UNIVERSIDAD AUTÓNOMA DE MADRID
ESCUELA POLITÉCNICA SUPERIOR

ROBUST SPEECH RECOGNITION UNDER BAND-LIMITED CHANNELS AND OTHER CHANNEL DISTORTIONS

Nicolás Morales Mombiela

Dissertation submitted to the Departamento de Ingeniería Informática
in Partial Fulfillment of the Requirements for the
Degree of Doctor of Philosophy

Universidad Autónoma de Madrid, Spain
June, 2007

Department:	Ingeniería Informática Escuela Politécnica Superior (EPS) Universidad Autónoma de Madrid (UAM), Spain
PhD. Thesis:	Robust Speech Recognition Under Band-limited Channels and other Channel Distortions
Author:	Nicolás Morales Mombiela Licenciado en Ciencias Físicas (UAM)
Supervisor:	Doroteo Torre Toledano Doctor Ingeniero de Telecomunicación (UPM) Associate Professor (UAM)
Co-supervisor:	Javier Garrido Salas Doctor en Ciencias Físicas (UAM) Associate Professor (UAM)
Year:	2007
Committee:	

Abstract

During the last decades, speech recognition has flourished both, for its potential applications and the improvement of recognition systems. Current Automatic Speech Recognition (ASR) systems are capable of high reliability when tested in controlled conditions (normally this implies high signal to noise ratio – SNR – and similar conditions in training and testing), but accuracy is significantly degraded when speech is subject to sources of variability. Among the possible sources of variability we study in this work those introduced by the transmitting channel, and more specifically those that completely remove parts of the spectrum (we call these band-limiting distortions).

Band-limiting distortions may appear, for example, in historical recordings, where due to the limited capabilities of recording equipment and storage units, low sampling frequencies may constrain the available bandwidth below 4 or even 2 kHz. In fact, the initial motivation for our study was the possibility of transcribing historical recordings from the 20th century, where a single speech file may even consist of several fragments with different bandwidths (this is part of the National Gallery of the Spoken Word project, NGSW). Telephone-transmitted signals are another example where speech is band-limited, and similarly, signals transmitted from on-board systems (like cars or aeroplanes) may present different band-limitations.

One of the keys to success in ASR is to achieve similar conditions between training data used to obtain reliable statistical acoustic models and test speech. In particular when speech is subject to distortions such as noise, convolutive filters, etc., models trained under different conditions (for example clean and full-bandwidth data) will suffer significant accuracy degradation. An important means of robustness is a reliable signal parameterizer that extracts relevant information from the linguistic/textual content of speech, while discarding irrelevant information (for the goal of ASR), pertaining to the mentioned typical sources of variability. However, when the parameterizer module itself is not able to remove all the undesired information that compromises ASR accuracy, it is possible to modify the acoustic models to match the conditions of input speech (model-side robustness) or conversely, distorted speech features may be modified to resemble the conditions of data used to train the acoustic models (feature-side robustness). In our work we study different implementations of the latter approach and compare results to those obtained with model-side robustness (the strategy typically used to deal with band-limiting distortions). We also study the possibility of combining feature-side and model-side approaches for increased accuracy.

Under a number of circumstances, feature compensation may be more convenient and may even outperform model-side solutions for ASR. For example, when the number of different distortions affecting test data is large our approach compensates speech from different distortions to resemble undistorted speech and allows keeping active a single and unmodified speech recognizer for which the process of compensation remains transparent. Additionally, feature compensation algorithms may be

trained with only limited amounts of data, while model-based robustness solutions tend to be more resource-demanding. Also, in portable devices where memory and computational loads are an important limitation, feature compensation offers a light and reliable solution, allowing to store a single set of acoustic models and performing ASR with only one recognition engine (we show that the memory space required to store corrector functions is at least between one and two orders of magnitude below that of full acoustic models).

In our work, we propose algorithms for feature compensation, whose common ground is the learning of a transformation between the distorted feature space (band-limited spectrum) and the undistorted space (full-bandwidth). In our experiments, distorted features are compensated using the appropriate transformations in order to obtain pseudo-undistorted features, so that they can be used for recognition with models trained with undistorted data. We propose different solutions that meet possible constraints of real systems, like availability or not of stereo-data for training (stereo data refers to speech samples recorded simultaneously in clean and band-limited environments), training data scarcity, data storage limitations, blind classification and compensation of multiple distortions, etc. A large number of experiments have been conducted and shed light in the potential performance of different settings and variations of the feature compensation approaches proposed, and on a variety of problems and conditions. Results are always compared to those of other possible solutions to the problem, consisting in classical robustness methods like Cepstral Mean Normalization (CMN) and model-side robustness (model adaptation and retraining). Evaluation is performed applying artificial band-limiting filters (low-pass and band-pass filters) on full-bandwidth data, as well as with real telephone data, which poses more challenging conditions due to the existence of multiple distortions (not only band-limitations, but also other types of convolutional filters and additive noises).

Our experiments show that performance of ASR using feature compensation is competitive in most of the scenarios considered with that of model-side approaches, typically employed to deal with bandwidth limitations (in most cases our proposed multivariate feature compensation outperforms standard model adaptation). Also, in the case of systems subject to multiple distortions, we show that feature compensation allows for a simpler implementation and produces better accuracy, too. Finally, in the case of training data scarcity, feature compensation has proved to be a very efficient solution. Additionally, feature compensation and model robustness techniques may be combined for improved system robustness.

Resumen

En las últimas décadas el campo del reconocimiento de voz ha alcanzado gran importancia gracias, en gran medida, a la aparición de nuevas aplicaciones y la mejora de las tasas de reconocimiento. Los sistemas de reconocimiento actuales ofrecen gran fiabilidad cuando se utilizan en condiciones controladas (lo que normalmente significa una buena relación señal-ruido y condiciones similares en las etapas de entrenamiento y evaluación), pero su grado de acierto se ve severamente reducido cuando la voz está sometida a fuentes de variabilidad. Entre las posibles causas de variabilidad, en este trabajo tratamos aquellas debidas a los canales transmisores, y en concreto a aquellos canales que eliminan completamente parte del espectro (distorsiones limitadoras de banda).

Podemos encontrar distorsiones limitadoras de banda, por ejemplo, en grabaciones históricas, en las que son habituales bajas frecuencias de muestreo que pueden llegar a restringir el ancho de banda por debajo de los 4 Khz. o incluso 2 Khz. (esto es debido a las limitadas prestaciones de equipos de grabación o soportes de almacenamiento antiguos). De hecho, una de las principales motivaciones de este trabajo fue la posibilidad de transcribir una biblioteca de archivos sonoros del siglo XX, en los cuales incluso se puede llegar a encontrar fragmentos con distinta calidad de señal (esto es parte del proyecto National Gallery of the Spoken Word [NGSW]). Otras situaciones en las que se pueden encontrar limitaciones en el ancho de banda son voz transmitida por un canal telefónico o voz transmitida desde sistemas integrados en vehículos, como en automóviles y aviones.

Uno de los aspectos fundamentales a la hora de obtener un sistema robusto es que la voz utilizada para entrenamiento de modelos estadísticos debe asemejarse lo más posible a la voz que se pretende reconocer. Si esto no se cumple, por ejemplo cuando la señal a reconocer está afectada por ruido, filtros convolutivos, etc., las tasas de acierto de un sistema entrenado en condiciones distintas (típicamente voz limpia y con un ancho de banda completo), se verán seriamente afectadas. Una medida eficaz de robustez es la utilización de un extractor de parámetros capaz de obtener una representación fiable de la información relevante de la señal, y que al mismo tiempo elimine toda la información inútil desde el punto de vista del reconocimiento de voz. Sin embargo, cuando el parametrizador no es capaz de eliminar todos los elementos de variabilidad externos al mensaje (que no hacen sino complicar el reconocimiento), se puede mejorar la fiabilidad del sistema mediante una de las siguientes estrategias: modificación de los modelos acústicos para acercarse a las condiciones de la voz a reconocer (robustez de modelos), o alternativamente, modificación de la voz que se quiere reconocer (o su representación paramétrica), de modo que resulte natural para el sistema (compensación de voz). En nuestro trabajo estudiamos diferentes técnicas de compensación de la representación paramétrica de la voz y comparamos nuestros resultados con los de técnicas de robustez de modelos, que suelen ser utilizadas en el caso de voz limitada en ancho de banda. También mostramos la posibilidad de combinar ambas estrategias para obtener mejores tasas de reconocimiento.

Existen múltiples situaciones en las que la compensación de parámetros puede ser más conveniente y obtener mayores tasas de acierto que las técnicas de robustez de modelos. Por ejemplo, cuando un sistema recibe voz con distintos anchos de banda, la compensación de parámetros permite la utilización de un sistema único y sencillo para el cual la compensación resulta transparente. Además, nuestras técnicas ofrecen gran rendimiento incluso cuando existen muy pocos datos de entrenamiento, mientras que las estrategias de modelos robustos suelen requerir mayores cantidades de datos. También, en dispositivos móviles, o aquellos en los que economizar en recursos de memoria y cómputo es una prioridad, las técnicas de compensación de parámetros pueden resultar muy provechosas porque no requieren gran espacio de almacenamiento y permiten la utilización de un único sistema de reconocimiento para una amplia gama de distorsiones (en nuestro trabajo mostramos que el espacio de almacenamiento es al menos entre uno y dos órdenes de magnitud menor que el de modelos acústicos completos).

Las técnicas de compensación de parámetros propuestas en esta tesis consisten en la aplicación de transformaciones entre los espacios limitado en banda y de ancho de banda completo; los vectores de parámetros extraídos de voz limitada en ancho de banda son compensados utilizando transformaciones adecuadas y de este modo se pueden utilizar para reconocimiento de voz mediante un sistema entrenado con voz que abarca el espectro completo. En la sección experimental hemos considerado situaciones reales como pueden ser la existencia o no de datos de entrenamiento estéreo (ficheros grabados simultáneamente en condiciones de espectro completo y espectro limitado), escasez de datos de entrenamiento, limitaciones de capacidad de almacenamiento, compensación con clasificación automática del ancho de banda, voz con múltiples distorsiones en distintos fragmentos, etc. La gran cantidad de pruebas realizadas también permite obtener una clara idea del potencial de las técnicas propuestas en diferentes situaciones y condiciones, y los resultados son comparados siempre con aquellos de técnicas clásicas, como Normalización de la Media Cepstral (CMN en sus siglas en inglés) y las citadas técnicas de robustez de modelos (adaptación de modelos o re-entrenamiento de modelos). En nuestros experimentos hemos considerado tanto filtrados frecuenciales artificiales aplicados sobre voz con el espectro completo, como datos telefónicos reales, que suponen un significativo aumento de la dificultad debido a la interacción de múltiples distorsiones (no solo limitaciones del ancho de banda, sino también otros filtros convolutivos, ruidos aditivos, etc.).

Nuestros resultados muestran que las técnicas de compensación de parámetros propuestas son en la mayoría de las situaciones consideradas competitivas con las de robustez de modelos (de hecho en la mayoría de los problemas considerados hemos obtenido mejores resultados que con técnicas de adaptación de modelos). También en los casos de voz sometida a múltiples distorsiones mostramos que nuestra solución se puede implementar de un modo mucho más sencillo que soluciones basadas en robustez de modelos, y con mejores tasas de acierto. Además en el caso de datos de entrenamiento escasos la compensación de parámetros se ha mostrado como una solución muy eficiente. Finalmente mostramos que ambas estrategias se pueden combinar para obtener mayores tasas de reconocimiento.

Acknowledgements

This work would not have been possible without the teachings, help and support of many people towards whom I have become very much in debt.

Most importantly I would like to thank my supervisor Doroteo Torre Toledano for everything he has contributed in all aspects of this work. For the last three years Doroteo has dedicated me as much time as a PhD candidate can expect. He has taught me and guided me through my research, while giving me absolute freedom and we have enjoyed countless scientific discussions.

I would also like to thank especially my co-supervisor Javier Garrido. With his knowledge, experience and dedication Javier has always helped me and given me advice whenever it was needed.

I have also had the enormous luck of collaborating with John Hansen during the course of two six-months stays in his research laboratories, first in Colorado and later in Texas. I thank John for his endless flow of ideas and the source of inspiration he has been for me.

In my five years in the Escuela Politécnica Superior of Universidad Autónoma de Madrid I have also greatly benefited from working with José Colás (who introduced me to the field of speech recognition) and Joaquín González.

As a result of my collaboration with different groups in Spain and the USA, many people have been involved in my work and have had an important influence in the development of this Thesis both for their scientific and moral support. I would like to thank my colleagues from HCTLab, Daniel, Sergio, Eduardo, Ricardo, Javier and Víctor with all of whom I have shared hours and hours of work and fun. Also I truly thank my friends from CSLR, Andi, Vanessa, Xiang-Xiang, etc. and those from CRSS, Ayako, Vinod, Wooil, Rongqing, Murat, Pongtep, Nitish, Vaishnevi, (the list goes on and on as the group is extremely active) and those from ATVS, Dani, Javi, Iñaki, Alex, and all the others.

In 2006 I also had the privilege of spending 6 months as an intern in IBM TJ Watson Research Center. I would like to thank my manager Yuqing Gao and my mentor Liang Gu, who greatly broadened my research interests and knowledge. They also gave me the opportunity to work on *real* projects and charged me with responsibilities that made my time there extremely exciting. In TJ Watson I also had the chance to work and interact with some of the most prominent researchers in the field, for whom I feel great admiration.

I would like to dedicate this work to my parents and my brother. They have given me love, values, confidence and they have, in summary, always been there. Also, I want to dedicate this to the friends with whom I have shared my life. I would like to thank Ana who is a model for me and someone from whom I have borrowed so many things, and Juan Manuel who also had a large influence in my life and with whom I have shared so many thoughts and good moments. A very special place is for Ayako who has always inspired me with her opinions and life attitude. Many thanks to Sara who has given me

strength with her positive reinforcement. And thanks to all my fiends from my undergraduate years, Tania, Olga, Alberto, Irene, Jorge, Elena, Rodri, Micky, to my *Erasmus* friends (all of you, the list is too long ☺) and my friends from here and there, María, Mavi, Ana V., Paola, Erin…thanks to everyone who made an effort to make things better.

My work was supported in its majority by the Spanish Ministry of Education through an *FPU* scholarship. I would also like to thank HCTLab and ATVS from Universidad Autónoma de Madrid and CRSS from the University of Texas at Dallas for their financial support in different moments in this Thesis.

Contents

List of Figures

List of Tables

List of Abbreviations and Acronyms

ASR: Automatic Speech Recognition.

BP300-3400Hz: Band-Pass filter with cut-off frequencies at 300 Hz and 3400 Hz.

CMN: Cepstral Mean Normalization.

EM: Expectation-Maximization.

FB: Full-Bandwidth.

FFT: Fast Fourier Transform.

GMM: Gaussian Mixture Model.

HMM: Hidden Markov Model.

HTK: Hidden Markov model ToolKit.

IEC: Individual Environment Class combination.

LATA: Local Access and Transport Area.

LDC: Linguistic Data Consortium.

LM: Language Model.

LP#kHz: Low-Pass filter with cut-off frequency at # kHz.

LPC: Linear Predictive Coefficients.

LVCSR: Large Vocabulary Continuous Speech Recognition.

MAP: Maximum A Posteriori.

MEC: Multi-Environment Class creation.

MFCC: Mel Frequency Cepstral Coefficients.

ML: Maximum Likelihood.

MLLR: Maximum Likelihood Linear Regression.

MMSE: Minimum Mean Squared Error.

MSE: Mean Squared Error.

NGSW: National Gallery of the Spoken Word.

NIST: National Institute of Standards and Technology.

PCA: Principal Component Analysis.

pdf: Probability Density Function.

PLP: Perceptual Linear Prediction.

PSTN: Public Switched Telephone Network.

RASTA: RelAtive SpecTrAl processing of speech.

RATZ: MultivaRiate-gAussian-based cepsTral normaliZation.

RMSE: Root Mean Squared Error.

SNR: Signal-to-Noise Ratio.

SPLICE: Stereo-based Piecewise LInear Compensation for Environments.

STFT: Short-Time Fourier Transform.

VTS: Vector Taylor Series.

1

Introduction

The term speech technology refers to several research fields that study coding, synthesis, recognition and processing of speech by computers; speech recognition, in particular, deals with the problem of automatically extracting the textual content of speech, allowing machines to understand humans using our most natural means of communication.

According to [Reddy, 1976], initial attempts on automatic speech recognition date from the 1950s when the first vowel and digit recognizers were created. Since the early 1970s, DARPA financed projects that promoted the interest of research in this field, but it has been during the last decades that speech recognition has flourished both, for its potential applications and the increasing accuracy of ASR systems. Among the possible applications of speech technologies, we mention the following:

- Telephone services such as banking, ticket selling, question-answering systems, etc.
- Dictation systems for personal computers.
- Increasing accessibility for handicapped people.
- Simplifying human-machine communication and allowing hands-free interaction and freedom of mobility for the user.
- Transcribing, indexing and searching audiovisual material (such as radio and TV recordings).

Current automatic speech recognition systems are capable of high reliability when tested in controlled conditions (normally this implies high SNR and similar conditions in training and testing). Figure 1.1 [Pallett, 2003], shows the historical evolution of the National Institute of Standards and Technology (NIST) evaluations. For example, as of 2003, broadcast news ASR leader systems achieved up to 90% accuracy rate for a vocabulary of up to 10,000 words (of course, accuracy depends on a large number of factors, including speaker characteristics, context of the conversation, vocabulary size, signal distortions, possibility or not of model adaptation, etc.); commercial systems have even been advertised as having a potential 99% when tested under ideal conditions (although such impressive numbers obviously depend on testing conditions). However, speech recognition systems are

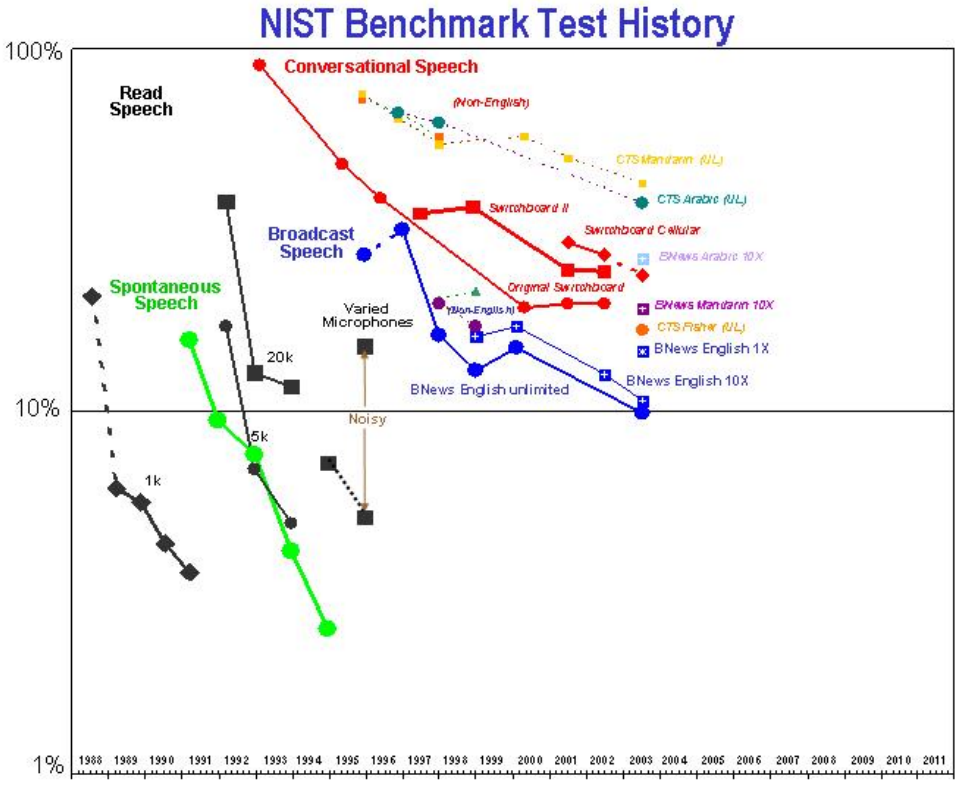


Figure 1.1 Summary of NIST's benchmark ASR test history (from [Pallet, 2003]).

still not fully integrated in our lives and in some cases they might even become a nuisance due to poor system performance (for example, automatic dialogue systems may frustrate users when accuracy is low or design deficient). Speech variability is the consequence of a combination of effects that may be generally classified as human related (physiological, cultural, behavioral, mood, etc.), environment related (such as background noises and reverberations) and transmitting-channel related (channel noise as well as convolutional distortions). All these sources of speech variability are capable of hiding or even obliterating the information contained in the speech signal and thus compromise understandability. However, it has been observed that under different signal distortions, human listeners' understanding lies at least one order of magnitude over machine understanding (at least as of 1997 [Lippmann, 1997]). This represents a major research challenge for the speech recognition community and explains why robustness has been a hot topic since the 1990s.

Among the mentioned types of distortions, in this Thesis we focus on channel distortions and more specifically on band-limiting channels, which completely remove parts of the spectrum of speech signals, causing the speech representation to differ very significantly from that of full-bandwidth signals. Band-limitations may appear in a variety of practical situations. For example, historical recordings typically present only the low-frequency portion of the spectrum due to low sampling frequencies of the recording equipment or storage materials. For example, recordings from the early 1900s stored in wax discs may be limited to a frequency range below 2 kHz [Hansen *et al.*, 2000]. Also, typical telephone channels impose a band-pass filter on the signal that allows transmission of most of the speech information but significantly modifies the characteristics of the spectrum. It may also be the case that on-board systems (mounted in cars, airplanes, etc.) transmit through band-limiting channels

[Abut *et al.*, 2004] [Denenberg *et al.*, 1993]. In other cases, the original sampling frequency may be low in order to save memory space and this will produce the same effect as a low-pass filter if speech is upsampled (this may happen, for example, in non-professional recordings, and those made with mobile devices).

Reconstruction of band-limited speech has been studied since the early 1990s for the purpose of expanding the bandwidth of telephone-transmitted signals in order to make them sound more natural [Cheng *et al.*, 1994] [Avendano *et al.*, 1995] [Yasukawa, 1996]. However, only recently has bandwidth extension been suggested as a means to perform ASR of band-limited speech using full-bandwidth acoustic models [Morales *et al.*, 2005a] [Seltzer *et al.*, 2005] [Kim and Hansen, 2006]. In the past, ASR applications dealt with band-limiting distortions by training specific acoustic models with data from the particular distortion, especially in the case of telephone speech, for which large amounts of training data are available. In this Thesis we propose an alternative solution: instead of modifying the system's acoustic models to fit the characteristics of distorted speech, we expand band-limited speech so that it will *look familiar* to acoustic models trained with full-bandwidth data. Our approach may be better in than model-side approaches in some cases, such as when adaptation data from a distorting channel is scarce, when a system receives speech from a variety of distortions or when the problem imposes constraints on memory or computational usage. In addition, signal compensation and model robustness approaches may be combined in order to obtain increased accuracy.

1.1 Thesis Goals

From an early stage our primary goals were:

- To model the effect of band-limiting distortions over speech signals and study how they impact the characteristics of the parameterized signal.
- To study practical system implementations and algorithms to deal with the effects of these distortions.
- To show explicitly the degradation of ASR accuracy rates as a result of channel distortions and how the proposed solutions improve baseline results. This is done by means of a thorough experimental section where a large number of configurations are evaluated.
- To study the relation between quality of signal restoration and ASR performance. Although the goal of feature compensation techniques is to reconstruct pseudo-undistorted speech representations, we argue that for the goal of speech recognition, perfect signal restoration is not absolutely necessary, as long as the decoder finds the correct state sequence.
- To propose solutions to real cases where channel distortions affect speech, such as telephone signals, spoken document retrieval of historical records, low-sampling frequency, etc.
- To study the possibility of using canonical models and front-end features that may be adapted to many different distortions by means of simple corrector algorithms. For example, when a system may receive speech from a variety of channels or environments, a framework based on a single acoustic model set and adequate feature compensation seems more efficient than handling multiple acoustic model sets.

- To evaluate performance of our proposed solutions on real data, for which distortions may be more complicated than simple band-limiting distortions.
- To provide qualitative and quantitative comparisons of memory and computation requirements of different robustness strategies. Although our research has not been dictated by the need of optimizing memory and computational requirements, these are important factors for potential implementation of the proposed solutions in real systems.

1.2 Overview of this Thesis

In Chapter 2 we present the general principles of current speech recognition technologies and a review of those robustness techniques that are relevant to this Thesis for their relation to our proposed solutions, or because they are employed in our experiments (for more extensive reviews on robustness, see for example [de Mori, 1998] [Huang *et al.*, 2001] [Josifovski, 2002] [Quatieri, 2002]).

In Chapter 3 we present the problem of band-limitations showing practical situations where they may arise. We also present a brief summary of spectral characteristics of phonemes in English which determines how each of these units will be affected by the removal of parts of the spectrum. This is followed by a summary of related work on techniques for bandwidth extension. Finally we propose our mathematical model of band-limiting distortions and present a discussion on how band-limitations affect cepstral features and their assumed decorrelation. This is the motivation for the family of multivariate feature compensation algorithms proposed later in the Thesis.

In Chapter 4 we describe the experimental framework, methodology and tools that are used in our work, as well as the speech corpora employed for evaluation.

Chapter 5 presents the mathematical foundations and practical issues on the proposed algorithms for feature compensation that will be later evaluated.

Chapter 6 introduces two of the most widely used model adaptation methods, Maximum Likelihood Linear Regression (MLLR) and Maximum a Priori adaptation (MAP). These are employed in later chapters for comparison with the proposed feature compensation methods and in order to obtain realistic performance in those comparisons, in this chapter we evaluate ASR performance for MLLR and MAP for different values of their tunable parameters.

Chapters 7 and 8 consist of experiments and discussion for the proposed compensation techniques using two different approaches. Evaluation is performed for a variety of settings and constraints that might arise in real situations (such as speech subject to unidentified distortions, limited amounts of adaptation data, etc.) and results are compared to those with other typical solutions to the problem of training-testing mismatch. We also make a qualitative and quantitative study on computational and memory requirements.

Conclusions and our main contributions are presented in Chapter 9. We also propose a few research lines for future work.

Finally, Appendix A describes the creation of a corpus designed to capture the characteristics of a single telephone channel, a requirement in some of our tests.

2

Speech Recognition Principles and Review of Related Robustness Methods

Speech recognition systems operating under ideal conditions (typically those of a laboratory where training and test data conditions are controlled) obtain high accuracy rates. However, reliability in real conditions is still an unsolved problem and one of the main reasons preventing this technology from becoming ubiquitous. This chapter starts with a brief presentation of an archetypal ASR system based on Hidden Markov Models (HMMs) and the main sources of signal degradation in typical applications. This is followed by a review of a selection of robustness strategies in state-of-the-art systems, which focuses on relevant techniques for speech affected by additive noises and convolutional filters caused by the transmitting channel (for more complete reviews on robustness see, for example [de Mori, 1998] [Huang *et al.*, 2001] [Josifovski, 2002] [Quatieri, 2002]).

2.1 Automatic Speech Recognition

Many system architectures have been historically proposed for ASR; however, typical state-of-the-art speech recognizers employ a structure based on 3 stages (Figure 2.1): a) feature extraction, b) pattern scoring and c) decision. In the following subsections we describe their main principles.

2.1.1 Signal Processing and Feature Extraction

The following steps take place in a typical front-end:

- The speech signal is sampled at a fixed rate, which limits the available frequency bandwidth to the range between 0 Hz to 1/2 of the sampling rate (Nyquist theorem [Nyquist, 1928] [Shannon, 1949]). As the temporal representation of speech is highly redundant and not robust enough for ASR, further processing is performed in order to obtain a more adequate representation.

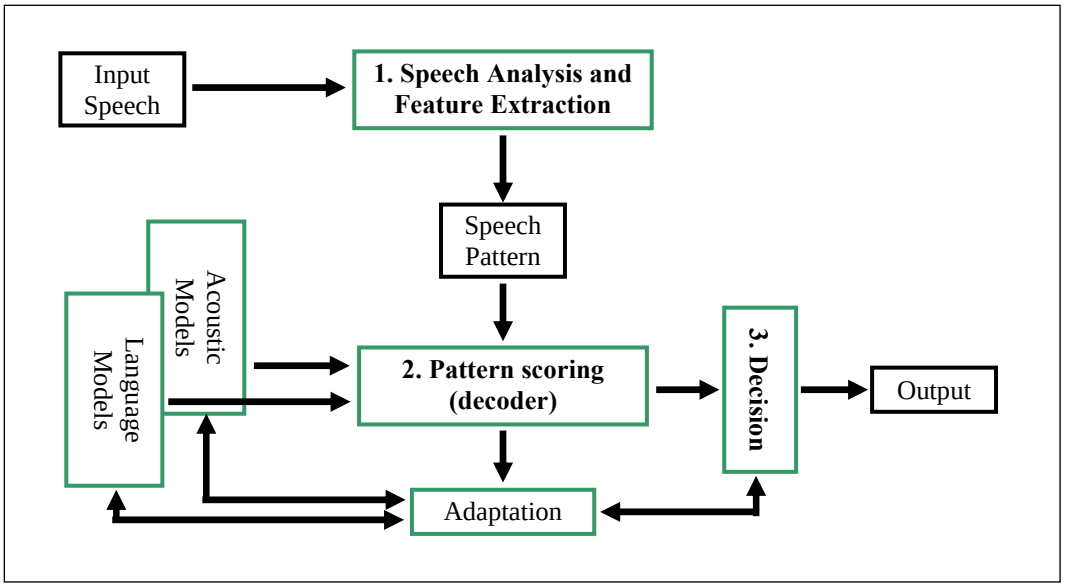


Figure 2.1 General architecture of a robust speech recognizer system, derived from figures in [Junqua and Haton, 1996; p.84] and [Huang *et al.*, 2001; p. 5].

- It is common practice to apply an enhancement technique called pre-emphasis, which boosts high frequencies in the speech signal to compensate the spectral tilt produced in the acoustic signal as a result of radiation from the lips. Pre-emphasis takes place also in the human ear [Moore, 2003; pp. 55-59] in order to flatten the spectrum and is generally considered to improve ASR performance. However, it should be noted that it may increase the SNR if too much noise is contained in the high-frequencies [Vergin *et al.*, 1996].
- The time-signal is divided into a sequence of windows or frames of short duration, each of which contains quasi-stationary parts of the signal (corresponding to quasi-static configurations of the vocal tract). In order to avoid border effects, these windowing functions normally approach zero-values near the borders (Hamming, Hanning, etc.). It is also usual to allow overlapping between consecutive windows in order to increase time resolution (typical window sizes are of the order of 25 ms and window shifts of the order of 10 ms).
- The next step is normally feature extraction. This plays a key role in system robustness and its discussion is deferred to Section 2.4.

2.1.2 Pattern scoring

In the pattern scoring module, speech frames are compared to a system's acoustic models. The probability of emission of individual frames is estimated for each model and results in the generation of the most-likely state sequence (or sequences) for an utterance. Different classification methods exist; for example, in the past Dynamic Time Warping, a dynamic time processing technique employing template-based models became a popular approach [Sakoe and Chiba, 1978] [Morales *et al.*, 2003]. Neural Networks are currently an active field of research and may be combined with HMM modeling [Schwarz *et al.*, 2004]. However, the most widely used acoustic models in current systems are HMMs, in which basic speech units (phonemes, syllables, words, etc.) are represented as a succession of states,

each of which is characterized by a probability of emission (modeled as a probability density function, pdf) and a probability of transition to other states. HMMs are further discussed in Section 2.1.4.

Just as human speech understanding is not solely based on acoustic information (context of speech, speaker characteristics, and other sources of information are critical for human understanding), pattern classification should not be limited to acoustic information. Therefore, robust systems introduce information at multiple levels of the search algorithm, so that the search is simplified or modified according to the available contextual information, Language Models (LMs), target language, etc. Additionally, as shown in Figure 2.1, model adaptation may also be done in real-time, during recognition. For example, acoustic models may be adapted to match a particular speaker's characteristics, or may compensate speech distortions. LMs may also be simplified if the topic of the message is identified.

2.1.3 Decision

The most likely state sequence (or sequences) is passed to application modules which in response take actions (decisions). The decision may be as simple as accepting the most likely textual transcription, but post-processing is also possible for better accuracy or user's experience. For example, when confidence measures are implemented, a system may take further steps towards confirmation of the original hypotheses [Wessel *et al.*, 2001].

Other options include processing of multiple candidates (lattices or N-best lists [Richardson *et al.*, 1995] [Nguyen *et al.*, 1994]), for example cascading several recognition modules or combining their outputs as is done in ROVER [Fiscus, 1997].

2.1.4 Hidden Markov Modeling with Gaussian Mixture pdfs

HMMs are the most widely used method for statistical acoustic modeling for ASR. Among their most relevant advantages, they do not require training data segmentation and allow great flexibility in model topology, integration of acoustic and semantic information, etc. [Rabiner, 1989]. HMMs consist of a doubly stochastic process: a hidden process that represents a sequence of states (in principle representing static configurations of the vocal tract), which in turn generates a sequence of observations (the speech sounds). Models are composed of a succession of states for which the following two sets of probabilities need to be defined (Figure 2.2): the probability of transition to other states and the probability of emission of observations. Transitions between states and in and out of the model are typically represented by transition matrixes for each model, while the most extended means for representing the probability of emission are Gaussian Mixture Models, GMMs (although other possibilities exist, such as Rayleigh Mixtures [Rivet *et al.*, 2007]).

HMMs are a well studied and understood tool. The following three problems arise in a speech recognizer [Rabiner, 1989]:

- The Learning Problem: given a sequence of observations what is the ideal set of parameters that maximize the joint probability of emission? The solution is given by the Baum-Welch algorithm (also known as Forward-Backward algorithm) [Baum *et al.*, 1970].

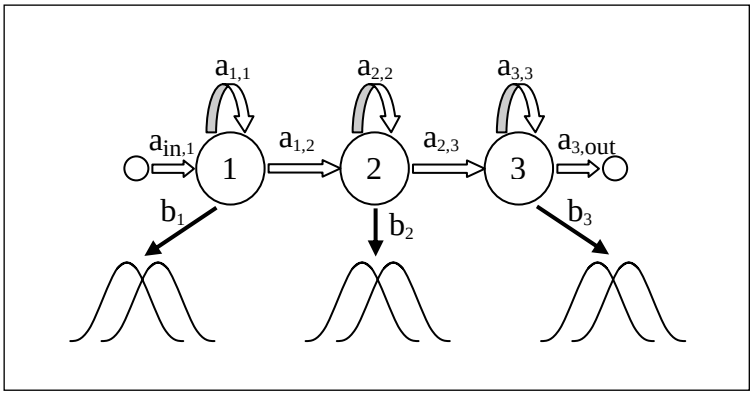


Figure 2.2 Schematic representation of an HMM model. $a_{x,y}$ represent transitions between states in the HMM and in and out of it, while b_x represents GMM pdfs. This model has 3 emitting states and transitions other than those specified have zero probability (although in a general representation it is possible to transition from any state to any other state and in and out of the model).

- The Decoding Problem: given a model and an observation sequence, what is the most likely state sequence? This is solved with the Viterbi algorithm [Viterbi, 1967].
- The Evaluation Problem: given a model and a sequence of observations, what is the probability that the model generated the observations? The solution is the Forward-Backward algorithm (actually the forward part of it).

When an utterance or chain of observations is passed to a recognizer, the best path, or state sequence is defined as:

$$\mathbf{X}_{best} = \max_{\mathrm{x}} \left\{ a_{x(0)x(1)} \prod_{t=1}^{T} b_{x(t)}\left(\mathbf{o}_t\right) \cdot a_{x(t)x(t+1)} \right\}, \tag{2.1}$$

where $\mathbf{X} = \left\{ x(0), x(1), \ldots, x(T) \right\}$ are sequences of states, $a_{x(t-1)y(t)}$ is the probability of transition from state x to state y, and $b_j\left(\mathbf{o}_t\right)$ is the probability of emission of observation $\mathbf{o}_t$ by state j. When pdfs are modeled using GMMs the probability of emission is computed as:

$$b_j\left(\mathbf{o}_t\right) = \prod_{s=1}^{S} \left[\sum_{m=1}^{M_{sj}} c_{jsm} N\left(\mathbf{o}_{st}; \boldsymbol{\mu}_{jsm}, \boldsymbol{\Sigma}_{jsm}\right) \right]^{\gamma_s}, \tag{2.2}$$

where S represents the different input streams (for example in audio-visual speech recognition at least two streams exist: audio features and visual features [Potamianos *et al.*, 2003]. In audio-only speech recognition it is also possible to derive multiple sets of features or streams from a single utterance; however in this Thesis a single stream is always used), γ_s is the weight of each stream, c_{jsm} the weight of each Gaussian mixture and N the Gaussian probability likelihood defined as:

$$N\left(\mathbf{o}; \boldsymbol{\mu}, \boldsymbol{\Sigma}\right) = \frac{1}{\sqrt{\left(2\pi\right)^n \left|\boldsymbol{\Sigma}\right|}} \cdot \exp\left(-\frac{1}{2} \cdot \left(\mathbf{o} - \boldsymbol{\mu}\right)^T \cdot \boldsymbol{\Sigma}^{-1} \cdot \left(\mathbf{o} - \boldsymbol{\mu}\right) \right). \tag{2.3}$$

2.2 Sources of Speech Variability

Given two utterances with the same linguistic or textual content, the widest definition of speech variability (for the problem of ASR) is any factor that causes the two utterances to be non-identical. Distortions are particular causes of variability introduced in the signal after the original message is

produced, but are not the only reason for variability, as this may be caused by different factors in the process of speech production. Sources of variability may be classified according to different criteria. Based on their cause they may be divided as [Junqua and Haton, 1996; pp. 125-189]:

- Production phenomena.
- Distortions introduced by the acoustic environment.
- Distortions produced by the transmission channel.

Figure 2.3, shows a general diagram of how variability appears in the process of voice transmission.

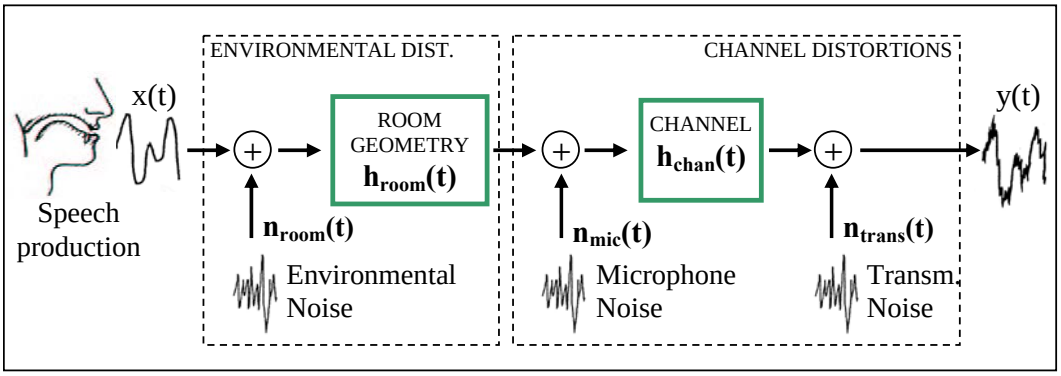


Figure 2.3 Model of speech variability due to environmental and channel distortions (based on a figure by Gallardo-Antolín [2002; p. 6]). Subindex *mic* stands for microphone, *trans* stands for transmission and *chan* for channel (convolutional filters due to the microphone and transmission channel are grouped into h_{chan}).

2.2.1 Production Phenomena

In the process of speech generation a number of circumstances may affect the speech signal. Physiological variability significantly alters the spectrum and is caused by the size of the vocal tract, length and width of the neck, etc. Sociolinguistic factors such as dialect, health or age, as well as speaking rate, mood, etc. also modify the speaker's voice characteristics (a well studied source of variability is the Lombard effect, that appears when speakers talk under stress [Gupta and Mermelstein, 1982] [Junqua, 1993] [Hansen, 1994] [Hansen and Cairns, 1995]). Additional sources of variability are the type of speech (isolated or continuous, read or spontaneous, etc.), the style imposed by the task or the context in which speech is produced. The linguistic context may also modify the intonation, articulation, etc.

Speech variability due to production phenomena is difficult to model using standard signal processing techniques. A very successful approach is by training speaker-independent models; using data from a sufficiently representative portion of the target speaker population it is possible to train models that represent the common characteristics of the speaker distribution, while ignoring the particularities of individual speakers. As for speech style and contextual situation, variability may be tackled by means of task constraints. A typical example is a speech dialog system that drives the user towards a determined style and reduces the vocabulary size. Appropriate LMs also help modeling variability caused by the context of speech.

2.2.2 Acoustic Environment

Distortions caused by the acoustic environment often include additive noises[1], as well as reverberations and reflections imposed by the geometry of the recording room [Huang *et al.*, 2001; pp. 477-486]. The ideal recording environment is an isolated anechoic chamber where only direct waves reach the microphone. However, real applications require in many cases deployment of ASR systems in a large variety of acoustic environments, which compromises their accuracy.

2.2.3 Channel Effects

These are distortions introduced by the recording and transmission equipment and include many linear or non-linear convolutional distortions as a result of A/D conversion, bandwidth limitations, non-flat frequency responses of microphone and transmission components, application of digital codecs, etc. In addition, channel distortions may also present themselves in the form of additive noises (very typical for example in data conversions and changes of channel of transmission).

