraaij, W., Kronenthal, M., Lathoud, G., Lincoln, M., Lisowska, A., McCowan, I., Post, W., Reidsma, D., and Wellner, P. (2005). The AMI Meeting Corpus: A Pre-Announcement. *Proc. Multimodal Interaction and Related Machine Learning Algorithms Workshop (MLMI–05), Springer LNCS 3869*, pages 28–39.

Chen, J., Xu, B., and Huang, T. (1998). A novel robust feature of speech signal based on the mellin transform for speaker-indipendent speech recognition. In *Proc. IEEE ICASSP*, volume 2, pages 629–632.

Chen, L., Rose, R. T., Parrill, F., Han, X., Tu, J., Huang, Z., Harper, M., Quek, F., McNeill, D., Tuttle, R., and Huang, T. (2006). VACE Multimodal Meeting Corpus. *Proc. Multimodal Interaction and Related Machine Learning Algorithms Workshop (MLMI–06), Springer LNCS 3869*, pages 40–51.

Chu, S. M., Marcheret, E., and Potamianos, G. (2005). Automatic Speech Recognition and Speech Activity Detection in the CHIL Smart Room. *Proc. Multimodal Interaction and Related Machine Learning Algorithms Workshop (MLMI–05), Springer LNCS 3869*, pages 332–343.

Cieri, C., Miller, D., and Walker, K. (2004). The Fisher Corpus: a Resource for the Next Generations of Speech-To-Text. In *Proc. LREC*, pages 69–71.

Claes, T., Dologlou, I., ten Bosch, L., and Coupernolle, D. V. (1997). New Transformations of Cepstral Parameters for Automatic Vocal Tract Length Normalization in Speech Recognition. In *Proc. Eurospeech*, volume 3, pages 1363–1366.

Cohen, J., Kamm, T., and Andreou, A. (1995). Vocal tract normalization in speech recognition: compensating for systematic speaker variability. *J. Acoust. Soc. Am.*, 97(5, Pt. 2):3246–3247.

Davis, K. H., Biddulph, R., and Balashek, S. (1952). Automatic Recognition of Spoken Digits. *Journal of the Acoustical Society of America*, 24(6):627–642.

Davis, S. B. and Mermelstein, P. (1980). Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-28:357–366.

Digalakis, V., Rtischev, D., and Neumeyer, L. (1995). Speaker Adaptation Using Constrained Estimation of Gaussian Mixtures. *IEEE Transactions on Speech and Language Processing*, 3(5):357–366.

Dognin, P. L. (2004). *A Band Pass Transform for Speaker Normalisation*. PhD thesis, University of Pittsburgh.

Droppo, J., Mahajan, M., Gunawardana, A., and Acero, A. (2005). How to Train a Discriminative Front End with Stochastic Gradient Descent and Maximum Mutual Information. In *Proc. ASRU*, pages 41–46.

Dusan, S. (2005a). Estimation of Speaker's Height and Vocal Tract Length from Speech Signal. In *Proc. Eurospeech*, pages 1989–1992.

Dusan, S. (2005b). Vocal Tract Length During Speech Production. In *Proc. Eurospeech*, pages 1366–1369.

Eide, E. and Gish, H. (1996). A parametric approach to vocal tract length normalization. In *Proc. IEEE ICASSP*, volume 1, pages 346–348.

Eskenazi, M. (1993). Trends in Speaking Styles Research. In *Proc. Eurospeech*, pages 501–509.

Evermann, G., Chan, H., Gales, M., Jia, B., Mrva, D., Woodland, P., and Yu, K. (2005). Training lvcsr systems on thousands of hours of data. In *Proc. IEEE ICASSP*, volume 1, pages 209–212.

Evermann, G. and Woodland, P. (2000). Posterior Probability Decoding, Confidence Estimation and System Combination. In *Proc. NIST Speech Transcription Workshop*.

Ezzaidi, H. and Rouat, J. (2000). Comparison of MFCC and Pitch Synchronous AM, FM Parameters for Speaker Identification. In *Proc. ICSLP*, pages 318–321.

Fant, G. (1966). A note on vocal tract size factors and non-uniform f-pattern scalings. *Speech Transactions Laboratory Quarterly Progress and Status Report*, 7(4):22–30.

Faria, A. and Gelbart, D. (2005). Efficient pitch-based estimation of vtln warp factors. In *Proc. Interspeech*, pages 213–216.

Ferguson, J. D. (1980). Hidden Markov Analysis: an Introduction. In *In J. D. Ferguson (Ed.) Hidden Markov Models for Speech, Institute for Defense Analyses*, pages 8–15.

Fiscus, J. G. (1997). A Post-processing System to Yield Reduced Word Error Rates: Recognition Output Voting Error Reduction (ROVER). In *Proc. IEEE Workshop on ASRU*, pages 347–352.

Fiscus, J. G., Ajot, J., and Garofolo, J. S. (2007). The Rich Transcription 2007 Meeting Recognition Evaluation. *Proceedings of the Rich Transcription 2007 Spring Meeting Recognition Evaluation, Springer LNCS*, pages 373–389.

Fiscus, J. G., Ajot, J., MIchel, M., and Garofolo, J. S. (2006). The Rich Transcription 2006 spring meeting recognition evauation. In *Machine learning for multimodal interaction: Proceedings of MLMI '06*, number 4299 in Lecture Notes in Computer Science, pages 309–322. Springer.

Fitch, W. and Giedd, J. (1999). Morphology and Development of the Human Vocal Tract: A Study Using Magnetic Resonance Imaging. *Journal of the Acoustical Society of America*, 106:1511–1522.

Fitch, W. T. S. (1994). *Vocal Tract Length Perception and the Evolution of Language*. PhD thesis, Department of cognitive and linguistic sciences, Brown University.

Fitt, S. (2000). Documentation and User Guide to Unisyn Lexicon and Post-Lexical Rules. Technical report, Centre for Speech Technology Research, University of Edinburgh.

Fletcher, H. (1940). Auditory Patterns. *Reviews of Modern Physics*, 12:47–65.

Forney, G. D. (1973). The Viterbi Algorithm. *Proceedings of the IEEE*, 61:268–278.

Fossler-Lussier, E. (2003). A Tutorial on Pronunciation Modeling for Large Vocabulary Speech Recognition. In *Text- and Speech-Triggered Information Access*, Lecture Notes in Computer Science, pages 38–77. Springer.

Gales, M. (1999). Semi-Tied Covariance Matrices for Hidden Markov Models. *IEEE Transactions on Speech and Audio Processing*, 7:272–281.

Gales, M. and Woodland, P. (1996). Mean and variance adaptation within mllr framework. *Computer Speech and Language*, 10:249–264.

Gales, M. and Woodland, P. (1998). Maximum Likelihood Linear Transformations for HMM-Based Speech Recognition. *Computer Speech and Language*, 12:75–98.

Garau, G., Renals, S., and Hain, T. (2005). Applying Vocal Tract Length Normalization to Meeting Recordings. In *Proc. Eurospeech*, pages 265–268.

Garofolo, J. S., Laprun, C. D., Mitchel, M., Stanford, V. M., and Tabassi, E. (2004). The NIST Meeting Room Pilot Corpus. In *Proc. LREC*.

Garrod, S. and Pickering, M. J. (2004). Why is conversation so easy? *Trends in Cognitive Sciences*, 8(1):8–11.

Gauvain, J. and Lee, C. (1994). Maximum A Posteriori Estimation for Multivariate Gaussian Mixture Observation of Markov Chains. *IEEE Transactions on Speech and Audio Processing*, 2(2):291–298.