The focus of this Thesis is on channel distortions and in particular on speech signals for which part of the available spectrum is completely removed. Other types of distortions are also considered when they are imposed by real applications (additive noise in telephone data, for example).

For the sake of notation throughout the rest of this Thesis, we define here the following concepts: by *clean speech* we refer to completely undistorted speech, i.e., full-bandwidth and assuming no distortions. By *distorted speech* or *limited-bandwidth speech* we refer to speech that went through a band-limiting channel, although other types of distortion such as additive noises due to the environmental the transmission channel or computation residuals will also affect the signal.

2.2.4 Model of the Distorted Signal

Figure 2.3 presented a general diagram of distortions that may affect the original signal, $x(t)$. In the time domain the signal reaching the ASR system, $y(t)$ may be expressed as:

$$y(t) = \left(\left(x(t) + n_{room}(t) \right) * h_{room}(t) + n_{mic}(t) \right) * h_{chan}(t) + n_{trans}(t), \tag{2.4}$$

where $*$ is the operation of convolution. The previous equation may be simplified as:

$$y = \left(x * h_{room} \right) * h_{chan} + \left(n_{room} * h_{room} \right) * h_{chan} + n_{mic} * h_{chan} + n_{trans} = x * h + n, \tag{2.5}$$

where variable *t* has been dropped for simplicity of notation. The resulting equation defines a general expression for speech signals affected by convolutional and additive distortions, and is in accordance with the widely accepted formulation by Acero [1990].

In the spectral domain, the relationship between the power density spectra is:

$$|Y(f)|^2 = |X(f)|^2 \cdot |H(f)|^2 + |N(f)|^2, \tag{2.6}$$

where it is assumed that the cross-correlation between the filtered signal and additive noise is null. The convolution becomes a multiplicative term in the frequency domain.

[1] Sometimes the word noise is used as a synonym of distortion and in that context the term convolutional noise may be used to refer, for example to telephone channels. However, in this Thesis noise always refers to additive distortions.

2.3 Classification of Robust Methods for ASR

When the distortions in Eq. (2.6) introduce significant variations on the original signal a large mismatch occurs between acoustic models trained with undistorted data and incoming distorted speech. The goal of robustness techniques is to reduce the resulting impact on ASR accuracy. Depending on the stage of the recognition process in which robust techniques are applied, they can be classified into three categories:

- Robust feature extraction.
- Speech and feature enhancement.
- Acoustic model adaptation.

The limits between these categories are not always clear and some approaches are considered hybrid. Also, robust recognition systems normally combine several of these techniques at different levels to obtain optimal results. In the following sections we show the principles of some representative techniques in each group and finally refer to other methods that do not fit in any of the mentioned categories.

2.4 Robust Feature Extraction

The first stage in the speech recognition process is the extraction of a set of reliable features from the original speech signal. For ASR the goal is to extract the phonetic content, and leave out, as much as possible, the sources of variability affecting speech. Additionally, feature extraction allows for removal of part of the redundancy present in the time-representation of the speech signal that does not help in ASR and increases computational cost.

2.4.1 Representations Based on Models of Speech Production

Simple models of speech production consider the vocal tract as a set of concatenated tubes excited by the vibration of the vocal chords and/or the turbulences produced by a constriction of the vocal tract. The concatenated tubes may, in turn, be modeled as an N pole digital filter, where in theory N tubes limit the number of complex conjugate pole pairs to $N/2$ and different poles may be identified with resonances or formants created when part of the energy is reflected back and forth in tube boundaries ([Huang *et al.*, 2001; pp. 284-290]). The transfer function of the vocal tract is then represented as:

$$H(z) = \frac{1}{1 - \sum_{i=1}^{P} a_i z^{-i}} = \frac{1}{A(z)}, \tag{2.7}$$

where p is the number of poles in the model (in [Huang *et al.*, 2001; p. 290] it is stated that in practice it is sufficient to use F_s + 2 poles, where F_s is the sampling frequency in kHz). According to this model, a sample of speech can be predicted as a linear combination of previous samples. The error in such prediction corresponds mainly to the excitation of the vocal tract and can be expressed as:

$$e(t) = s(t) - \tilde{s}(t) = s(t) - \sum_{i=1}^{P} a_i s(t-i). \tag{2.8}$$

As the configuration of the vocal tract evolves in time, the poles a_i will take different values. Thus, it is possible to extract a signal representation as the succession of the values of the first P poles, for speech frames, known as Linear Predictive Coefficients (LPC) [Makhoul, 1975]. Other popular representations are derived from the LPCs, for example Line Spectral Frequencies [Itakura, 1975] and Reflection Coefficients.

2.4.2 Perceptually-motivated Representations

This family of parameterizations is based on a Short-Time Fourier Transform (STFT) of the speech signal (practical implementations normally use the Fast Fourier Transform (FFT), followed by a bank of filters in an attempt to mimic the function of the cochlea in the human ear. Since it is normally assumed that only the amplitude (and not the phase) of the acoustic signal is required for speech recognition, the squared magnitude of the FFT is typically taken before applying the filterbank. It is also common practice to distribute the filterbank in a non-linear scale in the frequency axis (typically mel-scale), because experiments on human listeners suggest that the ear's resolution is approximately linear until 1000 Hz, and logarithmic for higher frequencies [Stevens and Volkmann, 1940]. In Figure 2.4, a typical mel-scaled filterbank is shown, where center frequencies for each filter are computed as:

$$Mel(f) = 2595 \cdot \log_{10}\left(1 + \frac{f}{700}\right). \tag{2.9}$$

A widely used parameterization known as Perceptual Linear Predictive Coefficients (PLP) uses the outputs of the filterbanks to compute LPC coefficients [Hermansky, 1990]. However, in state-of-the-art systems it is more popular to apply a cepstral transformation [Bogert *et al.*, 1963] over mel-spaced frequency channels, with several advantages: firstly, the logarithm of the filterbank outputs emulates the non-linear response of the human ear to the amplitude of the acoustic signals. Additionally it has been observed that the first cepstral vectors capture most of the variability in the speech signal and so, it is possible to reduce feature vector dimensionality while keeping most of the information. Finally, correlation between different cepstral features is small, so the covariance matrixes in HMM models of cepstral representations of speech may be assumed diagonal with only a small ASR performance loss (while allowing a great reduction in computational costs) [Pols, 1977; pp. 42-52] [Quatieri, 2002; pp. 253-261]. Davis and Mermelstein [1980] established the superiority of the MFCC family of algorithms over other families like LPC and Reflection Coefficients. They also noted that MFCCs better represent the perceptually relevant aspects of the short-time speech spectrum and are particularly adequate for optimal representation of consonantal spectra.

For a given frame, the real cepstral transformation is obtained as:

$$C_i = \sqrt{\frac{2}{N}} \sum_{j=1}^{N} \log\left(\left|X_j\right|^2\right) \cos\left(\frac{\pi i}{N}\left(j - 0.5\right)\right), \tag{2.10}$$

where X_j represents the output of filter j in the bank of filters (Figure 2.4).

All experiments in this Thesis use MFCCs as the basic parameterization.

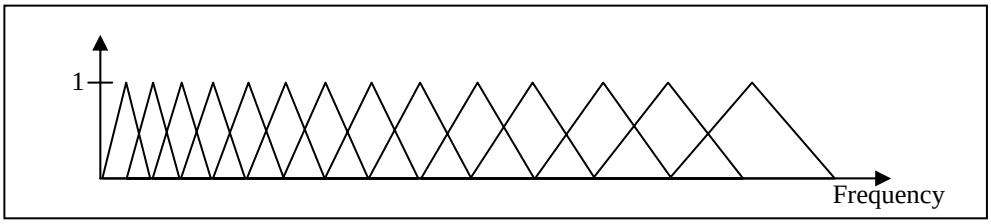


Figure 2.4 Mel-scale filterbank with equal maxima.

2.4.3 Temporal Evolution of Speech

Temporal changes in the spectra of speech play an important role in human perception, but HMM models assume each frame is independent of past observations. A solution to overcome this limitation of HMMs is to explicitly introduce the temporal evolution of the speech signal at the feature level. Thus, feature extractors for ASR typically append delta and delta-delta coefficients to the static coefficients (the term delta refers to the first derivative, or difference between speech frames) [Furui, 1986]. Another possibility is to use a parameterization that implicitly represents evolution such as Relative Spectral (RASTA) representations that include information related to signal variations that change at the typical rates of speech, while filtering out others that vary more slowly or quickly [Hermansky and Morgan, 1994].

2.4.4 Other Techniques for Feature Extraction

It is an accepted fact that a fixed-length signal windowing is not optimal, as it does not represent the non-uniform distribution of information in speech signals [Gallardo-Antolín, 2002; pp. 81-85]. Thus, new feature extraction approaches allowing for variable length windowing and wavelet transformations have been designed in an attempt to properly capture signal variations [Hermansky and Sharma, 1999].

Other authors have noted the independence of a typical feature extractor module from the actual task of pattern classification. In [Li and Stern, 2004] a linear transformation of log-spectral features was proposed in order to maximize the acoustic likelihood of the most likely state sequence.

While some of these novel techniques have produced encouraging results they incur in significant complexity increase and their usability in real applications is still unclear.

2.5 Speech and Feature Enhancement

We group in this category a loose range of techniques that introduce additional processing in the standard speech analysis and feature extraction modules. With the goal of increasing the quality of the speech signal (from the perspective of improving ASR), enhancement techniques may be implemented at different levels in a speech recognition engine. For example, enhancement prior to feature extraction is done in echo cancellation, for removing echoes that appear in telephone-like applications [Huang *et al.*, 2001; pp. 497-504]. Also, blind source separation provides significant robustness when the signal from the target speaker is mixed with other voices [Jutten and Herault, 1991] [Comon, 1994]. Enhancement may also be applied embedded in the feature extractor module as is done with Wiener filtering, spectral subtraction [Boll, 1979], CMN, histogram equalization, feature compensation (with

Gaussian Models or Neural Networks), etc. The following sections describe the techniques relevant to our work for their use in our experiments or because they are related to our proposed solutions.

2.5.1 Cepstral Mean Normalization

CMN is a technique for removing constant convolutive distortions in the speech signal [Atal, 1974]. The combination of simple implementation and significant accuracy increase at a very low computational cost explains its extensive use in ASR applications.

Working in a homomorphic feature vector space for the operation of convolution (for example MFCCs [Quatieri, 2002; pp. 253-261]), feature vectors are modified by a convolutional filter as:

$$\mathbf{y}_t = \mathbf{x}_t + \mathbf{h}\,. \tag{2.11}$$

The average of the signal for a period of time *T* is:

$$\overline{\mathbf{y}} = \frac{1}{T}\sum_{t=1}^{T}\mathbf{y}_t = \frac{1}{T}\sum_{t=1}^{T}\left(\mathbf{x}_t + \mathbf{h}\right) = \overline{\mathbf{x}} + \mathbf{h}\,, \tag{2.12}$$

where it is assumed that **h** is constant. The CMN version of **y** is computed by subtracting the time average from the observation in each individual frame:

$$\mathbf{y}_t^{CMN} = \mathbf{y}_t - \overline{\mathbf{y}} = \mathbf{x}_t + \mathbf{h} - \left(\overline{\mathbf{x}} + \mathbf{h}\right) = \mathbf{x}_t - \overline{\mathbf{x}} = \mathbf{x}_t^{CMN}\,. \tag{2.13}$$

From Eq. (2.13) CMN is shown to be immune to stationary convolutional distortions (e.g. when different types of microphone are used). As a result, two signals obtained by passing a common original signal through different stationary convolutional filters (microphones, for instance) will be identical after CMN is applied. There is however one limitation to its use in ASR and it is that the average contribution of speech in the fragment of an utterance over which CMN is computed should be near zero, so that only the contribution of the convolutional distortion is removed from each sample. On the contrary, if the mean is computed with only a few frames from a segment of an utterance with a stable spectral distribution, removal of the average of the signal would result in bringing all observations to almost null values, so most of the speech information would be removed along with the convolutional distortion. It has been shown empirically that CMN has not a negative impact on ASR provided that utterances are longer than 2-4 seconds [Huang *et al.*, 2001; pp. 522-523].

2.5.2 Feature Compensation with Gaussian Mixture Models

A drawback of many enhancement algorithms is that they assume uncorrelation between different regions of the spectrum (or between elements of a feature vector) and treat them independently. Independent transformations of speech characteristics may result in incongruence between different parts of the feature vectors that represent the speech signal, which would lead to an unpredictable (and error-full) behavior of the decoder module.

In this section we introduce a family of algorithms that classify speech frames into clusters and modify their features using appropriate transformations. Although, Gaussian Mixture Models could be used for compensation of different types of features, we derive their formulas for the case of our target parameterization, MFCCs.

Assuming the model of environmental and channel distortions Eq. (2.6) and a feature extractor module based on a bank of filters, the output of each filter takes the form:

$$\left|Y\left(f_j\right)\right|^2 = \left|X\left(f_j\right)\right|^2 \cdot \left|H\left(f_j\right)\right|^2 + \left|N\left(f_j\right)\right|^2, \tag{2.14}$$

where f_j is the center frequency of filter *j* in the bank of filters. Now, taking logarithms:

$$\ln\left|Y\left(f_j\right)\right|^2 = \ln\left|X\left(f_j\right)\right|^2 + \ln\left|H\left(f_j\right)\right|^2 + \ln\left(1+\exp\left(\ln\left|N\left(f_j\right)\right|^2 - \ln\left|X\left(f_j\right)\right|^2 - \ln\left|H\left(f_j\right)\right|^2\right)\right). \tag{2.15}$$

Without loss of generality, we may apply an additional transformation **C** to the feature vectors such that, for example a vector of undistorted speech would be:

$$\mathbf{x} = \mathbf{C}\left(\ln\left|X\left(f_0\right)\right|^2, \quad \ln\left|X\left(f_1\right)\right|^2, \quad \ldots, \quad \ln\left|X\left(f_M\right)\right|^2\right). \tag{2.16}$$

Similarly, the feature vector of distorted speech is:

$$\mathbf{y} = \mathbf{C}\left(\ln\left|Y\left(f_0\right)\right|^2, \quad \ln\left|Y\left(f_1\right)\right|^2, \quad \ldots, \quad \ln\left|Y\left(f_M\right)\right|^2\right), \tag{2.17}$$

and similar equations may be stated for the convolutive filter **h** and noise **n**. When the target features are MFCCs, matrix **C** in Eqs. (2.16) and (2.17) corresponds to the Discrete Cosine Transform, while for Mel Frequency Energy features (MFE), it would simply be the identity matrix **I**. Inserting these equations into Eq. (2.15) we can express the undistorted feature vectors in terms of the other components:

$$\mathbf{x} = \mathbf{y} - \mathbf{h} - \mathbf{C}\ln\left(1 - e^{\mathbf{C}^{-1}(\mathbf{n}-\mathbf{y})}\right). \tag{2.18}$$

The Minimum Mean Squared Error estimation of the clean signal is:

$$\mathbf{x}^{MMSE} = \mathbf{y} - \mathbf{h} - \mathbf{C}E\left\{\ln\left(1 - e^{\mathbf{C}^{-1}(\mathbf{n}-\mathbf{y})}\right)\right\}. \tag{2.19}$$

When enough data is available from both the undistorted and distorted spaces it is possible to model their relation directly from Eq. (2.19); however, the non-linearity in this equation complicates an explicit modeling.

As shown in [Huang *et al.*, 2001; p. 526] a popular approach is to model to probability distributions of undistorted and distorted features as mixtures of Gaussians. For example, for distorted features:

$$p\left(\mathbf{y}\right) = \sum_{k=1}^{K} p\left(\mathbf{y}\middle|k\right) \cdot P\left(k\right) = \sum_{k=1}^{K} N\left(\mathbf{y};\boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k\right) \cdot P\left(k\right), \tag{2.20}$$

where $P\left(k\right)$ is the a-priori probability of mixture k in the GMM. If **x** and **y** are assumed jointly Gaussian within a cluster of data pairs k, the conditional expectation of clean feature vectors given the distorted vectors and the cluster is, then[2] (see [Buera *et al.*, 2007] for a slightly different approach):

$$E\left\{\mathbf{x}\middle|\mathbf{y},k\right\} = \boldsymbol{\mu}_{X,k} + \boldsymbol{\Sigma}_{XY,k}\left(\boldsymbol{\Sigma}_{Y,k}\right)^{-1}\left(\mathbf{y} - \boldsymbol{\mu}_{Y,k}\right) = \mathbf{B}_k\mathbf{y} + \mathbf{b}_k, \tag{2.21}$$

[2] Note that in Eq. (2.21) *X* and *Y* represent multivariate distributions of clean data and distorted data respectively, while **x** and **y** represent samples of such distributions.

and the joint pdf is:

$$p(\mathbf{x},\mathbf{y}) = \sum_{k=1}^{K} p(\mathbf{x},\mathbf{y}|k)\cdot P(k) = \sum_{k=1}^{K} p(\mathbf{x}|\mathbf{y},k)\, p(\mathbf{y}|k)\cdot P(k) =$$

$$= \sum_{k=1}^{K} N(\mathbf{x};\mathbf{B}_k\mathbf{y}+\mathbf{b}_k,\Gamma_k)\cdot N(\mathbf{y};\boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k)\cdot P(k), \qquad (2.22)$$

where Γ_k is the covariance matrix of the conditional probability of $\mathbf{x}$ and $\mathbf{y}$, and class *k***.**

Estimation of undistorted features from distorted features may be obtained using two different criteria: Maximum Likelihood (ML) and Minimum Mean Squared Error (MMSE).

ML estimation of full-bandwidth features consists of finding the value **x** that maximizes the joint probability in Eq. (2.22). Thus:

$$\mathbf{x}^{ML} = \arg\max_{\mathbf{x}} \{p(\mathbf{x},\mathbf{y})\} = \arg\max_{\mathbf{x}} \left\{ \sum_{k=1}^{K} N(\mathbf{x};\boldsymbol{B}_k\mathbf{y}+\mathbf{b}_k,\Gamma_k)\cdot N(\mathbf{y};\boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k)\cdot P(k) \right\}. \qquad (2.23)$$

The exact computation of the maximum for each sample is possible but time-consuming. However, when the Gaussians for different values of *k* do not present too much overlapping, a close approximation to the maximum in Eq. (2.23) will be obtained by maximizing for the best possible *k*. As $P(k)$ and $N(\mathbf{y};\boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k)$ are independent of **x**, the maximum probability is obtained maximizing term $N(\mathbf{x};\boldsymbol{B}_k\mathbf{y}+\mathbf{b}_k,\Gamma_k)$, and thus,

$$\mathbf{x}^{ML}(\mathbf{y}) = \mathbf{B}_q\mathbf{y}+\mathbf{b}_q, \qquad (2.24)$$

where *q* is the class for which:

$$q = \arg\max_{k} \{N(\mathbf{x};\boldsymbol{B}_k\mathbf{y}+\mathbf{b}_k,\Gamma_k)\cdot N(\mathbf{y};\boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k)\cdot P(k)\} =$$

$$= \arg\max_{k} \left\{ \frac{1}{\sqrt{(2\pi)^n\cdot|\Gamma_k|}}\cdot N(\mathbf{y};\boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k)\cdot P(k) \right\}, \qquad (2.25)$$

where we used: $N(\mathbf{x}^{ML}=\boldsymbol{B}_q\mathbf{y}+\mathbf{b}_q;\boldsymbol{B}_q\mathbf{y}+\mathbf{b}_q,\Gamma_q) = \frac{1}{\sqrt{(2\pi)^n\cdot|\Gamma_q|}}\cdot\exp\left(-\frac{1}{2}\cdot(\boldsymbol{0})^T\cdot\Gamma_q^{-1}\cdot(\boldsymbol{0})\right)$, for the class that maximizes the probability, and so the exponential is equal to 1.

When the MMSE criterion is used instead, the goal is to find the function mapping the distribution of limited bandwidth data to full bandwidth data that minimizes the squared error. In [Huang *et al.*, 2001; pp.102-104] it is shown that this function is given by the conditional expectation. Therefore, for a particular observation **y**:

$$\mathbf{x}^{MMSE} = E\{\mathbf{x}|\mathbf{y}\} = \sum_{k=1}^{K} P(k|\mathbf{y})\cdot E\{\mathbf{x}|\mathbf{y},k\} = \sum_{k=1}^{K} P(k|\mathbf{y})\cdot(\mathbf{B}_k\mathbf{y}+\mathbf{b}_k). \qquad (2.26)$$

This is similar to the ML expression in Eq. (2.24), but here the compensation is a combination of contributions from all classes that partition the feature space, instead of only that for the most likely class.

Feature compensation with GMMs has been previously used for speech recognition of noisy speech. Different implementations use a variety of methods for estimating GMMs and the transformation matrix $\mathbf{B}_k$ and offset vector $\mathbf{b}_k$ for each Gaussian mixture (normally computed using stereo-data mapping techniques, or mapping of data distributions). In RATZ, it was proposed to compute different corrector functions for each state in the set of acoustic models of an ASR engine [Moreno, 1996]. In SPLICE the feature space was partitioned using a set of GMMs trained independently from the acoustic models. Thus, based on the amount of adaptation data and the complexity of the distortion it is possible to modify the number of partitioning classes for optimal performance [Droppo *et al.*, 2001]. Buera *et al.* [2007] propose modeling the feature space as GMMs for both the clean and distorted spaces, and associating Gaussians from one and the other space prior to compensation. Also, in [Afify *et al.*, 2007], clean and distorted features are concatenated prior to feature space partitioning. In this way, it is expected to find clusters of data in the undistorted space that remain close in the distorted space and may therefore be compensated using the same corrector functions. A similar approach to SPLICE is followed in [Raj *et al.*, 2004], but here the target speech representation are spectral features, thus removing the extra complication of the cosine transformation and allowing for straightforward exploitation of the redundancy of different spectral regions in the reconstruction process.

As an important part of the algorithms proposed for feature compensation of band-limited speech are inspired in these methods for noise robustness the topic of Gaussian-based compensation is later extended in our work.

2.6 Model-side Robustness

In this section we make a brief summary on model-side robustness with emphasis on techniques used for comparison with our proposed feature-side algorithms or closely related with them. In general, model-side methods require more training or adaptation data than feature-side solutions, but also may bring better accuracy.

2.6.1 Model Retraining

When large amounts of training data from the target distortion are available an obvious option is to train new models. Also, when recording sufficient amounts of data is not practical but the transfer function of the distortion in the target environment is known, new data may be generated by applying the transfer function to clean data.

Model retraining is the most resource and time consuming option; new models need to be trained and generally they need to be fine-tuned. Also, for applications where speech may come from different sources, each with their own distortions, multiple recognizer systems need to be run in parallel, so the computational cost is multiplied several times. However, model retraining is also in many cases the option that provides the best possible performance and in our experiments we use it to set an upper-bound on ASR accuracy.

2.6.2 Multi-style Training

Sometimes speech variability is constrained and enough data is available for a general representation of speech under the different conditions that may affect speech in a particular application. In these cases, it is possible to obtain robust models with the usual training (Baum-Welch [Rabiner, 1989]), by pooling together all data from the different sources in order to obtain robust models. A typical application is training of speaker-independent models with data from multiple speakers, but the same principle may be used for training models robust to a variety of environmental and channel conditions. As the characteristics of distortions grow more dissimilar the variability in the signal due to channel characteristics grows larger and ASR will benefit from an increase in the number of mixtures per model, so as to obtain more accurate modeling of all the speech variability.

2.6.3 Acoustic Model Adaptation

A less resource-demanding solution is to modify models trained with clean data and adapt them to the target condition. Different methods exist whose relative performance depends on the particular task constraints (memory requirements, level of variability and required speed) and the amount of adaptation data available. Adaptation techniques are generally divided into two categories [Gales, 1997]: linear and non-linear transformations. Non-linear transformations require storage of complete adapted HMM sets and need more adaptation data than linear transformations, but may outperform these. On the contrary, linear transformations allow the storage of the transformation matrixes only and are capable of impressive performance with limited amounts of data). Linear transformations may be performed over the mean vectors of Gaussian mixtures, their covariance matrixes or both. When the transformation matrixes are the same for means and covariances we talk of constrained model adaptation, while if they are independent we call it unconstrained adaptation.

2.6.3.1 Constrained Linear Adaptation

Constrained transformations of models have the peculiarity that the same matrix is used for transformation of the vector of means and covariance matrix. The general form of Constrained Linear Adaptation (CLA) [Digalakis *et al.*, 1995] is:

$$\boldsymbol{\mu}_k^A = \mathbf{A}_k \cdot \boldsymbol{\mu}_k^C - \mathbf{b}_k \,, \tag{2.27}$$

$$\boldsymbol{\Sigma}_k^A = \mathbf{A}_k \cdot \boldsymbol{\Sigma}_k^C \cdot \left(\mathbf{A}_k\right)^T, \tag{2.28}$$

where $\boldsymbol{\mu}_k^C$ and $\boldsymbol{\Sigma}_k^C$ are respectively, the vector of means and matrix of covariance for a particular Gaussian mixture *k* that belongs to a particular state in the HMM trained with clean data. The means vector and covariance matrix adapted to the new condition are $\boldsymbol{\mu}_k^A$ and $\boldsymbol{\Sigma}_k^A$. It is common practice to group Gaussian mixtures in clusters, each of which will follow a different transformation (in this way it is possible to adapt mixtures for which no adaptation data is available).

Linear adaptation methods are trained using an EM approach, where the models are modified iteratively so as to maximize the probability of emission of the new data distribution.

An interesting property of CLA is that it may be performed almost identically in the model or feature sides. In order to show this, we insert Eqs. (2.27) and (2.28) into Eq. (2.3). The probability of an observation with adapted models for a Gaussian mixture *k* is then:

$$N\left(\mathbf{o};\boldsymbol{\mu}^{A},\boldsymbol{\Sigma}^{A}\right)=\frac{1}{\sqrt{\left(2\pi\right)^{n}\left|\mathbf{A}\cdot\boldsymbol{\Sigma}^{C}\cdot\mathbf{A}^{T}\right|}}\cdot \\ \cdot\exp\left(-\frac{1}{2}\cdot\left(\mathbf{o}-\mathbf{A}\cdot\boldsymbol{\mu}^{C}+\mathbf{b}\right)^{T}\cdot\left(\mathbf{A}\cdot\boldsymbol{\Sigma}^{C}\cdot\mathbf{A}^{T}\right)^{-1}\cdot\left(\mathbf{o}-\mathbf{A}\cdot\boldsymbol{\mu}^{C}+\mathbf{b}\right)\right), \quad (2.29)$$

where superindex *k* has been dropped for simplicity (but it should be noted that Eq. (2.29) is the likelihood for an observation and a Gaussian mixture *k*). This may be re-arranged using a few standard properties of matrix operations [Spiegel and Abellanas, 1998; pp. 112-113][3]:

$$N\left(\mathbf{o};\boldsymbol{\mu}^{A},\boldsymbol{\Sigma}^{A}\right)=\frac{1}{\left|\mathbf{A}\right|}\cdot\frac{1}{\sqrt{\left(2\pi\right)^{n}\left|\boldsymbol{\Sigma}^{C}\right|}}\cdot \\ \cdot\exp\left(-\frac{1}{2}\cdot\left(\mathbf{A}^{-1}\mathbf{o}-\boldsymbol{\mu}^{C}+\mathbf{A}^{-1}\mathbf{b}\right)^{T}\cdot\boldsymbol{\Sigma}_{C}^{-1}\cdot\left(\mathbf{A}^{-1}\mathbf{o}-\boldsymbol{\mu}^{C}+\mathbf{A}^{-1}\mathbf{b}\right)\right), \quad (2.30)$$

and so:

$$N\left(\mathbf{o};\boldsymbol{\mu}^{A},\boldsymbol{\Sigma}^{A}\right)=\frac{1}{\left|\mathbf{A}\right|}\cdot N\left(\mathbf{A}^{-1}\cdot\mathbf{o}+\mathbf{A}^{-1}\cdot\mathbf{b};\boldsymbol{\mu}^{C},\boldsymbol{\Sigma}^{C}\right)=\left|\mathbf{A}^{*}\right|\cdot N\left(\mathbf{A}^{*}\cdot\mathbf{o}+\mathbf{b}^{*};\boldsymbol{\mu}^{C},\boldsymbol{\Sigma}^{C}\right), \quad (2.31)$$

where we defined $\mathbf{A}^{-1}=\mathbf{A}^{*}$ and $\mathbf{A}^{-1}\cdot\mathbf{b}=\mathbf{b}^{*}$. Therefore, $\mathbf{A}^{*}$ and $\mathbf{b}^{*}$ may be identified with the corrector coefficients $\mathbf{B}_{k}$ and offset $\mathbf{b}_{k}$ in Eq. (2.21) (remember that we dropped index *k* here) and so CLA is identical to feature compensation except for the term $\left|\mathbf{A}^{*}\right|$ in Eq. (2.31).

2.6.3.2 MLLR and MAP Adaptation

MLLR [Leggetter and Woodland, 1995] and MAP [Gauvain and Lee, 1994] are probably the two most widely extended adaptation techniques (MLLR performs linear adaptation and MAP non-linear adaptation). Both of them are employed in this Thesis for comparison with the proposed algorithms and their description as well as extensive experiments on performance is deferred to Chapter 6.

2.6.4 Other Model-side Robustness Techniques

A number of other approaches exist for model-side adaptation. Parallel Model Combination (PMC) estimates Gaussian models of clean speech and noise independently and combines them in order to obtain models of noisy speech [Gales, 1995].

Another popular approach is Vector Taylor Series (VTS). Here, the non-linearity in Eq. (2.19) that relates clean and distorted features is approximated by means of a polynomial vector Taylor series and is used to estimate new values of model parameters for distorted speech from those trained with clean speech [Moreno *et al.*, 1996].

[3] The following matrix properties were used:
$\left(\mathbf{A}^{T}\right)^{-1}=\left(\mathbf{A}^{-1}\right)^{T}$, $\left(\mathbf{A}\cdot\mathbf{B}\right)^{-1}=\mathbf{B}^{-1}\cdot\mathbf{A}^{-1}$, $\left(\mathbf{A}\cdot\mathbf{B}\right)^{T}=\mathbf{B}^{T}\cdot\mathbf{A}^{T}$, $\left|\mathbf{A}\cdot\mathbf{B}\right|=\left|\mathbf{B}\right|\cdot\left|\mathbf{A}\right|$, $\left|\mathbf{A}^{T}\right|=\left|\mathbf{A}\right|$.

2.7 Hybrid Methods

There also exist solutions that employ feature compensation-like approaches embedded in the decoder module. In [Cooke *et al.*, 2001] two methods called marginalization and data imputation are proposed for feature compensation over spectral features. In the first one decoding is made with only those features that are classified as reliable. In the second one, unreliable components are estimated by MAP or MMSE from the reliable components at the level of HMM states. This is similar to the feature compensation method proposed in [Raj *et al.*, 2004]; however, in the latter, estimation is made at the level of Gaussian classes trained independently from the decoder module and therefore compensation may be made independently from this module.

SPLICE with uncertainty decoding makes use of SPLICE feature mapping and additionally modifies the pattern matching criterion in the decoder module based on the degree of certainty in the feature compensation [Droppo *et al.*, 2002] [Deng *et al.*, 2004]. In practice this approach results in typical SPLICE feature compensation, plus the introduction into the decoder module of a term that describes the uncertainty of the compensation. This term is added to the covariance matrixes of acoustic models prior to likelihood computation, so that those MFCC features for which compensation was unclear will see an important increase in the width of their Gaussian distribution, or equivalently will be less relevant for likelihood computation.

Another option for hybrid robustness is feature compensation and retraining of acoustic models (or model adaptation) using compensated features. In this way, new models become resistant to the possible artificial distortions and defects introduced in the process of data enhancement.

2.8 Other Robustness Techniques

There are other robustness techniques that do not fall strictly into any of the 3 categories previously described – feature extraction, speech enhancement and model-side robustness. Among them, and for their significance in our work we mention a successful method for combining transcription outputs from multiple recognizers ROVER [Fiscus, 1997] and a method for reducing the mismatch between features and models, particularly when non-stationary noises corrupt the original signal by adding a well-behaved masking noise to both training and test data sets [van Compernolle, 1987] [Claes and van Compernolle, 1996] [Morales *et al.*, 2007c].

Another promising solution for robustness to environmental and channel distortions, but out of the scope of this Thesis is audio-visual speech recognition (see for example [Potamianos *et al.*, 2003] for a review on state-of-the-art systems and [Hennecke *et al.*, 1996] for an introduction to feature extraction and fusion techniques).

3

Band-limiting Distortions

For ASR, full-bandwidth speech is normally considered to span the range of frequencies between 0 Hz and 8 kHz. This assumption is based both on experiments on human's perception and on ASR performance comparison between systems with different available bandwidths. A study in [ITU, 1993] shows clearly user's preference for signals with a frequency range 0.05-7 kHz, compared to signals with the typical telephone channel bandwidth. Also, for the goal of ASR, systems using a complete spectrum typically outperform those with band-limited data (as shown in the experimental section in Chapter 8). For these reasons, whenever possible, acoustic models should be trained with data spanning the frequency range 0-8 kHz. However, it may be the case that the target speech utterances for a particular application present some type of band-limitation, typically imposed by the transmitting channel, recording conditions, etc. In these cases, feature vectors from band-limited data will differ significantly from those used to train the system and the mismatch will cause an important degradation in recognition accuracy. Experiments on human listeners show that the redundancy of speech across different frequency bands allows understandability even for very narrow band-pass filters. Between the 30s and 50s, Fletcher studied understandability of meaningful speech sounds (intelligibility) and nonsense sounds (articulation), under different distortions typically motivated by telephone distortions in The Bell Telephone Laboratories[1]. He showed that understandability is almost unaffected when speech is low-pass filtered or high-pass filtered with a cut-off frequency of 1500 Hz. In [Warren *et al.*, 1995] it was shown that human understandability of sentences heard through spectral slits of 1/3 octave width and steep slopes is over 90% for bands centered in the region 1100-2100 Hz, showing that the most relevant speech information is contained in that spectral region. Another interesting result is that slits of 1/3 octave centered respectively at 6 kHz and 370 Hz allowed less than 25% understandability each, while when combined they produced 77% understandability. This seems to indicate that human listeners perform multiple sub-band-like recognition processes and combine the information from each of these for improved performance.

[1] A very interesting review on Fletcher's work may be found in [Allen, 1994].

All these observations indicate that removal of parts of the spectrum is not critical for understandability of speech. However, ASR systems are very sensitive to this kind of mismatch and very important degradation may occur even when the removed spectral regions are small. In this chapter we start with examples of practical situations where band-limitations affect speech signals. Then we present a summary of the spectral characteristics of phonetic units in English, which determines how they will be affected by band-limitations. We present also a brief review of related work on bandwidth extension and a mathematical model on the impact of these distortions in the speech signal. Finally, we discuss on how band-limitations affect the degree of correlation of the speech signal parameterization employed in this Thesis, the Mel Frequency Cepstral Coefficients (MFCC) and how this affects our compensation strategy.

3.1 Examples of Band-limiting Environments

There are many real situations where the environment, transmitting channel or audio equipment may impose band-limiting distortions on speech. A few examples are:

- Signals transmitted through the telephone line are band-pass filtered (typical telephone ranges are 300 Hz - 3400 Hz). The channel conditions vary from one call to another, making the problem even more difficult to solve [Jankowski, 1991] [Yasukawa, 1996] [Moreno and Stern, 1994] [ITU-T, 2001].
- Speech signals may be sampled at low frequencies due to limitations of the recording devices employed, for example in historical recordings [NGSW] [Hansen *et al.*, 2004] [Kim and Hansen, 2006] or with the goal of saving memory space or reducing CPU usage, especially in small portable devices like PDAs or cellular phones [Pearce, 2000] [Moyal, 2005]. In many cases, band-limited signals may be mixed with high-quality recordings, for example in documentaries, broadcast news, etc.; a situation that further complicates ASR. In Figure 3.1 we show one such case, where a historical recording is followed by the voice of the anchor that spans the entire spectrum up to 8 kHz.

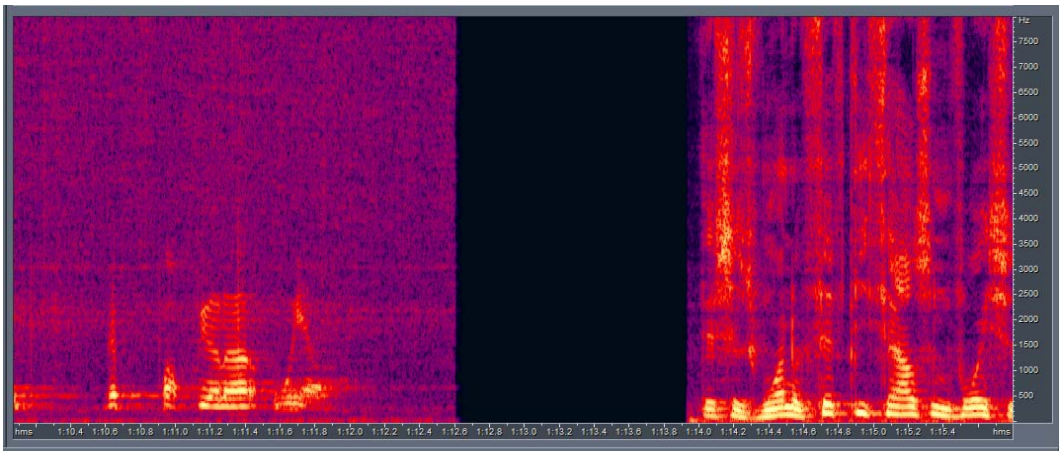


Figure 3.1 An example of a file from the NGSW collection with a historical recording from 1912, where the available bandwidth is below 2500 Hz (a speech on September 22nd by presidential candidate to the US government of the Progressive Party, Theodore Roosevelt). This historical recording is followed by the voice of the anchor that spans the entire spectrum up to 8 kHz.

- Speech systems installed on-board vehicles such as those in cars, airplanes, etc., may transmit in limited frequency ranges [Abut *et al.*, 2004] [Denenberg *et al.*, 1993].
- For the goal of human understandability it could be useful to remove spectral parts of speech highly contaminated by noise, as understandability is still high for narrow band-pass filters [Allen, 1994] [Warren *et al.*, 1995]. However, if the modified speech is used for ASR it needs to be repaired, or otherwise compensated.

In addition to these, other related situations are:

- Digital *codecs* may adapt the quality of compression to the needs of the communication channel [Lilly and Paliwal, 1996] [Euler and Zinke, 1994]. Thus, different distortions affect the original signal in a dynamic process and degrade recognition systems' performance. Robustness approaches for GSM-coded data may be found, for example, in [Huerta, 2000] and [Gallardo-Antolín *et al.*, 2005].
- When the signal is transmitted under aggressive environmental conditions it might be affected in a number of ways; for example, transmission through the atmosphere or underwater may cause attenuation of particular regions of the spectrum [Ott, 1977].

It should be noted that the last two examples normally imply very complex distortions, for which specific robustness solutions need to be applied and are out of the scope of this Thesis.

A typical solution to band-limited speech is retraining new models, as is normally done in telephone applications, for which large amounts of data are usually available. However, in some cases, training new models is not the best option. A few examples are the following:

- Sometimes it is difficult to collect enough data from the new environment to properly train new models. As shown in Chapter 8, for limited amounts of training data, feature compensation could be better than model-side retraining or model adaptation [Morales *et al.*, 2007b].
- In other cases, the number of possible distortions affecting speech is large and it is not practical to train separate recognition systems for each condition [Morales *et al.*, 2007a].
- Additionally, even when the number of possible distortions is fixed and relatively small (for example the range of possible sampling frequencies is normally reduced to fixed numbers: 8 kHz, 16 kHz, 22 kHz, etc.), the limited memory and CPU resources available for certain devices such as portable systems make desirable a small footprint recognizer system, and so, model adaptation or feature compensation would be preferred over using multiple acoustic model sets.

3.2 Spectral Characteristics of Phonemes in English

In this Thesis we propose general methods to compensate band-limiting channels irrespective of the actual cut-off frequencies and phonetic content of speech. In doing so, we obtain global solutions that are simple to implement. However, an interesting extension to this work would be a study on the effects of different channel limitations on particular phonemes, and specific solutions for them.

Although this task is out of the scope of this Thesis, we present here a summary on the spectral characteristics of phonemes and discuss in general terms what frequency ranges are more necessary for ASR [Junqua and Haton, 1996; pp. 10-20] [Rogers, 2000] [Huang *et al.*, 2001; pp. 36-51].

Vowels are characterized by the lack of a significant constriction of the air flow in the vocal tract. Different sounds are created by modifying the position of the articulators (primarily the oral cavity), which change the resonances of the vocal tract and are typically manifested in the apparition of a few formants. Many authors consider the first two formants discriminative enough for vowel characterization and although their spectral outcome depends on the complete vocal tract, F1 is normally related to the length of the pharyngeal tube (up to the constriction in the vocal tract) and F2 to the oral cavity. In English the first two formants of any vowel fall between 300 and 2300 Hz, and the third one remains below 3000 Hz. Therefore, we should expect vowels to be relatively resistant to low-pass filters with cut-off frequencies over 3 kHz.