Ghulam, M., Fukuda, T., Horikawa, J., and Nitta, T. (2004). A Noise-Robust Feature Extraction Method based on Pitch-synchronous ZCPA for ASR. In *Proc. ICSLP*, pages 133–136.

Giuliani, D. and Brugnara, F. (2006). Acoustic Model Adaptation with Multiple Supervision. In *Proc. TC-STAR Workshop on Speech-to-Speech Translation*, pages 151–154.

Godfrey, J. J., Holliman, E. C., and McDaniel, J. (1992). SWITCHBOARD: Telephone Speech Corpus for Research and Development. In *Proc. ICASSP*, volume 1, pages 517–520.

Gopinath, R. A. (1998). Maximum Likelihood Modeling with Gaussian Distributions for Classification. In *Proc. ICASSP*, volume 2, pages 661–664.

Gu, L. and Rose, K. (2001). Perceptual Harmonic Cepstral Coefficients for Speech Recognition in Noisy Environments. In *Proc. IEEE ICASSP*, volume 1, pages 125–128.

Haeb-Umbach, R. (1999). Investigations on Inter-Speaker Variability in the Feature Space. In *Proc. ICASSP*, volume 1, pages 397–400.

Hain, T., Burget, L., Dines, J., Garau, G., Karafiat, M., Lincoln, M., Leewven, D. V., and Wan, V. (2007a). The 2007 ami(da) system for meeting transcription. *Proceedings of the Rich Transcription 2007 Spring Meeting Recognition Evaluation, Multimodal Technologies for Perception of Humans: International Evaluation Workshops CLEAR 2007 and RT 2007*, pages 414–428.

Hain, T., Burget, L., Dines, J., Garau, G., Karafiat, M., Lincoln, M., McCowan, I., Moore, D., Wan, V., Ordelman, R., and Renals, S. (2005a). The 2005 AMI system for the transcription of speech in meetings. In *Machine learning for multimodal interaction: Proceedings of MLMI '05*, number 3869 in Lecture Notes in Computer Science, pages 450–462. Springer.

Hain, T., Burget, L., Dines, J., Garau, G., Karafiat, M., Lincoln, M., McCowan, I., Moore, D., Wan, V., Ordelman, R., and Renals, S. (2005b). The Development of the AMI System for the Transcription of Speech in Meetings. In *Machine learning for multimodal interaction: Proceedings of MLMI '05*, number 3869 in Lecture Notes in Computer Science, pages 344–356. Springer.

Hain, T., Burget, L., Dines, J., Garau, G., Karafiat, M., Lincoln, M., Vepa, J., and Wan, V. (2006). The ami meeting transcription system: Progress and performance. *Springer LNCS, Proceedings of the Rich Transcription 2006 Spring Meeting Recognition Evaluation*, pages 419–431.

Hain, T., Burget, L., Dines, J., Garau, G., Karafiat, M., Lincoln, M., Vepa, J., and Wan, V. (2007b). The AMI System for the Transcription of Speech in Meetings. In *Proc. IEEE ICASSP*, volume 4, pages 357–360.

Hain, T., Dines, J., Garau, G., Karafiat, M., Moore, D., Wan, V., Ordelman, R., and S.Renals (2005c). Transcription of conference room meetings: An investigation. In *Proc. Eurospeech*, pages 1661–1664.

Hain, T., Woodland, P., Evermann, G., Gales, M., Moore, G. X. L., Povey, D., and Wang, L. (2005d). Automatic Transcription of Conversational Telephone Speech. *IEEE Transactions on Speech and Audio Processing*, 3:1173–1185.

Hain, T., Woodland, P., Niesler, T., and Whittaker, E. (1999). The 1998 HTK system for transcription of conversational telephone speech. In *Proc. IEEE ICASSP*, volume 1, pages 57–60.

Haverinen, H. and Kiss, I. (2003). On-Line Parametric Histogram Equalization Techniques for Noise Robust Embedded Speech Recognition. In *Proc. Interspeech*, pages 3061–3064.

Hermansky, H. (1990). Perceptual Linear Predictive Analysis of Speech. *The Journal of the Acoustical Society of America*, 87:1738–1752.

Hilger, F. and Ney, H. (2006). Quantile Based Histogram Equalisation for Noise Robust Large Vocabulary Speech Recognition. *IEEE Transactions on Audio, Speech, and Language Processing*, 14:845–854.

Hillard, D., Hoffmeister, B., Ostendorf, M., Schlüter, R., and Ney, H. (2007). iROVER: Improving System Combination with Classification. In *Proc. NAACL-HLT*, pages 65?–68.

Hoffmeister, B., Plahl, C., Fritz, P., Heigold, G., Loof, J., Schlüeter, R., and Ney, H. (2007). Development of the 2007 RWTH Mandarin LVCSR System. In *Proc. ASRU*, pages 455–460.

Holmes, W. J. (2000). Improving the Representation of Time Structure in Front-Ends for Automatic Speech Recognition. In *Proc. ICSLP*, volume 2, pages 1073–1076.

Hunt, M. J. (1979). A Statistical Approach to Metrics for Word and Syllable Recognition. In *Proc. JASA*, volume 66, page S35.

Irino, T., Minami, Y., Nakatani, T., Tsuzaki, M., and Tagawa, H. (2002). Evaluation of a speech recognition/generation method based on HMM and STRAIGHT. In *Proc. ICSLP*, volume 4, pages 2545–2548.

Irino, T. and Patterson, R. (2002). Segregating information about the size and shape of the vocal tract using a time-domain auditory model: the stabilized wavelet-Mellin transform. *Speech Communication*, 36:181–203.

Irino, T. and Patterson, R. D. (1999). Stabilised Wavelet Mellin Transform: an Auditory Strategy for Normalising Sound-Source Size. In *Proc. Interspeech*, pages 1899–1902.

Janin, A., Baron, D., Edwards, J., Ellis, D., Gelbart, D., Morgan, N., Peskin, B., Pfau, T., Shriberg, E., Stolcke, A., and Wooters, C. (2003). The ICSI meeting corpus. In *Proc. IEEE ICASSP*, volume 1, pages 364–367.

Jelinek, F., Bahl, L. R., and Mercer, R. L. (1975). Design of a Linguistic Statistical Decoder for the Recognition of Continuous Speech. *IEEE Transactions on Information Theory*, IT-21(3):250–256.

Kakita, K. (1996). Inter-speaker interaction of F0 in dialogs. In *Proc. ICSLP*, volume 2, pages 689–692.

Kawahara, H. (2007). Getting started with STRAIGHT in command mode. Available at: `http://www.indiana.edu/~acoustic/s522/gettingStartedV40_006b.pdf`.

Kawahara, H., Masuda-Katsuse, I., and de Cheveigné, A. (1999). Restructuring Speech Representations Using Pitch Adaptive Time-Frequency Smoothing and Instantaneous-Frequency-Based F0 Extraction: Possible Role of Repetitive Structure in Sounds. *Speech Communication*, 27.

Kim, D., Gales, M., Hain, T., and Woodland, P. (2004a). Using VTLN for broadcast news transcription. In *Proc. ICSLP*, pages 1953–1956.

Kim, S., Eriksson, T., Kang, H., and Youn, D. H. (2004b). A Pitch Synchronous Feature Extraction Method for Speaker Recognition. In *Proc. IEEE ICASSP*, volume 1, pages 405–408.

Kirchhoff, K., Fink, G. A., and Sagerer, G. (2000). Conversational Speech Recognition using Acoustic and Articulatory Input. In *Proc. IEEE ICASSP*, volume 3, pages 1435–1438.

Knapp, C. H. and Carter, G. C. (1976). The Generalised Correlation Method for Estimation of Time Delay. *IEEE Transactions on Acoustics, Speech and Signal Processing*, ASSP-24(4):320–327.