Consonants are characterized by a constriction of the air flow at some point in the vocal tract. Generally they are divided into two main categories based on whether there is vibration of the vocal cords (voiced) or not (voiceless or unvoiced) and several subcategories according to the positioning of the main obstruction and manner of articulation. Their frequency distribution is more variable than for vowels, but the following characteristcs are typically accepted.

Stops are transient and dynamic sounds. They start with an obstruction phase in which no energy is observed (voiceless stops: /p/, /t/, /k/), or there is only a little low-frequency energy due to the vibration of the vocal cords (voiced stops: /b/, /d/, /g/). They continue with a burst of energy where the spectrum can reach high frequencies and finish with a friction noise and a period of aspiration. These sounds are highly influenced by the vowel that follows and their spectra are variable.

Fricatives are characterized by a narrow but not complete obstruction of the vocal tract. Phonemes /s/, /sh/, /z/ and /zh/ are special because they present high energy in the higher frequencies up to 8 kHz. Phonemes /f/, /v/ and /th/ have most of their energy in intermediate frequencies and at least some energy in high frequencies, while /dh/ keeps most of its energy in the lower frequencies. Due to the presence of significant energy in the higher frequencies, fricatives should be the phonemes more drastically affected by low-pass filters and low sampling rates.

Nasals are created by the excitation of the nasal cavity, while the oral cavity is completely closed at some point. Coupling of the oral and nasal cavities produces characteristic resonances and antiresonances, where the first formant is typically around 500 Hz, and the second formant is around 2500-3000 Hz.

Semivowels present only a partial constriction of the vocal tract. *Liquids* /l/ and /r/ are quite vowel-like and present well-defined formants. For example, the first formants for clear /l/ (as in “lake”) are typically located around the following frequencies: F1: 300 Hz, F2: 1600 Hz and F3: 2500 Hz, while typical formants for dark /l/ (as in “milk”) are: F1: 400 Hz, F2: 600 Hz and F3: 2500 Hz. *Glides* /w/ and /y/ are more difficult to characterize.

Africates are a combination of a stop and a fricative. In English two such sounds exist: /ch/ and /jh/.

From the previous discussion, it may be concluded that apart from some fricatives, and possibly the burst in stops, most of the speech content is above 250 Hz and below 4000 Hz. In the light of this observation, the standard telephone channel seems to be a reasonable choice (typically a pass-band filter with cut-off frequencies 300 Hz and 3400 Hz), as most of the speech content is preserved. From an evolutionary point of view [Yilmaz, 1967], human's speech ability seems to be avoiding the insertion of discriminative information in the low-frequencies, where it could interfere with many stationary noises in nature. It is also no surprise that the human ear presents a larger spectral resolution in the low-frequencies, precisely where most of the speech information is contained [Stevens and Volkmann, 1940].

Finally, for the purpose of signal restoration, we expect to be able to reconstruct most of the information, provided that the removed spectrum in the region 300-4000 Hz is not too large, for two reasons: a) the mel-scale used for filter positioning assures that most of the channels (below 4000 Hz) remain untouched, and b) the information contained below 4000 Hz should allow in most cases for identification and approximate reconstruction of the complete spectrum. In the case of low-pass filters, plosives and fricatives should be the phonemes most affected.

3.3 Related Work

Bandwidth extension has been widely studied with the purpose of reconstructing full-bandwidth signals that sound more natural. Only recently has this problem been specifically addressed for the task of ASR.

3.3.1 Bandwidth Extension for Subjective Quality Improvement

This type of studies has been motivated by the goal of building full-spectrum signals from telephone channel-transmitted speech. Enhancement techniques usually employ LPC-based analysis of the signal to create an extended spectral envelope from the restricted-bandwidth version and produce an excitation signal for that envelope [Cheng *et al.*, 1994] [Yasukawa, 1996] [Chennoukh *et al.*, 2001] [Jax and Vary, 2003] [Qian and Kabal, 2004]. These techniques require an important effort in order to obtain reconstructed signals that sound natural, because the human ear is very sensitive to artifacts in bandwidth-extended speech.

In principle, such signal restoration techniques could also be used for robust ASR. However, the effort needed to reconstruct the complete and naturally sounding time-signal is unnecessary for speech recognition. For example, restoration of the phase is critical for subjective quality, but the typical parametric representations of speech for ASR (MFCCs, PLP, etc.) ignore the phase. Thus, in our case, this effort is not justified (it should be noted however, that recent studies on human listener's [Liu *et al.*, 1997] [Shi *et al.*, 2006] as well as on ASR systems [Schlüter and Ney, 2001] have shown that phase information may have an important role in understandability, especially for non-stationary parts of the signal like plosives, and when the SNR is low).

3.3.2 Bandwidth Extension for Automatic Speech Recognition

When full-bandwidth acoustic models are used with band-limited speech the mismatch degrades system's performance. With the goal of ASR robustness, in this Thesis we propose to perform

bandwidth extension in the space of features, where the task should be simpler than in the time-signal space. An additional advantage is that feature reconstruction may be inserted seamlessly in state-of-the-art recognizers as an extra module between the feature extractor and decoder modules, while time-signal restoration may be difficult to implement in certain tasks, for example in distributed systems.

The interest of the speech community in signal restoration for the particular problem of band-limiting distortions has been raised in recent years. Seltzer *et al.* [2005] proposed the combination of two types of compensation in a single framework; for noise compensation a simple offset is used as in typical SPLICE and for bandwidth-extension a linear compensation is proposed, as it was previously suggested in our own work [Morales *et al.*, 2005a] [Morales *et al.*, 2005b]. In [Kim and Hansen, 2006] the masking approach originally presented for compensation of additive noise ([Raj *et al.*, 2004]) is modified to fit band-limiting distortions. In the original work by Raj *et al.* the values of spectral energies in observations of corrupted speech give the upper bound for reconstructed data. However, as explained by Kim and Hansen, in band-limiting distortions this assumption is no longer valid.

A related application has been recently considered, too; if full-bandwidth data is scarce and band-limited speech is available, full-bandwidth models could be trained using a combination of real full-bandwidth data and reconstructed data obtained by expansion of band-limited speech (for example telephone data). In [Seltzer and Acero, 2005], a limited amount of full-bandwidth data is used together with large amounts of bandwidth-extended data and the usual Baum-Welch training strategy follows in order to obtain full-bandwidth acoustic models. However, this approach artificially reduces the covariance of Gaussian mixtures, because features are reconstructed in a deterministic way. In an extension of this work [Seltzer and Acero, 2007], a modification of the Expectation-Maximization (EM) algorithm was proposed for training HMMs using the combination of full-bandwidth and limited bandwidth features, but introducing this time the uncertainty associated with missing data in the HMM covariance matrixes. Although they obtained promising results, it is unclear how performance compares to the more standard approach of training models with limited bandwidth data and adapting them using full-bandwidth data and acoustic model adaptation techniques.

3.4 A Mathematical Model of the Effect of Band-limiting Distortions on Cepstral Features

Following the notation in Eq. (2.14) purely convolutional distortions may be expressed as follows for each output in a bank of filters:

$$\left|Y\left(f_j\right)\right|^2 = \left|X\left(f_j\right)\right|^2 \cdot \left|H\left(f_j\right)\right|^2 = \left|Y_j\right|^2 = \left|X_j\right|^2 \cdot h_j \,, \tag{3.1}$$

where the change of notation has been introduced for clarity in the following discussion.

Typically, the transfer function of a microphone is not flat in the entire spectrum, resulting in the attenuation of the signal in some spectral regions. In those cases, CMN is a good solution to restore the attenuated portions of speech, which effectively should result in the application of the inverse of the transfer function of the microphone h_j.

The singularity of band-limiting distortions compared to other convolutional distortions is that in band-limiting distortions parts of the spectrum are completely removed, while in general convolutional

distortions it is normally assumed that the spectrum is multiplied by a non-zero value. For complete removal of bands Eq. (3.1) remains true, but h_j follows a particular form:

$$\left|Y_j\right|^2 = \left|X_j\right|^2 \cdot h_j, \quad \text{where} \quad \begin{cases} h_j = 0 & if\ j \in F \\ h_j = 1 & if\ j \notin F \end{cases}, \tag{3.2}$$

where F represents the set of channels in the filterbank removed by the bandwidth-limitation. Note that this is an approximation since we assume constant and binary values for the band-limitation in the outputs of the bank of filters. Complete removal of bands means that normalization techniques that search for the inverse transfer function of the channel in order to restore the original signal are not useful anymore, because for completely removed channels the inverse of the transfer function h_j does not exist. This mathematical conclusion is also intuitive, as information from removed channels is completely lost; instead, reconstruction of removed parts of the spectrum should be attempted with information from the remaining parts (assuming that there exists correlation between spectral regions).

As most state-of-the-art recognition systems employ feature parameterizations of the family of MFCCs (Section 2.4.2), particular emphasis is made on this type of features in this Thesis, so that the proposed algorithms may be implemented in a straightforward manner. Following is a mathematical model relating MFCCs of full-bandwidth speech and band-limited speech.

The general definition of MFCCs is [Young *et al.*, 2005; p.61]:

$$x_i(t) = \mathbf{C}\left(\log\left(\left|X_j(t)\right|^2\right)\right) = \sqrt{\frac{2}{N}}\sum_{j=1}^{N}\log\left(\left|X_j(t)\right|^2\right)\cos\left(\frac{\pi i}{N}(j-0.5)\right), \tag{3.3}$$

where i is the order of the MFCC coefficient[2] and we have introduced explicitly the time index t to denote that we are working with observations. The DCT transformation matrix $\mathbf{C}$ is expanded on the right-hand side of the equation, for clarity. We may rewrite Eq. (3.3) as:

$$x_i(t) = \sum_{j=1}^{N}\log\left(\left|X_j(t)\right|^2\right)\cdot C_{ij}, \quad \text{where} \quad C_{ij} = \sqrt{\frac{2}{N}}\cdot\cos\left(\frac{\pi i}{N}(j-0.5)\right). \tag{3.4}$$

Similarly, MFCC features of band-limited speech are:

$$y_i(t) = \sum_{j=1}^{N}\left(\log\left(h_j(t)\cdot\left|X_j(t)\right|^2 + e_j(t)\right)\right)C_{ij}, \tag{3.5}$$

where the term e_j accounts for residual computational errors and inaccuracies of the model. When the original signal is affected by additive noise it may also be included in this term, but in our model, we assume the signal's SNR is large.

[2] MFCC stands for Mel Frequency Cepstrum Coefficient. Thus, instead of MFCC coefficients it would be more appropriate to say MFC Coefficients. However, for clarity MFCC will be maintained as a full unit of meaning throughout this Thesis.

The difference between equivalent MFCCs from speech recorded simultaneously in full-bandwidth and band-limited environments is obtained from Eqs. (3.4) and (3.5):

$$x_i(t) - y_i(t) = \sum_{j=1}^{N} \left[\log\left(\left| X_j(t) \right|^2 \right) - \log\left(h_j(t) \cdot \left| X_j(t) \right|^2 + e_j(t) \right) \right] C_{ij} \,. \tag{3.6}$$

Therefore, we can write the full-bandwidth MFCC in terms of the band-limited MFCC and a correction term. In the following equation, the sum over all the channels in the filterbank is decomposed into 2 terms corresponding to those channels affected by the bandwidth restriction and those channels that remain intact:

$$\begin{aligned} x_i(t) = y_i(t) + \Bigg[& \sum_{\substack{j=1 \\ j \notin F}}^{N} \left[\log\left(\left| X_j(t) \right|^2 \right) - \log\left(h_j(t) \cdot \left| X_j(t) \right|^2 + e_j(t) \right) \right] + \\ & + \sum_{\substack{j=1 \\ j \in F}}^{N} \left[\log\left(\left| X_j(t) \right|^2 \right) - \log\left(h_j(t) \cdot \left| X_j(t) \right|^2 + e_j(t) \right) \right] \Bigg] \cdot C_{ij} \,, \end{aligned} \tag{3.7}$$

where F represents the group of channels affected by the bandwidth limitation.

For bandwidth limited speech (i.e., complete removal of spectral bands), we assume $h_j \to 1$ for unmodified parts of the spectrum and $h_j \to 0$ for removed channels (and for expanded regions in upsampled data). For the channels not affected by the bandwidth limitation, $e_j \ll h_j \cdot \left| X_j \right|^2$, and $h_j \cdot \left| X_j \right|^2 \to \left| X_j \right|^2$, so the sum over the unaffected channels in Eq. (3.7) can be discarded. Also, as $h_j \to 0$ for the channels removed, in this case $h_j \cdot \left| X_j \right|^2 \ll e_j$ and we can therefore approximate the full-bandwidth MFCCs as:

$$x_i(t) \approx y_i(t) + \sum_{\substack{j=1 \\ j \in F}}^{N} \left[\log\left(\left| X_j(t) \right|^2 \right) - \log\left(e_j(t) \right) \right] \cdot C_{ij} \tag{3.8}$$

Again assuming the term e_j (the output of filter j in the range of frequencies removed in band-limited speech) negligible compared to $\left| X_j \right|^2$ (the output of filter j in full-bandwidth speech), we obtain a further simplification:

$$x_i(t) \approx y_i(t) + \sum_{\substack{j=1 \\ j \in F}}^{N} \left[\log\left(\left| X_j(t) \right|^2 \right) \right] \cdot C_{ij} \,. \tag{3.9}$$

This equation reflects the relation between the full-bandwidth and limited-bandwidth features but does not seem to help in estimating the full-bandwidth features because the sum is over the output of the filters in the missing parts of the spectrum.

In order to solve this apparent dead end, we make use of the central idea of bandwidth extension: different parts of the spectrum are highly correlated, the correlation being defined by the configuration adopted by the vocal tract in order to produce different sounds. Under this assumption and for a class of sounds k that share similar acoustic characteristics (similar configuration of the vocal tract), the

filterbank outputs in the removed regions of the spectrum when input is full-bandwidth speech are related to those in the available parts (a_m) by some unknown function E_j^k:

$$\widehat{X_j} \approx E_j^k\left(X_{a_1},\ldots,X_{a_M}\right); \quad j \in F \quad and \quad a_m \notin F\,. \tag{3.10}$$

Alternatively, we could use as the estimators the values of the MFCCs of restricted bandwidth speech that have an approximately direct relation (Eq. (3.5)) with the filterbank outputs that appear as parameters of the mapping functions in Eq. (3.10):

$$\widehat{X_j} \approx G_j^k\left(\mathbf{y}\right); \quad j \in F\,. \tag{3.11}$$

Finally, by inserting this into Eq. (3.9) and combining G_j^k with the logarithm and C_{ij} term into the new functions H_j^k we obtain a new equation:

$$x_i\left(t\right) \approx y_i\left(t\right) + \sum_{\substack{j=1\\ j\in F}}^{N}\left[\log\left(\left|X_j\left(t\right)\right|^2\right)\right]\cdot C_{ij} = y_i\left(t\right) + \sum_{\substack{j=1\\ j\in F}}^{N} H_j^k\left(\mathbf{y}\right) = J_j^k\left(\mathbf{y}\left(t\right)\right). \tag{3.12}$$

Eq. (3.12) establishes that full-bandwidth features belonging to a particular class of sounds, *k*, may be approximately reconstructed using unknown functions of the band-limited features, $J_j^k\left(\mathbf{y}\left(t\right)\right)$. Therefore, two questions arise at this point: how to partition speech into classes of sounds and what is the form of the $J_j^k\left(\mathbf{y}\left(t\right)\right)$ functions. These questions are addressed in Chapter 4.

3.5 Cepstral Decorrelation and Band-limiting Distortions

One of the reasons for the extended use of MFCCs in speech recognition is that they are considered a quasi-uncorrelated set of parameters. This justifies the use of acoustic models with diagonal covariance matrices, which highly reduce computational costs (at a low accuracy cost). From the point of view of compensation of distorted feature, uncorrelation suggests that in practice no information pertaining to a given MFCC coefficient may be obtained from any other coefficient. In Chapter 5, compensation formulas are presented rigorously and the general expressions take the form of multivariate compensations (also introduced in Chapter 2 for noise compensation). For example, the ML form of the compensation is:

$$\mathbf{x}^{ML}\left(\mathbf{y},k\right) = \mathbf{B}_k\mathbf{y} + \mathbf{b}_k\,, \tag{3.13}$$

where a full compensation matrix $\mathbf{B}_k$ indicates multivariate compensation. However, in the light of the feature decorrelation it seems that a simplification of the feature compensation formulas from full compensation matrixes (multivariate compensation) to diagonal compensation matrixes (univariate compensation) would be possible with very little impact in performance. In fact, this is normally done in feature compensation for noise-corrupted speech reconstruction [Droppo *et al.*, 2001] [Buera *et al.*, 2007], and has also been done for band-limiting distortions [Morales *et al.*, 2005b] [Seltzer *et al.*, 2005]. However, we show in this section that this simplification is less justified for band-limiting distortions. Our discussion starts with the derivation of the property of *quasi*-uncorrelation for MFCC features in full-bandwidth speech. We explain some theoretical indications that point towards a higher correlation of MFCCs of band-limited speech and show an empirical example that supports our prediction. We

finally justify our hypothesis that based on the larger degree of MFCC feature correlation, feature compensation of band-limited speech should employ non-diagonal compensation matrixes.

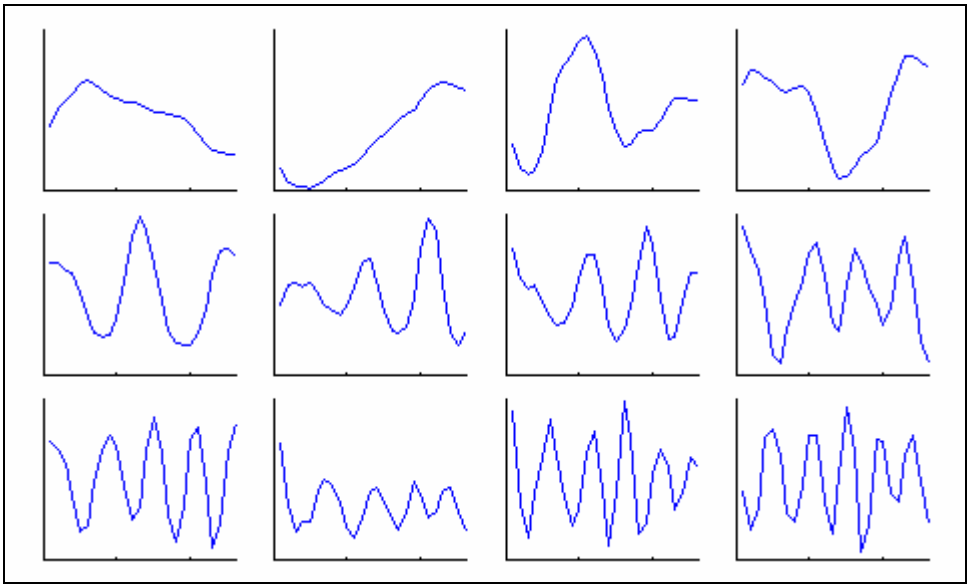
Dimensionality reduction techniques are a set of algorithms that search for the optimal representation of a particular distribution with maximal compactness. For example, Principal Component Analysis (PCA) [Duda *et al.*, 2000; pp. 115-117], is a technique that extracts the directions of maximum variability of a particular set of data samples. In practice, these are the eigen-vectors of the sample covariance matrix of the distribution, and their eigen-values represent the variability captured by each direction. Therefore, an efficient way for compressing information with minimal loss of data variability is a change of basis where the vectors of the new basis are the first eigen-vectors of the sample covariance matrix. An additional property of these vectors is that they are uncorrelated and orthogonal (these seem to be reasonable properties for a representation that allows dimensionality reduction). However, PCA computation can be expensive. Moreover, it depends on the considered set of samples; therefore the vector basis would be subject to variability and changes in the environment would require a change of basis. On the contrary, it would be desirable to use a fixed basis that nevertheless has similar properties to PCA-derived vectors. This fixed transformation came to be MFCCs.

The origin of MFCC coefficients is closely tied to PCA. The first author to propose a transformation of a spectral representation of speech in terms of sinusoidal functions was Yilmaz [1967] [1972], based on psychoacoustic observations and on the property of orthogonality of a sinusoidal basis. However, it was Pols [1977; pp. 42-52] who realized that for a particular set of speech utterances, PCA-derived vectors resembled the shapes of sinusoidal functions. He then studied the variability captured by each PCA-derived eigen-vector and each MFCC and observed that they were similar, too. In the first two rows of Table 3.1 we show the results of similar experiments to those by Pols regarding the relation between MFCCs and PCA-derived eigen-vectors in a full-bandwidth (FB) distribution using TIMIT data. Similarly to results in Pols' work, we observe that the variability captured by vectors with the same index in the MFCC and PCA bases is similar. Also, in Figure 3.2 a) we show the shapes of the first eigen-vectors which resemble, to a certain extent, sinusoidal functions.

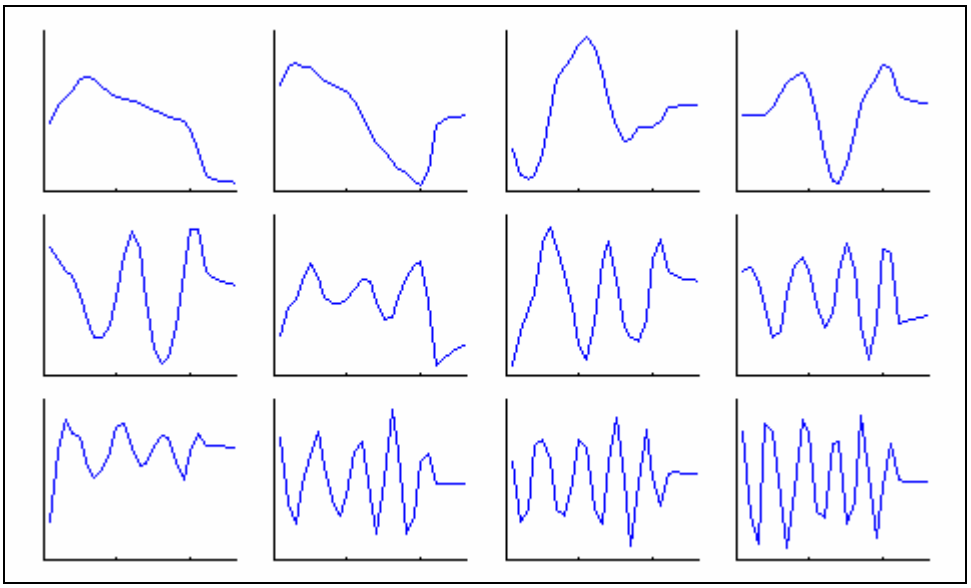
When a band-limiting condition is considered the situation is slightly changed because the energy in missing parts of the spectrum will be several dBs below that in the unfiltered bands. Thus, extraction of MFCCs over band-limiting channels can also be viewed as a transformation of full-bandwidth speech where the sinusoidal functions are flattened in the filtered regions. For example, in Figure 3.3 we show the theoretical cosine transformation vectors of orders 1 and 3 on top and an equivalent transformation including band-pass filtering with cut-off frequencies 300-3400 Hz (BP300-3400Hz) in

Vector index→ **Vector basis ↓**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
MFCC FB	66.90	17.68	3.93	2.73	1.87	1.30	0.95	0.82
PCA FB	71.52	15.27	3.16	2.80	1.55	0.96	0.83	0.58
MFCC BP300-3400	80.85	7.14	3.38	2.07	1.69	1.00	0.78	0.68
PCA BP300-3400	63.86	16.24	6.46	3.58	1.35	2.14	1.36	0.76

Table 3.1 Relative variance captured by PCA-derived eigen-vectors or MFCC transformation vectors for full-bandwidth and band-pass filtered data. Vectors in both bases seem to capture a similar amount of variation in the case of full-bandwidth speech, but not so much in band-limited speech.



a)



b)

Figure 3.2 First 12 eigen-vectors for a log-mel Frequency Energy representation of a) full-bandwidth and b) limited-bandwidth speech (BP300-3400 Hz). The first set presents a resemblance with sinusoidal functions, while the second set shows deviations, especially near the borders.

the bottom. In this case the basis that defines the MFCC transformation over band-limited speech is not orthogonal anymore and therefore, we expect MFCCs to resemble less clearly PCA-derived features and present increased correlation. The plot is a simplification because the result of the logarithm in the filtered regions has been zeroed. In reality the logarithm is never 0, but a very significant decimation of the energy exists in band-limited regions, so the orthogonality of the transformation is still broken. In the last two rows of Table 3.1 we show that the relationship between the variability captured by eigen-vectors derived from band-limited TIMIT and MFCCs is less obvious than that for full-bandwidth speech, and similarly, in Figure 3.2 b) the shapes of the eigen-vectors are less similar to sinusoidals, particularly near the borders.

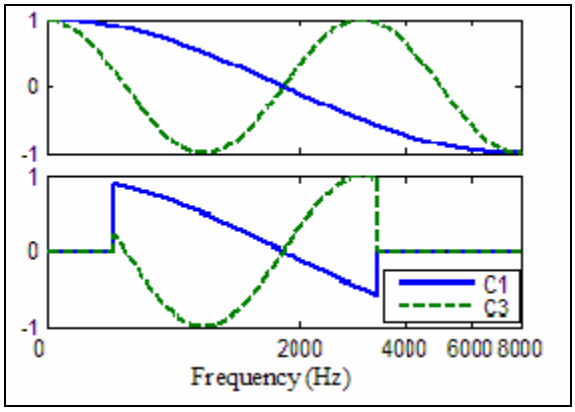


Figure 3.3 Cepstral transformations of orders 1 and 3 for full-bandwidth (top) and limited-bandwidth speech (bottom; 300-3400Hz band-pass filter). The band-limited transformation basis is no longer orthogonal. The plot is for MFCC of mel spectrum prior to log computation.

In order to verify empirically our hypothesis that MFCCs of band-limited speech are more correlated than those of full-bandwidth speech, we define the following measure of non-diagonality of a covariance matrix:

$$nonDiag = \sum_{i}^{staticMFCCs} \sum_{j,\, j \neq i}^{MFCCs} \delta_{ij} \,, \tag{3.14}$$

where

$$\delta_{ij} = \begin{cases} 1 & if\ corr\_coef(i,j) \equiv \dfrac{\mathrm{cov}(i,j)}{\sqrt{\mathrm{cov}(i,i)\cdot \mathrm{cov}(j,j)}} \geq \tau \\ 0 & otherwise \end{cases} . \tag{3.15}$$

This metric is invariant for scaling of MFCC features and establishes that a significant correlation exists if the covariance between two coefficients, normalized by their standard deviations (i.e. the correlation coefficient) is larger than a threshold. It should be noted that this measure is quantized in the sense that coefficients with a significant correlation are given a score of 1 and others, a score of 0. This seems more interesting to us than a continuous measure such as multiplication of correlation coefficients; the quantized measure tells us how many features are highly correlated, while in a continuous measure, very low correlation between two terms could hide the fact that an important number of other coefficients are correlated.

Experiments setting the threshold to $\tau = 5$ and using all TIMIT training data to compute the covariance matrix produced a non-diagonality of 51 for full-bandwidth MFCC coefficients, 108 for LP4kHz coefficients and 110 for BP300-3400Hz coefficients (actual numbers vary for different values of τ, but the tendency is steady and clear). It should be noted that in all three feature spaces considered all the coefficients contributing to the measure of non-diagonality had $p < 0.01$. The p-value of each coefficient indicates the probability of obtaining the observed correlation coefficient, when the actual correlation is null (small values of p indicate statistical significance of the correlation coefficient).

The previous argumentation seems to indicate that when the distortion affecting speech is of a band-limiting type, better reconstruction performance may be obtained by multivariate compensation. In order to evaluate this hypothesis, in Chapter 8 we compare performance of feature compensation using the univariate and multivariate approaches.

4

A Unified Framework for Feature Compensation

In this chapter we develop a unified framework for feature compensation and present a series of strategies and algorithms for different stages of the feature compensation process. Applications, experiments and discussion are deferred to Chapters 7 and 8.

As discussed in Section 3.4 and shown in Eq. (3.9) direct reconstruction of the full-bandwidth features from restricted-bandwidth speech frames is not possible, in principle, because bandwidth restriction implies loss of information. However, an important redundancy exists between different frequency bands of the speech signal. Therefore, by learning the relation between the available and unavailable parts of the speech signal we expect to be able to estimate full-bandwidth feature vectors from band-limited vectors, as shown in Eq. (3.12).

Our general feature compensation framework is presented in Figure 4.1 In the training stage the feature space is first partitioned into clusters of data (partitioning classes). It is assumed that data within a cluster in the distorted feature space suffered a similar transformation as a result of the distortion and so, to a certain extent, full-bandwidth data will show a similar clustering structure. Feature compensation aims at creating the opposite effect for each cluster, so as to reconstruct full-bandwidth features, and for this goal, a set of corrector functions is trained for each cluster, mapping observations from the distorted space to the full-bandwidth space. In the compensation stage distorted speech feature vectors are imputed to one of the partitioning classes (ML compensation) or several of them (MMSE compensation) and corrected with the corresponding corrector functions.

This chapter is organized following the structure suggested in Figure 4.1: first we describe two different strategies for partitioning the feature space (Phoneme-based and Gaussian mixture-based). Second, we propose two different methodologies for training corrector functions depending on whether or not stereo-data is available (in this Thesis we define stereo data as speech for which simultaneous recordings are available from the restricted-bandwidth environment and the full-bandwidth

environment). Third, we show different possibilities for feature compensation and finally a post-processing solution is proposed for improving the quality of compensation removing abrupt changes in the compensation sequence using a smoothing function.

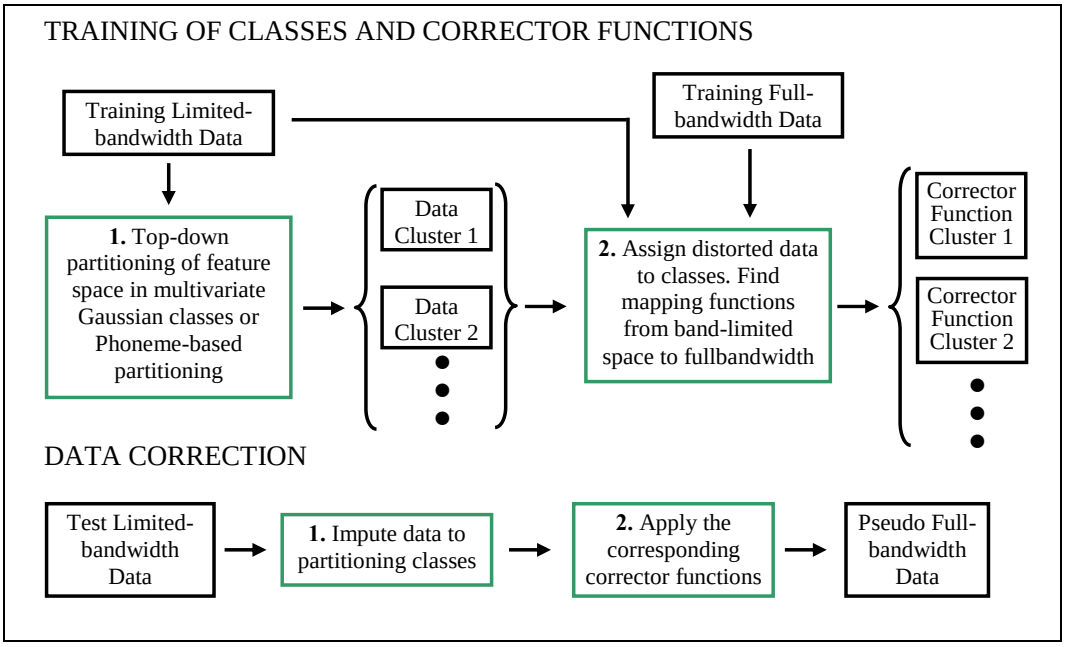


Figure 4.1 Schematic representation of the proposed architecture for training of classes and corrector functions and for compensation of band-limited feature vectors to generate pseudo full-bandwidth feature vectors.

4.1 Phoneme-based Partitioning

Phoneme-based partitioning of the feature space assumes that realizations of any given phoneme are affected similarly by a particular distortion and may therefore be compensated using the same corrector functions. As suggested in Eq. (3.12) compensation of a frame should be made according to its classification in a group of frames with similar energy distribution C and it seems reasonable to assume, at least as a first attempt, that phonemes are an appropriate classification criterion. Therefore, the feature space is divided according to the phonetic classification of observations (a knowledge-based criterion) using phonetically aligned labels of training data (alternatively automatic phonetic alignment may be made instead).

4.2 Data-driven Gaussian Class-based Partitioning

Alternatively, it is possible to avoid the use of linguistic knowledge and labels and define a partitioning of the acoustic feature space based on data-driven criteria. Instead of making decisions according to a-priori knowledge that may not adequately fit actual observations, data-driven methods allow the system to more *freely* make the most appropriate decisions for the desired goal. Therefore, in addition to removing the need of a-priori knowledge, these criteria, may address particular problems in a more direct way than knowledge-based criteria. One such criterion exploited in this Thesis is Gaussian class-based feature space partitioning, where multivariate Gaussians define regions of the feature space where data share a similar spectral distribution. The following two sections explore two strategies for Gaussian class partitioning depending on whether stereo data is available, or not.

4.2.1 Non-stereo Data Partitioning

The method follows an EM strategy and may be considered a simplification of Baum-Welch, which is typically used for training acoustic HMM models [Rabiner, 1989] [Young *et al.*, 2005; pp. 114-131]. The only requirement for training data is that it must sufficiently represent the speaker-independent distorted feature space; on the contrary, no labeling pertaining speaker identity or phonetic content is required.

An initial cluster is defined as a Gaussian distribution with mean $\boldsymbol{\mu}_{0,0}$ and covariance $\boldsymbol{\Sigma}_{0,0}$, representing the mean vector and covariance matrix of the whole data distribution. This initial cluster is divided into two by perturbing the mean vector as +/- η times the vector of standard deviations, where η is a perturbation factor (0.2 in all our experiments) and in subsequent steps the number of Gaussians is increased by splitting one Gaussian mixture each time, according to a partitioning criterion. The process is shown here schematically:

1. **Initialization**: Find $\boldsymbol{\mu}_{0,0}$ and $\boldsymbol{\Sigma}_{0,0}$ of the whole data. This is so far the only class, so we set class 0 as the class to split in the first iteration.
2. **Split class**: The new class is stored in position *M*. Assuming class *y* was designated as the class to split in iteration *M*-1, classes *M* and *y* of the new iteration *M* are set to:

$$\begin{aligned} \boldsymbol{\mu}_{M,y} &= \boldsymbol{\mu}_{(M-1),y} + \eta \cdot \boldsymbol{\sigma}_{(M-1),y} \\ \boldsymbol{\mu}_{M,M} &= \boldsymbol{\mu}_{(M-1),y} - \eta \cdot \boldsymbol{\sigma}_{(M-1),y} \end{aligned}; \qquad \boldsymbol{\Sigma}_{M,M} = \boldsymbol{\Sigma}_{M,y} = \boldsymbol{\Sigma}_{(M-1),y}, \tag{4.1}$$

   where $\boldsymbol{\mu}_{s,r}$ is the mean vector of Gaussian mixture *r* in iteration *s* and $\boldsymbol{\sigma}_{s,r}$ the vector of standard deviations, obtained directly as the squared roots of diagonal elements in the covariance matrix. The rest of classes remain untouched:

$$\boldsymbol{\mu}_{M,z} = \boldsymbol{\mu}_{(M-1),z}; \quad \boldsymbol{\Sigma}_{M,z} = \boldsymbol{\Sigma}_{(M-1),z} \qquad \textit{when} \quad z \neq y \quad \textit{and} \quad z \neq M\,. \tag{4.2}$$

3. **Re-evaluate classes**: Observations are reassigned to the class for which likelihood is maximal and mean vectors and covariance matrixes are re-calculated. This is repeated a number of times every time a new class is introduced (3 times in our experiments).
4. **Find class to split**: If the number of classes is smaller than the specified final number of classes, find the class to split according to an established criterion and go to step 2. In our experiments the criterion is to split the mixture with larger internal variability, although other criteria have been explored with similar results.

4.2.2 Stereo Data Partitioning

In the previous section we presented non-stereo partitioning, a technique that assumes that clusters group together data for which the distortion caused a similar transformation. However, when stereo data is available, it is possible to try a more explicit approach with the goal of identifying clusters that actually followed a similar transformation. In [Afify *et al.*, 2007] it was proposed to use a supervector of features for each observation, where full-bandwidth features and band-limited features are concatenated. In this manner, we may use the same algorithm that was proposed in the previous section for a slightly different partitioning; as the feature super-vectors contain stereo realizations of full-

bandwidth and distorted-bandwidth observations, the resulting clusters should group together data for which features are close both in the full-bandwidth and limited-bandwidth spaces.

Additionally, a different method based on minimizing the Mean Squared Error (MSE) of predictions could also be considered, although its implementation is left for future work. Step 4 in the partitioning algorithm presented in the previous section may be modified so that the criterion for choosing the class to partition at each stage would be governed by the minimization of the MSE between full-bandwidth features and reconstructed features using a development set. In each iteration it is possible to compute the MSE after splitting any of the existing classes, and the set of children clusters that minimizes MSE would be used for the next iteration. This implementation in its complete form requires following steps 2 and 3 in each iteration for any set of possible children clusters before MSE can be computed and thus, highly increases computation time.

4.3 Stereo Data Training of Corrector Functions

Corrector functions are trained with the goal of optimal data mapping between the limited-bandwidth and full-bandwidth features. When stereo data is available mapping is possible using Linear Least Squares curve fitting techniques. For example, for univariate polynomial compensation, the coefficients for each corrector class were obtained independently for each element in the feature vector by fitting pairs of observations (full-bandwidth and limited-bandwidth) to a polynomial curve [Press *et al.*, 1992; pp. 656-680]. In Figure 4.2 we show an example of data mapping and the resulting curve for a particular Gaussian class and MFCC coefficient C2. For each point in the plot the x-coordinate corresponds to the restricted-bandwidth MFCC (Low-Pass filter with cut-off frequency 4kHz; LP4kHz) and the y-coordinate is the full-bandwidth MFCC in a given frame.

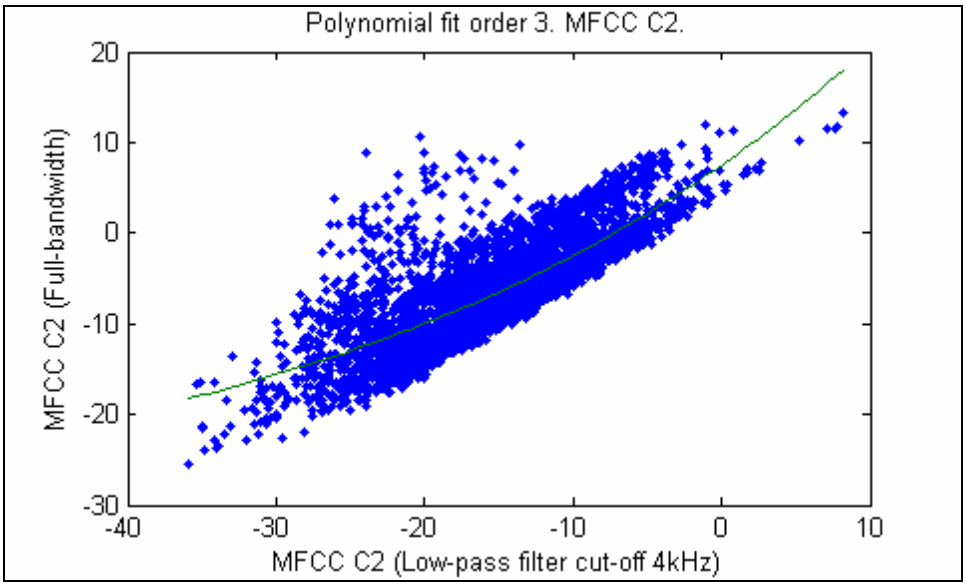


Figure 4.2 Mapping of LP4kHz data to full-bandwidth data for MFCC C2 in a particular Gaussian class. The plot also shows a third order polynomial fit.

In multivariate compensation we used a step-wise strategy that successively introduces new compensation features in a multivariate fit using analysis of variances [MATLAB ANOVAS]. In this way it is possible to study the evolution of the Root MSE (RMSE) of the multivariate fit for each new coefficient introduced and the number of final coefficients may be set to guarantee convergence of this measure. For example, Figure 4.3 shows the evolution of the RMSE for a multivariate fit of full-bandwidth MFCC C2 using MFCC coefficients from the limited-bandwidth distribution. Not surprisingly the first coefficient inserted, after the offset is C2. Going from a simple offset to a fit with a single coefficient reduces RMSE from 6.28 to 3.58 and results in the same compensation applied in univariate linear compensation. However, the inclusion of the next 4 coefficients (C1, C3, C6 and C11) further reduces the RMSE to 3.19, which seems to indicate that significant benefits may be obtained by applying multivariate compensation. On the contrary inclusion of additional coefficients offers little improvement, suggesting that in this case, the multivariate fit may be truncated after the first 6 terms. Moreover, inclusion of extra coefficients might cause over-fitting and have a negative impact in reconstruction of test data unseen during training.