Krauss, R. M. and Pardo, J. S. (2006). Speaker perception and social behaviour: Bridging social psychology and science. *In P.A.M. Van Lange (Eds): Bridging Social Psychology: Benefits of Transdisciplinary Approaches*. `http://www.columbia.edu/$\sim$rmk7/PDF/Bridges.pdf`.

Kumar, N. and Andreou, A. G. (1998). Heteroscedastic Discriminant Analysis and Reduced Rank HMMs for Improved Recognition. *Speech Communication*, 26:283–297.

Lee, K. (1990). Context-Dependent Phonetic Hidden Markov Models for Speaker-Independent Continuous Speech Recognition. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 38(4):599–609.

Lee, L. and Rose, R. (1996). Speaker normalisation using efficient frequency warping procedures. In *Proc. IEEE ICASSP*, volume 1, pages 353–356.

Legetter, C. J. and Woodland, P. C. (1994). Speaker Adaptation of Continuous Density HMMs Using Multivariate Linear Regression. In *Proc. ICSLP*, pages 451–454.

Legetter, C. J. and Woodland, P. C. (1995). Flexible Speaker Adaptation Using Maximum Likelihood Linear Regression. In *Proc. Eurospeech*, pages 110–115.

Levinson, N. (1947). The Wiener RMS Error Criterion in Filter Design and Prediction. *Journal of Mathematics and Physics*, 25:261–278.

Lowerre, B. T. and Reddy, R. (1976). The HARPY Speech Recognition System: Performance with Large Vocabularies. *The Journal of the Acoustical Society of America*, 60(51):510–511.

Magimai-Doss, M., Stephenson, T. A., Ikbal, S., and Bourlard, H. (2004). Modelling auxiliary features in tandem systems. In *Proc. ICSLP*.

Mangu, L., Brill, E., and Stolcke, A. (2000). Finding Consensus Among Words: Lattice-Based Word Error Minimisation. *Computer Speech and Language*, 14(4):373–400.

McCowan, I., Bengio, S., Gatica-Perez, D., Lathoud, G., Monay, F., Moore, D., Wellner, P., and Bourlard, H. (2003). Modeling Human Interactions in meetings. In *Proc. ICASSP*.

McCowan, I. A. (2001). *Robust Speech Recognition using Microphone Arrays*. PhD thesis, Queensland University of Technology, Australia.

McCowan, I. A., Krishna, M. H., Gatica-Perez, D., Moore, D. C., and Ba, S. (2005). Speech Acquisition in Meetings with an Audio-Visual Sensor Array. In *Proc. ICME*, pages 1382–1385.

McDonough, J. (1998). Speaker normalisation with all-pass transforms. In *Proc. ICSLP*. Paper number 869.

Mertins, A. and Rademacher, J. (2005). Vocal Tract Length Invariant Features for Automatic Speech Recognition. In *Proc. ASRU*, pages 308–312.

Miguel, A., Lleida, E., Rose, R., Buera, L., and Ortega, A. (2005). Augmented Space Acoustic Decoding for Modeling Local Variability in Speech. In *Proc. Eurospeech*, pages 3009–3012.

Miguel, A., Lleida, E., Rose, R., Buera, L., Saz, O., and Ortega, A. (2008). Capturing Local Variability for Speaker Normalisation in Speech Recognition. *IEEE Transactions on Audio, Speech, and Language Processing*, 16:578–593.

Mitchel, M., Ajot, J., and Fiscus, J. (2006). The NIST Meeting Room Corpus 2 Phase 1. In *MLMI, Springer LNCS 4299*, pages 13–23. Springer.

Mohri, M., Riley, M., Hindle, M., Ljolje, A., and Pereira, F. (1998). Full Expansion of Context-Dependent Networks in Large Vocabulary Speech Recognition. In *Proc. ICASSP*, pages 665–668.

Molau, S., Hilger, F., and Ney, H. (2003). Feature Space Normalisation in Adverse Acoustic Conditions. In *Proc. ICASSP*, volume 1, pages 656–659.

Molau, S., Kanthak, S., and Ney, H. (2000). Efficient Vocal Tract Normalization in Automatic Speech Recognition. In *Proc. Elektronische Sprachsignalverarbeitung (ESSV)*, pages 209–216.

Molau, S., Pitz, F., and Ney, H. (2001). Histogram Based Normalisation in the Acoustic Feature Space. In *Proc. ASRU*, pages 21–24.

Moore, D. C. and McCowan, I. A. (2003). Microphone Array Speech Recognition: Experiments on Overlapping Speech in Meetings. In *Proc. ICASSP*, volume 5, pages 497–500.

Mostefa, D., Moreau, N., Choukri, K., Potamianos, G., Chu, S. M., Tyagi, A., Casas, J. R., Turmo, J., Cristoforetti, L., Tobia, F., Pnevmatikakis, A., Mylonakis, V., Talantzis, F., Burger, S., Stiefelhagen, R., Bernardin, K., and Rochet, C. (2007). The CHIL Audiovisual Corpus for Lecture and Meeting Analysis inside Smart Rooms. *Language Resources and Evaluation*, 41(3–4):389–407.

NIST (2004). Spring 2004 (RT-04S) Rich Transcription Meeting Recognition Evaluation Plan. Available at: `http://www.nist.gov/speech/tests/rt/2004-spring/documents/rt04s-meeting-eval-plan-v1.pdf`.

Odell, J. J., Valtchev, V., Woodland, P. C., and Young, S. J. (1994). A One Pass Decoder Design for Large Vovabulary Speech Recognition. In *Proc. ARPA Spoken Language Technology Workshop*, pages 405–410.

Paul, D. B. and Baker, J. M. (1992). The Design for the Wall Street Journal-based CSR Corpus. In *Proc. ICSLP*, pages 899–902.

Pickering, M. J. and Garrod, S. (2004). Towards a mechanistic psychology of dialogue. *Behavioural and Brain Sciences*, 27:169–226.

Picone, J. (1993). Signal Modeling Techniques in Speech Recognition. *IEEE Proceedings*, 81:1215–1247.

Pitz, M. (2005). *Investigations on Linear Transformations for Speaker Adaptation and Normalisation*. PhD thesis, RWTH Aachen.

Pitz, M. and Ney, H. (2005). Vocal Tract Normalisation equals Linear Transformation in Cepstral Space. *IEEE Transactions on Speech, and Audio Processing*, 13(5):930–944.

Plante, F., Meyer, G. F., and Ainsworth, W. A. (1995). A Pitch Extraction Reference Database. In *Proc. Eurospeech*, pages 837–840.

Poritz, A. B. (1988). Hidden Markov Models: a Guided Tour. In *Proc. ICASSP*, volume 1, pages 7–13.

Povey, D. (2003). *Discriminative Training for Large Vocabulary Continuous Speech Recognition*. PhD thesis, Cambridge University Engineering Dept.

Quatieri, T. F. (2001). *Discrete Time Speech Signal Processing, Principles and Practice*. Prentice Hall.

Rabiner, L. (1989). A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. *Proc. of the IEEE*, 77(2).

Rabiner, L. R., Cheng, M. J., Rosenberg, A. E., and McGonegal, C. A. (1976). A Comparative Performance Study of Several Pitch Detection Algorithms. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-24(5).

Rao, A. V., Linden, J., Gersho, A., Cuperman, V., and Heidari, R. (2003). Pitch Adaptive Windows for Improved Excitation Coding in Low-Rate Celp Coders. *IEEE Transactions on Speech and Audio Processing*, 11:648–659.

Richardson, F., Ostendorf, M., and Rohlichek, J. R. (1995). Lattice-Based Search Strategies for Large Vocabulary Recognition. In *Proc. ICASSP*, pages 576–579.