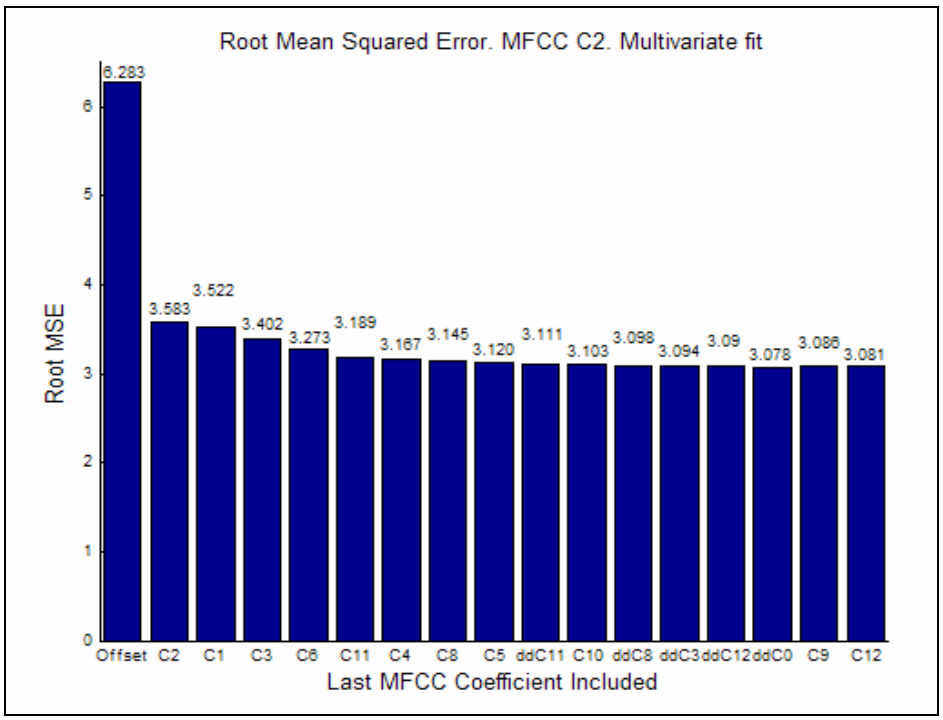


Figure 4.3 Evolution of RMSE for stepwise multivariate fit of full-bandwidth MFCC C2 to limited-bandwidth MFCCs (for a low-pass filter, cut-off frequency 4kHz). Values shown are the RMSE after the coefficient indicated in the x-axis is introduced in the multivariate fit (C2, C1, etc. indicate static MFCC coefficients of orders 2, 1, etc., respectively. 'dd' coefficients are second-order derivatives. 'd' coefficients would be first order derivatives but none is among the best first 17 candidates shown in the plot).

4.4 Non-stereo Data Training of Corrector Functions

In situations where collecting stereo data is complicated, it is still possible to train corrector functions that globally map the distribution of band-limited data to that of full-bandwidth data. Success relies on the assumption that both distributions sufficiently represent their respective feature spaces. An iterative method based on EM is used to obtain an optimal mapping between both spaces. We assume that K clusters of data partitioning the filtered space have been found following either Phoneme-based or

Gaussian class-based approaches, as shown in Sections 4.1 and 4.2. In the most general case the vector of means $\boldsymbol{\mu}_{Y,k}$ and matrix of covariance $\boldsymbol{\Sigma}_{Y,k}$ of a cluster *k* in the band-limited feature space are related to their full-bandwidth equivalents as:

$$\boldsymbol{\mu}_{X,k} = \boldsymbol{\mu}_{Y,k} + \mathbf{r}_k \,, \tag{4.3}$$

$$\boldsymbol{\Sigma}_{X,k} = \boldsymbol{\Sigma}_{Y,k} + \mathbf{R}_k \,. \tag{4.4}$$

For the moment no constraints are imposed on the correction factors $\mathbf{r}_k$ and $\mathbf{R}_k$ and they may take any values. This assures generality, because the vector of means and matrix of covariance for a cluster in the band-limited space may take any form in the full-bandwidth space.

The EM training strategy searches for optimizing the distribution of emission probability in the full-bandwidth space:

$$p(\mathbf{x}) = \sum_{k=1}^{K} N(\mathbf{x}; \boldsymbol{\mu}_{X,k}, \boldsymbol{\Sigma}_{X,k}) \cdot P(k) \,. \tag{4.5}$$

It can be shown that maximizing Eq. (4.5) for all available data is equivalent to iteratively maximizing an auxiliary function *Q* defined as [Moreno, 1996] [Huang *et al.*, 2001; pp. 170-172]:

$$Q(\phi, \overline{\phi}) = \sum_{t=1}^{T} \sum_{k=1}^{K} \frac{p(\mathbf{x}_t, k | \phi)}{p(\mathbf{x}_t | \phi)} \log\left(p(\mathbf{x}_t, k | \overline{\phi}) \right), \tag{4.6}$$

where ϕ is the set of old Gaussian classes in the full-bandwidth space (defined by their vectors of means and covariance matrixes in the full-bandwidth space, or equivalently by the transformations of the band-limited classes, $\mathbf{r}_k$ and $\mathbf{R}_k$ as in Eqs. (4.3) and (4.4)) and $\overline{\phi}$ the new set. Substituting Eqs. (4.3) and (4.4) in Eq. (4.6), expanding the term of posterior probability and simplifying, we obtain:

$$\begin{aligned} Q(\phi, \overline{\phi}) = const. + \sum_{t=1}^{T} \sum_{k=1}^{K} p(k | \mathbf{x}_t, \phi) \cdot \\ \cdot \left\{ -\frac{1}{2} \log\left| \boldsymbol{\Sigma}_{Y,k} + \overline{\mathbf{R}}_k \right| - \frac{1}{2} \left(\mathbf{x}_t - \boldsymbol{\mu}_{Y,k} - \overline{\mathbf{r}}_k \right)^T \left(\boldsymbol{\Sigma}_{Y,k} + \overline{\mathbf{R}}_k \right)^{-1} \left(\mathbf{x}_t - \boldsymbol{\mu}_{Y,k} - \overline{\mathbf{r}}_k \right) \right\}. \end{aligned} \tag{4.7}$$

Now, differentiating in terms of $\overline{\mathbf{r}}_k$ and $\overline{\mathbf{R}}_k$ and equating to zero, we obtain solutions that maximize *Q*. The solution for full covariance matrixes and compensation matrixes is complicated, but a great simplification occurs if both are assumed diagonal. In that case:

$$\frac{\partial Q}{\partial \overline{\mathbf{r}}_k} = \sum_{t=1}^{T} p(k | \mathbf{x}_t, \phi) \cdot \left(\boldsymbol{\Sigma}_{Y,k} + \overline{\mathbf{R}}_k \right)^{-1} \cdot \left(\mathbf{x}_t - \boldsymbol{\mu}_{Y,k} - \overline{\mathbf{r}}_k \right) = 0 \,, \tag{4.8}$$

$$\begin{aligned} \frac{\partial Q}{\partial \overline{\mathbf{R}}_k} = -\frac{1}{2} \cdot \sum_{t=1}^{T} p(k | \mathbf{x}_t, \phi) \cdot \\ \cdot diag\left\{ \left(\boldsymbol{\Sigma}_{Y,k} + \overline{\mathbf{R}}_k \right)^{-1} - \left(\mathbf{x}_t - \boldsymbol{\mu}_{Y,k} - \overline{\mathbf{r}}_k \right)^T \cdot \left(\boldsymbol{\Sigma}_{Y,k} + \overline{\mathbf{R}}_k \right)^{-2} \cdot \left(\mathbf{x}_t - \boldsymbol{\mu}_{Y,k} - \overline{\mathbf{r}}_k \right) \right\} = \mathbf{0} \,, \end{aligned} \tag{4.9}$$

where *diag* (**A**) is a vector containing diagonal elements in matrix **A**, and **0**, a vector of zeros.

Solving for the corrector coefficients:

$$\overline{\mathbf{r}}_k = \frac{\sum_{t=1}^{T} p\left(k\left|\mathbf{x}_t,\phi\right.\right)\cdot\mathbf{x}_t}{\sum_{t=1}^{T} p\left(k\left|\mathbf{x}_t,\phi\right.\right)} - \boldsymbol{\mu}_{Y,k}, \tag{4.10}$$

$$\overline{\mathbf{R}}_k = \frac{\sum_{t=1}^{T} p\left(k\left|\mathbf{x}_t,\phi\right.\right)\cdot\mathbf{I}\odot\left\{\left(\mathbf{x}_t-\boldsymbol{\mu}_{Y,k}-\overline{\mathbf{r}}_k\right)\cdot\left(\mathbf{x}_t-\boldsymbol{\mu}_{Y,k}-\overline{\mathbf{r}}_k\right)^T\right\}}{\sum_{t=1}^{T} p\left(k\left|\mathbf{x}_t,\phi\right.\right)} - \boldsymbol{\Sigma}_{Y,k}, \tag{4.11}$$

where $\odot$ is the product of two matrixes term by term.

These solutions are equivalent to those in RATZ, originally designed for noise compensation, with the difference that classes in RATZ were defined in the clean space [Moreno, 1996].

Let's assume now that the effect of the distortion in each class is simply an offset for each observation. Then the vectors of means and covariances are modified as:

$$\boldsymbol{\mu}_{X,k} = E\left\{\mathbf{x}_k\right\} = E\left\{\mathbf{b}_k+\mathbf{y}_k\right\} = \mathbf{b}_k+\boldsymbol{\mu}_{Y,k}, \tag{4.12}$$

$$\boldsymbol{\Sigma}_{X,k} = E\left\{\left(\mathbf{x}_k-\boldsymbol{\mu}_{X,k}\right)^2\right\} = E\left\{\left(\mathbf{b}_k+\mathbf{y}_k-\mathbf{b}_k-\boldsymbol{\mu}_{Y,k}\right)^2\right\} = E\left\{\left(\mathbf{y}_k-\boldsymbol{\mu}_{Y,k}\right)^2\right\} = \boldsymbol{\Sigma}_{Y,k}. \tag{4.13}$$

Identifying Eqs. (4.12) and (4.13) with Eqs. (4.3) and (4.4) we obtain:

$$\mathbf{r}_k = \mathbf{B}_k, \tag{4.14}$$

$$\mathbf{R}_k = \mathbf{0}. \tag{4.15}$$

Therefore, the extra constraint of simple offsets suppresses Eq. (4.11) in the EM algorithm, so covariance matrixes are not updated anymore and $\mathbf{r}_k$, as defined in Eq. (4.10) will be the offset vector to apply for feature compensation.

Alternatively, we may use a model of univariate linear compensation. Then:

$$\boldsymbol{\mu}_{X,k} = E\left\{\mathbf{x}_k\right\} = E\left\{\mathbf{b}_k+\mathbf{B}_k\cdot\mathbf{y}_k\right\} = \mathbf{b}_k+\mathbf{B}_k\cdot\boldsymbol{\mu}_{Y,k}, \tag{4.16}$$

where $\mathbf{B}_k$ is diagonal. Similarly, for the covariance matrix:

$$\boldsymbol{\Sigma}_{X,k} = E\left\{\left(\mathbf{x}_k-\boldsymbol{\mu}_{X,k}\right)^2\right\} = E\left\{\left(\mathbf{b}_k+\mathbf{B}_k\cdot\mathbf{y}_k-\mathbf{b}_k-\mathbf{B}_k\cdot\boldsymbol{\mu}_{Y,k}\right)^2\right\} = \mathbf{B}_k^2\cdot\boldsymbol{\Sigma}_{Y,k}. \tag{4.17}$$

Now, we identify Eqs. (4.16) and (4.17) with Eqs. (4.3) and (4.4), getting:

$$\mathbf{b}_k = \mathbf{r}_k-\left(\mathbf{B}_k-\mathbf{I}\right)\cdot\boldsymbol{\mu}_{Y,k}, \tag{4.18}$$

$$\mathbf{B}_k^2 = \mathbf{I}+\mathbf{R}_k\cdot\boldsymbol{\Sigma}_{Y,k}^{-1}, \tag{4.19}$$

where $\mathbf{b}_k$ and $\mathbf{B}_k$ are the terms that will be used for feature compensation.

4.5 Feature Compensation

4.5.1 General Formulas

Independently of the strategy chosen for partitioning the feature space, as shown in Eq. (3.12) compensation of distorted features is made for corrector function of class *k* as:

$$\mathbf{x}_k(t) \approx \hat{\mathbf{x}}_k(t) = J^k(\mathbf{y}(t)). \tag{4.20}$$

In Section 2.5.2 we showed that for each class, full-bandwidth features may be estimated from band-limited features as:

$$\hat{\mathbf{x}}_k(t) = \mathbf{B}_k\mathbf{y} + \mathbf{b}_k \tag{4.21}$$

(this expression was derived for Gaussian class-based compensation, but it may also be applied to Phoneme-based compensation).

When compensation matrix $\mathbf{B}_k$ is full, compensation will be multivariate, as each MFCC coefficient is reconstructed using a linear combination of all the others. On the contrary, when it is forced to be diagonal, compensation is univariate. A particular case of compensation is SPLICE [Droppo *et al.*, 2001], where $\mathbf{B}_k = \mathbf{I}$.

In our experiments we also explore an extension of univariate feature compensation in which we try to approximate the missing information as a polynomial series of the distorted MFCC values in order to take into account non-linear dependencies between the full-bandwidth and distorted features. In this case, instead of Eq. (4.21), compensation would be made as [Morales *et al.*, 2005b]:

$$\hat{x}_k^i(t) = b_{k,0}^i + b_{k,1}^i \cdot y_i(t) + b_{k,2}^i \cdot y_i^2(t) + \ldots + b_{k,N}^i \cdot y_i^N(t) = \sum_{n=0}^{N} b_{k,n}^i \cdot y_i^n(t), \tag{4.22}$$

where $b_{p,n}^i$ is the polynomial coefficient of order *n*, class *p* and MFCC coefficient *i*. Also, y_i^n is the value of MFCC coefficient *i* to the *n* power.

4.5.2 Compensation with Gaussian-based Classes

When the ML approach is used for estimation, it was shown in Section 2.5.2 that assuming little overlapping between Gaussians, reconstructed features take the form:

$$\hat{\mathbf{x}}^{ML}(t) = \hat{\mathbf{x}}_q(t), \tag{4.23}$$

where *q* is the most likely class as defined in Eq. (2.25). In our experiments this equation is simplified assuming that the determinant of the matrix of covariances of the conditional probability Γ_k is approximately the same for different classes. Then, *q* in Eq. (4.23) is:

$$q = \arg\max_k \left\{ P(k) \cdot N(\mathbf{y}(t); \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right\}. \tag{4.24}$$

Instead of using a hard decision as suggested by ML, we may use the MMSE solution, so that compensation is made as a combination of corrections from multiple classes, as shown in Eq. (2.26):

$$\hat{\mathbf{x}}^{MMSE}(t) = \sum_{k=1}^{K} P(k|\mathbf{y}(t)) \cdot \hat{\mathbf{x}}_k(t), \tag{4.25}$$

where $P(k|\mathbf{y}(t))$ is the probability of the distorted feature vector belonging to class *k*, normalized to the unit for the sum of all classes.

4.5.3 Compensation with Phoneme-based Classes

When classes are defined in terms of phonetic units estimation is made as in Eq. (4.23) with the contribution of a single corrector class. However, contrary to the case of Gaussian-based classes where the most likely class may be easily identified as in Eq. (4.24), here pre-imputation is not straightforward; observations have to be identified with a phonetic class prior to compensation and this type of information is obviously unavailable in ASR tasks because finding the phoneme sequence is precisely the problem to solve. In the following sections we propose a number of practical implementations addressing this *chicken and egg problem* [Morales *et al*., 2005a].

4.5.3.1 Oracle Phoneme-specific Compensation

Oracle Phoneme-specific correction employs time-aligned phonetic transcriptions of test data in order to apply the appropriate corrector function to each frame. Phonetic transcriptions for test data will not be available in real scenarios, but evaluation using an oracle solution provides us with an upper bound on performance for Phoneme-based compensation.

4.5.3.2 General Compensation

Here all the distorted feature space is pooled together in a single class. Thus, strictly speaking it should not be termed Phoneme-based partitioning (no partitioning at all takes place), but it may be understood as a simplification of Phoneme-based approaches, with the advantage that phonetic labels are not needed anymore. General feature compensation is a global transformation of the feature space and its generality and simplicity make it comparable to CMN.

4.5.3.3 Two-stage Compensation

Here we propose compensation in two-passes, as shown in Figure 4.4. The first stage consists of General compensation, as in the previous section, followed by a pass of the ASR phonetic recognizer that provides a transcript for the test utterance. In the second stage, the generated phonetic transcription is employed for Phoneme-specific correction as in the oracle method. An extension is to generate a transcript lattice or several candidate transcripts in the first stage, allowing for different Phoneme-specific corrections of the distorted feature vector. In this case, the ASR final decision is chosen as the one with maximum probability likelihood.

4.5.3.4 Compensation Embedded in the Decoder Module

A natural solution to the pre-imputation problem in Phoneme-based compensation is to introduce feature compensation into the decoder module. Compensation is applied embedded in the Viterbi decoding process prior to likelihood computation for each pair observation-acoustic model. For example, for a given observation, Phoneme-specific compensation for class /h/ is applied before computing the likelihood probability against states that belong to acoustic model /h/. Similarly, for acoustic model /p/ compensation is applied with corrector functions trained for class /p/, etc. The goal

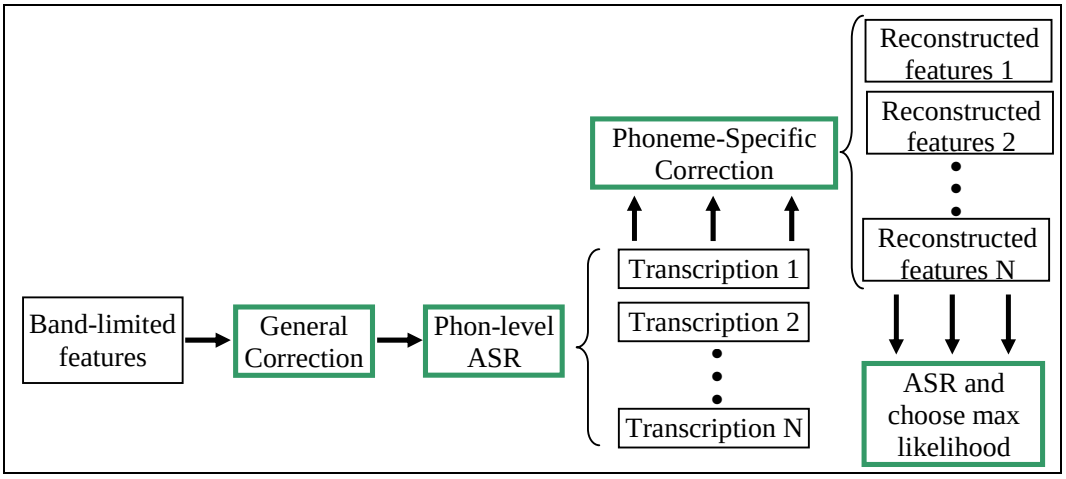


Figure 4.4 Two-stage Phoneme-based feature compensation.

is for the corrector function to behave as a *secret code*: starting from a highly mismatched situation (limited-bandwidth speech and full-bandwidth models) the correct compensation will bring the feature vector to the best possible match, while compensations from incorrect phonemes will bring features to an undetermined and still mismatched location in the feature space. In other words, the best acoustic model should be that for which better probability likelihood is obtained for a given frame after the corresponding Phoneme-specific correction is applied.

Compared to previously discussed Phoneme-based compensation approaches, compensation embedded in the decoder module does not require pre-imputation of speech frames to phonemes; instead, Phoneme-based compensation is applied using the corresponding Phoneme-specific corrector function prior to likelihood computation. In terms of computational efficiency, this method requires compensation of the whole utterance using each available corrector function. However, it removes the necessity of multiple recognition passes, required by two-stage compensation.

We note the similarity between this approach and Constrained Linear Adaptation, as described in Section 2.6.3.1.

4.5.4 Multi-environment Compensation

In previous sections it was assumed that the type of distortion affecting speech was fixed and specific corrector functions had been trained to compensate its effect. However, in practical situations, a single system may receive speech coming from a variety of sources, each of which may present a different band-limitation and in many occasions no a-priori knowledge on the particular distortion will be available. For example, speech downloaded from the Internet may have been recorded at different sampling rates and different corrections should be used for each of them.

A possible solution is the introduction of a frequency analysis module prior to feature compensation (or prior to the parameterizer module itself). Alternatively, we propose an automatic distortion identification method embedded in our general feature compensation framework [Morales *et al.*, 2006]. The idea is again that even if the frequency information is mixed in the cepstral representation of speech, it nevertheless leaves a distinct footprint in the complete feature vector that can be used to identify the type of distortion.

In Eq. (2.20) the probability distribution of band-limited features was represented as:

$$p(\mathbf{y}) = \sum_{k=1}^{K} p(\mathbf{y}|k) \cdot P(k) = \sum_{k=1}^{K} N(\mathbf{y}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \cdot P(k), \tag{4.26}$$

where K is the set of classes partitioning a single distorted space. This may be extended now to the case of multiple distorting environments as:

$$p(\mathbf{y}) = \sum_{d=1}^{D} \sum_{k^d=1}^{K^d} p(\mathbf{y}|k^d) \cdot P(k^d|d) \cdot P(d) = \sum_{d=1}^{D} \sum_{k^d=1}^{K^d} N(\mathbf{y}; \boldsymbol{\mu}_{k^d}, \boldsymbol{\Sigma}_{k^d}) \cdot P(k^d|d) \cdot P(d). \tag{4.27}$$

In Eq. (4.27) d is the band-limitation in a set of D possible distortions, $P(d)$ is the a-priori probability of finding distortion d and $P(k^d|d)$ the probability of a particular Gaussian in the set of Gaussians of distortion d. The final expression may be simplified by converting the sets of classes $\Psi^d = \{k^d\}$ from each particular environment d into a super set of classes from all environments:

$$\Psi = \{\psi^d\} = \Psi^1 \cup \Psi^2 \cup \ldots \cup \Psi^D. \tag{4.28}$$

Thus, Eq. (4.27) is transformed into a new equation similar to Eq. (4.26), where the set of classes originally belonging to the set of a fixed distortion is substituted by a superset of classes from multiple distortions:

$$p(\mathbf{y}) = \sum_{\psi} N(\mathbf{y}; \mu_\psi, \Sigma_\psi) \cdot P(\psi). \tag{4.29}$$

In synthesis, classes are created independently for each possible distortion and they are combined into a super set of classes for automatic multi-environment compensation. In our experiments we call this *Individual Environment Class combination, IEC*. Alternatively, training data from multiple environments may be pooled together from the start and classes will then be trained in a multi-distortion mode (similarly to what is done in training speaker-independent acoustic models, where data from all available speakers is pooled together). We call this approach *Multi-Environment Class creation, MEC*. In principle, MEC should result in a smoother representation of the multi-environment space; compensation classes may be created grouping together data from multiple distortions, where appropriate, while the method should converge to IEC if multi-distortion classes are inappropriate. However, IEC allows for explicit distortion-based partitioning of the space because data from different distortions is kept separate. Experiments in Section 8.4.1 demonstrate the superiority of IEC, as it results in a partitioning method directed towards a characterization of the feature space based on the type of distortion. This shows the convenience of combining a data-driven approach (automatic partitioning) with a-priori knowledge (information on the distortion affecting each utterance).

4.6 Smoothing the Compensation

One limitation of the proposed compensation methods is that contiguous frames identified as belonging to different clusters of data are consequently corrected with different sets of corrector functions, and this may lead to sudden changes in the correction applied to consecutive frames. Sometimes these abrupt changes may be motivated by an actual event in the speech signal, such as the burst in a stop, but in many occasions, abrupt changes in the reconstructed feature vector are caused by an error in the

a-priori choice of phoneme or Gaussian cluster (typically the impact of errors will be more significant in ML than in MMSE compensation). Classification errors normally have a small duration, producing high-frequency noise in the correction applied to the features, a situation that is clearly undesirable and could degrade overall ASR performance (this is especially harmful for dynamic features obtained by regression of reconstructed static features). We define the correction value for MFCC coefficient *i* as the difference between the input band-limited value and the reconstructed value:

$$Correction_i(t) = \hat{x}_i(t) - y_i(t). \tag{4.30}$$

One way of reducing the effect of artificial noise caused by errors in cluster classification is to smooth the sequence of correction values. Several types of low-pass filtering were evaluated for smoothing the correction sequence; initially an IIR filter with two poles in the same position in the Z-plane, similar to that used by Droppo *et al*. [2001], was used:

$$H(z) = \frac{a}{\left(z^{-1} - p^{-1}\right)(z - p)} = \frac{-a \cdot p \cdot z^{-1}}{\left(1 - p \cdot z^{-1}\right)^2}, \tag{4.31}$$

where *a* is the amplification factor that should be adjusted for a normalized filter gain and the root location *p* determines the cut-off frequency of the filter. However, other types of filters, including non-linear filters, such as median filters proved to be more successful, perhaps due to their ability to remove short-term deviations (lasting one or a few frames) while maintaining longer abrupt changes that may be more representative of actual transitions between acoustic events.

In Figure 4.5 we show plots corresponding to ideal compensation (this is the oracle solution computed as the difference between distorted and full-bandwidth features), compensation using Gaussian class-based univariate ML compensation and the same compensation after smoothing using a median filter and a linear filter as in Eq. (4.31). The plots show that the compensation sequence contains some high-frequency variations that are probably due to errors in class imputation rather than to changes in the speech features. Smoothing with a linear filter effectively removes these rapid variations, but cannot follow abrupt and sustained changes due to changes in the speech content. Smoothing with a median filter is a better option since it removes the rapid variations while being able to follow abrupt and sustained changes. In the experimental section smoothing provides consistent improvements in ASR accuracy, indicating the convenience of such approach for removing artificial rapid variations that probably affect more seriously dynamic features.

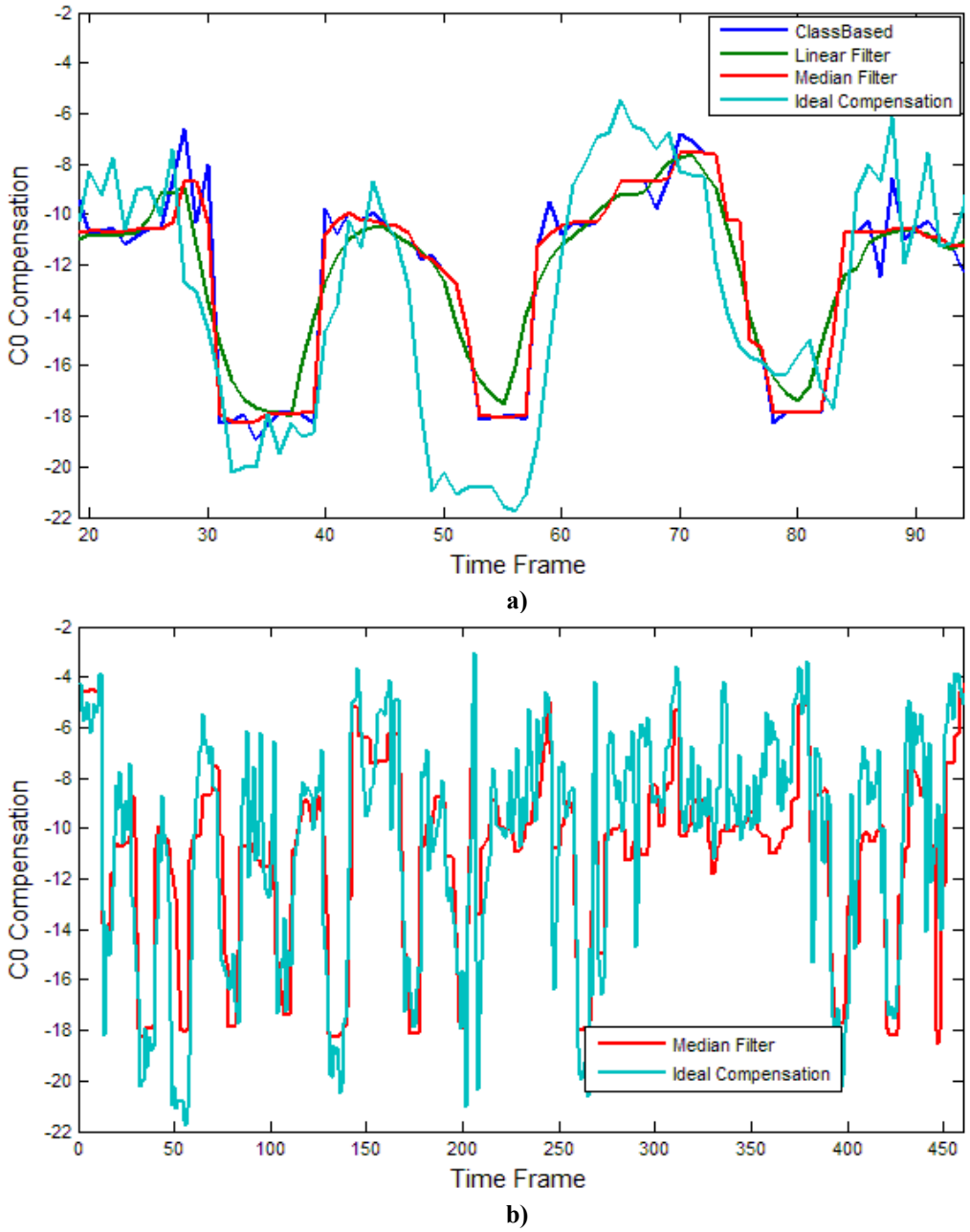


Figure 4.5 Analysis of the evolution of the correction value for MFCC coefficient C0, frame by frame for a given utterance. Subplot a) is a small-duration segment showing the ideal correction values (the actual difference for each frame between the full-bandwidth and LP4kHz MFCC coefficients), class-based generated correction values as well as the result of smoothing the sequence of correction values using a linear filter and a median filter. Subplot b) shows the whole utterance values of ideal and class-based median-smoothed correction values.

5

Speech Tools and Experimental Framework

This chapter describes the tools and methodology for system training and performance evaluation of the experiments conducted in this Thesis. The primary aim in our work is improving speech recognition performance when training data is full-bandwidth and clean, and test data is band-limited (and possibly subject to noises). As we are interested in the improvement rather than in absolute performance and as the techniques proposed in this Thesis can be applied to sophisticated recognition systems as well as simpler ones, we decided to keep our experimental prototype system simple. Therefore, most of our experiments use a context-independent phoneme-based HMM recognizer with a bigram phonetic LM. State-of-the-art recognizers use more complex phonetic models (context-dependent), vocabularies that limit the amount of possible phonetic sequences and word grammars or word LMs that limit the amount of possible word sequences. These add-ons normally improve accuracy, but necessarily increase the effort needed for development, tuning and evaluation. Moreover, dedicated vocabulary lists, word grammars and LMs are application-dependent, which complicates evaluation of results. However, we expect the results obtained in our simple framework to generalize well to other more sophisticated frameworks, and for the purpose of validating this point we have conducted a few experiments on other systems.

5.1 The HTK Development Toolkit

HTK stands for Hidden Markov Models ToolKit [Young *et al.*, 2005]. It is a set of modules and source code files in the C programming language, specifically designed for experimental development of ASR systems based on HMMs. HTK is probably the most widely used platform in the speech recognition research community. It shares the principles of commercial systems such as IBM's ViaVoice [ViaVoice], or Dragon Naturally Speaking [Dragon] (both currently distributed by Nuance), and performance is also comparable (with the difference that commercial systems are finished products and are highly optimized, while HTK is only a research and development platform). Other development

platforms used for research and development (even in production systems) are Sphinx, from Carnegie-Mellon University [Sphinx] and SONIC from Colorado University [SONIC].

In this Thesis HTK is used for signal pre-processing, parameterization, acoustic model training, model adaptation, LM development, speech recognition tests and performance evaluation. Also, the sources of the decoder module were modified for feature compensation embedded in the decoder module (Section 7.1.4).

5.2 Signal Parameterization

Throughout this Thesis experiments employ the following front-end, unless otherwise specified. Pre-processing uses pre-emphasis filtering ($\alpha = 0.97$) and the signal is analyzed using Hamming windows of length 25 ms with 10 ms window shifts. Thirteen MFCC coefficients including C0 and their respective first and second order derivatives (for a total of 39 features) are computed from a filter-bank of 26 triangular filters uniformly distributed in the mel-frequency scale in the region 0-8 kHz. Delta coefficients (derivatives) are computed following the HTK definition [Young *et al.*, 2005; pp. 63-64]:

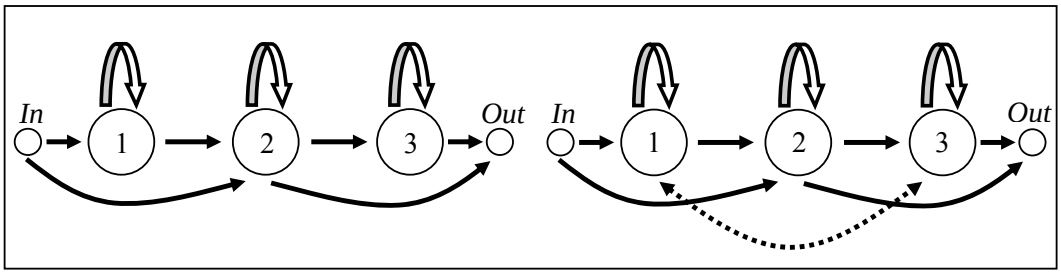
$$\Delta C_t = \frac{\sum_{\theta=1}^{\Theta} \theta \cdot \left(C_{t+\theta} - C_{t-\theta} \right)}{2 \cdot \sum_{\theta=1}^{\Theta} \theta^2}, \tag{5.1}$$

where we set $\Theta = 2$.

5.3 Acoustic Models

Our speech recognizer employs context-independent HMM acoustic models, with a basic topology as shown in Figure 5.1 a). It is a 3-state, left-to-right model (Bakis topology), with additional transitions from the initial non-emitting state, *In*, to the second emitting state, and from the second emitting state to the final non-emitting state, *Out* (these additional transitions allow for better modeling of very short productions typically found in conversational speech). This is similar to a topology suggested by Rabiner [1989] and is also recommended in [Young *et al.*, 2005; pp. 30-34]. Initial and final non-emitting states in each model (represented in the figure with small circles *In* and *Out*) are imposed for model connection in HTK and do not participate in decoding [Young *et al.*, 2005; pp. 30-34].

Evaluation in English employs TIMIT and its derived databases (NTIMIT, CTIMIT, UAM-TIMIT), described in Section 5.7.1. We train the set of 45 different phoneme events suggested in [Lee



a) **b)**

Figure 5.1 Model topology for regular phoneme models (a), and special topology for silence models with extra transitions (b). Non-emitting states (small circles *In* and *Out*) are necessary for model concatenation.

Phon	TIMIT Abs (%)	Modif. TIMIT Abs.	Conv. Eng. %	Phon	TIMIT Abs (%)	Modif. TIMIT Abs.	Conv. Eng. %	Phon	TIMIT Abs (%)	Modif. TIMIT Abs.	Conv. Eng. %
iy	6953 (5.05)	4626	3.90	**ow**	2136 (1.55)	1653	1.98	**d**	3548 (2.58)	2432	3.33
ih	5051 (3.67)	4248	6.13	**l**	5801 (4.22)	4425	3.77	**dx**	2709 (1.97)	1864	1.07
eh	3853 (2.80)	3277	3.53	**el**	951 (0.69)	951	0.25	**g**	2017 (1.47)	1191	1.18
ae	3997 (2.90)	2292	2.44	**r**	6539 (4.75)	4681	3.87	**p**	2588 (1.88)	2588	1.79
ix	8642 (6.28)	7370	0.70	**y**	1715 (1.25)	995	1.09	**t**	4364 (3.17)	3948	5.88
ax	3610 (2.62)	3535	7.65	**w**	3140 (2.28)	2216	2.77	**k**	4874 (3.54)	3794	3.10
ah	2306 (1.68)	2266	1.56	**er**	5453 (3.96)	4138	2.23	**z**	3773 (2.74)	3682	2.75
uw	2463 (1.79)	1952	1.20	**m**	4027 (2.93)	3566	3.06	**zh**	151 (0.11)	149	0.09
uh	535 (0.39)	500	0.91	**n**	8039 (5.84)	6896	6.87	**v**	1994 (1.45)	1994	1.74
ao	2940 (2.14)	1865	1.05	**en**	723 (0.53)	630	0.61	**f**	2216 (1.61)	2215	1.55
aa	3064 (2.23)	2256	1.79	**ng**	1368 (0.99)	1220	1.08	**th**	751 (0.55)	745	0.70
ey	2282 (1.66)	2271	1.69	**ch**	822 (0.60)	820	0.54	**s**	7475 (5.43)	6176	4.61
ay	2390 (1.74)	1934	2.97	**jh**	1209 (0.88)	1013	0.56	**sh**	2238 (1.63)	1317	0.56
oy	684 (0.50)	304	0.08	**dh**	2826 (2.05)	2376	3.14	**hh**	2111 (1.53)	1660	1.31
aw	729 (0.53)	728	0.64	**b**	2181 (1.58)	2181	1.90	**ax_h**	375 (0.27)	357	N/A

Table 5.1 Phonetic distribution in the training partition of TIMIT and conversational US English. Column *TIMIT* shows absolute values (relative in parenthesis). *Modif. TIMIT* shows absolute occurrences for a modified version of TIMIT where repeated prompts are removed. *Conv. Eng.* shows relative occurrence in casual conversational English, derived from [Mines *et al.*, 1978]. Phoneme /ax_h/ is not considered in the reference, so no information is available on its frequency in conversational English.

and Hon, 1989], and widely accepted when TIMIT is used (Table 5.1). Following the suggestion in the same reference, for ASR evaluation we map the 45 phonemes to a subset of 38.

For evaluation in Castilian Spanish we used Albayzin (described in Section 5.7.2) and a minimal set of the 23 phonemes that are strictly required to recognize words in Castilian Spanish was trained. This phoneme set was evaluated in phonetic recognition experiments on read and spontaneous speech [Toledano *et al.*, 2005] and has also been used in the development of the largest phoneme and syllable frequency inventory for spontaneous Castilian Spanish [Moreno *et al.*, 2006]. Table 5.2 shows the phoneme set used as well as absolute and relative frequencies in Albayzin and spontaneous Castilian Spanish. Original labeling in Albayzin uses a set of 31 allophones that was mapped to our subset of 23 phonemes.

In addition to phoneme models, three silence models are trained in each set of HMMs: starting silence, ending silence and short-pause. The topology of the first two is identical to that of phoneme models, with additional transitions between emitting states 1 and 3 (Figure 5.1 b)), allowing for extra flexibility in the process of modeling silence, which is close to a random acoustical event. The short-pause model has only one emitting state, so as to better modeling its typical short duration.

Phon	Albayzin Abs (%)	Conv. Spa. %	Writt. Spa. %	Phon	Albayzin Abs (%)	Conv. Spa. %	Writt. Spa. %	Phon	Albayzin Abs (%)	Conv. Spa. %	Writt. Spa. %
a	30384 (15.58)	12.27	12.89	**i**	11568 (5.93)	7.22	7.59	**p**	4056 (2.08)	2.74	2.73
b	10464 (5.36)	2,50	2.55	**x**	1080 (0.55)	0.62	0.77	**r**	888 (0.46)	0.42	0.99
θ	2712 (1.39)	1.52	2.00	**k**	6936 (3.56)	4.49	3.80	**ɾ**	9288 (4.76)	5.12	6.19
tf	1080 (0.55)	0.30	0.18	**l**	12816 (6.57)	4.51	5.46	**s**	13560 (6.95)	8.11	7.33
d	7320 (3.75)	4.36	5.42	**m**	6000 (3.08)	3.15	2.76	**t**	7728 (3.96)	4.52	4.31
e	23712 (12.16)	15.12	12.74	**n**	13872 (7.11)	7.05	7.09	**u**	6888 (3.53)	3.14	3.04
f	960 (0.49)	0.50	0.92	**ŋ**	1008 (0.52)	0.19	0.31	**ʎ**	2664 (1.37)	0.83	0.53
g	2520 (1.29)	0.91	1.04	**o**	17544 (8.99)	10.38	9.32				

Table 5.2 Phonetic distribution in the training partition of Albayzin, conversational and written Castilian Spanish. Column *Albayzin* shows absolute values (relative in parenthesis). Phoneme frequencies in Castilian Spanish derived from [Moreno *et al.*, 2006].

5.4 Language Models

Most of the experiments in this Thesis use bigram phoneme-based LMs. Usage of such simple LMs keeps our experiments application-independent, does not incur in significant increase in recognition time and boosts ASR rates (compared to use of no LM at all), making our results more significant and showing that the performance improvement with our proposed signal processing approaches is complementary and not overlapping with the gains obtained by using LMs.

In our experiments we use back-off bigrams. This type of LM calculates the probability of any phoneme following any given previous phoneme based on the frequency of occurrence of that particular phoneme sequence in the training material (phonemes are substituted by words as the basic unit, in word-based LMs). A little probability is discounted from each observed sequence of phonemes to allow recognition of transitions unseen during training, and these unseen transitions are backed-off to the unigram probability.

Phoneme-based grammars in TIMIT and Albayzin were trained using the phonetic transcriptions in the training partition. In Albayzin, no overlapping exists between prompts in the training and testing partitions, so the method is perfectly valid. On the contrary, in TIMIT there is an overlapping imposed by the design of the corpus as 2 phrases are spoken by each speaker (SA utterances; see Section 5.7.1). Thus, by using all the prompts in the training partition, the most likely phonetic sequences in prompts SA1 and SA2 will be favored, while others will be slightly penalized. In Table 5.3 we show accuracy for a phonetic recognizer and two types of LM; the first one trained with all prompts in the training partition ad the second one ignoring SA utterances. As it could be expected, using all training data for training the LM favors recognition of SA files and penalizes no-SA files. It may also be observed that performance is always better for SA files, because acoustic models for the phonemes in these utterances have more training data with exactly the same co-articulation. Therefore, for evaluation we could have left out SA utterances. Nevertheless, after realizing that performance is not too importantly affected by the LM and that in our experiments we are interested in relative performance between different approaches and not absolute performance, we decided to use all the test partition for evaluation (this favors comparison with other published works that typically use this same partition) and an LM trained with all prompts in the training partition.

Training word-based LMs we used both training and testing prompts for computing the probability of word sequences. Using information from the testing condition for training is admittedly unfair, but we consider it necessary in the case of word-based LMs; leaving out prompts in the test partition and training an LM with the very limited amount of text in the training partition would have strongly and artificially penalized the actual transcriptions. On the contrary, using test transcriptions for

Test data →	**All files**		**SA files**		**no SA files**	
Training LM ↓	**%Corr.**	**%Acc.**	**%Corr.**	**%Acc.**	**%Corr.**	**%Acc.**
All training	75.40	71.18	88.44	85.72	72.28	67.70
no SA	74.86	70.77	84.26	81.37	72.61	68.24

Table 5.3 ASR performance for a phonetic recognizer with context-independent models and two different LMs, for TIMIT. The first row shows results using a LM trained with all prompts in the test partition, as is used in the rest of this Thesis. The second row shows results for a LM trained without the SA files. Performance is given for all files as well as divided according to whether test files are SA or not.

training the LM makes our experiments with word-based grammars somewhat optimistic. In any case, our interest is not in absolute performance itself, but in the comparison of different results using the same LM, so the effect of the LM will approximately cancel out in the comparison.

5.5 Training and Test Strategies

The training partitions in TIMIT and Albayzin are used for training speaker and gender-independent models for both corpora (different model sets for each corpus). Baum-Welch is used to estimate model parameters and Gaussian mixtures are incremented one by one until the desired final number is attained (in our experiments we use 45 Gaussian mixtures per model, 15 per state).

Tests are conducted with the whole test partitions unless otherwise specified. For model adaptation techniques, band-limited features are passed to a decoder module that uses adapted models. On the contrary, when feature compensation is used distorted features are compensated and then passed to a decoder where models are trained with clean and full-bandwidth speech.

5.6 Scoring Measures

Scoring uses a dynamic programming approach in order to find the best possible alignment between a recognizers' output transcript and a reference label file [Young *et al.*, 2005; pp. 277-280]. The following two standard scoring measures are employed:

- Percent Correct is:

$$\%\text{Correct}=100\% \cdot \frac{C}{N} = 100\% \cdot \frac{N - S - D}{N}, \tag{5.2}$$

- and Accuracy is defined as:

$$\%\text{Accuracy}=100\% \cdot \frac{C - I}{N} = 100\% \cdot \frac{N - S - D - I}{N}, \tag{5.3}$$

where C represents correct hypotheses (words in word-based recognition, or phonemes in phoneme-based recognition), I are insertions, S represents substitutions, D stands for deletions and N is the total number of units in the reference transcripts (i.e., $N = C + S + D$).

For speech recognition, the weights given to the LM and penalty insertion factor are tunable parameters. A recognizer may then be set to increment the number of correct units at the cost of incrementing also the number of insertions. Therefore, a compromise must be found, and accuracy as defined in Eq. (5.3) typically represents a more descriptive performance measure.

5.7 Speech Corpora

Following is a description of the two corpora used for evaluation in this Thesis: TIMIT (and other corpora derived from it) for English and Albayzin for Castilian Spanish.

5.7.1 TIMIT and Related Databases

The DARPA TIMIT Acoustic-Phonetic Continuous Speech Corpus (commonly known as TIMIT) is one of the most widely used databases in the English language [Fisher *et al.*, 1986]. It consists of short

sentences of read speech under clean conditions, spoken by 630 speakers covering the dialectal variability in the USA. Each speaker reads 10 sentences belonging to 3 different categories: SA are 2 sentences read by every speaker and designed to capture dialectal differences. SX are phonetically diverse sentences designed to provide a good coverage of pairs of phones, with extra occurrences of phonetic contexts thought to be either difficult or of particular interest; a total of 450 different prompts exist, each being read 7 times by different speakers and each speaker reading 5 of them. Finally, SI sentences are used only once in the whole database and each speaker reads 3 of them (1890 different prompts exist).

In our experiments we follow the suggestion of 462 mixed-genre speakers for training and 168 for test. Training of the acoustic models is done using all 4620 files in the training partition, including the two SA utterances by each speaker. This could lead to unbalanced training of those phonemes present in SA utterances, compared to others non-present (Table 5.3), but we decided to keep them in order to increase the dialectal richness of our models. In Table 5.1 we show the relative occurrence of phonemes in TIMIT, as well as that observed in real conversational English, derived from a study by Mines *et al.*, [1978]. There only seems to be a remarkable difference on phoneme occurrences for /ih/ and /ix/ that could be due to different labeling conventions.

A number of corpora derived from TIMIT are available for research [LDC]. In this Thesis we use NTIMIT [Jankowski, 1991], the telephone version of TIMIT. We also employ UAM-TIMIT [Morales *et al.*, 2007b], a variation on NTIMIT we created by sending the original TIMIT through the telephone network in a single call (Appendix A; we are currently in contact with the Linguistic Data Consortium, LDC, for publication of this corpus). Finally, several corpora were derived from TIMIT by applying artificial band-pass and low-pass filters to the original data.

5.7.2 Albayzin

This database of Castilian Spanish actually consists of three different corpora, of which we only used in this Thesis the acoustic-phonetic corpus. This corpus is similar in design and recording conditions to TIMIT. For training we use a subset of the training partition comprising 140 speakers, each pronouncing 25 utterances randomly chosen from a list of 200 utterances. The test partition is the same proposed in the corpus documentation, consisting of 40 speakers, each pronouncing 50 utterances from a list of 500 utterances and there is no overlapping between phrases in the training and test corpora. The set of phonemes employed is given in Table 5.2.

Albayzin was used for the purpose of validating our results in a wider range of situations, by testing the same solutions used for TIMIT in a different database and language. This shows that the improvements obtained are not coincidental or language-dependent.

6
Model Adaptation Techniques

Model adaptation is a widely used strategy for compensation of signal variability due to speaker characteristics, noise or convolutional distortions, etc. Model adaptation strategies were initially developed for adaptation of speaker-independent acoustic models to particular users [Grenier, 1980] [Shikano *et al.*, 1986]. However, the same approaches used for speaker adaptation can be applied to environment adaptation [Matassoni *et al.*, 2000]. In this Thesis model adaptation is used for two different purposes: firstly, it gives us a measure of comparison with our proposed feature compensation methods and secondly, it may be used in combination with feature compensation for improved system robustness. In addition to model adaptation, many experiments in Chapters 7 and 8 include also results with new models trained from scratch using data from the target environment. We call these *matched models* and can be considered as the most extreme case of model adaptation.

In this chapter we give extensive results for ASR performance with the goal of understanding the influence of tunable parameters in the two most extended model adaptation algorithms found in the literature: MLLR [Leggetter and Woodland, 1995] [Leggetter, 1995] and MAP [Gauvain and Lee, 1994]. However, it is not the goal of this Thesis to explore new model adaptation techniques so the cited algorithms are used in their usual forms.

6.1 MLLR Adaptation

MLLR is a technique that estimates a set of linear transformations for the parameters of a multi-mixture HMM system [Legetter and Woodland, 1995]. For each mixture in a set of HMMs the original vector of mean coefficients ξ, is modified as:

$$\boldsymbol{\mu} = \mathbf{W} \cdot \xi, , \tag{6.1}$$

where $\xi = \left[1 \; \mu_1 \; \mu_2 \; \ldots \; \mu_N\right]$, $\boldsymbol{\mu}$ is the new vector of means and $\mathbf{W}$ is the transformation matrix, defined as:

$$\mathbf{W} = \left[\mathbf{b} \; \mathbf{A}_{NN}\right], \tag{6.2}$$

where **b** is an offset bias vector and $\mathbf{A}_{NN}$ a squared matrix with size *N x N* (*N* being the dimension of Gaussian mixtures, i.e. the dimension of the feature space).

A similar expression for MLLR adaptation of the covariance matrix exists [Gales *et al.*, 1996], but in our work we only used mean vectors' adaptation as in the original formulation by Leggetter and Woodland [1995]. In this regard, an informal experiment was conducted on adaptation data from TIMIT LP4kHz data, that produced no significant differences: using filtered test data and adapted models, accuracy was 68.38% for means-only adaptation and 68.50% for means and covariance matrix adaptation.

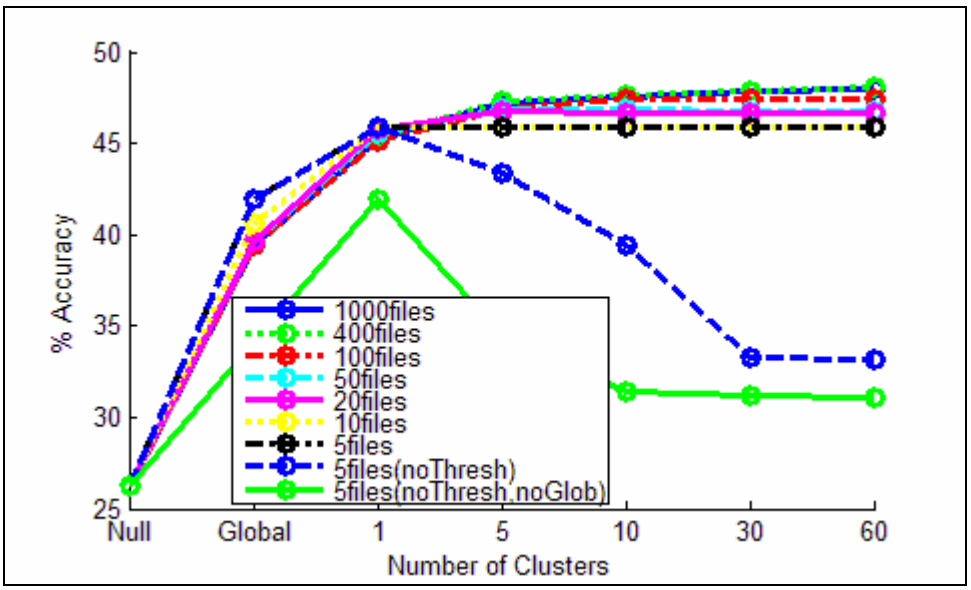
When adaptation data is scarce adaptation of individual mixtures in HMM acoustic models would cause overfitting of the adaptation matrixes to the adaptation data (even for some mixtures no adaptation data at all would be available), reducing performance on test data unseen during adaptation. Therefore, in MLLR, output distributions are clustered together into groups of Gaussians that are assumed to follow a similar transformation as a result of the new condition. Clustering is made through a regression class tree and a minimum cluster occupancy threshold is set, so that clusters below a minimum amount of data are adapted with data from their parent nodes in the tree [Young *et al.*, 2005; pp. 132-143].

The first stage in adaptation is a frame-to-state alignment of the adaptation data from the target environment. However, alignment uses prior models (in our case full-bandwidth models), incurring in misalignments of adaptation data and resulting in sub-optimal adaptation. In order to reduce this effect, it is possible to apply global MLLR (single-cluster MLLR) prior to regular cluster-based MLLR.

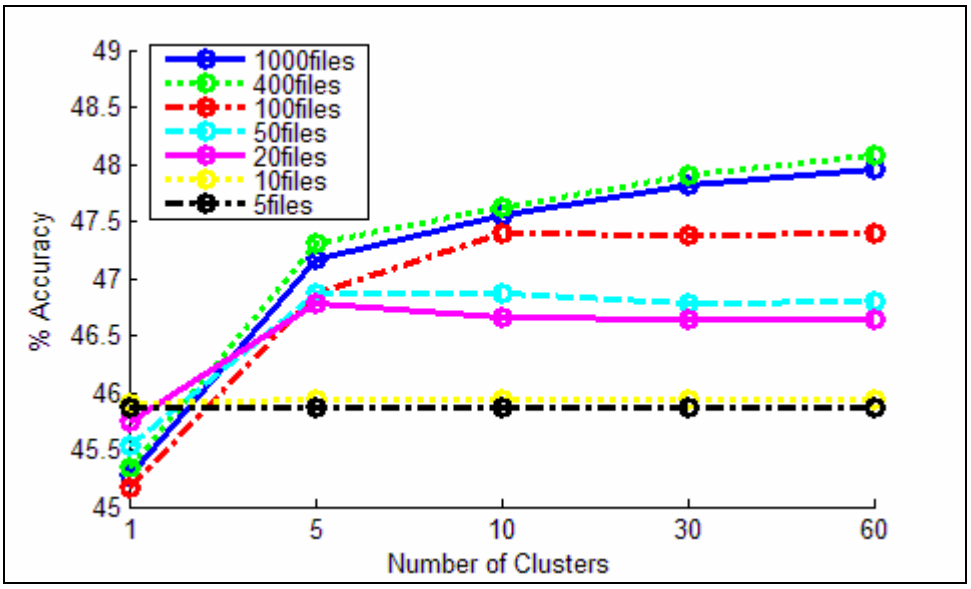
In Table 6.1 and Figure 6.1 we show ASR accuracy for models trained with TIMIT data and adapted for recognition of test data from NTIMIT with different amounts of adaptation data. The first two columns in the Table show the convenience of global MLLR as a first step of adaptation to assure proper data alignment. Rows 2 and 3 compare performance of global MLLR only and global MLLR followed by MLLR with only one cluster of data. As global MLLR is actually MLLR with one cluster, the difference between Rows 2 and 3 is whether one or two passes of global MLLR are made. The significant accuracy increase in the latter shows the advantage of an initial pass of global MLLR. Comparison between Columns 2 and 3, shows the importance of setting a minimum occupancy

Number of Adapt. Files → / **Mode ↓**	**5 no thresh no Global**	**5 no thresh**	**5**	**10**	**20**	**50**	**100**	**400**	**1000**
No adapt	26.27	26.27	26.27	26.27	26.27	26.27	26.27	26.27	26.27
Global MLLR	----	41.92	41.92	40.64	39.62	39.60	39.42	39.63	39.41
1 cluster	**41.92**	**45.87**	**45.87**	45.91	45.75	45.54	45.16	45.35	45.27
5 clusters	34.49	43.40	45.87	**45.94**	**46.78**	**46.87**	46.86	47.30	47.17
10 clusters	31.41	39.41	45.87	45.93	46.66	46.86	**47.39**	47.62	47.56
30 clusters	31.15	33.27	45.87	45.93	46.64	46.78	47.37	47.90	47.81
60 clusters	31.12	33.13	45.87	45.94	46.64	46.79	47.39	**48.08**	**47.95**

Table 6.1 Percent accuracy with MLLR adaptation for different numbers of clusters and amount of adaptation files. First row is No adaptation and second row is global MLLR. Columns show the number of adaptation files randomly chosen from NTIMIT. The first 2 columns use no cluster occupation threshold, while the rest use a 700-frames threshold. All results except the first column use global MLLR prior to cluster-based MLLR. For each amount of training data the best result is highlighted.



a)



b)

Figure 6.1 Accuracy with TIMIT acoustic models MLLR-adapted for NTIMIT data. Plots show percent accuracy vs. different MLLR settings as described in Table 6.1. Subfigure b) is a zoom in the area of best performance. Numerical values are given in Table 6.1.

threshold for regression tree clusters when data is scarce (the threshold is set to 700 frames in our experiments). The rest of columns use the same minimum occupancy threshold, which causes performance convergence for fixed and small amounts of data and large number of clusters. It is also observed that performance reaches its peak for approximately 400 adaptation files and 30 clusters, although more adaptation data or clusters do not affect performance negatively. A somehow surprising observation is that when only global MLLR is used performance decreases for more adaptation data. For example from 5 to 10 files there is more than 1% absolute difference. However, this seems to be an effect of the random election of adaptation files and not a general tendency: in effect, the same type of global MLLR adaptation using files 6 to 10 in the adaptation list (those files contained in the list of 10 adaptation files and not in the list of 5 files) generated 37.15% accuracy compared to 41.92% using the first 5 files. When more adaptation data is used, performance converges gradually.

6.2 MAP Adaptation

MAP is a Bayesian adaptation strategy that combines prior models with accumulation statistics from the new data to obtain adapted models. Bayesian adaptation was first proposed for speech by Brown *et al.* [1983], but the usual reference for adaptation of the probabilities of emission of HMM models is the formulation by Gauvain and Lee [1994]. Specifically, the vectors of means of Gaussian mixtures are modified as:

$$\boldsymbol{\mu}^{MAP} = \frac{N}{N+\tau} \cdot \overline{\boldsymbol{\mu}} + \frac{\tau}{N+\tau} \cdot \boldsymbol{\mu}^{prior}, \tag{6.3}$$

where $\boldsymbol{\mu}^{MAP}$ is the adapted Gaussian mean, $\overline{\boldsymbol{\mu}}$ the statistical mean from the new data and $\boldsymbol{\mu}^{prior}$ the original mixture's vector of means. N is the occupation likelihood of the adaptation data for the mixture and τ is the weighting parameter for the prior knowledge.

Unlike MLLR, MAP adaptation does not make use of mixture clustering and each mixture is adapted using only data imputed to it. In order to avoid under-adaptation due to data scarcity, in this Thesis MAP adaptation uses MLLR-adapted models as a prior, so that the good properties of each adaptation method are combined (also assuring proper data alignment for MAP). In Table 6.2 and Figure 6.2 we show performance for different values of τ, the weighting of the a-priori knowledge and different amounts of training data, for models trained with TIMIT and adapted for NTIMIT.

As expected, when large amounts of data are available for adaptation, the weight of prior knowledge may be minimized (performance is stable for values smaller than τ = 20). On the contrary, when data is scarce more weight should be given to prior models.

Number of Adapt. Files → / **Prior Weight τ ↓**	**5**	**10**	**20**	**50**	**100**	**400**	**1000**
0	34.71	37.55	42.04	45.60	47.84	49.53	49.77
4	46.28	45.38	46.62	47.15	48.41	**49.72**	**49.83**
10	**46.30**	46.09	46.88	47.29	**48.44**	49.71	49.74
20	46.16	**46.12**	**46.99**	**47.30**	48.43	49.56	49.70
100	45.97	45.99	46.75	47.05	48.02	49.23	49.46
1e3	45.88	45.93	46.67	46.84	47.53	48.42	48.62
1e4	45.87	45.94	46.65	46.79	47.39	47.95	47.97
1e5	45.87	45.93	46.64	46.79	47.37	47.91	47.81

Table 6.2 Performance of MAP adaptation for different values of prior knowledge weight and data availability. The best-performing value of τ for each fixed amount of training data is highlighted.

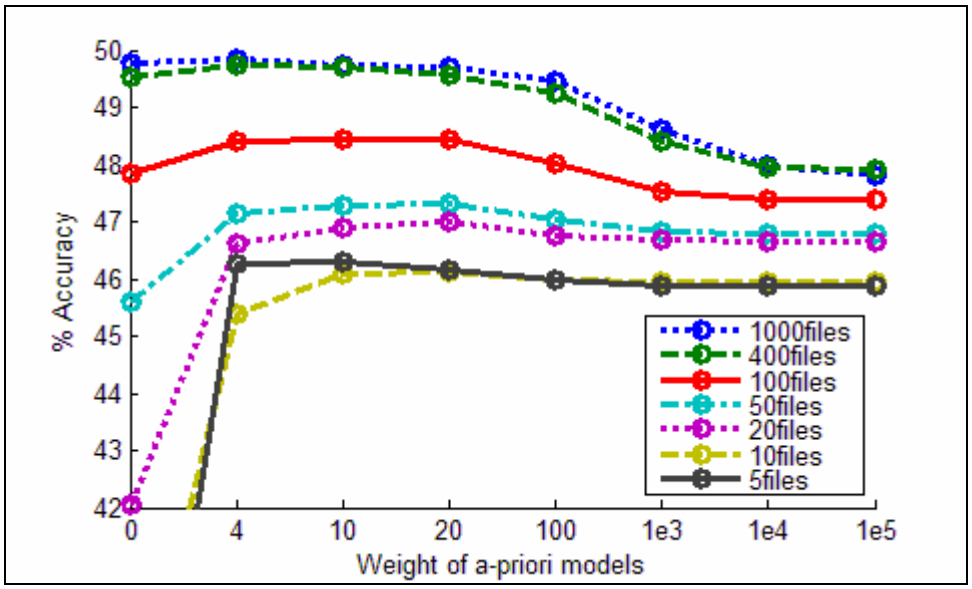


Figure 6.2 Accuracy of TIMIT models adapted using MAP for NTIMIT data. Prior models are 30-cluster MLLR models as described in Section 6.1. Numerical values are given in Table 6.2.

6.3 Summary and Conclusions

The goal of adaptation in this Thesis is mostly for comparison. For the sake of clarity, unless otherwise specified, experiments in Chapters 7 and 8 use a fixed adaptation strategy and parameter values. These are chosen to set performance in the stability regions shown in Figures 6.1 and 6.2, so that little variations will only cause small performance differences without statistical significance: global MLLR is followed by 28-cluster MLLR with minimum occupancy threshold and MAP adaptation with $\tau = 12$.

Unless otherwise stated, model adaptation in the following chapters employs all data in the TIMIT training partition, which assures that performance is saturated. However, the assumption of large corpora for adaptation is oftentimes non-realistic (also, in the case of large corpora, models can be trained from scratch instead of using adaptation). For this reason, in Section 8.4.3, performance of ASR systems using feature compensation or model adaptation is compared for different amounts of data availability.

Finally, we remind the reader that the NTIMIT corpus was used for setting the tunable parameters in adaptation. The reason is that this corpus shows an important variability due to the wide variety of telephone channels it comprehends. Therefore, it poses the most difficult case for model adaptation and a configuration in the region of stable performance in NTIMIT will also be in the stable region for TIMIT, UAM-TIMIT, or the corpora derived from TIMIT by applying artificial filters.

7

Evaluation of Phoneme-based Feature Compensation

In this chapter we present practical implementations and experimental results on feature compensation using Phoneme-based partitioning of the feature space, as described in Sections 4.1 and 4.5.3.

Training of Phoneme-based compensation functions requires phonetically labeled training data so that each Phoneme-based corrector function is trained using only data belonging to the target phoneme. This condition is fulfilled in many databases or otherwise an approximation may be obtained by means of automatic phonetic alignment of textual labels. However, the main problem in implementing this family of algorithms is during the compensation stage, as it is a requirement to identify the phonetic content prior to compensation, and consequently prior to speech decoding. In the following section we present different approaches for application of Phoneme-based compensation and later on we present experimental results for each of these.

7.1 Compensation Approaches for Phoneme-based Partitioning

In Section 4.5.3 we presented a number of practical solutions for the problem of identification of the phonetic sequence in a speech utterance, prior to ASR. Here, we briefly summarize them.

7.1.1 Oracle Phoneme-specific Compensation

In this mode we make use of real phonetic transcriptions of test utterances, so as to apply the appropriate corrector function to each frame. It is a simple configuration that shows the maximum compensation capability of this approach. However, in real situations phonetic transcriptions of test utterances are not available prior to decoding, so other strategies need to be used.

7.1.2 General Compensation

In this case, the same set of corrector functions is applied to all speech frames under a given distortion. The effect is a global compensation of the channel and noise distortions affecting speech. This is a simple approach, similar in spirit to spectral subtraction or CMN.

7.1.3 Two-stage Compensation

We propose here a method for cascading two processes of compensation-decoding (Figure 4.4). In the first one, General compensation is applied to speech of unknown textual content and the result is sent to the decoder that generates an initial transcription. The second stage applies Phoneme-dependent correction according to the generated phonetic transcription.

When enough computational power is available, the proposed method may be extended by the generation of an N-best candidate transcription or lattice in the first stage. Each of these phonetic transcription candidates may be used to obtain a different sequence of corrected features and each may be passed to a full-bandwidth speech recognizer. In our experiments we employ 3-best candidate transcriptions and the winning sequence is chosen as the one with maximum likelihood (alternatively final transcriptions could be combined using, for example, a ROVER scheme [Fiscus, 1997]).

7.1.4 Compensation Embedded in the Decoder Module

Application of feature compensation can also be applied embedded in the recognizer module, prior to computation of likelihood, as explained in Section 4.5.3.4. This requires modification of the code in the decoding module, but it is advantageous in that it removes the need for phonetic labeling of speech. In a nutshell, likelihood of a feature vector corresponding to a speech frame of unknown phonetic content for a Gaussian mixture from the acoustic model of phoneme /*x*/ is computed using a modified version of the vector compensated using the corrector functions of phoneme /*x*/. The idea is for the corrector functions to act like *magic keys*, so that when the correction from the true phoneme is applied and likelihood is evaluated with the models from the same phoneme, likelihood will be maximal.

7.2 Experimental Results

We divide our experiments in two sections. In the first one we apply Phoneme-based feature compensation in an independent module inserted in a speech recognizer between the feature extractor and decoder modules and evaluate performance using Oracle compensation, General compensation and 2-stage compensation. In the second section we evaluate Phoneme-based feature compensation embedded in the decoder module.

7.2.1 Results with Non-embedded Compensation

In Table 7.1 and Figure 7.1 we show performance for different approaches to the problem of band-limitations in a variety of distortions.

Using full-bandwidth models for ASR with band-limited speech (*No compensation*) significantly degrades performance compared to the case of full-bandwidth speech (*FB*). Although the same performance may not be expected, because part of the speech information is lost due to bandwidth-

Mode	Distortion	% Corr.	% Acc.	Distortion	% Corr.	% Acc.
No Compensation	**FB**	75.40	71.18			
Matched CMN		75.71	71.61			
No Compensation	**LP6kHz**	64.32	58.30	**BP300-3400 Hz**	41.13	32.67
Model Adaptation		75.46	70.85		70.63	64.90
CMN		74.30	69.95		60.91	54.71
Oracle Compensation		75.62	71.76		74.31	71.01
General Correction		74.02	70.14		53.31	48.96
2-stage		74.63	70.69		54.44	50.18
No Compensation	**LP4kHz**	55.93	44.67	**LP2kHz**	30.45	26.10
Model Adaptation		73.57	68.64		63.48	57.96
CMN		68.00	62.28		51.70	45.63
Oracle Compensation		75.35	71.44		74.52	70.32
General Correction		67.54	62.75		47.75	42.60
2-stage		69.05	64.31		48.91	44.42

Table 7.1 ASR performance using Phoneme-based compensation for a variety of band-limiting distortions.

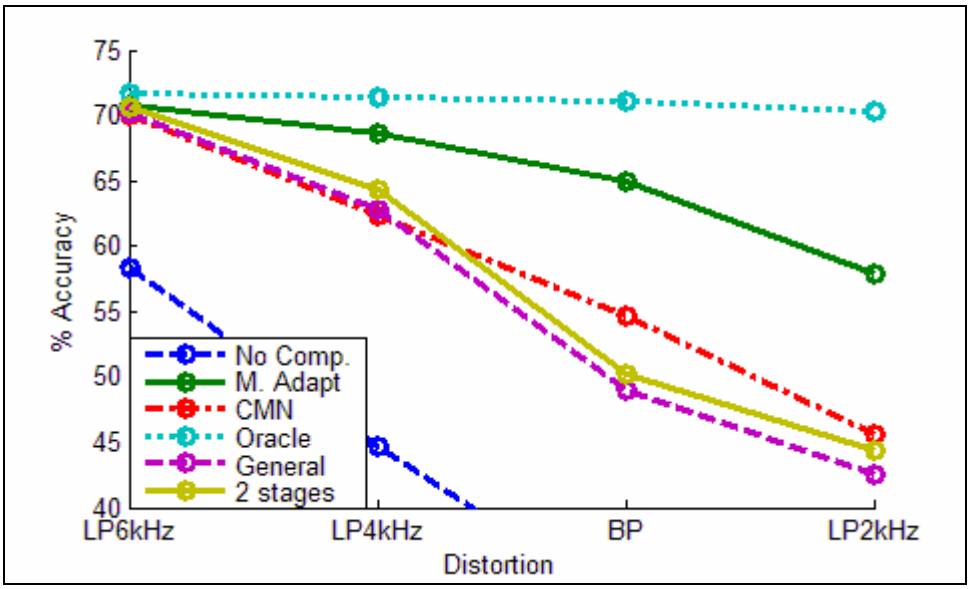


Figure 7.1 ASR accuracy using Phoneme-based compensation for a variety of band-limiting distortions. Data points correspond to Table 7.1.

limitations, we hope to be able to obtain significant performance increases by means of robustness techniques.

Oracle feature compensation shows that if the appropriate corrector function was applied in each frame of band-limited data, there would be almost no accuracy loss, even for the case of a band-limitation as severe as LP2kHz. As previously indicated, this is not applicable in real tests, because Oracle compensation makes use of phonetic labeling of speech prior to ASR. The consequence of using the true transcriptions is that for each frame, acoustic features are brought to the region of space where acoustic models' pdfs find their maxima for each individual phoneme, and so Oracle performance is not realistic. Nevertheless, results show the potential of feature compensation and in particular of Phoneme-based partitioning of the feature space.

Two-stage compensation outperforms General compensation in all cases, showing the advantage of the second pass of ASR in this approach. When the distortion is moderate (LP6kHz and LP4kHz), performance using 2-stage Phoneme-based compensation is comparable to that of model adaptation.

However, for more severe distortions, performance is clearly inferior, due to low performance of General compensation that fails to produce an acceptable labeling to use in the second stage. In fact, General Compensation is so prone to errors when the distortion is strong that 2-stage compensation is also outperformed by CMN in these severe distortion cases.

7.2.2 Results with Compensation Embedded in the Decoder Module

In this section we compare performance of Phoneme-based compensation as an autonomous module or embedded in the decoder module. In Table 7.2 and Figure 7.2 we show results for this approach using univariate, as well as multivariate feature compensation. Compensation embedded in the decoder module is inserted in the Forward search algorithm [Rabiner, 1989], and for each acoustic model we use a different compensation.

During the Forward pass it is still unknown what will be the winner phoneme sequence and this makes it impossible to compute dynamic features from regression of the static ones (or it would require major changes in the search algorithm). For this reason, compensation embedded in the decoder module uses compensation of dynamic features (instead of obtaining them by regression of static features), and for the sake of comparison results for Oracle compensation and General compensation in Table 7.2 do so too (this explains differences with Table 7.1).

In Table 7.2, we include ASR performance as well as the average log-likelihood per utterance in

Mode		**Distortion**	**% Corr.**	**% Acc.**	**Average log-likelihood (x100)**
No Compensation		**FB**	75.40	71.18	-233
Univariate	Oracle	**LP6kHz**	75.42	71.58	-227
	General correction		73.96	70.15	-231
	Embedded		72.48	66.88	-227
Multivariate	Oracle		75.03	70.43	-264
	General correction		74.64	69.98	-264
	Embedded		75.19	70.92	-229
Univariate	Oracle	**LP4kHz**	74.53	70.93	-220
	General correction		67.02	62.60	-226
	Embedded		67.67	61.92	-219
Multivariate	Oracle		75.34	71.01	-245
	General correction		70.70	65.92	-249
	Embedded		72.48	67.49	-220
Univariate	Oracle	**BP300-3400 Hz**	71.68	68.62	-211
	General correction		52.57	48.35	-219
	Embedded		42.00	38.17	-205
Multivariate	Oracle		75.08	70.89	-238
	General correction		64.40	59.47	-243
	Embedded		66.94	61.50	-211
Univariate	Oracle	**LP2kHz**	69.98	66.30	-208
	General correction		46.97	41.87	-216
	Embedded		49.14	44.00	-203
Multivariate	Oracle		73.51	69.53	-251
	General correction		51.19	46.21	-256
	Embedded		56.58	51.52	-204

Table 7.2 ASR performance using Phoneme-based compensation embedded in the decoder, or outside of the decoder (Oracle and General correction). Results for outside-of-the-decoder strategies differ from those in Table 7.1, as here we compensate dynamic features instead of obtaining them by regression of static features.

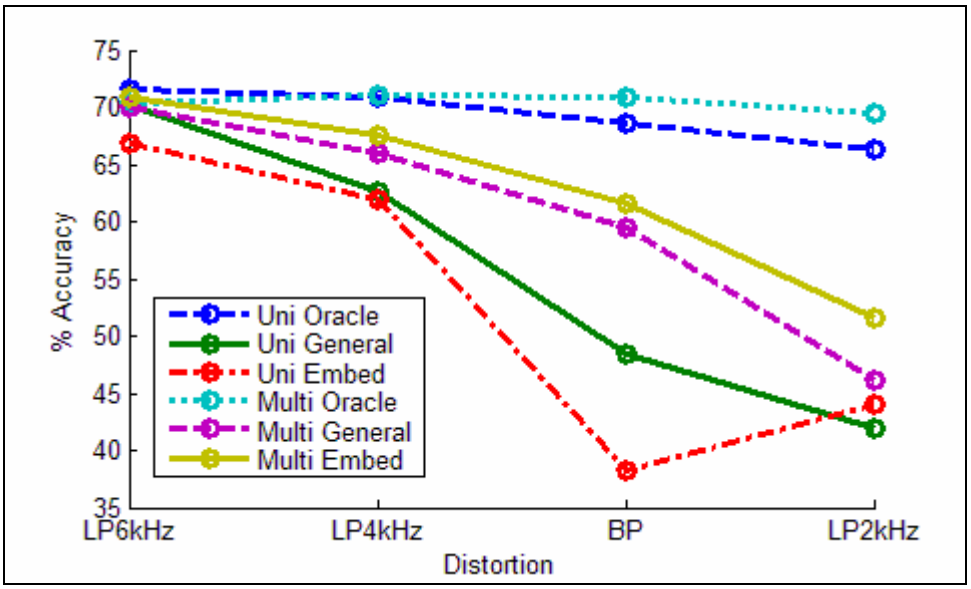


Figure 7.2 ASR accuracy comparison for Phoneme-based compensation embedded in the decoder module or outside of it for a variety of band-limiting distortions. Data points correspond to Table 7.2.

ASR. As it could be expected, compensation embedded in the decoder presents, for each distortion the maximum average likelihood. The reason is that for each frame this approach chooses the best matching phoneme (in reality this is not a frame by frame decision, because the LM also influences the result, but the final result is most importantly affected by this maximization criterion). Nevertheless, maximization of average likelihood does not result in the best ASR performance; for example, ASR performance with Oracle compensation is clearly superior in spite of smaller likelihood. This behavior may be explained by the following limit case: take two different phonemes for which training data for each feature do not show significant correlation between full-bandwidth and band-limited features. Then, it could well happen that the polynomial fits resemble straight and horizontal lines. In that case, compensation of any observation with these corrector functions would result in a final value corresponding to the means of the full-bandwidth features in the training data, and so the decoder would be choosing which of these two phonemes is more likely, based on the values of these means, and independently of the observation.

The described situation poses a major problem to feature compensation embedded in the decoder module, because if compensated values are independent of the original values, the corrector functions will not behave anymore as *magic keys*. In order to reduce this problem, we propose the use of multivariate feature compensation. As the fit now depends on a larger number of coefficients in the feature vector (and more importantly those that explain most of the variability in data), compensated features will be less subject to that problem. Following the metaphor of the magic keys, this is equivalent to extending security to a set of multiple keys. Interestingly, this new solution not only boosts performance compared to univariate compensation, but also approaches the performance of model adaptation (see ASR result for model adaptation in Table 7.1; for moderate distortions performance with multivariate compensation embedded in the decoder is almost identical, and even in the case of the BP300-3400Hz distortion, accuracy is only 3% absolute worse. Only in the case of telephone data the difference is larger). Advancing into next chapter's results, performance is also

comparable to that with multivariate feature compensation using Gaussian-based classes (see results in Table 8.7). This shows that this approach is an appropriate solution for the problem of a-priori imputation of frames to phoneme classes in Phoneme-based compensation.

7.3 Summary and Conclusions

In this chapter we have explored performance of Phoneme-based feature compensation for a number of band-limiting distortions. Oracle compensation using the actual phonetic labeling of test data showed the potential upper limit performance of ASR with feature compensation, which is very close to that using the actual full-bandwidth features.

The two techniques proposed for real implementations of feature compensation as an independent module (General compensation and 2-stage compensation) obtained encouraging results for moderate distortions, but failed in experiments on strong distortions and were clearly outperformed by model adaptation.

Alternatively, compensation embedded in the decoder module has the drawback that it requires a slight modification of the architecture of the decoder module, but it proved to be the best solution for Phoneme-based compensation, especially in the case of multivariate compensation, where it showed results comparable to model adaptation.

8

Evaluation of Gaussian Class-based Feature Compensation

We present in this chapter experiments on practical implementations of Gaussian class-based feature compensation, as described in Sections 4.2 and 4.5.2. In our experiments we compare performance with other robustness methods such as CMN and model-side compensation, as well as Phoneme-based compensation presented in the previous chapter. This chapter contains the main results of this Thesis (hence its length), allowing for an easy overall comparison of the results attained with all the methods considered in this work. Compared to Phoneme-based partitioning, Gaussian class-based compensation does not require phonetic labeling of the target speech, which posed the main constraint on the use of those techniques. The simplicity of use of the new approach and its superior performance explain why results are given here for a much larger set of conditions and constraints than in the previous chapter.

8.1 Overview

Training of Gaussian-class based compensation consists of two stages (Figure 4.1). In synthesis, the first step is to create feature-space partitioning Gaussian classes in an automatic process using the EM algorithm and secondly, for each Gaussian class created in this way, a set of corrector functions is trained. In this chapter we show several implementations for both stages based on the methodology proposed in Chapter 4 and we evaluate performance in situations that recreate potential constraints on realistic situations.

Figure 8.1 shows a schema of the experimental categories considered in this chapter. Four categories are considered (anti-clockwise in the figure, starting from the top left corner). Firstly, we provide the reader with a global idea of performance of a baseline version of Gaussian-based compensation compared to other approaches for a variety of distortions. This baseline version uses non-stereo based partitioning of the feature space, stereo-based training of corrector functions and feature compensation using univariate polynomial expansion of distorted features. Results are given

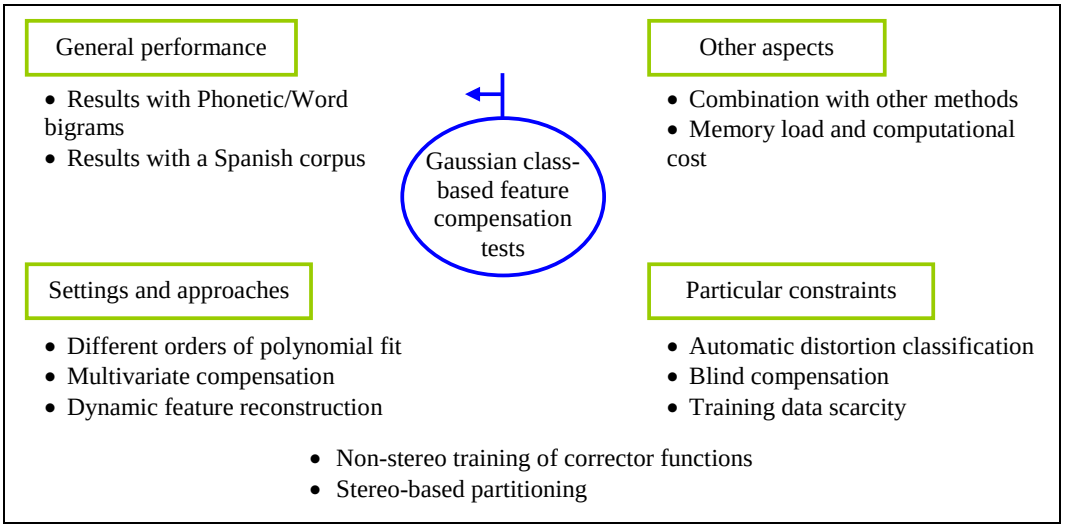


Figure 8.1 Schema of sections and experimental categories considered in this chapter.

both for phoneme-based and word-based LMs, as well as for English and Spanish corpora, to show the validity of our solutions in different systems. Later sections use phoneme-based LMs and are only evaluated in an English corpus (TIMIT [Fisher *et al.*, 1986] and its derived corpora NTIMIT [Jankowski *et al.*, 1990] and UAM-TIMIT [Morales *et al.*, 2007b]). In Section 8.3 we evaluate different implementations and settings of Gaussian class-based feature compensation. Specifically we evaluate performance of univariate polynomial expansion for different orders of the polynomial fit, we compare multivariate and univariate compensation and discuss on the convenience of compensating dynamic features or computing them using the typical regression of static features (using reconstructed static features). In addition we propose two variants of the usual procedure for partitioning the feature space and training corrector functions, depending on whether or not stereo-data is used for each of these stages. The standard approach in our work has been using stereo information for training corrector functions but not for partitioning the feature space. However, particular applications may impose constraints on the use or not of stereo-data. Therefore, these two aspects are in part considered variations of the proposed approach and in part particular constraints (hence they are placed between these two experimental categories in Figure 8.1).