Robinson, D. W. and Dadson, R. S. (1956). A Redetermination of the Equal Loudness Relations for Pure Tones. *British Journal of Applied Physics*, 7:166–181.

Robinson, T., Fransen, J., Pye, D., Foote, J., and Renals, S. (1995). WSJCAM0: A British English Speech Corpus for Large Vocabulary Continuous Speech Recognition. In *Proc. IEEE ICASSP*, pages 81–84. Available at:`http://citeseer.ist.psu.edu/robinson95wsjcam.html`.

Saon, G., Padmanabhan, M., and Gopinath, R. (2002). Eliminating Inter-Speaker Variability Prior to Discriminant Transforms. In *Proc. ICASSP*, volume 1, pages 73–76.

Saon, G., Padmanabhan, M., Gopinath, R., and Chen, S. (2000a). Maximum Likelihood Discriminant Feature Spaces. In *Proc. ICASSP*, volume 2, pages 129–132.

Saon, G., Padmanabhan, M., Gopinath, R., and Chen, S. (2000b). Maximum Likelihood Modeling with Gaussian Distributions for Classification. In *Proc. ICASSP*, volume 2, pages 1129–1132.

Schlüter, R., Bezrukov, I., Wagner, H., and Ney, H. (2007). Gammatone Features and Feature Combination for Large Vocabulary Speech Recognition. In *Proc. IEEE ICASSP*, volume 4, pages 649–652.

Schlüter, R., Zolnay, A., and Ney, H. (2006). Feature combination using linear discriminant analysis and its pitfalls. In *Proc. Interspeech*. Paper 1077.

Schukat-Talamazzini, E., Hornegger, J., and Niemann, H. (1995). Optimal Linear Feature Transformations for Semi-Continuous Hidden Markov Models. In *Proc. ICASSP*, pages 369–372.

Schwartz, R., Chow, Y., imball, O., Roucos, S., Krasner, M., and Mahoul, J. (1985). Context-Dependent Modeling for Acoustic-Phonetic Recognition of Continuous Speech. In *Proc. ICASSP*, pages 1205–1208.

Sena, A. D. and Rocchesso, D. (2004). A Fast Mellin Transform with applications in DAFX. In *Proc. 7th International Conference on Digital Audio Effects*, volume 2007, pages 65–69.

Smith, D. R. R. and Patterson, R. D. (2005). The Interaction of Glottal Pulse Rate and Vocal-Tract Length in Judgements of Speaker Size, Sex and Age. *JASA, Acoustical Society of America Journal*, 118:3177–3186.

Smith, D. R. R., Patterson, R. D., Turner, R., Kawahara, H., and Irino, T. (2005). The Processing and Perception of Size Information in Speech Sounds. *J. Acoust. Soc. Am.*, 117(1):305–318.

Stephenson, T. A., Escofet, J., Magimai-Doss, M., and Bourlard, H. (2002). Dynamic Bayesian Network based Speech Recognition with Pitch and Energy as Auxiliary Variables. In *Proc. IEEE Workshop in Neural Networks for Signal Processing*, pages 637–646.

Stolcke, A., Wooters, C., Mirghafori, N., Pirinen, T., Bulyko, I., Gelbart, D., Graciarena, M., Otterson, S., Peskin, B., and Ostendorf, M. (2004). Progress in meeting recognition: The icsi-sri-uw spring 2004 evaluation system. *Proceedings of the Rich Transcription 2004 Spring Meeting Recognition Evaluation, NIST IEEE ICASSP 2004 Meeting Recognition Workshop*, pages 26–384.

Talkin, D. (1995). A robust algorithm for pitch tracking (RAPT). In Kleijn, W. B. and Paliwal, K. K., editors, *Speech Coding and Synthesis*, pages 495–518. Elsevier.