In Section 8.4 we deal with problems that could arise in particular applications. Namely, automatic distortion classification, where we show how the feature compensation framework may be used in an ASR system receiving speech from a variety of distortions for which training data is available, blind compensation, where the previous situations is extended to distortions for which no training data is available and training data scarcity, where we show performance as a function of the amount of training data available.

Finally, in Section 8.5 we study the possibility of combining feature compensation with other robustness methods (CMN, model adaptation and model retraining) and in Section 8.6 a comparative study is made on memory load and computational costs of our algorithms and other robust techniques.

8.2 General Performance of Gaussian Class-based Compensation

In this section we show ASR performance with a baseline implementation of Gaussian class-based compensation (non-stereo based space partitioning and feature compensation using univariate stereo-based linear compensation of first order). A variety of real and artificial distortions are considered and for each of them specific corrector function sets are trained. Performance is compared with that using Phoneme-based feature compensation, described in the previous chapter, as well as other robustness methods, in order to give an idea of the potential of Gaussian-based compensation, and establish a basis for later improvements and task constraints. The following two sections evaluate performance using a phonetic bigram LM and a word bigram LM, in English (TIMIT) and Spanish Castilian (Albayzin).

8.2.1 Results with Phonetic Bigram LM

8.2.1.1 Results for LP4kHz Filter

Table 8.1 and Figure 8.2 show ASR performance for TIMIT test data distorted with an LP4kHz filter. Both, model adaptation and feature compensation use all data in the training partition. Full-bandwidth models are used for feature compensation and are also the seeds for model adaptation. Row A shows results for full-bandwidth test data and sets the maximum performance goal (a non-realistic goal when only distorted data is available for ASR because some information is irreversibly lost, at least for some phonetic classes). Row B shows that when band-limited data is passed directly to full-bandwidth models performance is significantly degraded and this motivates the use of robustness methods. Row C shows results for model adaptation (global MLLR followed by 28-class MLLR, followed by MAP), Row D those with new models trained with data from the target distortion and Row E shows performance when CMN is applied to both, training and test data. Rows F to H show performance of different types of Phoneme-based compensation, and Row I uses Gaussian Class-based Compensation. The last four rows apply a smoothing over the compensation sequence in order to remove errors caused by acoustic class misclassification errors, as explained in Section 4.6 (we use a median filter of size 5 frames). Gaussian class-based correction uses 32 classes and first order polynomial correction.

	Key	Mode	% Corr.	% Acc.
	A	Full-bandwidth Test Data	75.40	71.18
Classical Approaches	**B**	No Compensation	55.93	44.67
	C	Model Adaptation	73.57	68.64
	D	Matched Models	74.73	69.33
	E	CMN	68.00	62.28
Polynomial Corrector Functions	**F**	Oracle Phoneme Corr.*	75.35	71.44
	G	Polynomial Gen. Corr.	67.54	62.75
	H	2-stage Corr.	69.05	64.31
	I	Gauss. Class Based Corr.	72.41	66.97
Polynomial Corrector Functions and Smoothing	**J**	Oracle Phoneme + Smooth *	75.25	71.22
	K	Polynomial Gen. + Smooth	68.04	63.01
	L	2-stage + Smooth	69.27	64.40
	M	Gauss. Class Based + Smooth	72.61	67.45

Table 8.1 Performance of classical approaches and proposed feature compensation methods for an artificial LP4kHz filter. Rows F and J marked with asterisks show results using Oracle Phoneme-based compensation that cannot be applied in real cases.

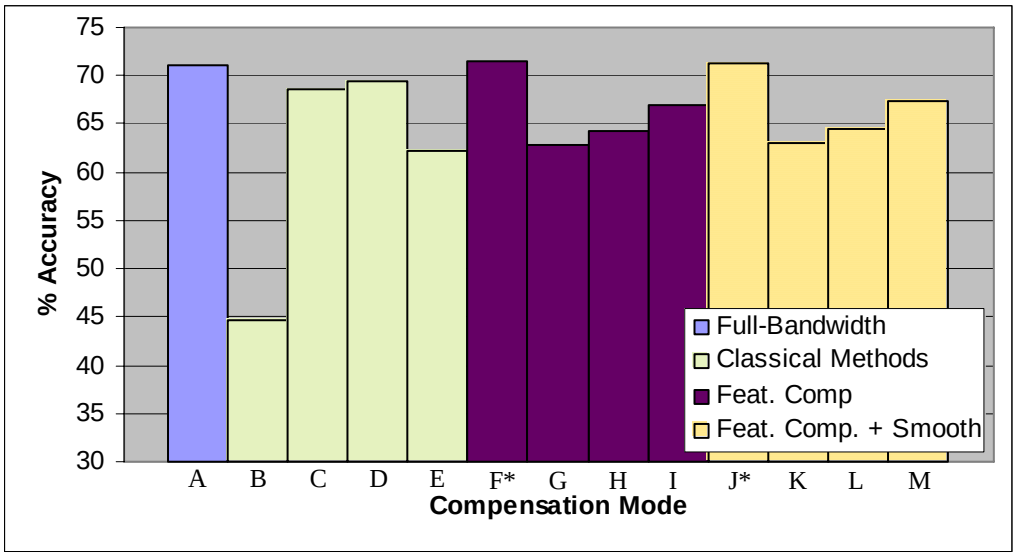


Figure 8.2 Comparison of ASR accuracy using multiple robustness approaches on TIMIT LP4kHz. Capital letters in the x-axis correspond to *key* column in Table 8.1. Bar A shows performance on undistorted TIMIT test data, for reference. Bars F and J use Oracle Phoneme-based compensation and cannot be applied in real cases.

Comparison of polynomial correction methods shows the superiority of Gaussian class-based partitioning of the space compared to practical implementations of Phoneme-based compensation (Phoneme-based Oracle compensation is not applicable in real cases). This is a very promising result, because Gaussian-based compensation is also simpler to implement than Phoneme-based approaches. Performance is also significantly better than with CMN and close to that with model-side methods (model adaptation and model retraining). It should be noted too, that smoothing the compensation sequence always improves performance (except in the Oracle case), although the gain is very moderate.

8.2.1.2 Results for Different Band-limiting Distortions

Table 8.2 and Figure 8.3 show results for the following artificial bandwidth-limiting distortions: LP6kHz, LP4kHz, LP2kHz and BP300-3400Hz, the latter representing the theoretical bandwidth of a landline telephone channel. In addition, two real telephone speech corpora are considered: NTIMIT, where TIMIT data has been passed through a variety of telephone calls with multiple characteristics [Jankowski *et al.*, 1990] and UAM-TIMIT, a corpus designed for experiments in this Thesis, where TIMIT files were passed through a telephone channel in a single call, resulting in a common channel distortion for all files [Morales *et al.*, 2007b]. Feature compensation, model adaptation and matched models have the same settings as in the previous section.

Feature compensation performance is significantly better than the baseline *No Compensation* for each distortion considered. Compared to matched models and model adaptation, performance depends on the complexity of the task; when the distortion is not severe (LP6kHz and LP4kHz), feature compensation performs similarly to model adaptation. However, more restrictive band-limitations and real channels, that also introduce noise in the signal (as is the case with NTIMIT and UAM-TIMIT), degrade performance of feature compensation approaches. Nevertheless, the comparison made is not absolutely fair: the number of free parameters in model adaptation is much larger than in feature compensation (non-diagonal versus diagonal compensation matrixes), and the whole training partition

in TIMIT has been used for adaptation, which favors approaches with more free parameters. In Sections 8.3 and later we show how performance of feature compensation may be improved, and analyze performance in terms of the number of free parameters and amount of available training data for different approaches. Also, it is worth noting that the comparison between Gaussian class-based feature compensation and CMN is always clearly in favor of class-based compensation with the exception of NTIMIT. In this case, CMN has the advantage that the compensation is done independently for each utterance, while in Gaussian class-based feature compensation the same compensation is applied to all utterances.

Mode	Distortion	% Corr.	% Acc.	Distortion	% Corr.	% Acc.
No Compensation	**LP6kHz**	64.32	58.30	**BP300-3400 Hz**	41.13	32.67
Model Adaptation		75.46	70.85		70.63	64.90
Matched Models		75.45	71.03		71.86	65.73
CMN		74.30	69.95		60.91	54.71
Class-based + Smooth		74.92	70.63		66.48	60.18
No Compensation	**LP4kHz**	55.93	44.67	**UAM-TIMIT**	30.98	21.23
Model Adaptation		73.57	68.64		62.63	58.26
Matched Models		74.73	69.33		69.10	61.80
CMN		68.00	62.28		51.59	46.98
Class-based + Smooth		72.61	67.45		56.61	49.72
No Compensation	**LP2kHz**	30.45	26.10	**NTIMIT**	36.15	26.27
Model Adaptation		63.48	57.96		55.96	50.71
Matched Models		68.67	61.57		62.45	53.76
CMN		51.70	45.63		39.62	34.05
Class-based + Smooth		55.73	49.53		40.84	34.14

Table 8.2 Performance of Gaussian class-based correction with smoothing of corrector sequence, compared to matched models, model adaptation and no compensation for multiple real and artificial band-limiting distortions.

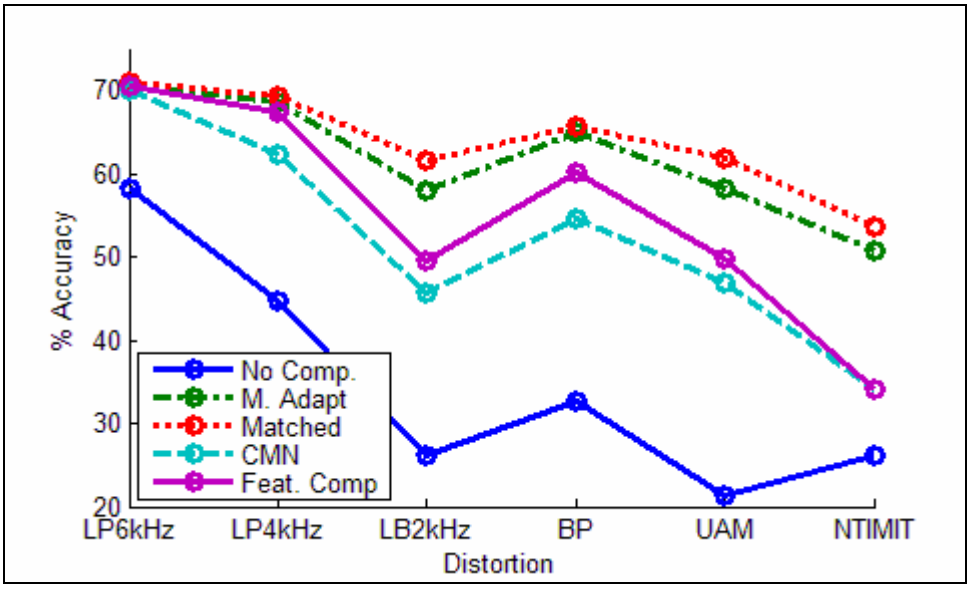


Figure 8.3 Accuracy of different approaches for a variety of distortions. Data points correspond to Table 8.2.

8.2.2 Results with Word-based Bigram LM

Most of the experiments in following sections use phoneme-based LMs. These are less task-dependent than word-based LMs and therefore, make simpler the analysis of results that also become more general. However, as ASR systems normally use word-based grammars it is interesting to verify the validity of the proposed methods using this type of LM.

As in the previous section we show performance for a variety of strategies for TIMIT distorted with an LP4kHz filter, using this time a word-level LM (Table 8.3 and Figure 8.4). Results are similar to those with a phoneme-based LM. It is interesting however, that feature compensation outperforms model adaptation using this type of LM (and is close to matched models). Although this result depends on the weight given to the LM in the decoder, we observed in our tests this tendency of bringing results of Gaussian class based compensation closer to those of model-side robustness (sometimes outperforming them). We hypothesize that feature compensation is more prone to errors of small duration than model-side approaches, but the impact of this type of errors is less important for ASR when word-based LMs are employed (this is similar to the effect caused by smoothing of the corrector sequence in that it reduces the impact of short duration misclassification errors).

	Set	Mode	% Corr.	% Acc.
	A	Full-bandwidth Test Data	79.94	78.68
Classical Approaches	**B**	No Compensation	31.61	24.09
	C	Model Adaptation	73.00	72.41
	D	Matched Models	75.85	75.08
	E	CMN	65.27	62.24
Polynomial Corrector Functions	**F**	Oracle Phoneme Corr.*	77.73	76.69
	G	Polynomial Gen. Corr.	70.07	67.61
	H	2-stage Corr.	71.58	69.50
	I	Gauss. Class-based Corr.	75.62	74.01
Polynomial Corrector Functions and Smoothing	**J**	Oracle Phoneme + Smooth*	78.17	76.90
	K	Polynomial Gen. + Smooth	70.22	67.61
	L	2-stage + Smooth	71.78	69.77
	M	Gauss. Class-based + Smooth	75.54	73.94

Table 8.3 ASR performance using a word-based LM. Data is TIMIT LP4kHz. Oracle compensation is marked with asterisks.

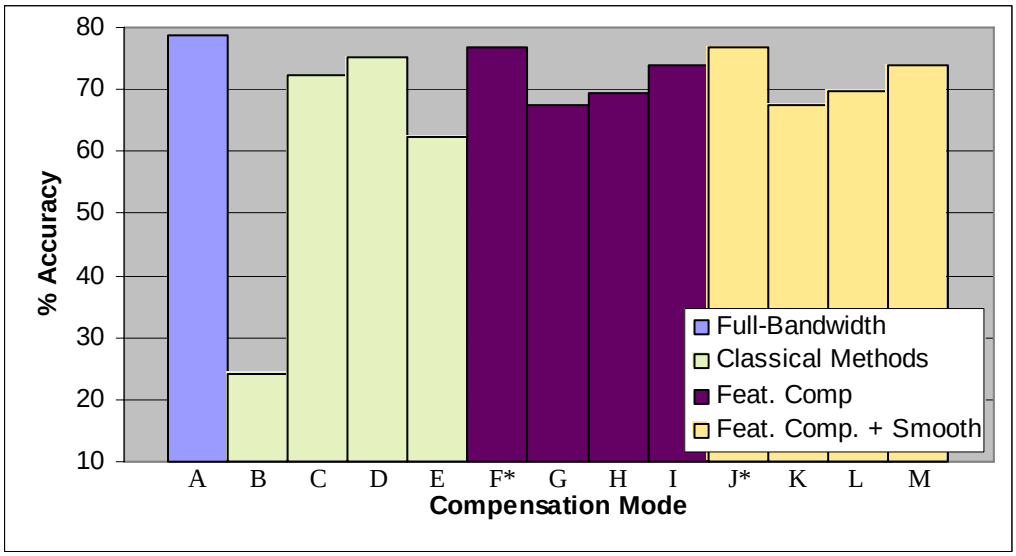


Figure 8.4 Comparison of ASR accuracy using multiple robustness approaches on TIMIT LP4kHz. Capital letters in the x-axis correspond to *key* column in Table 8.3. Bar A shows performance on undistorted TIMIT test data, for reference. Oracle compensation is marked with asterisks.

8.2.3 Results on a Spanish Corpus

Here we show the independence of our results with regard to the target language by studying performance with a corpus of Castilian Spanish (Albayzin [Moreno *et al.*, 1993]). Tables 8.4 and 8.5 show recognition using phoneme and word-based LMs for an LP4kHz filter. Results are consistent with those for TIMIT in English.

	Mode	% Corr	%Acc.
	Full-bandwidth Test Data	78.13	75.43
Classical Approaches	No Compensation	54.37	49.07
	Matched Models	76.30	73.31
Polynomial Corrector Functions	2-stage	71.44	69.59
	Class-based	74.38	70.78
Polynomial Corrector Functions and Smoothing	2-stage + Smooth	72.76	69.71
	Class-based + Smooth	74.48	71.31

Table 8.4 ASR performance for several robustness approaches in Albayzin for a phoneme-based LM and LP4kHz distortion.

	Mode	% Corr	%Acc.
	Full-bandwidth Test Data	80.96	76.21
Classical Approaches	No Compensation	22.78	-13.85
	Matched Models	75.95	69.75
Polynomial Corrector Functions and Smoothing	2-stage + Smooth	68.19	58.77
	Class-based + Smooth	74.48	67.36

Table 8.5 ASR performance for several robustness approaches in Albayzin for a word-based LM and LP4kHz distortion.

8.3 Experiments with Different Settings and Approaches

The following sections explore performance of different modifications and configurations of the basic setting for Gaussian-based feature compensation.

8.3.1 Different Orders of Polynomial Fit

In Section 4.5.1, we proposed feature mapping using a polynomial series with an infinite number of coefficients, as in Eq. (4.22), which for practical implementations must be truncated at a particular order. This can also be considered an extension of the simple formulation of compensation using a diagonal compensation matrix (polynomial series of order 1), that was derived in Section 2.5.5. Here we study performance for different orders of the polynomial series.

In Figure 8.5 we show the evolution of RMSE on training data, for a particular Gaussian class and MFCC C2, as the order of the polynomial fit is increased. Using a first order polynomial fit reduces RMSE very significantly, but larger order fits do not yield significant improvements (compare for example with a similar figure of RMSE for multivariate fit in Figure 4.3). The fact that this figure is computed for RMSE using training data and the small improvement obtained in larger order fits, suggests that larger order fits will not provide important benefits for ASR on test data. In fact, fits of different orders are very close in the most populated regions of each class, differing only in the outlier areas, where higher order fits can be highly unstable. Outlier observations compensated with high-order polynomial series have a high risk of being sent to regions of the spectrum very far from their real location, causing major errors in ASR.

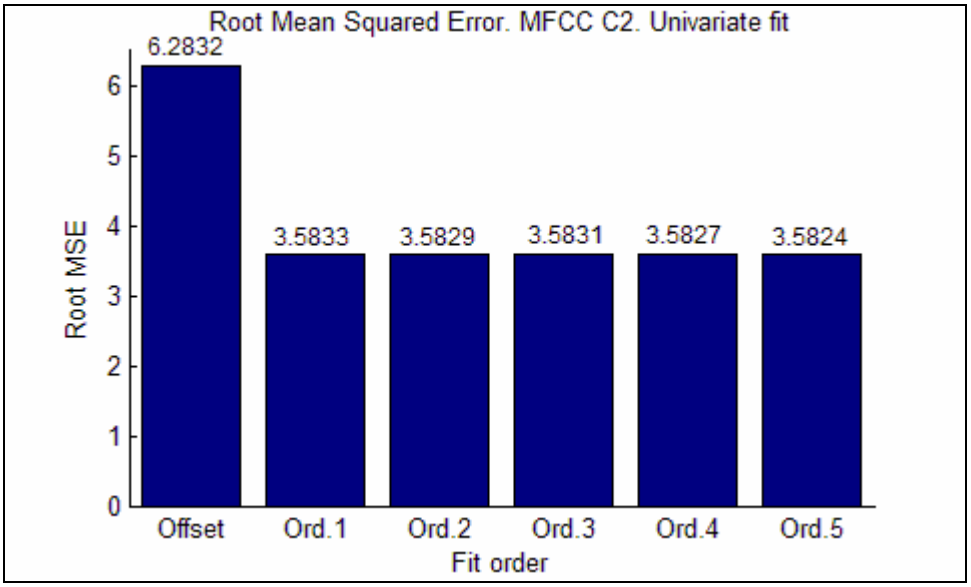


Figure 8.5 Evolution of RMSE for stepwise univariate estimation of full-bandwidth MFCC C2 to limited-bandwidth MFCC C2 (TIMIT LP4kHz). Values shown are for the order of polynomial fit indicated in the x-axis.

In Table 8.6 we show recognition results for different orders of the polynomial fit on TIMIT LP4kHz using a word-based LM. We also show performance for correction using simply an offset as in SPLICE (equivalent to a polynomial series of first order where the coefficient of order 1 is fixed to 1.00). Results show that there is not a big influence of the order of the polynomial fit in ASR

Order	% Corr.	% Acc.
Offset	74.84	71.48
Zero Order Fit	20.02	7.49
First Order Fit	74.78	71.51
Second Order Fit	74.70	71.65
Third Order Fit	75.08	71.99
Fourth Order Fit	74.79	71.54
Fifth Order Fit	74.68	71.60

Table 8.6 Performance of corrector functions with different orders of the polynomial correction using TIMIT data distorted with an LP4kHz filter and a word-based LM.

performance, which apparently justifies the simplification generally made in algorithms like SPLICE where simple offset corrections are applied in univariate correction (in our experiments we generally used a polynomial series of first order). Concerning the extremely low performance of the zero fit order (compensation using the average value of MFCC coefficients in the full-bandwidth space), it should be noted that here (as in most other experiments) we compensated static features only, while dynamic features were reconstructed using a regression of the static ones (see Section 8.3.3 for more details). When compensation is made using the average values for each class, imputation of several consecutive frames to the same class will cause dynamic features to be equal to zero, which is obviously very confusing for the decoder.

8.3.2 Multivariate Feature Compensation

So far, Gaussian-based compensation was done using the simplification to diagonal compensation matrix $\mathbf{B}_k$ as in Eq. (4.21), which effectively produces univariate compensations (each MFCC coefficient is compensated using a linear function of its observed distorted value). Nevertheless, in Section 3.5, we showed that for band-limited distortions this simplification could be less justified than for other types of distortion such as additive noise. In this section we compare performance of both approaches for a variety of distortions.

In Table 8.7 and Figure 8.6 we compare accuracy with univariate and multivariate compensation using a phoneme-based LM. Results show that a very significant improvement may be obtained by

Mode	Distortion	% Corr.	% Acc.	Distortion	% Corr.	% Acc.
No Compensation	**Full-Band**	75.40	71.18			
No Compensation	**LP6kHz**	64.32	58.30	**BP300-3400 Hz**	41.13	32.67
Model Adapt		75.46	70.85		70.63	64.90
Matched		75.45	71.03		71.86	65.73
Univariate-32		74.88	70.65		65.63	58.46
Multivariate-32		75.22	70.95		69.29	63.44
Univariate-256		---	---		68.08	60.78
Multivariate-256		---	---		70.62	64.79
No Compensation	**LP4kHz**	55.93	44.67	**UAM-TIMIT**	30.98	21.23
Model Adapt		73.57	68.64		62.63	58.26
Matched		74.73	69.33		69.10	61.80
Univariate-32		72.41	66.97		56.03	49.14
Multivariate-32		73.16	68.46		62.53	56.78
Univariate-256		---	---		60.32	53.38
Multivariate-256		---	---		64.67	58.79

Table 8.7 ASR performance using univariate and multivariate Gaussian-based compensation for different distortions. Results are compared to no compensation and model-side approaches. In univariate and multivariate compensation the number that follows indicates the amount of classes employed for band-limited space partitioning.

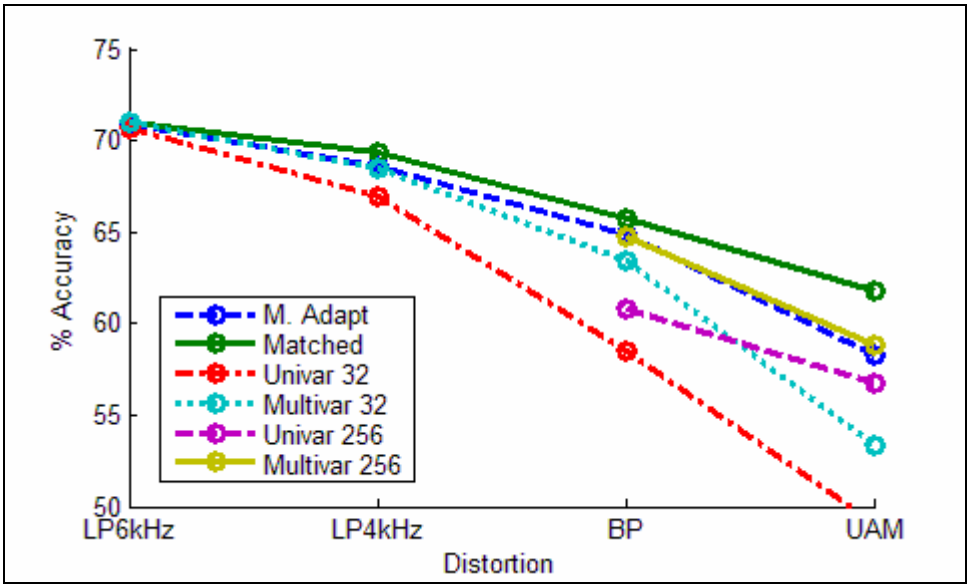


Figure 8.6 Accuracy of different approaches for a variety of band-limiting channels. Data corresponds to Table 8.7.

going from univariate to multivariate feature compensation. For the more complicated distortions BP300-3400Hz and UAM-TIMIT, we show accuracy when 32 and 256 corrector classes are employed, and in all cases, multivariate compensation offers undeniable accuracy increase (the comparison is even more favorable to multivariate compensation when distortions are more severe). Even more importantly, in all the considered distortions, performance of multivariate feature compensation is similar to that of model adaptation and only slightly worse than model retraining.

8.3.3 Reconstruction of Dynamic Features

Dynamic features may be reconstructed using two different approaches: they can be either corrected using feature compensation or computed from reconstructed dynamic features using the usual regression formula defined in Eq. (5.1).

Table 8.8 shows the average Mahalanobis distances in a frame by frame comparison between reconstructed data from the test partition in TIMIT LP4kHz and the original full-bandwidth data. This was computed individually for each MFCC coefficient and here we show results grouped into static and first and second order derivatives, as well as the added total. Results are given for both univariate and multivariate compensation using feature compensation of dynamic features (columns 1 and 3) or recomputation from static features (columns 2 and 4).

Mahalanobis Distances	**Univariate dynamic compensation ($x10^{-2}$)**	**Univariate regression ($x10^{-2}$)**	**Multivariate dynamic compensation ($x10^{-2}$)**	**Multivariate regression ($x10^{-2}$)**
Static MFCCs	0.7848	0.7848	0.7091	0.7091
Δ MFCCs	0.8624	0.8180	0.7193	0.7234
ΔΔ MFCCs	0.8534	0.8582	0.7393	0.7526
Added	2.501	2.461	2.168	2.185
ASR accuracy	66.15	66.83	68.22	68.46

Table 8.8 Average Mahalanobis distance between TIMIT LP4kHz and the actual full-bandwidth features. Results are given for univariate and multivariate compensation, where dynamic features are computed using either feature compensation or regression from reconstructed static features. ASR accuracy with each set of features is also shown.

It is interesting from this table that while computing dynamic features using corrector functions reduces the difference between reconstructed and original full-bandwidth data for the majority of individual groups of features (the only exception being delta features in univariate compensation), ASR accuracy is better when dynamic features are reconstructed by regression of static features. The first observation is not surprising because compensation of dynamic features aims at reducing MMSE between band-limited and full-bandwidth distributions. However, ASR accuracy results show that minimizing the difference does not assure optimal recognition results. The reason may be the incongruence between static and dynamic features when dynamic features are computed using corrector functions; there is a mismatch between acoustic models trained with data for which dynamic features are defined as a function of static features and feature vectors where dynamic features are reconstructed by compensation of band-limited dynamic features. In the rest of experiments in this Thesis feature compensation is done for static features only, and dynamic features are always obtained by regression of the static ones.

8.3.4 Stereo-based Feature Space Partitioning

Experiments in Gaussian-based compensation outside of this section employ non-stereo feature space partitioning as described in Section 4.2.1. However, in Section 4.2.2 we argued that the method is sub-optimal in the sense that it is independent of the goal of feature compensation for ASR. Namely, non-stereo partitioning assumes that groups of data that are close together in the band-limited space have followed the same transformation and were also close in the full-bandwidth space. In stereo-based partitioning we attempt to find clusters of data closely tied in both the full-bandwidth and limited-bandwidth spaces by using super-vectors, where the full-bandwidth and limited-bandwidth features are concatenated.

In Table 8.9 we compare performance of *Non-Stereo Partitioning*, as is normally done in our work, and two different versions of compensation using stereo-based partitioning. Differently from what is done in other experiments in this Thesis, where class partitioning is done with our own software, here class partitioning was made using HTK software and a slightly different partitioning criterion, which explains the small performance differences with other experiments (for example

Mode	Distortion	% Corr. 32class	% Acc. 32class	%Corr 256class	% Acc. 256class
No Compensation	**Full-Band**	75.40	71.18	75.40	71.18
No Compensation	**UAM-TIMIT**	30.98	21.23	30.98	21.23
Non-Stereo Partition		54.55	48.07	60.51	53.63
Stereo Partition Half		55.31	48.63	58.99	52.03
Stereo Partition Full iter. 0		59.29	48.02	63.25	51.23
Stereo Partition Full iter. 1		60.05	48.12	64.17	51.09
Stereo Partition Full iter. 2		60.21	48.10	64.43	51.13
Stereo Partition Full iter. 3		60.23	48.08	64.41	51.05
No Compensation	**LP4kHz**	55.93	44.67	55.93	44.67
Non-Stereo Partition		71.87	66.80	72.85	67.79
Stereo Partition Half		71.97	66.96	72.60	67.60
Stereo Partition Full iter. 0		72.21	66.66	72.59	67.19
Stereo Partition Full iter. 1		72.33	66.77	73.01	67.34
Stereo Partition Full iter. 2		72.27	66.61	72.98	67.25
Stereo Partition Full iter. 3		72.24	66.58	72.98	67.26

Table 8.9 ASR performance for univariate feature compensation using stereo and non-stereo partitioning.

Table 8.7). *Stereo Partition Half* designates the case where partitioning classes are created using super-vectors of concatenated features, but training of corrector functions and feature compensation uses only band-limited features. Finally, *Stereo Partition Full* represents the configuration where the three steps (class creation, corrector function training and feature compensation) employ concatenated vectors. This is an iterative process because the full-bandwidth feature vectors of distorted speech need to be computed in each step using the correction from the previous step.

Non-Stereo Partitioning and Stereo Partition Half produce similar results, which indicates that using stereo data only for class creation does not produce better modeling of the feature space. However, when stereo-data is also employed for training the corrector functions (Stereo Partition Full), Percent Correct is substantially improved, but surprisingly accuracy is worsened, indicating that the number of insertions is substantially increased. Most probably, an adjustment of the insertion penalty would make accuracy for Stereo Partition Full better than for Stereo Partition Half, but we have decided to keep this parameter fixed in all our phoneme-based LM tests. As for the ideal number of iterations, one iteration seems to provide convergence (in accordance to results in [Afify *et al.*, 2007] for noise robustness). This is not surprising because each iteration uses the full-bandwidth features estimated in the previous iteration, and therefore the choice favored by these will be in accordance to that suggested by the band-limited features.

In summary, the performance difference seems to be motivated not by better partitioning of the feature space, but by whether corrector functions are created using concatenated super-vectors or simple band-limited vectors and Stereo Partitioning Full is probably benefiting from more accurate correction.

8.3.5 Non-stereo Training of Compensator Functions

A drawback of the feature compensation algorithms previously described is the need of stereo data for training the compensation. However, when stereo data is not available it is possible to train a global mapping between the full-bandwidth and distorted distributions, as shown in Section 4.4. In this case, all constraints on the training data disappear: in general Gaussian class-based feature compensation does not require any type of labeling, and when the constraint of stereo data is also removed, the only requirement is the availability of data from the distorted environment (even test data itself may be used for online adaptation). In this section we compare performance using stereo-data and non-stereo data training of the corrector functions (Sections 4.3 and 4.4).

Table 8.10 and Figure 8.7 show ASR performance for non-stereo training of corrector functions. Test data is TIMIT LP4kHz, for which accuracy using stereo data training of corrector functions was 66.97% (Table 8.1). Results are given for two different modes: Gaussian classes defined in the full-bandwidth space (FB), or in the band-limited space (LB), as is normally done in other experiments in this Thesis. The first option could be useful when a single system is subject to multiple distortions, because it allows storage of a unique set of corrector functions (that need to be modified following Eqs. (4.3) and (4.4) prior to imputation of distorted feature vectors to classes). Additionally, for the case of classes defined in the band-limited space we show performance using liner compensation as in Eqs. (4.18) and (4.19).

Mode →	FB space - Offset		LB space - Offset		LB space - Linear	
Iterations ↓	% Corr	% Acc	% Corr	% Acc	% Corr	% Acc
0	66.36	56.84	68.63	56.61	56.08	41.76
1	68.93	58.33	69.58	63.46	67.84	58.11
2	70.10	61.23	70.41	64.34	69.62	61.89
3	70.55	62.13	70.90	64.71	70.26	63.40
4	70.73	62.39	**71.14**	**64.75**	70.56	63.83
5	70.92	62.60	71.24	64.67	**70.70**	**63.87**
6	70.88	62.57	71.13	64.40	70.60	63.62
7	71.01	62.74	71.00	64.13	70.43	63.32
8	**71.04**	**62.80**	70.92	64.00	70.32	63.08
9	70.98	62.76	70.81	63.84	70.26	62.96
10	70.93	62.69	70.70	63.70	70.14	62.72

Table 8.10 System performance for different non-stereo compensation approaches and number of iterations of the EM algorithm. Data is from TIMIT LP4kHz.

In Section 4.4 we showed that non-stereo mapping is an iterative process. The full-bandwidth and limited bandwidth data distributions are imputed to Gaussian classes, using in each pass the current compensator functions. Imputation, in turn, determines the new values of compensator functions.

Results clearly show that modeling classes in the limited-bandwidth space offers much better performance and quicker convergence than doing so in the full-bandwidth space. Thus, in real applications this approach should be used unless saving memory space is a priority, in which case a single set of full-bandwidth space classes can be used for an unlimited number of distortions. Regarding the comparison of compensation using a linear mapping, or simple offsets, the latter seems better (this is somehow surprising because in linear mapping more free parameters exist and so, mapping could be more precise). Although more work needs to be done, it is our impression that linear mapping is being penalized by the fact that in Eq. (4.19), the sign of $\mathbf{B}_k$ is undefined. In our experiments we used the plus sign because in stereo data training of corrector functions the majority of corrector terms were positive, but obviously this is not the best criterion.

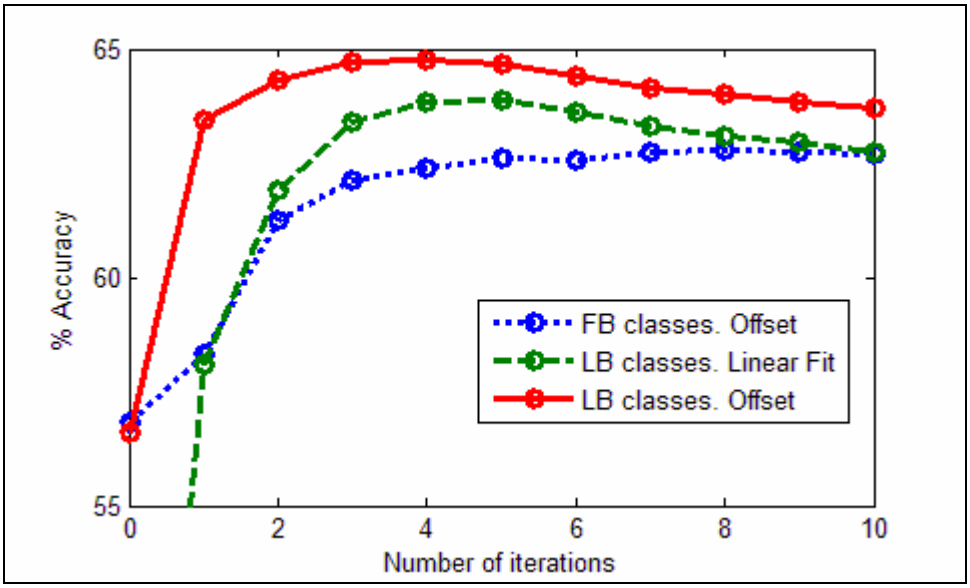


Figure 8.7 Accuracy vs. number of iterations in three different non-stereo compensation modes for TIMIT LP4kHz. Data corresponds to Table 8.10.

Probably the most interesting conclusion from these experiments is that performance is close to that using stereo-based Gaussian class compensation (66.97% accuracy in the same conditions), and superior to CMN (62.28%).

8.4 Experiments on Possible Constraints Imposed by Particular Applications

In this category of experiments we show performance of different approaches under several constraints that might arise in real applications. Firstly, we show how feature compensation may be used in ASR systems for which input data may present different band-limitations and how the system is able to automatically classify the distortions and consequently apply the appropriate compensation. Later, we extend this framework to a situation where data from the actual band-limitations is not available during training, but the effect of the distortion may be approximated by a combination of others that are available for training. Finally, we show performance of feature compensation and model-side robustness techniques as a function of the amount of available band-limited data for adaptation.

8.4.1 Feature Compensation with Automatic Distortion Classification

In this section we study compensation of data from multiple distorting environments using a single feature compensation framework. This approach allows the system to automatically detect and compensate the type of distortion of incoming speech, keeping active a single set of acoustic models at all times. The only constraint is that the distortion is among a closed set of distortions and data is available for training the corrector functions (in Section 8.4.2 this constraint is removed).

Two different strategies may be employed for multi-environment Gaussian class creation (see Section 4.5.4): sets of classes from individual environments may be combined in a super set (*IEC*), or Gaussian classes can be trained combining data from multiple environments (*MEC*). In IEC input speech frames from an unidentified distortion should be first classified among one of the possible distortions (the criterion is simply the distortion of the Gaussian class for which likelihood is maximal) and then corrected with classes from the appropriate subset. In MEC, however, the concept of individual distortion disappears and classes model a multi-distortion environment.

To illustrate this scenario we show in Figure 8.8 a spectrogram from a file in TIMIT where different distortions are applied (chunks of random size are distorted by a randomly chosen bandwidth limitation among LP6kHz, LP4kHz, LP2kHz or no distortion). Below the spectrogram we show results of automatic classification using classes combined following IEC, and the same after smoothing the environment classification. Smoothing is based on the assumption that the band-limitation affecting a portion of speech will probably last for at least several observations, so for each frame, a window of length N is created and the distortion that represents the statistical mode in the whole window is declared the best guess (in the figure N=21 frames, equivalent to 210 ms, a lapse of time small enough to assume band-limitation stability). This smoothing should not be confused with that defined in Section 4.6 that applies to the sequence of corrector values; here we smooth the distortion classification.

From Figure 8.8 it is clear that for this particular utterance automatic classification is highly reliable and almost perfect after smoothing is applied. In Table 8.11 we show complete results for the

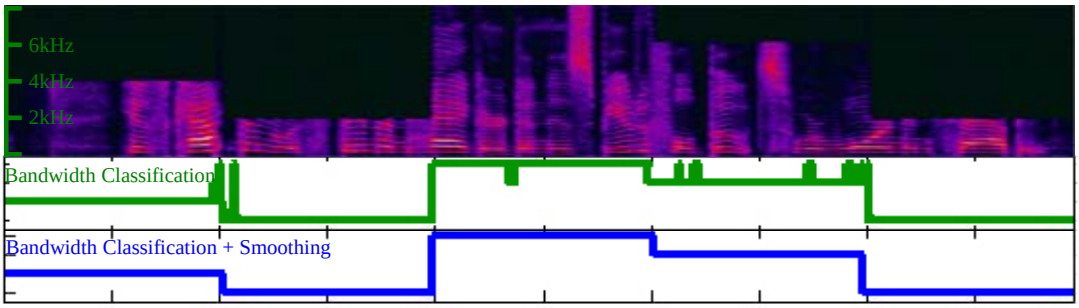


Figure 8.8 Spectrogram of file DR1_FAKS0_SI1573 from TIMIT after random length filtering with randomly chosen low-pass filters. Below the spectrogram we show classification outputs with and without smoothing.

Detected Chan → / Input Channel ↓	FB (%)	LP6kHz (%)	LP4kHz (%)	LP2kHz (%)
FB	91.58 **(98.81)**	6.57 **(0.99)**	1.61 **(0.18)**	0.25 **(0.02)**
LP6kHz	5.07 **(0.18)**	92.85 **(99.43)**	1.94 **(0.38)**	0.14 **(0.01)**
LP4kHz	1.25 **(0.01)**	0.46 **(0.00)**	97.96 **(99.95)**	0.33 **(0.03)**
LP2kHz	0.19 **(0.00)**	0.03 **(0.00)**	0.02 **(0.00)**	99.76 **(100.00)**

Table 8.11 Input distortion classification rates as percentage of frames classified. Results in parentheses and bold are obtained using smoothing of the distortion classification.

whole TIMIT database under the four environments considered, that confirm the reliability of automatic distortion classification. Therefore an integrated system that first classifies the distortion and then compensates its effects on the features will perform almost as well as a system where a-priori information of the type of distortion for each observation is given to the system.

In Table 8.12 and Figure 8.9 we show accuracy using both, IEC and MEC. We stress the fact that for both IEC and MEC exactly the same feature compensation algorithm has been used for all the different bandwidth limitations, so the only difference is in the space partitioning strategy. We have also tested an oracle form of IEC assuming perfect identification of the band-limiting distortion in order to analyze the loss of performance due to inaccuracies in distortion classification. We also show performance using CMN and model adaptation. It should be noted that the comparison with model adaptation is biased because we assume that the type of distortion is known and ASR uses only the appropriate acoustic models in each case (it is similar in that sense to the oracle form of IEC). Implementations without this knowledge would require a prior classification module and would result in a slight accuracy loss. Also, if the distortions are subject to rapid changes, a system running with multiple acoustic model sets in parallel would require an effort in integration of the models or outputs of each set, while the proposed feature compensation methods would be directly applicable to those situations.

Regarding the relative performance of IEC and MEC methods, better accuracy is obtained with the former, which indirectly shows that the method for class partitioning is sub-optimal: assuming optimality, MEC should be at least as good as IEC, because inter-distortion classes should only be created if a performance increase was possible; otherwise MEC should converge to IEC. The reason for this anomaly is the fact that the partitioning criterion is independent of the goal of feature vector

reconstruction. On the contrary, combination of classes from individual environments implicitly drives the system to the goal of reconstruction of distorted data, because classes are divided according to the distortion that produced the alteration on feature vectors.

A relevant observation is that IEC with smoothing of the classification is practically as powerful as Oracle IEC, which means that classification of bandwidth limitation is close to perfect. Our results in Table 8.12 and Figure 8.9 are for univariate compensation and therefore are slightly worse than with model adaptation. However, with exactly the same automatic bandwidth classification method we can also apply multivariate compensation, which as we showed previously (and also in following sections), generates similar ASR performance as model adaptation. In addition, IEC with smoothing and multivariate feature compensation is a more powerful solution to the problem of bandwidth limitations than model adaptation, because compensation is made outside of the decoder (and so it only requires to store a single set of full-bandwidth HMM models) and can seamlessly handle different and even time-varying bandwidth limitations with an accuracy close to model adaptation.

Distortion	Compensation	% Corr.	% Acc.
Full Bandwidth	IEC	75.22	70.79
	IEC + smoothing	75.33	71.08
	MEC	74.43	69.52
	No Compensation	75.40	71.18
LP6kHz	IEC	74.81	70.48
	IEC + smoothing	74.84	70.62
	MEC	74.22	69.45
	Oracle IEC	74.88	70.67
	Model Adaptation	75.46	70.85
	CMN	74.30	69.95
	No Compensation	64.32	58.30
LP4kHz	IEC	72.31	66.81
	IEC + smoothing	72.32	67.10
	MEC	71.19	64.68
	Oracle IEC	72.32	67.11
	Model Adaptation	73.57	68.64
	CMN	68.00	62.28
	No Compensation	55.93	44.67
LP2kHz	IEC	55.07	48.22
	IEC + smoothing	55.01	48.21
	MEC	55.60	47.33
	Oracle IEC	55.01	48.21
	Model Adaptation	63.48	57.96
	CMN	51.70	45.63
	No Compensation	30.45	26.10

Table 8.12 ASR for different filters using univariate feature compensation with automatic distortion classification, compared to Oracle IEC and recognition with model adaptation, CMN and no compensation. Automatic classification results are given for IEC (with and without smoothing of the classification decision) and MEC methods.

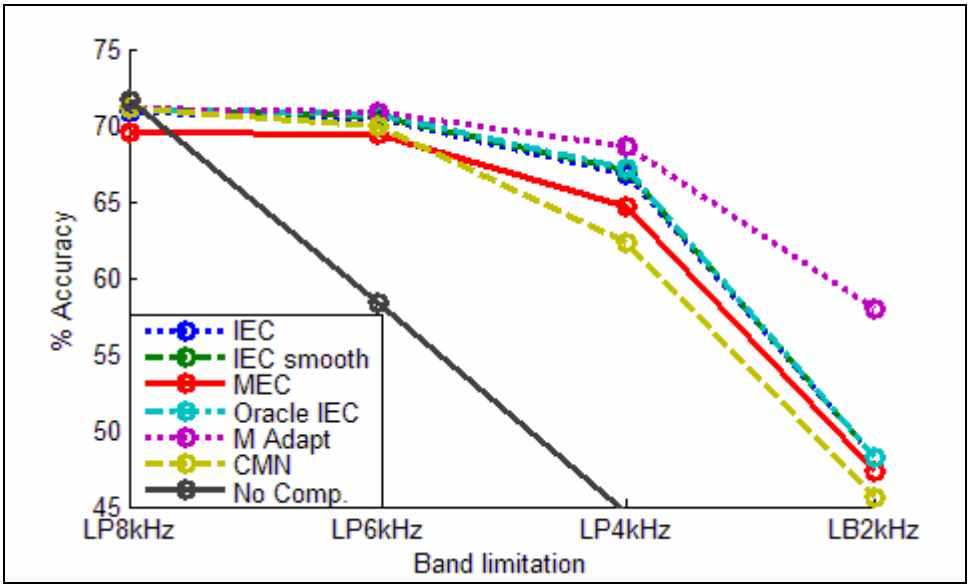


Figure 8.9 ASR accuracy for automatic feature compensation and other approaches, for a variety of distortions. Data points are shown in Table 8.12.

8.4.2 Blind Compensation of Unseen Distortions

Here, the automatic distortion classification framework proposed in the previous section is extended to a continuous range of low-pass filters, while training data is still restricted to a limited number of environments. In other words, distortions unseen during training are compensated using corrector functions for which data is available. Gaussian classes and corrections are defined using the IEC strategy described in the previous section.

Two experimental setups are considered; in the first one, the same four band-limitations considered in the previous section are used for training classes: LP6kHz, LP4kHz, LP2kHz and no distortion. Feature compensation is done with data from the low-pass filtering environments observed in training as well as intermediate low-pass filters unseen during training (see Table 8.13 for cut-off frequencies, where *O#:* represent distortions for which data is available during training and *U#:* are those for which no training data exists). In the second setup 8 low-pass filters are used for training with cut-off frequencies uniformly distributed in a mel-scale and we compensate data from these low-pass filters, as well as those with intermediate cut-off frequencies (on a mel-scale). The main difference between the two setups is the relative distance between the distortions available for training.

In Table 8.13 we show classification accuracy for the two proposed configurations. The column labeled *Hit* corresponds to proper classification and *+/- 1* includes classification as the immediately prior or posterior low-pass filter. In the case of filters unobserved during training, column *+/- 1* means that the distortion is successfully classified as belonging to the immediately superior or inferior observed distortions (the system cannot score hits for distortions unseen during training).

Percentage of hits for observed distortions is higher in Setup 1 because the number of choices for the system is smaller. However, it should be noted that including the immediately consecutive distortions in Setup 2 brings accuracy of classification of observed distortions above 96% in all cases and this without using smoothing of the classification discussed in the previous section.

Setup 1			Setup 2		
Input Bandwidth	**Hit**	**+/- 1**	**Input Bandwidth**	**Hit**	**+/- 1**
O1: 8000 Hz	91.58	98.14	O1: 8000 Hz	68.78	97.70
U1: 7000 Hz	---	97.67	U1: 7588 Hz	---	97.20
O2: 6000 Hz	92.85	99.86	O2: 7196 Hz	58.35	96.92
U2: 5000 Hz	---	82.83	U2: 6823 Hz	---	86.01
O3: 4000 Hz	97.96	98.75	O3: 6467 Hz	83.20	97.78
U3: 3000 Hz	---	50.87	U3: 6128 Hz	---	93.25
O4: 2000 Hz	99.76	99.78	O4: 5805 Hz	77.52	96.42
			U4: 5497 Hz	---	93.27
			O5: 5204 Hz	88.07	98.45
			U5: 4925 Hz	---	96.96
			O6: 4659 Hz	88.26	98.69
			U6: 4405 Hz	---	96.66
			O7: 4164 Hz	90.52	99.74
			U7: 3933 Hz	---	99.78
			O8: 3714 Hz	98.66	99.91

Table 8.13 Input distortion classification accuracy (in percentage of frames) using automatic environment classification in feature compensation. For observed bandwidths 'Hit' corresponds to correct choice and '+/-1' is the sum of 'Hit' plus the cases where 1st choice belongs to the immediately previous or posterior observed distortions. For unobserved bandwidths '+/-1' shows the percentage of times when the 1st choice belongs to the immediately previous and posterior observed distortions.

Misclassification as the immediate distortions should not degrade too importantly the results of the correction and so, system accuracy should not be severely affected. On the contrary, unseen distortions, are identified correctly in a much larger proportion in Setup 2, particularly in the lower frequencies, showing that a sufficient resolution is needed for successful identification of unseen distortions (of course at the cost of increasing computation time). The following experiment shows ASR accuracy using Setup 2.

In Tables 8.14 a) and b), and Figure 8.10, results are shown for channels seen and unseen during training, respectively. Feature compensation is performed using IEC plus smoothing and multivariate feature compensation. Once again, we present results with a supervised form of this method in which the distorting channel is assumed to be known to the system in advance and an unsupervised form in which automatic bandwidth detection is applied to all the different bandwidth conditions. Also, in these experiments we use MMSE compensation as in Eq. (4.25), which allows the combination of compensations produced by classes from different environments. Thus, unseen environments can be compensated combining the corrections trained for different distortions. We compare these results to those using model adaptation, where we assume a supervised framework in which the band-limiting channel is known in advance.

The main conclusion from this experiment is that our proposed feature compensation approach outperforms model adaptation (particularly for not very strong bandwidth limitations) or performs only slightly worse (for very limited bandwidths) with the additional advantage of being able to seamlessly compensate any bandwidth limitation (either seen or unseen during training of the compensation), while allowing the decoder to work with a single set of full-bandwidth acoustic models. It may also be observed that our feature compensation approach seems to be slightly less sensitive to mismatch in the bandwidth between training and test conditions than model adaptation techniques.

Input Bandwidth	Feat. C. Unsup.	Feat. C. Sup.	Model Adapt
O1: 8000 Hz	71.02	71.18	69.78
O2: 7196 Hz	70.77	71.14	69.81
O3: 6467 Hz	70.38	70.91	69.64
O4: 5805 Hz	70.19	70.64	69.34
O5: 5204 Hz	69.15	69.77	68.65
O6: 4659 Hz	68.27	68.70	68.29
O7: 4164 Hz	66.55	67.18	67.46
O8: 3714 Hz	65.49	65.68	66.45

a)

Input Bandwidth	Feat. C. Unsup.	M. Adapt Previous	M. Adapt Posterior
U1: 7588 Hz	70.92	69.78 (56.30)	69.79 ----
U2: 6823 Hz	70.43	67.30 (61.08)	69.43 (68.25)
U3: 6128 Hz	70.02	68.84 (55.38)	69.28 (62.07)
U4: 5497 Hz	69.26	67.84 (57.54)	67.94 (59.92)
U5: 4925 Hz	68.11	67.62 (54.06)	67.34 (53.85)
U6: 4405 Hz	66.06	67.08 (54.37)	65.66 (53.10)
U7: 3933 Hz	64.48	65.96 ----	65.04 (48.74)

b)

Table 8.14 ASR accuracy for a) observed and b) unobserved channels during training. In a) performance of supervised and unsupervised feature compensation is almost identical, showing that automatic distortion classification is successful. For distortions unobserved during training no matching models exist, so results are given for the immediately superior and inferior models. In parenthesis accuracy using the acoustic models from two distortions above or below the actual one (for example, when input distortion is U1 and models trained with O3 instead of O2 are used accuracy goes from 65.04% to 48.74%). This shows the high cost in accuracy for classification errors.

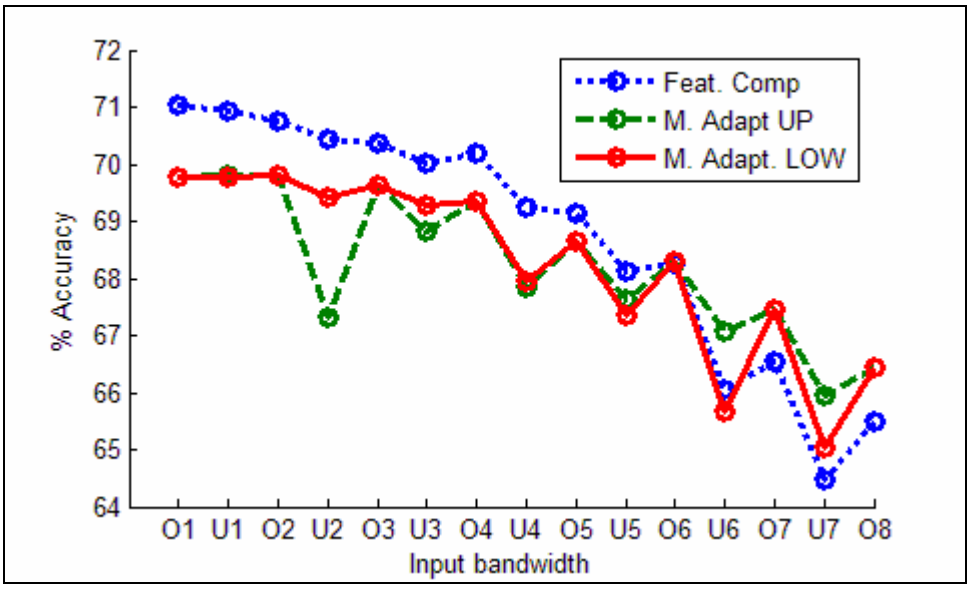


Figure 8.10 ASR accuracy for feature compensation and model adaptation. The x-axis indicates the filter corrupting data. Distortions labeled O# are observed during training, and those labeled U# are unobserved.

In Figure 8.8 we showed a particularly complicated situation where a single utterance may be subject to different distortions of random length (observed and unobserved distortions in Setup 2). In such cases feature compensation may be employed in a straightforward manner (in fact without any modification) because feature compensation automatically classifies the channel bandwidth and is independent from the decoder module. Model-side approaches, on the contrary require more elaborate system modifications. In order to illustrate the situation, we show in Table 8.15 a comparison of ASR performance on a version of TIMIT filtered as in Figure 8.8 with all the observed and unobserved distortions in Setup 2. Models adapted with data from a single distortion are clearly unsatisfactory, because whatever distortion is used, most of the times the actual distortion of incoming data will be unmatched. A better solution is model adaptation using data from all

Mode	% Corr.	% Acc.
Feature Compensation	**73.54**	**68.47**
Model Adapt: O1	53.60	48.61
Model Adapt: O2	57.30	51.98
Model Adapt: O3	59.34	53.23
Model Adapt: O4	60.62	53.95
Model Adapt: O5	62.20	54.68
Model Adapt: O6	60.93	53.01
Model Adapt: O7	63.52	55.35
Model Adapt: O8	62.55	54.24
Model Adapt: All	**68.04**	**61.32**

Table 8.15 ASR performance on TIMIT corpus distorted with all low-pass distortions considered in experimental Setup 2 in different fragments of random size. Feature compensation uses acoustic models trained with full-bandwidth data and feature compensation using IEC blind classification. Model Adapt results are given for acoustic models adapted with data from a particular distortion among the observed ones and in the final row for models adapted with data from all the observed distortions.

observed distortions (last row in Table 8.15), but this approach is nevertheless clearly outperformed by feature compensation (even in this case where univariate feature compensation was employed).

In summary, in situations where a system needs to process speech coming from channels with different (possibly unknown) bandwidths, and even in cases in which channel bandwidth changes in time, feature compensation has proved to be a very powerful solution. In the most simple situation of a fixed and known band-limiting distortion, our feature compensation methods outperform (or perform as well as) model adaptation approaches. In more complex cases in which model adaptation is very difficult to apply, such as unknown channel bandwidth or time-varying channel bandwidth, our feature compensation approaches can automatically detect the channel bandwidth and compensate it at the feature level, while keeping the decoder module untouched (it can still operate in the usual way for full-bandwidth speech with just the full-bandwidth acoustic models and unaware of the distortions), and still perform as accurately or even better than in a supervised model adaptation framework.

8.4.3 On Available Training Data and Number of Corrector Classes

One of the main advantages of model adaptation or feature compensation over model retraining is the need for less adaptation or training data. In Chapter 6 we showed that even for very limited amounts of data, model adaptation offers significant accuracy increase (of course this is a known fact that has been shown in the past by different authors, for example in [Leggetter and Woodland, 1995]) and the same property would be desirable for feature correction methods. In this section we study the relation between available training data and feature compensation performance.

Similarly to the case of model adaptation, data availability determines the number of adaptation or corrector classes for optimal reconstruction performance. The first set of experiments shows ASR accuracy for univariate feature compensation of order 1 (Table 8.16 and Figure 8.11a) and multivariate feature compensation (Table 8.17 and Figure 8.11b) for TIMIT LP4kHz and different numbers of classes and training files available. The specified number of training files is used for both, class creation and training of the compensation.

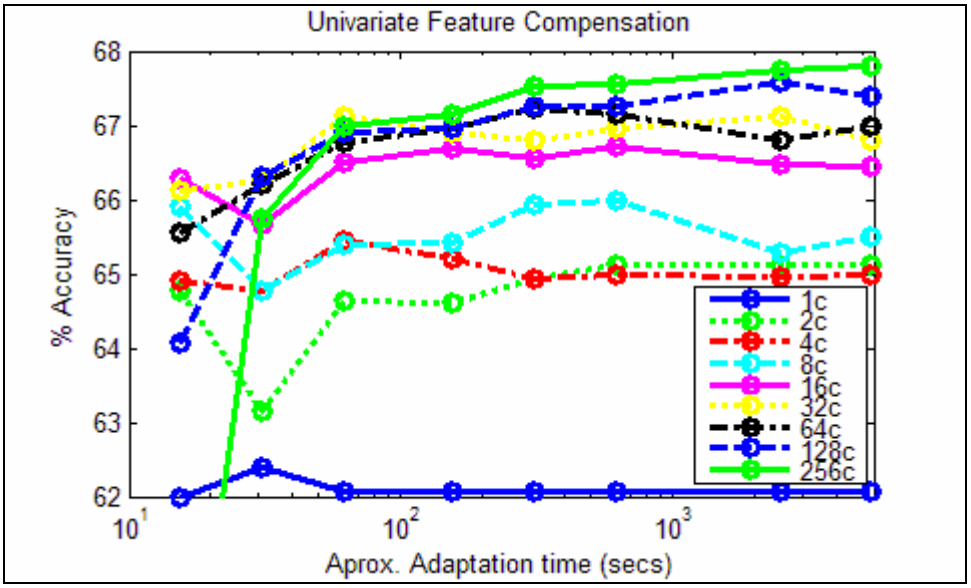
Number of adapt. files → / Number of classes ↓	5	10	20	50	100	200	800	1718
1 class	62.00	62.38	62.08	62.08	62.08	62.08	62.08	62.08
2 classes	64.76	63.16	64.63	64.61	64.94	65.13	65.12	65.11
4 classes	64.90	64.78	65.45	65.19	64.92	65.00	64.97	64.98
8 classes	65.90	64.76	65.38	65.41	65.94	65.98	65.29	65.49
16 classes	**66.28**	65.65	66.49	66.69	66.56	66.72	66.48	66.45
32 classes	66.12	66.27	**67.12**	66.91	66.80	66.95	67.13	66.80
64 classes	65.56	66.21	66.78	66.97	67.24	67.14	66.80	67.00
128 classes	64.06	**66.31**	66.91	66.95	67.25	67.27	67.58	67.39
256 classes	57.95	65.75	66.98	**67.15**	**67.53**	**67.55**	**67.73**	**67.80**

Table 8.16 ASR accuracy of univariate feature compensation for TIMIT LP4kHz and different number of classes and adaptation files available. The best result is highlighted for each amount of adaptation files.

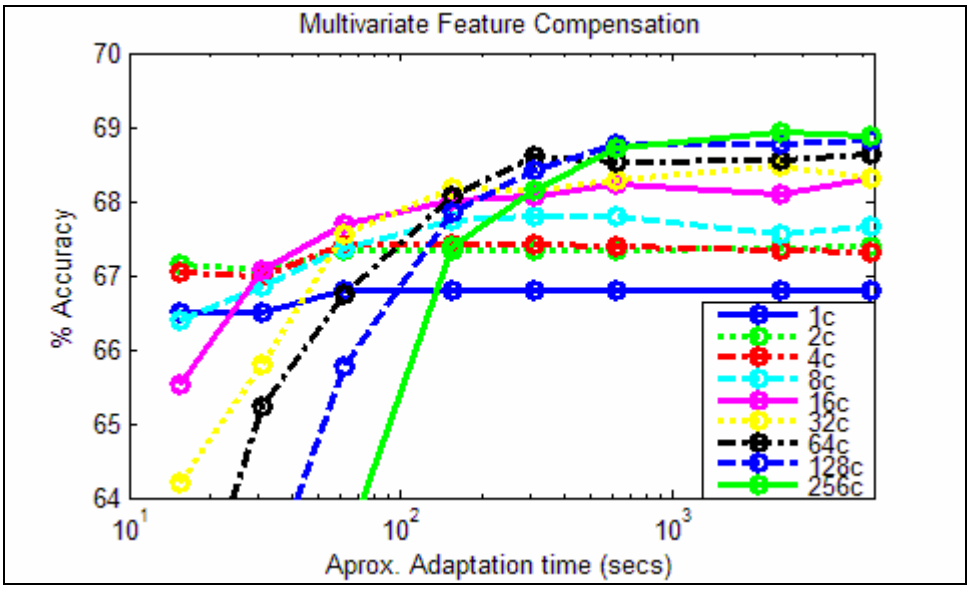
Number of adapt. files → / Number of classes ↓	5	10	20	50	100	200	800	1718
1 class	66.51	66.49	66.79	66.79	66.79	66.79	66.79	66.79
2 classes	**67.16**	67.06	67.34	67.35	67.33	67.35	67.36	67.38
4 classes	67.03	66.98	67.41	67.41	67.43	67.38	67.33	67.31
8 classes	66.39	66.85	67.36	67.75	67.79	67.81	67.56	67.66
16 classes	65.54	**67.08**	**67.70**	68.02	68.08	68.24	68.10	68.30
32 classes	64.20	65.81	67.55	**68.17**	68.14	68.29	68.46	68.30
64 classes	61.96	65.22	66.74	68.06	**68.61**	68.53	68.56	68.63
128 classes	54.21	62.73	65.77	67.84	68.41	**68.77**	68.78	68.82
256 classes	27.46	56.15	63.27	67.38	68.15	68.72	**68.94**	**68.89**

Table 8.17 ASR accuracy of multivariate feature compensation for TIMIT LP4kHz and different number of classes and adaptation files available. The best result is highlighted for each amount of adaptation files.

Several interesting conclusions may be drawn. Firstly, the superiority of multivariate compensation over univariate compensation is proved again. The number of free parameters to adjust in multivariate compensation is several times larger than in univariate compensation (depending on the average number of coefficients used for multivariate compensation). Thus, we may, for example, compare performance of univariate compensation with 256 classes and multivariate with 32 classes (same number of free parameters if 8 coefficients are used in average for multivariate compensation), where the latter outperforms univariate compensation both, in the saturated region and when training data is scarce. In terms of computational efficiency, multivariate and univariate compensation are similar for a fixed number of classes (as will be shown in Section 8.6.2, online compensation time is not highly increased using multivariate instead of univariate compensation. Increasing the number of classes is much more expensive). In this sense, it would make more sense to compare performance of univariate and multivariate compensation for the same number of classes, and in this case the difference is much larger (note that these experiments are for a LP4kHZ distortion and the differences would be even larger for more severe distortions, as was shown in Table 8.7). Finally, from Figure 8.11, it may be observed that for any number of classes, differences in performance when there is enough training data are smaller in multivariate compensation than in univariate compensation, probably because the number of free parameters is larger (multivariate compensation with 1 class has about the same number of free parameters than univariate compensation with 8 classes).



a)



b)

Figure 8.11 ASR accuracy of a) univariate and b) multivariate feature compensation for TIMIT LP4kHz and different number of corrector classes and amounts of adaptation data.

A general conclusion valid for both approaches is that feature adaptation performance is degraded due to overfitting to the training data, when the number of classes trained is large and the amount of training data is limited. The same effect was observed in our experiments in Chapter 6 for MLLR model adaptation (Table 6.1) and the solution should be the same: setting minimum occupancy thresholds for Gaussian classes, so that the number of classes is kept low in conditions of data scarcity. In our experiments we have used thresholds for model adaptation, but not for feature compensation.

In the following experiment we compare performance of Gaussian-based correction and model adaptation for different amounts of training data. Gaussian-based correction results are given for univariate (32 classes) and multivariate (2, 32 and 256 classes) modes (Table 8.18 and Figure 8.12). Three model adaptation strategies are considered: global MLLR (glob), 28-class MLLR (MLLR) and MAP. They are combined in several ways: a) glob + MLLR, b) MAP and c) glob + MLLR + MAP.

Number of adapt. files → **Mode ↓**	**5**	**10**	**20**	**50**	**100**	**200**	**800**	**1718**
Univariate 32 class	66.12	66.27	67.12	66.91	66.80	66.95	67.13	66.80
Multivariate 2 class	**67.16**	**67.06**	67.34	67.35	67.33	67.35	67.36	67.38
Multivariate 32 class	64.20	65.81	**67.55**	**68.17**	68.14	68.29	68.46	68.30
Multivariate 256 class	27.46	56.15	63.27	67.38	**68.15**	**68.72**	**68.94**	**68.89**
glob + MLLR	66.08	66.90	66.96	67.14	67.31	67.91	68.27	68.30
MAP	46.67	50.34	52.30	54.73	57.84	59.93	61.81	62.16
glob + MLLR + MAP	66.02	66.77	66.50	66.45	66.66	67.58	68.38	68.61

Table 8.18 Comparison of ASR accuracy for feature compensation and model adaptation methods with scarce adaptation data for TIMIT LP4kHz and different number of classes and available training files. For each number of training files the best solution is highlighted.

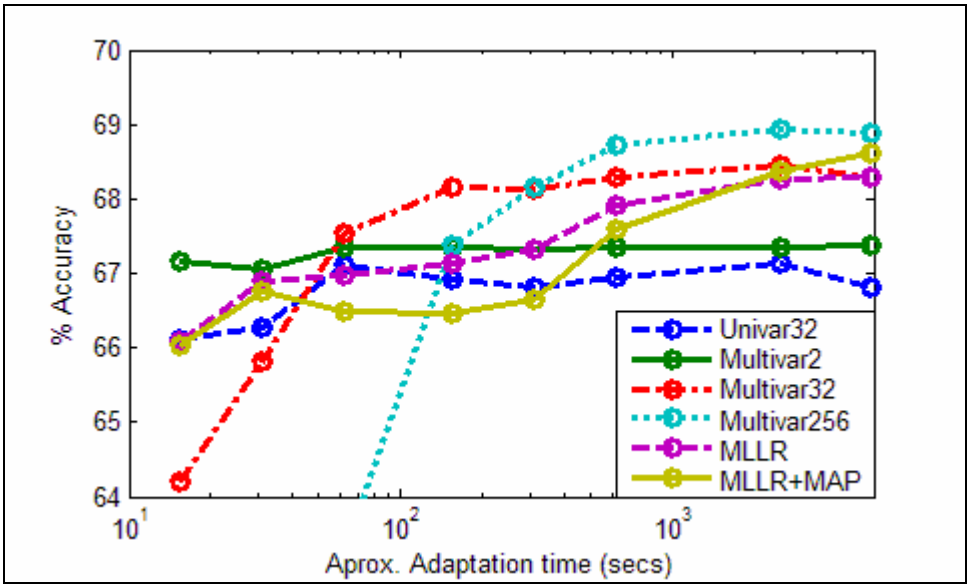


Figure 8.12 ASR accuracy vs. available adaptation data for feature compensation and model adaptation. Data corresponds to Table 8.18. Adaptation time is approximate for an average file length of 3.1 seconds.

In Figure 8.12 and Table 8.18 three regions are observed:

a) For very limited amounts of data (below 50 seconds) multivariate feature compensation using 2 classes outperforms all the other approaches and clearly beats model adaptation. On the other hand, multivariate compensation with 32 classes is worse, but improves very fast and compensation with 256 classes performs very poorly. This undoubtedly reflects the problem of overfitting to a small number of training data. Model adaptation techniques using a minimum occupancy threshold (MLLR) perform only slightly worse than multivariate compensation with 2 classes, showing good robustness against training data scarcity. Clearly, our multivariate feature compensation results could be improved by including a minimum occupancy threshold in order to limit the number of Gaussian classes to use, according to the amount of training data available.

b) Between 50 and 400 seconds the figure shows better performance of multivariate feature compensation using 32 classes (while performance for 2 classes is saturated and performance for 256 classes is rising quickly).

c) When the amount of training data is large (over 500 seconds) all the proposed methods seem to be converging to their respective points of saturation. In the considered setup multivariate feature compensation with 256 classes outperforms all the other approaches.

In summary, it has been shown that if a minimum occupancy threshold is applied to multivariate feature compensation in order to limit the number of Gaussian classes in accordance to the amount of data available, this approach is a valid alternative to model adaptation for bandwidth limitations (in our experiments a better alternative), even in the case of data scarcity; a situation that could well happen in a scenario of multiple distortions as those discussed in Sections 8.4.1 and 8.4.2.

8.5 Combination of Robustness Approaches

In this section we propose the combination of robustness approaches for increased ASR accuracy. In particular we propose two types of combination: a) feature compensation over CMN features and b) model retraining or model adaptation using compensated features. In Table 8.19 results are shown for a variety of band-limitations.

From the table, it is clear that feature compensation over CMN feature vectors does not produce significant differences compared to feature compensation only. This result seems to indicate that feature compensation is already capable of incorporating the effect of CMN and no further improvements may be obtained by combination of these two methods.

However, it is clear that model retraining and model adaptation on feature-compensated data improves performance compared to using independently model-side robustness or feature compensation. Combination of feature-side and model-side approaches performs better than feature compensation only because the system may adapt itself to the artificial effects introduced in the process

Mode	Distortion	% Corr.	% Acc.	Distortion	% Corr.	% Acc.
No Compensation	**Full-Band**	75.40	71.18			
CMN		75.71	71.61			
No Compensation	**LP6kHz**	64.32	58.30	**BP300-3400 Hz**	41.13	32.67
CMN		74.30	69.95		60.91	54.71
MLLR+MAP Adapt		75.46	70.85		70.63	64.90
Matched		75.45	71.03		71.86	65.73
Multivariate		75.22	70.95		70.62	64.79
CMN + Multivariate		75.47	71.22		70.12	64.31
Mvariate + glob Adapt		75.21	70.94		70.67	64.75
Mvariate + MLLR		75.28	71.04		70.75	65.00
Mvariate + MLLR+MAP		75.30	71.11		70.66	65.14
Mvariate + Retrain		75.61	71.14		73.05	66.87
No Compensation	**LP4kHz**	55.93	44.67	**UAM-TIMIT**	30.98	21.23
CMN		68.00	62.28		51.59	46.98
MLLR+MAP Adapt		73.57	68.64		62.63	58.26
Matched		74.73	69.33		69.10	61.80
Multivariate		73.16	68.46		64.67	58.79
CMN + Multivariate		73.14	68.34		64.80	58.66
Mvariate + glob Adapt		73.56	68.80		65.98	59.32
Mvariate + MLLR		73.72	69.00		66.44	59.89
Mvariate + MLLR+MAP		73.80	69.20		66.25	60.12
Mvariate + Retrain		74.99	69.83		71.32	63.96

Table 8.19 Comparison of different individual approaches (CMN, matched models, model adaptation and multivariate feature compensation) with combined methods. Multivariate compensation results for BP300-3400Hz and UAM-TIMIT are with 256 classes, while those for LP6kHz and LP4kHz are with 32 classes. Relevant comparisons are shaded using the same colors.

of feature compensation. Compared to model-side robustness only, the improvement is probably related to the reduction of data variability caused by feature compensation. This is a very important result because it represents another utility of our feature compensation methods for bandwidth compensation; up to this point we had compared our feature compensation methods against model-side methods (particularly model adaptation) and had shown its superiority in some situations. However, there are other situations very well suited to model adaptation (such as situations where band limitation is done by an approximately fixed and well known channel as is the case with a landline telephone channel). This result shows that our feature compensation methods can also be applied in these cases, not as an alternative to, but complementing model adaptation or model retraining with substantial improvements over only model adaptation or retraining (for UAM-TIMIT using the two approaches combined yields an improvement of ~2% absolute accuracy over using only mode adaptation or retraining only).

8.6 Analysis of Memory and Computational Costs

This section shows a qualitative study on computational and memory requirements of model-side and feature-side approaches. It should be noted that a wide variety of settings exists for these techniques and thus, memory and computation load may vary significantly. Moreover, the model-side approaches tested in this Thesis employ HTK tools, which although not extremely optimized are probably more time-efficient than our own algorithms for which saving time or memory was not a priority during development. For these reasons, results in this section are interesting for a qualitative comparison, but are probably biased against our own algorithms.

8.6.1 Memory Cost

A complete HMM system stored in memory is approximately a collection of the following floating point numbers:

$$HMM\ \ engine = Models \cdot \left(States \cdot Mixtures \cdot 2 \cdot Dimension + States^2 \right), \tag{8.1}$$

where factor 2 is because we store the vector of means and vector of variances (assuming diagonal covariance) and $States^2$ accounts for the transition matrixes. When several HMM model sets are stored for use in multi-environment applications (for example in model retraining or MAP adaptation), the memory cost is multiplied by the number of environments.

In the case of MLLR adaptation, the adapted HMM sets can be stored in full with the same computational cost as in Eq. (8.1) for each set, or alternatively, the base HMM model set can be stored along with adaptation matrixes for each class and distortion, saving space when several distortions exist:

$$MLLR\ system = HMM\ \ engine + Distortions \cdot Clusters \cdot Dimension^2 . \tag{8.2}$$

Similarly, in feature compensation, we store a single HMM model set and a set of Gaussian classes and transformation matrixes. In univariate compensation the cost is:

$$\begin{aligned} Univar\ comp = HMM\ \ engine + Distortions \cdot Clusters \cdot \\ \cdot \left\{ 2 \cdot Dimension + \frac{Dimension}{3} \cdot \left(fitOrder + 1 \right) \right\}, \end{aligned} \tag{8.3}$$

where $2 \cdot Dimension$ represents the part due to means and covariances for each class and the rest is for corrector functions of static features.

In multivariate compensation the cost is:

$$Multivar\ comp = HMM\ engine + Distortions \cdot Clusters \cdot \cdot \left\{ 2 \cdot Dimension + \frac{Dimension}{3} \cdot \left(Average\ fit\ size \right) \right\}, \tag{8.4}$$

where *Average fit size* defines the number of coefficients used in average for compensation using multivariate compensation.

In Table 8.20 we give numbers to equations from (8.1) to (8.4), for a hypothetical context-independent phoneme-based HMM engine with 50 acoustic models, each with 3 states, 15 mixtures per state and 39 features per observation. The first column represents the cost of storing whole HMMs. The second is for MLLR adaptation assuming 30 clusters. The third column shows values for univariate compensation with 30 clusters and the fourth column multivariate compensation with 30 clusters and an average 8 terms in multivariate compensation. *D* represents the number of distortions the system may encounter.

Full HMMs	MLLR	Univar Feat. Comp.	Multivar Feat. Comp.
$D \cdot$ 1.76e+5	1.76e+5 + $D \cdot$ 4.60e+4	1.76e+5 + $D \cdot$ 3.12e+3	1.76e+5 + $D \cdot$ 5.46e+3

Table 8.20 Comparison of number of parameters stored for different robustness methods. *D* represents the number of distortions considered for the system.

Even in this case of context-independent models, feature compensation offers important memory savings compared to storing full sets of HMMs and to storing MLLR transforms for each distortion, particularly for systems subject to a variety of possible distortions. For systems using context-dependent models as is normally the case in Large Vocabulary Continuous Speech Recognition (LVCSR), these differences would be even larger [Morales, 2004].

8.6.2 Computational Cost

Computational costs may be divided into two categories depending on whether they can be pre-computed offline or if they need to be done online. The first category is normally included in system training and although reducing computational demands is always desirable it is normally not critical in this stage. On the contrary, saving time in online operations is usually more important.

Training new models for each environment requires running multiple iterations of the Baum-Welch algorithm [Rabiner, 1989]. This process is very costly, but it may be optimized by pruning and other simplifications.

Training MLLR requires imputation of observations in the training set to a Gaussian mixture in the HMM system (this is normally done running recognition prior to adaptation, or passing labels to the recognizer and performing alignment). Updating the mean vectors is then a low-consuming process.

In feature compensation the first step is creation of Gaussian classes. This process is simpler than Baum-Welch training and does not require data labels. Once classes are computed, feature compensation functions are trained for each of them in a process consisting of two stages: first, each

observation is imputed to the Gaussian class for which likelihood is maximal and second, linear regression or multivariate regression is used for finding the corrector coefficients.

As for online operations, when model-based approaches are used for robust recognition there is no computational cost, as long as only one recognition system is used. However, when multiple recognizers are run simultaneously (for recognition of data from unknown environments) computation time is multiplied by the number of distortions. Another possibility for model robustness is increasing the number of Gaussian mixtures, which could be useful when multi-environment training is considered, and this would also increase computation time.

In feature compensation input features need to be compensated in real time, increasing recognition time.

Given the complexity of offline and online operations, no explicit analysis on the number of operations will be made. On the contrary, we consider a practical comparison using HTK tools for model-side operations and our own code for feature compensation in a Pentium IV at 3 GHz and 1 GB of RAM (Table 8.21). In the first column we show the time required for training acoustic models from scratch (using the whole training partition of TIMIT), MLLR adaptation and Gaussian class creation for 30 and 256 clusters (using approximately $1/6^{th}$ of the training partition). In the second column we show the time-cost for computation of corrector functions for univariate and multivariate compensation and in the third column, the cost of online feature compensation is given in average time per file.

The cost of online operations in feature compensation is small in general. Only in the case of multivariate compensation with 256 classes there is a significant cost: an average $1/20^{th}$ of an utterance's real time. However, when multiple distortions exist the cost of feature compensation would most likely be smaller than that of using multiple decoders with specific models for each environment.

Offline operations				**Online operations**	
Full HMMs train (~ 14510s)	9605				
MLLR Adapt* (30 clusters)	236	**Computation of Corrector Functions ***		**Avg. correction time per file (Average file length is 3.1s)**	
Gauss. class creation* (30 class)	270	**30 class Univariate**	55	**30 class Univariate**	0.0149
		30 class Multivariate**	1066	**30 class Multivariate**	0.0190
Gauss. class creation* (256 class)	8277	**256 class Univariate**	155	**256 class Univariate**	0.1190
		256 class Multivariate**	9100	**256 class Multivariate**	0.1560

*Available training data is ~2480s.
** Multivariate corrector functions are computed with Matlab, increasing computation time.

Table 8.21 Comparison of computation time (in seconds) for offline and online operations for different robustness approaches.

8.7 Summary and Conclusions

In this chapter we have evaluated performance of ASR systems using a recognition engine trained with full-bandwidth speech and tested with band-limited data over which Gaussian class-based feature compensation is applied. Gaussian-based compensation proved to be a better method than Phoneme-

based compensation (studied in Chapter 7), both in terms of system accuracy and simplicity of implementation.

Results are shown for different types of distortions, including real telephone data, and in all cases, feature compensation was competitive with model-side approaches (in our experiments multivariate feature compensation employing a large number of classes outperformed standard model adaptation using MLLR and MAP; Sections 8.3.2 and 8.4.3). We also showed the generality of feature compensation by testing and validating our results for a word-based LM (Section 8.2.2) and for a Castilian Spanish corpus (Section 8.2.3).

Different feature compensation modes and settings have been evaluated. The best performing setup is multivariate feature compensation using non-stereo partitioning of the feature space, stereo-based training of the corrector functions and reconstruction of static features only (Sections 8.3.2 and 8.3.3). However, we also showed that it is possible to train corrector functions without the constraint of stereo-data at only a small cost in accuracy (Section 8.3.4). Based on the availability of training data, more or less classes should be used, and we hypothesized that application of a minimum occupancy threshold would solve the problem of overfitting observed when a large number of classes is trained using only a limited amount of training data from a distortion. In our experiments, feature compensation outperformed model adaptation for all amounts of training data assuming that the number of corrector classes is chosen according to the amount of training data (Section 8.4.3).

We studied a particular situation where different band-limitations corrupt speech utterances (time-varying distortions). In this case, corrector functions trained for compensation of different distortions may be used to compensate multiple-distortion feature vectors so that the decoder module remains unaltered and unaware of the multiple distortions affecting speech. We argued that the same problem would require a more sophisticated approach for model-side solutions (probably running multiple recognizers or modifying the decoder module) and in our experiments feature compensation offered much better performance (even when test data is affected by band-limitations unseen during training; Sections 8.4.1 and 8.4.2).

It has been shown that model-side and feature-side robustness methods could be combined for better performance than that with any of the individual approaches (Section 8.5), thus expanding the utility of our proposed feature compensation methods for situations well suited for model-side robustness.

Finally, we made a qualitative analysis of memory and computational costs for the different approaches considered and we argued that when systems are subject to multiple distortions, feature compensation would be very competitive in both aspects (Section 8.6).

9

Summary, Conclusions and Future Work

This Thesis studies feature compensation of band-limited speech, where parts of the signal spectrum are completely removed, and this with the goal of ASR. Our work started with a review of related robustness techniques in ASR and the proposition of a model of the effect of band-limiting distortions over speech features. Several strategies for compensation were proposed and their practical implementations were described. Finally, an extensive experimental and discussion section showed how the proposed techniques perform and how they compare to or complement other classical techniques for a wide variety of conditions and constraints.

In this chapter we summarize the main results of our research, outline the main contributions of this Thesis and finish with suggestions for future work.