Üebel, L. F. and Woodland, P. C. (1999). An Investigation into Vocal Tract Length Normalisation. In *Proc. Interspeech*, pages 2527–2530.

Umesh, S., Cohen, L., Marinovic, N., and Nelson, D. (1996). Frequency-warping in speech. In *Proc. ICSLP*, volume 1, pages 414–417.

Umesh, S., Cohen, L., Marinovic, N., and Nelson, D. (1999). Scale transform in speech analysis. *IEEE Transactions on Speech and Audio Processing*, 7(1):40–45.

Viterbi, A. J. (1967). Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Algorithm. *IEEE Transactions on Information Theory*, IT-13:260–269.

Wakita, H. (1977). Normalisation of Vowels by Vocal-Tract Length and its Application to Vowel Identification. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 25(2).

Wan, V. and Hain, T. (2006). Strategies for Language Model Web Data Collection. In *Proc. ICASSP*, volume 1, pages 1069–1072.

Wassner, H. and Chollet, G. (1996). New Cepstral Representation Using Wavelet Analysis and Spectral Transformation for Robust Speech Recognition. In *Proc. ICSLP*, volume 1, pages 260–263.

Wegmann, S., McAllaster, D., Orloff, J., and Peskin, B. (1996). Speaker normalisation on conversational telephone speech. In *Proc. IEEE ICASSP*, volume 1, pages 339–341.

Weintraub, M., Taussig, K., and Snodgrass, A. (1996). Effect of Speaking Style on LVCSR Performance. In *Proc. ICSLP*, pages 16–19.

Welling, L., Ney, H., and Kanthak, S. (2002). Speaker adaptive modeling by vocal tract normalisation. *IEEE Transactions on Speech and Audio Processing*, 10:415–426.

Westphal, M. (1997). The Use of Cepstral Means in Conversational Speech Recognition. In *Proc. European Conference on Speech Communication and Technology, EUROSPEECH*, pages 1143–1146.

Woodland, P. C., Gales, M. J. F., Pye, D., and Young, S. J. (1997). The Development of the 1996 HTK Broadcast News Transcription System. In *Proc. of the Speech Recognition Workshop*, pages 73–78.

Young, S., Evermann, G., Gales, M., Hain, T., Kershaw, D., Liu, X., Moore, G., Odell, J., Ollason, D., Povey, D., Valtchev, V., and Woodland, P. (2006). *The HTK book (v3.4)*. Cambridge University Engineering Department.

Young, S. J., Odell, J. J., and Woodland, P. C. (1994). Tree-Based State Tying for High Accuracy Acoustic Modeling. In *Proc. HLT Workshop*, pages 307–312.

Young, S. J., Russell, N. H., and Thornton, J. H. S. (1989). Token Passing: a Simple Conceptual Model for Connected Speech Recognition Systems. Technical Report CUED/F-INFENG/TR38, University of Cambridge.

Yu, H. and Waibel, A. (2000). Streamlining the Front End of a Speech Recogniser. In *Proc. ICSLP*, volume 1, pages 353–356.

Zhan, P. and Waibel, A. (1989). Vocal Tract Length Normalization for Large Vocabulary Continuous Speech Recognition. Technical report, CMU Language Technologies Institute.

Zhan, P. and Westphal, M. (1997). Speaker normalization based on frequency warping. In *Proc. IEEE ICASSP*, volume 2, pages 1039–1042.

Zhu, Q., Chen, B., Morgan, N., and Stolcke, A. (2004). On using MLP Features in LVCSR. In *Proc. Eurospeech*, pages 921–924.

Zilca, R., Navratil, J., and Ramaswamy, G. N. (2003). Depitch and the Role of Fundamental Frequency in Speaker Recognition. In *Proc. IEEE ICASSP*, volume 2, pages 81–84.

Zolnay, A., Kocharov, D., Schlüter, R., and Ney, H. (2007). Using multiple acoustic feature sets for speech recognition. *Speech Communication*, 49:514–525.