9.1 Summary of Results

The outline of this dissertation has been designed with the goal of organizing our ideas, mathematical developments and experiments in a rational and comprehensible way. However, the process in which this work evolved was obviously less structured as is more or less the case in all research works, where findings influence ideas and ideas make more findings possible. In this section we summarize our results in a pseudo-chronological manner that will allow the reader to follow the flow of thoughts that guided us in the course of our work.

The topic of feature compensation for band-limiting channels was suggested by Prof. Hansen in our collaboration in 2004. This seemed to be an interesting approach to the problem of ASR of historical spoken document retrieval, as part of the NGSW project [NGSW] [Hansen, *et al.*, 2004], dedicated to the creation of an online library of spoken word collections of the 20th century. Historical recordings are typically band-limited due to limitations of the audio devices used at the time of recording, and their characteristics are most of the times different from one recording to another. Thus,

training specific models for historical recordings may not be a valid option. In addition, time-varying band-limitations and rapid changes in the spectral characteristics of speech were observed in cases such as documentaries, broadcast news, etc., where historical recordings or telephone-transmitted speech from correspondents are mixed with high-quality recordings. In all these situations, model retraining or even adaptation may be problematic due to training data scarcity and the need to integrate recognition of multiple distortions in a single system. Alternatively, feature compensation may solve some of these problems, or may be combined with model robustness techniques.

We first evaluated Phoneme-based partitioning of the feature space. The mathematical model presented in Section 3.4 was developed and Phoneme-based partitioning seemed a reasonable method for dividing the feature space into regions with similar spectral characteristics. In order to evaluate the potential of this technique, we first evaluated an Oracle approach. This is not directly applicable to real cases because it requires phonetic labeling of speech, which is the goal of ASR. Nevertheless, results with Oracle feature compensation were very promising (even for very important distortions such as LP2kHz) and motivated further research. Real implementations were created in the form of General compensation and 2-stage compensation whose performance was promising and moderately close to that with model adaptation for small distortions.

As an alternative to Phoneme-based partitioning of the feature space, which is a knowledge-based criterion, we proposed the use of a data-driven method. Thus, we borrowed the concept of Gaussian-based feature compensation previously employed for noisy speech and adapted it to the case of band-limitations. Mathematical foundations and practical issues related to this approach are shown in Section 5.5. This solution resulted in a significant performance improvement and motivated a large number of experiments that show the performance of different settings for feature compensation as well as other techniques, such as CMN, model adaptation and model retraining. These results constitute the core of the experimental section and are one of the main contributions of this Thesis, as they provide us with an important knowledge on the potential of the different approaches using always a common experimental framework. Among the most important results are those for time-varying environmental distortions, where it was shown that when the band-limitation affecting speech may vary in time (even rapidly and abruptly), Gaussian-based feature compensation allows identification and compensation in a unified framework. This is even possible with automatic identification of the type of band-limitation, at the cost of very little accuracy decrease (Section 8.4.1), and even more interestingly, when the actual distortion is not available during training, but may be roughly represented by a combination of other distortions (Section 8.4.2). Additionally, we showed that when adaptation data is scarce, performance with feature-side techniques is similar or even better than model-side methods (Section 8.4.3).

At that point in our research we realized that probably univariate feature compensation was not the optimal solution. We had made the observation that the information lost in the removed channels could be repaired using redundant information contained in the remaining channels. Nevertheless, MFCC coefficients are the result of a combination of all the filterbank outputs, and they are typically assumed uncorrelated. Therefore, in the past it had been typical to use univariate feature compensation of MFCCs. However, we showed (both theoretically and empirically) that MFCCs of band-limited speech are more correlated than those of full-bandwidth speech, and so in our case it makes sense to

use multivariate compensation (Section 3.5). Experimental results with multivariate feature compensation boosted performance and constitutes one of the main practical results of our work. Performance of multivariate feature compensation proved to be very similar to that of model-side approaches and as previously stated may be very beneficial in practical situations where model-side approaches are difficult to apply such as multi-environment situations or time-varying channels (Section 8.3.2).

The success of the proposed techniques in artificial band-limitations encouraged us to evaluate their performance on real distortions, and in particular in telephone channels. For this purpose we employed NTIMIT [Jankowski *et al.*, 1990], the telephone version of TIMIT. However, results were not completely satisfactory, not so much for the difficulty of a real telephone channel, but for the variability observed in different files in NTIMIT. The reason for this variability is the fact that each utterance is transmitted through an individual telephone call with a particular geographical destination (thus each call constitutes a different transmission channel). In order to avoid this inconvenience, we created UAM-TIMIT [Morales *et al.*, 2007b]. This corpus is also a re-recording of TIMIT through a telephone channel, but the whole corpus is transmitted in a single telephone call. With this, we were able to evaluate performance using a real, but unique telephone distortion. Results were now close to those of an artificial band-pass filter BP300-3400Hz, the theoretical bandwidth of a telephone channel.

In the final stages of our work we evaluated the possibility of combining different robustness approaches. Combination of CMN and feature compensation did not outperform results with feature compensation only, indicating that feature compensation is sophisticated enough to include the benefits of CMN. On the contrary, combination of feature-side and model-side techniques (model adaptation or retraining) offered significant improvements over each of the individual methods, especially for more severe distortions. This expanded the utility of the our feature compensation approaches, which therefore are not only an alternative to model-side robustness techniques for situations in which the latter are difficult to apply, but also a complement to these techniques for situations well suited to them.

In summary, the methods proposed to compensate bandwidth limitations as a result of this Thesis can be considered in two ways with respect to the more classical solution of model retraining and model adaptation:

- They can be considered an alternative to model-side compensation of band-limited speech. In cases where plenty of data is available for training, performance of our feature-based methods is slightly inferior to model-side compensation. However, when adaptation or retraining data is scarce, our methods seemed to outperform model adaptation and model retraining. Moreover, our methods have the ability to blindly identify and compensate unknown bandwidth limitations, even for rapidly time-varying bandwidth limitations. Finally, our methods have a clear advantage for embedded systems because we can use a single model set trained on full bandwidth speech and do recognition on a wide variety of bandwidths with a single model set, thus allowing for small footprint recognizers.
- They can also be considered a complement to model-side compensation of band-limited speech. Even if there is enough data for model adaptation or model retraining and the

bandwidth is fixed and known, the combination of our feature-based compensation methods and model-side techniques yields better performance than either approach alone.

9.2 Major Contributions

In this section we summarize the major contributions of this Thesis:

- We have presented a novel mathematical model for band-limiting distortions and their effect on cepstral feature speech representations. This model has allowed us to infer compensation strategies that have been later exploited (Section 3.4).
- We have predicted theoretically and showed empirically that MFCC coefficients of band-limited speech are more correlated than those of full-bandwidth speech (Section 3.5). This novel result suggested a new feature compensation approach, multivariate compensation, that provides a significant accuracy increase over univariate compensation, typically used in compensation of noise-corrupted speech (this has been empirically verified in the experimental Sections 8.3.2 and 8.4.3).
- A variety of feature compensation techniques have been proposed for the problem of band-limitations. These roughly consist of classification of distorted speech into classes and application of corrector functions. Two major techniques have been proposed for partitioning, one is knowledge-based (Phoneme-based partitioning; Sections 4.1 and 4.5.3 and Chapter 7) and the other data-driven (Gaussian-based partitioning; Sections 4.2 and 4.5.2 and Chapter 8). The former is, to the best of our knowledge, absolutely novel. The latter is based on ideas borrowed from the field of feature compensation for robustness against additive noise, but in this Thesis it has been applied for the first time to the problem of bandwidth limitation. Two correction strategies have also been proposed: univariate polynomial compensation and multivariate compensation (Section 4.5.1). The first one is similar to the feature compensations used in the field of feature compensation for robustness against additive noise, but has been extended to higher orders in this Thesis. The latter (novel again) has been motivated by our theoretical explanation of the increased correlation between MFCC features of band-limited speech and has proved to clearly outperform univariate compensation of band-limited features.
- Extensive experiments have been conducted in a variety of conditions that have allowed us to obtain a clear idea of the advantages and disadvantages of the proposed methods (Chapters 7 and 8). Comparisons have always been made with other robustness methods. Of particular interest is the study of situations where our feature compensation approach could be more appropriate than model-side solutions, such as systems subject to time-varying bandwidths, bandwidths unseen during training, limited amounts of adaptation data, etc. (Section 8.4).
- We have combined our feature-based compensation with model-based compensation showing experimentally that our feature-based compensation methods can be seen not only as alternatives to model-side compensation, but also as complements, achieving better recognition accuracy when both compensation approaches are combined (Section 8.5). In this

way, our feature-based compensation techniques have proved to be useful not only in special situations not very well suited to model-side techniques, but also in more common situations where the bandwidth limitation is known and fixed.

- A corpus has been designed and recorded during the course of this research that may help other researchers for evaluation of ASR and related tasks on telephone channel speech (Annex A). At the time of writing we are pursuing publication of this corpus through the LDC.

9.3 Future Work

This Thesis may find a continuation in a number of research lines. We consider the following of special interest:

- In the experimental section of this work we restricted ourselves to compensation of MFCC features. This is motivated by a desire to simplify the use of these techniques in state-of-the-art recognizers, most of which use this type of parameterization. However, in future studies it would be worth studying compensation in non-cepstral spaces where the spectral characteristics of speech would be more evident [Kim and Hansen, 2006].
- In a similar direction, it is interesting to further explore the possibility of canonical speech representations optimal for robustness against band-limiting distortions. Typically, one of the two elements compared in the pattern matching module (acoustic models or input speech) is fixed and the other is modified and accommodated to match the fixed element. However, another possibility would be to design a canonical space where the characteristics of both parts would be more easily matched. An example of such approach is CMN, but, as shown in our experiments it is not powerful enough.
- In the discussion section we argued that the method employed for partitioning the feature space was sub-optimal in the sense that it was not aimed at the goal of feature compensation using linear combinations of the available features. Stereo-based partitioning and other partitioning criteria directed to the goal of feature compensation may significantly improve modeling of the distortions and consequently the quality of reconstruction. An interesting possibility would be to combine knowledge-based and data-driven methods to perform a partitioning of the space oriented towards the goal of feature compensation.
- In our experiments, accuracy of feature compensation on NTIMIT was far worse than that on data subject to artificial filters, or even UAM-TIMIT. The reason is most likely the channel variability present in NTIMIT, as a result of it being recorded in many different calls. It would be interesting to try new solutions in order to improve accuracy in this difficult task. For example, if it was possible to group together utterances in NTIMIT according to similar channel characteristics we could employ IEC multi-environment partitioning (as shown in Section 4.5.4), which was a better approach than MEC, currently being used for NTIMIT.
- In Section 3.2 we presented the spectral characteristics of groups of phonemes and argued that depending on the type of band-limitation some phonetic classes could be more affected than others (informal analysis of confusion matrices, not presented in our work, has shown this

predictable result, too). For example, fricatives may contain important differentiating information in higher frequencies that will disappear when the signal is subject to low-pass filters. Although specific measures for particular phonetic classes have not been implemented, this would probably allow for significant improvements.

- MFCC features are a convenient and easy-to-use representation of speech that has proved to be quite robust. However, part of the information in the original signal that MFCCs reject could be useful for particular tasks. For example, it has been shown that phase information (unavailable in the MFCCs because they use power spectra) may provide significant benefits for understandability. Experiments on human listeners showed that the phase is particularly important when the SNR of noise corrupted speech is low [Liu *et al.*, 1997] [Shi *et al.*, 2006]. Tests on ASR systems also proved that accuracy could be improved by combination of MFCC features and phase information [Schlüter and Ney, 2001]). Thus, it would be interesting to study the possibility of using more advanced front-ends that include phase or other forms of information that might be useful for feature compensation, too.

Additionally and although more vague than the previous ideas, we also believe that in order to improve ASR robustness, parts of the typical system architecture could be modified. Although it is unclear whether ASR systems should follow the same principles as the human ear, this is a convenient source of inspiration. Among the almost countless possibilities we mention the following with particular focus in band-limitations:

- Multiple-level and multiple-pass ASR systems with feedback. Human's understanding is a complex process where different levels of information are analyzed and it is possible to return to previous parts of a message that were incorrectly classified (for example, keywords help reducing the target vocabulary). Additionally when the characteristics of the speaker are identified, users are able to compensate parts of the speech according to their experience in similar situations (dialectal differences, for example). Also, humans seem to be able to compensate (or ignore) missing parts of the spectrum, or give less weight to parts of the speech that are less discernable, as a result of a particular distortion.
- Based on the observation that human listeners are capable of understanding sub-bands of speech or combine them for better performance [Warren *et al.*, 1995] [Allen, 1994], we also propose multi-band recognition as a promising approach to deal with band-limited speech [Bourlard and Dupont, 1997] [Hermansky *et al.*, 1996].

9.4 Main Publications

The following is a list of the main publications made in the course of this Thesis, constituting the basis of this dissertation:

- N. Morales, J.H.L. Hansen and D.T. Toledano. MFCC compensation for improved recognition of filtered and band-limited speech. *Proceedings ICASSP'05, volume 1, pages 521-524*. March, 2005.
- N. Morales, D.T. Toledano, J.H.L. Hansen, J. Colás and J. Garrido. Statistical class-based MFCC enhancement of filtered and band-limited speech for robust ASR. *Proceedings Interspeech'05, pages 2629-2632*. September, 2005.
- N. Morales, D.T. Toledano, J.H.L. Hansen, J. Garrido and J. Colás. Unsupervised class-based feature compensation for time-variable bandwidth-limited speech. *Proceedings ICASSP'06, volume 1, pages 533-536*. May, 2006.
- N. Morales, D.T. Toledano, J.H.L. Hansen and J. Colás. Blind feature compensation for time-variant band-limited speech recognition. *IEEE Signal Processing Letters, volume 14, issue 1, pages 70-73*. January, 2007.
- N. Morales, D.T. Toledano, J.H.L. Hansen and J. Garrido. Multivariate cepstral feature compensation on band limited data for robust speech recognition. *Proceedings NODALIDA'07, pages 144-151*. May, 2007.
- N. Morales, J.H.L. Hansen, D.T. Toledano and J. Garrido. MFCC enhancement techniques for robust speech recognition of bandwidth limited speech. *Speech Communication*. Major revision in progress.
- N. Morales, D.T. Toledano, J.H.L. Hansen and J. Garrido. Feature compensation as an alternative or complement to model adaptation and model retraining for bandwidth-limited speech. *In preparation for IEEE Transactions on Audio, Speech and Language Processing.*
- N. Morales, J. Tejedor, D. T. Toledano and J. Garrido. UAM-TIMIT: Sending TIMIT through a real and single telephone channel. *In preparation for Conference on Language Resources and Evaluation 2008*.

Although not exactly part of this Thesis, another work in ASR robustness is:

- N. Morales, L. Gu and Y. Gao. Adding noise to improve noise robustness in speech recognition. *Proceedings Interspeech'07, in print*. August, 2007.

Annex A
UAM-TIMIT: Generation of a Single-channel Telephone Corpus

In this annex we describe the motivation and process of creation of the corpus UAM-TIMIT [Morales *et al.*, 2007b]. The database is derived from the TIMIT corpus [Fisher *et al.*, 1986], by passing the original speech files through the Public Switched Telephone Network (PSTN) [ITU-T, 2001]. The whole database (6300 utterances and a total size of approximately 323 minutes) was re-recorded in a single telephone call, with the goal of maintaining the channel conditions as stable as possible in the derived corpus.

We are currently in contact with the LDC for the distribution of UAM-TIMIT.

A.1 Motivation

One of the challenges of this Thesis was the application of the proposed feature compensation techniques to real data. Experimentation in constrained artificial conditions, allows precise control of the different distorting conditions (channel distortions, additive noise, task complexity, etc.) and simplifies the analysis of results. In addition, it is easy to generate data affected by a particular artificial distortion by simply applying the desired filter on undistorted data. However, when the proposed algorithms are evaluated on data affected by real distortions, multiple effects are typically combined, complicating the analysis of results. In particular, when stereo-based compensation techniques were evaluated on real telephone data (for this purpose we used NTIMIT database, the PSTN version of TIMIT [Jankowski *et al.*, 1990] [Jankowski, 1991]), we observed a very significant performance degradation compared to an equivalent simulated condition (artificial band-pass filter of speech). Additive noise is an unavoidable nuisance in telephone communications and an obvious source of degradation, but stereo-based compensation techniques have proved in the past their ability to compensate noisy speech, provided that the SNR is high and the noise type stable [Moreno, 1996] [Droppo *et al.*, 2001]. Thus, we suspected that other unaccounted factors were in play.

Stereo-based compensation is trained with speech recorded simultaneously in the distorted and clean (full-bandwidth) environments, so that the system learns the relation between the two spaces and accordingly compensates data. As explained in Chapter 4, data from well-defined distorting environments should be used for training corrector functions appropriate to each distortion. However, in the case of NTIMIT the assumption of well-defined environment is not fulfilled, because the corpus was designed to characterize the PSTN standard in the widest range of possible conditions; utterances in the original TIMIT corpus were re-recorded after sending them in individual telephone calls involving all sorts of network combinations: long-distance calls, short-distance calls, local or transferred calls and a large number of receiving stations, or even telephone companies. Therefore, channel conditions vary greatly from one call to another and so, while NTIMIT is useful for tests on multiple telephone conditions, it does not present a well-defined and steady distortion. As shown in Section 4.5.4, multiple environment compensation is also possible, but we have observed that better compensation performance may be obtained when compensation classes are trained using well defined distorting environments. These classes may then be combined in a super set of classes at a later stage (we called this IEC).

While NTIMIT presented this inconvenience, it was still desirable to test our methods on real telephone data, where time-variable additive distortions exist (even the convolutional effect of the channel may vary slightly across time). To this purpose we designed UAM-TIMIT, a database built following the spirit of NTIMIT, but with the additional goal of obtaining a single and well-defined distorting environment. This is achieved by re-recording the whole corpus in a single telephone call. In addition, great care was taken during post-processing of the database for optimal alignment with the original TIMIT database, a desirable feature for stereo-based training and something that was not perfectly achieved in NTIMIT.

The rest of this chapter is a short summary of the characteristics of NTIMIT followed by the description of methodology and design specifications of the new corpus, UAM-TIMIT.

A.2 Methodology and Characteristics of NTIMIT

The PSTN is the sum of the world's public circuit-switched telephone networks, ruled by the standards of the ITU-T [ITU-T, 2001]. In the United States the network is organized in Local Access and Transport Areas (LATAs). When a call is made between 2 numbers in the same LATA the call is solely handled by the local telephone company. On the contrary, a call connecting different LATAs, is first handled by the originating local company, then by a long-distance carrier and finally by the receiving local company, in what is termed a long-distance call.

In NTIMIT calls originate from the NYNEX Science and Technology Laboratories in White Plains, New York. Half of the utterances were transmitted within the local LATA and the other half reached one of 10 selected LATAs. Thus, a wide range of channel conditions are represented in the database.

A known issue with NTIMIT is a small misalignment between the original corpus and their re-recorded equivalents. As explained in [Jankowsky *et al.*, 1990], for the purpose of alignment with TIMIT, sets of clicks were appended at the start and end of utterances. However, it is reported that this

method produced misalignments in approximately 10% of utterances. Misaligned utterances had to be retransmitted, and four passes were necessary for correct alignment of the complete corpus. Nevertheless, there have been some minor issues with the resulting corpus, such as incomplete utterances (at least 3 in the training set and 6 in the test set[1]) and a small misalignment difficult to perceive by human listeners. The average misalignment we have detected by maximizing the cross-correlation between TIMIT and NTIMIT files is 59.53 ± 14.98 samples per utterance. Thus, if the signal is parameterized using a window size of 25 ms the misalignment represents 14.88% of the window size. For many applications, this misalignment is irrelevant, but for training of corrector functions, system performance might be degraded. In order to gain insights on the impact that such misalignment may have on compensation using stereo-based techniques we designed the following experiment: misalignment was calculated for each file in NTIMIT (compared to its equivalent in TIMIT) and the same misalignment was applied to the corresponding file in UAM-TIMIT. Results in Table A.1 show that there is a significant degradation in ASR performance for a system were multivariate feature compensation is made using compensation classes trained with misaligned data.

Corpus	Compensation mode	% Corr.	% Acc.
Aligned UAM-TIMIT	Multivariate-32	62.53	56.78
	Multivariate-256	64.67	58.79
Misaligned UAM-TIMIT	Multivariate-32	61.45	55.78
	Multivariate-256	63.91	58.07

Table A.1. Speech recognition performance for multivariate compensation trained with two versions of UAM-TIMIT: the first one is perfectly aligned with TIMIT and the other has the same misalignment as NTIMIT.

A.3 Methodology of UAM-TIMIT

The following shows the process of preparation, execution and post-processing of the corpus UAM-TIMIT.

A.3.1 Dialogic Switchboard and Telephone Channel

Speech data was sent through the telephone network and recorded by means of a Dialogic D/41JCT-LS switchboard [D41JCT-LS datasheet]. One of the 4 integrated telephone lines was used as the caller end (sending speech data to the network) and another one as the receiving end (recording data). The process was handled using a voice platform designed in our group [Tomico *et al.*, 2003], capable of executing a variety of automatic telephone services. The telephone interface server module was developed using Intel's Dialogic System Release 5.1.1 software for Windows [Intel Dialogic]. The two lines were inside the building of the Escuela Politécnica Superior-UAM and the call was therefore handled locally.

In addition to the fact that the entire original corpus is passed in a single call, in our case we use the switchboard to generate directly the telephone signal at the calling end. On the contrary, in NTIMIT the signal was passed to the telephone line using a handset and an artificial mouth, placed in an acoustically isolated room [Jankowski *et al.*, 1990]. Thus, NTIMIT contains also the important convolutional distortions caused by the telephone microphone.

[1] This problem was reported by PhD. B. Pellom.

A.3.2 Data Preparation and Recording

A single audio file was created by concatenation of each file in the TIMIT corpus. However, given the limitation of total recording time in the Dialogic switchboard of a maximum of 6000 seconds, the process was divided in 4 fragments of approximately 4800 seconds (total size of TIMIT is around 19380 seconds). Each of the four chunks was preceded by two calibration tones: a fixed 1000 Hz tone and a linearly varying tone from 0 to 4000 Hz, both with a duration of 4 seconds (this is similar to the calibration tones existing in NTIMIT for each LATA). All 4 fragments were sent through the line in a single call, i.e., no hang-up was made between recording of each of them. Thus, in principle, the channel should remain stable throughout the duration of the recording of the whole database, aside from temporal variations. Nevertheless, in order to quantify these variations, we decided to introduce the aforementioned calibration tones at the start of each fragment and they may be used for assessment on possible variations on the channel conditions.

A.3.3 Post-processing Alignment with TIMIT

Utterances in the new corpus were originally obtained by splicing the single-call-recorded file according to the file sizes in TIMIT. However, we observed that this method generated small misalignments due to the slight difference between play output and recording sampling rates (the difference is approximately 1 sample every 16 seconds, or 1 sample every 128000 recorded samples; this is irrelevant for a particular utterance, but makes impossible alignment of the whole corpus with this method). Therefore, a more sophisticated approach was employed; in the release version of UAM-TIMIT alignment was made individually for each utterance by maximizing the cross-correlation function computed between the original TIMIT file and recorded file (Matlab function *xcorr*, from the Signal Processing Toolbox was used [MATLAB SPT]). With this method alignment errors were reduced to ± 1 sample per utterance.

A.4 Summary of Distortions Present in UAM-TIMIT

There are two main sources of distortion:

a) Digital/Analog - Analog/Digital conversions: The original digital signal is converted to analogical format by the D/A converter in the Dialogic switchboard, introducing a non-linear distortion not present in the original TIMIT files. Similarly, at the receiving end an additional distortion is introduced by the A/D converter. In an unpublished experiment in our group, we observed that for a system trained with a given set of utterances, ASR using re-recorded data using a SoundBlaster 5.1 Soundcard did not suffer a significant degradation compared to original data (in this experiment we used CMN and no adaptation of models).

b) Channel effects: These depend on the specific characteristics of the telephone channel used. Given the complexity of the network and the large number of hardware elements involved and causes of distortions (network load, weather conditions, etc.), these effects are difficult to predict and the best way to characterize them is by means of calibration tones or the stereo data itself.

References

H. Abut, J.H.L Hansen and K. Takeda (eds.) (2004). *DSP for in-vehicle and mobile systems*. Kluwer/Springer-Verlag. November, 2004.

A. Acero (1990). Acoustical and environmental robustness in automatic speech recognition. *PhD. Dissertation, Electrical and Computer Engineering Department. Carnegie Mellon University, Pittsburgh*. September, 1990.

M. Afify, X. Cui, Y. Gao (2007). Stereo-based stochastic mapping for robust speech recognition. *Proceedings ICASSP'07, volume 4, pages 377-380*. April, 2007.

J. B. Allen (1994). How Do Humans Process and Recognize Speech? *IEEE Transactions on Speech and Audio Processing, volume 2, issue 4, pages 567-577*. October, 1994.

B.S. Atal (1974). Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification. *Journal of the Acoustical Society of America, volume 55, pages 1304-1312*. June, 1974.

C. Avendano, H. Hermansky and E.A. Wan (1995). Beyond Nyquist: towards the recovery of broad-bandwidth speech from narrow-bandwidth speech. *Proceedings Interspeech'95, volume 1, pages 165-168*. September, 1995.

L.E. Baum, T. Petrie, G. Soules and N. Weiss (1970). A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *Annals of Mathematical Statistics, volume 41, issue 1, pages 164-171*. February, 1970.

B. Bogert, M. Healy and J. Tukey (1963). The quefrency alanysis of time-series for echoes. *Proceedings Symposium on Time Series Analysis, pages 209-243*. 1963.

S. Boll (1979). Suppression of acoustic noise in speech using spectral subtraction. *IEEE Transactions on Acoustics, Speech, and Signal Processing, volume 27, issue 2, pages 113-120*. April, 1979.

H. Bourlard and S. Dupont (1997). Subband-based speech recognition. *Proceedings ICASSP'97, volume 2, pages 1251-1254*. April, 1997.

P. Brown, C.-H. Lee and J. Spohrer (1983). Bayesian adaptation in speech recognition. *Proceedings ICASSP'83, volume 8, pages 761-764.* April, 1983.

L. Buera, E. Lleida, A. Miguel, A. Ortega and O. Saz (2007). Cepstral vector normalization based on stereo data for robust speech recognition. *IEEE Transactions on Audio, Speech and Language Processing, volume 15, issue 3, pages 1098-113*. March, 2007.

Y.M. Cheng, D. O'Shaughnessy and P. Mermelstein (1994). Statistical recovery of wideband speech from narrowband speech. *IEEE Transactions on Speech and Audio Processing, volume 2, issue 4, pages 544-548*. October, 1994.

S. Chennoukh, A. Gerrits, G. Miet and R. Sluijter (2001). Speech enhancement via frequency bandwidth extension using line spectral frequencies. *Proceedings ICASSP'01, volume 1, pages 665-668*. May, 2001.

T. Claes and D. van Compernolle (1996). SNR-normalisation for robust speech recognition. *Proceedings ICASSP'96, volume 1, pages 331-334*. May, 1996.

P. Comon (1994). Independent component analysis, a new concept? *Signal Processing, volume 36, issue 3, pages 287-314*. April, 1994.

D. van Compernolle (1987). Increased noise immunity in large vocabulary speech recognition with the aid of spectral subtraction. *Proceedings ICASSP'87, volume 12, pages 1143-1146*. April, 1987.

M. Cooke, P. Green, L. Josifovski and A. Vizinho (2001) Robust automatic speech recognition with missing and unreliable acoustic data. *Speech Communication, volume 34, issue 3, pages 267-285*. June, 2001.

D41JCT-LS datasheet. Available at: http://download.intel.com/design/telecom/prodbref/6925.pdf

S. Davis and P. Mermelstein (1980). Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Transactions on Acoustics, Speech and Signal Processing, volume 28, issue 4, pages 357-366*. August, 1980.

L. Denenberg, H. Gish, M. Meteer, T. Miller, J.R. Rohlicek, W. Sadkin and M. Siu (1993). Gisting conversational speech in real time. *Proceedings ICASSP'93, volume 2, pages 131-134*. April, 1993.

L. Deng, J. Droppo and A. Acero (2004). Dynamic compensation of HMM variances using the feature enhancement uncertainty computed from a parametric model of speech distortion. *IEEE Transactions on Speech and Audio Processing, volume 13, issue 3, pages 412-421*. May, 2005.

V.V. Digalakis, D. Rtischev and L.G. Neumeyer (1995). Speaker adaptation using constrained estimation of Gaussian mixtures. *IEEE Transactions on Speech and Audio Processing, volume 3, issue 5, pages 357-366*. September, 1995.

Dragon. Dragon Naturally Speaking 9 SDK. Available at: http://www.nuance.com/naturallyspeaking/sdk

J. Droppo, A. Acero and L. Deng (2001). Evaluation of the SPLICE Algorithm on the Aurora2 database. *Proceedings Eurospeech'01, pages 217-220*. September, 2001.

J. Droppo, A. Acero and L. Deng (2002). Uncertainty decoding with SPLICE for noise robust speech recognition. *Proceedings ICASSP'02, volume 1, pages 57-60*. May, 2002.

R.O. Duda, P.E. Hart and D.G. Stork (2000). *Pattern Classification*. Wiley-Interscience, 2nd Edition. October, 2000.

S. Euler and J. Zinke (1994). The influence of speech coding algorithms on automatic speech recognition. *Proceedings ICASSP'94, volume 1, pages 621-624*. April, 1994.

J.G. Fiscus (1997). A post-processing system to yield reduced word error rates: recognizer output voting error reduction (ROVER). *Proceedings ASRU'97, pages 347-354.* December, 1997.

W.M. Fisher, R. Doddington and K.M. Goudie-Marshall (1986). The DARPA speech recognition research database: specifications and status. *Proceedings of the DARPA workshop on Speech Recognition, pages 93-99*. February, 1986.

S. Furui (1986). Speaker-independent isolated word recognition using dynamic features of speech spectrum. *IEEE Transactions on Acoustics, Speech and Signal Processing, volume 34, issue 1, pages 52-59.* February, 1986.

M.J.F. Gales (1995). Model-based techniques for noise robust speech recognition. *PhD. Dissertation, Cambridge University*. September, 1995.

M.J.F. Gales, D. Pye and P. Woodland (1996). Variance compensation within the MLLR framework for robust speech recognition and speaker adaptation. *Proceedings ICASSP'96, volume 3, pages 1832-1835.* October, 1996.

M.J.F. Gales (1997). Maximum likelihood linear transformations for HMM-based speech recognition. *Technical Report, TR 291, Cambridge University*. May, 1997.

A. Gallardo-Antolín (2002). Reconocimiento de habla robusto frente a condiciones de ruido aditivo y convolutivo. *PhD. Dissertation, Electrical Engineering, Universidad Politécnica de Madrid, Spain*. 2002.

A. Gallardo-Antolín, C. Peláez-Moreno and F. Díaz-de-María (2005). Recognizing GSM digital speech. *IEEE Transactions on Speech and Audio Processing, volume 13, issue 6, pages 1186-1205*. November, 2005.

J.L. Gauvain and C.H. Lee (1994). Maximum a posteriori estimation for multivariate Gaussian observations of Markov chains. *IEEE Transactions Speech and Audio Processing, volume 2, issue 2, pages 291-298*. April, 1994.

Y. Grenier (1980). Speaker adaptation through canonical correlation analysis. *Proceedings ICASSP'80, volume 5, pages 888-891*. April, 1980.

V. Gupta and P. Mermelstein (1982). Effects of speaker accent on the performance of a speaker-independent, isolated-word recognizer. *Journal of the Acoustical Society of America, volume 71, pages 1581-1587*. June, 1982.

J.H.L. Hansen (1994). Morphological constrained feature enhancement with adaptive cepstral compensation (MCE-ACC) for speech recognition in noise and Lombard effect. *IEEE Transactions on Speech and Audio Processing, volume 2, issue 4, pages 598-614*. October, 1994.

J.H.L. Hansen and D.A. Cairns (1995). ICARUS: Source generator based real-time recognition of speech in noisy stressful and Lombard effect environments. *Speech Communication, volume 16, issue 4, pages 391-422*. June, 1995.

J.H.L. Hansen, B. Zhou, M. Akbacak, R. Sarikaya and B. Pellom (2000). Audio stream phrase recognition for a national gallery of the spoken word: "one small step". *Proceedings ICSLP'00, volume 3, pages 1089-1092*. October, 2000.

J.H.L. Hansen, R. Huang, P. Mangalath, B. Zhou, M. Seadle and J. Deller (2004). SPEECHFIND: spoken document retrieval for a national gallery of the spoken word. *NORSIG'04, pages 1-4*. June, 2004.

M.E. Hennecke, D.G. Stork and K.V. Prasad (1996). Visionary speech: Looking ahead to practical speechreading systems. *Speechreading by Humans and Machines, D. G. Stork and M. E. Hennecke, (eds.), pages 331-349. Springer-Verlag*. 1996.

H. Hermansky (1990). Perceptual linear predictive (PLP) analysis of speech. *Journal of the Acoustical Society of America, volume 87, issue 4, pages 1738-1752*. April, 1990.

H. Hermansky and N. Morgan (1994). RASTA processing of speech. *IEEE Transactions on Speech and Audio Processing, volume 2, issue 4, pages 578-589*. October, 1994.

H. Hermansky, S. Tibrewala and M. Pavel (1996). Towards ASR on partially corrupted speech. *Proceedings ICSLP'96, pages 462-465*. October, 1996.

H. Hermansky and S. Sharma (1999). Temporal patterns (TRAPs) in ASR of noisy speech. *Proceedings ICASSP'99, volume 1, pages 289-292*. March, 1999.

X. Huang, A. Acero and H.W. Hon (2001). *Spoken language processing: a guide to theory, algorithm and system development*. Prentice Hall. April, 2001.

R. Huerta (2000). Speech recognition in mobile environments. *PhD. Dissertation, Electrical and Computer Engineering Department. Carnegie Mellon University, Pittsburgh*. April, 2000.

Intel Dialogic. Intel Dialogic System Release 5.1.1 for Windows. Information available at: http://resource.dialogic.com/telecom/support/releases/winnt/Sr511/index.htm

F. Itakura (1975). Line spectrum representation of linear predictor coefficients of speech signals. *Journal of the Acoustical Society of America, volume 57, supplement S1, page S35*. 1975.

ITU (1993). Paired comparison test of wideband and narrowband telephony. Technical Report COM 12-9-*E, ITU*. March, 1993.

ITU-T (2001). Transmission performance characteristics of pulse code modulation channels. *Recommendation G. 712 (11/01)*.

C. Jankowski, A. Kalyanswamy, S. Basson and J. Spitz (1990). NTIMIT: a phonetically balanced, continuous speech, telephone bandwidth speech database. *Proceedings ICASSP'90, volume 1, pages 109-112*. April, 1990.

C. Jankowski (1991). The NTIMIT speech database. *Printed documentation with NTIMIT CD-ROM*. January, 1991.

P. Jax and P. Vary (2003). On artificial bandwidth extension of telephone speech. *Signal Processing, volume 83, issue 8, pages 1707-1719*. August, 2003.

L. Josifovski (2002). Robust automatic speech recognition with missing and unreliable data. *PhD. Dissertation, Computer Science Department, Sheffield University*. August, 2002.

J.C. Junqua (1993). The Lombard reflex and its role on human listeners and automatic speech recognizers. *Journal of the Acoustical Society of America, volume 93, issue 1, pages 1304-1312*. January, 1993.

J.C. Junqua and J.P. Haton (1996). *Robustness in Automatic Speech Recognition*. Kluwer Academic Publishers. 1996.

C. Jutten and J. Herault (1991). Blind separation of sources, part I: an adaptive algorithm based on neuromimetic architecture. *Signal Processing, volume 24, issue 1, pages 1-10*. July, 1991.

W. Kim and J.H.L. Hansen (2006). Missing-feature reconstruction for band-limited speech recognition. *Proceedings Interspeech'06, pages 2306-2309*. September, 2006.

LDC. Linguistic Data Consortium. Available at: http://www.ldc.upenn.edu/

K.F. Lee and H.W. Hon (1989). Speaker-independent phone recognition using hidden Markov models. *IEEE Transactions on Acoustics, Speech, and Signal Processing, volume 37, issue 11, pages 1641-1648*. November, 1989.

C.J. Leggetter (1995). Improved acoustic modeling for HMMs using linear transformations. *PhD. Dissertation, Cambridge University*. February, 1995.

C.J. Leggetter and P.C. Woodland (1995). Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models. *Computer, Speech and Language, volume 9, issue 2, pages 171-185*. April, 1995.

X. Li and R.M. Stern (2004). Parallel feature generation based on maximizing normalized acoustic likelihood. *Proceedings Interspeech'04, pages 953-956*. October, 2004.

B.T. Lilly and K.K. Paliwal (1996). Effect of speech coders on speech recognition performance. *Proceedings ICSLP'96, volume 4, pages 2344-2347*. October, 1996.

R.P. Lippmann (1997). Speech recognition by machines and humans. *Speech Communication, volume 22, issue 1, pages 1-15*. July, 1997.

L. Liu, J. He and G. Palm (1997). Effects of phase on the perception of intervocalic stop consonants. *Speech Communication, volume 22, issue 4, pages 403-417*. September, 1997.

J. Makhoul (1975). Linear prediction: a tutorial review. *Proceedings of the IEEE, volume 63, issue 4, pages 561-580*. April, 1975.

M. Matassoni, M. Omologo and D. Giuliani (2000). Hands-free speech recognition using a filtered clean corpus and incremental HMM adaptation. *Proceedings ICASSP'00, volume 3, pages 1407-1410*. June, 2000.

MATLAB ANOVAS. Documentation for Matlab's Analysis of Variances. Available at: http://www.mathworks.com/access/helpdesk/help/toolbox/stats/f75080.html#zmw57dd0e6551

MATLAB SPT. Documentation for Matlab's Signal Processing Toolbox. Available at: http://www.mathworks.com/access/helpdesk/help/toolbox/signal/

M.A. Mines, B.F. Hanson and J.E. Shoup (1978). Frequency of occurrence of phonemes in conversational English. *Language and Speech, volume 21, issue 3, pages 221-241*. July-September, 1978.

B.C.J. Moore (2003). *An introduction to the psychology of hearing*. Fifth edition. Academic Press, San Diego. January, 2003.

N. Morales, V. Tomico, E. Campos, J. Tejedor, D. Bolaños, S. Jiménez, J. Garrido and J. Colás (2003). Vocal navigation web page using pattern comparison techniques. *Proceedings m-ICTE 2003, pages 1525-1529*. December, 2003.

N. Morales (2004). Word-isolated long-vocabulary speaker-independent automatic speech recognition systems. *Advanced Studies Dissertation, Universidad Autónoma de Madrid*. June, 2004.

N. Morales, J.H.L. Hansen and D.T. Toledano (2005a). MFCC compensation for improved recognition of filtered and band-limited speech. *Proceedings ICASSP'05, volume 1, pages 521-524*. March, 2005.

N. Morales, D.T. Toledano, J.H.L. Hansen, J. Colás and J. Garrido (2005b). Statistical class-based MFCC enhancement of filtered and band-limited speech for robust ASR. *Proceedings Interspeech'05, pages 2629-2632*. September, 2005.

N. Morales, D.T. Toledano, J.H.L. Hansen, J. Garrido and J. Colás (2006). Unsupervised class-based feature compensation for time-variable bandwidth-limited speech. *Proceedings ICASSP'06, volume 1, pages 533-536*. May, 2006.

N. Morales, D.T. Toledano, J.H.L. Hansen and J. Colás (2007a). Blind feature compensation for time-variant band-limited speech recognition. *IEEE Signal Processing Letters, volume 14, issue 1, pages 70-73*. January, 2007.

N. Morales, D.T. Toledano, J.H.L. Hansen and J. Garrido (2007b). Multivariate cepstral feature compensation on band limited data for robust speech recognition. *Proceedings NODALIDA'07, pages 144-151.* May, 2007.

N. Morales, L. Gu and Y. Gao (2007c). Adding noise to improve noise robustness in speech recognition. *Proceedings Interspeech'07, in print*. August, 2007.

A. Moreno, D. Poch, A. Bonafonte, E. Lleida, J. Llisterri, J.B. Mariño and C. Nadeu (1993). Albayzin speech database: design of the phonetic corpus. *Proceedings Eurospeech'93, pages 175-178*. September, 1993.

A. Moreno, D.T. Toledano, N. Curto and R. Torre (2006). Inventario de frecuencias fonémicas y silábicas del castellano espontáneo y escrito. *Proceedings IV Jornadas de Tecnología del Habla, pages 77-80*. November, 2006.

P.J Moreno and R.M. Stern (1994). Sources of degradation of speech recognition in the telephone network. *Proceedings ICASSP'94, volume 1, pages 109-112*. April, 1994.

P.J. Moreno, B. Raj and R.M Stern (1996). A vector Taylor series approach for environment independent speech recognition. *Proceedings ICASSP'96, volume 2, pages 733-736*. May, 1996.

P.J. Moreno (1996). Speech recognition in noisy environments. *PhD. Dissertation, Electrical and Computer Engineering Department. Carnegie Mellon University, Pittsburgh*. April, 1996.

R. de Mori (1998). *Spoken dialogues with computers*. Academic Press, 1998.

A. Moyal (2005). Using a DSP-based speech recognition engine in telephony applications. *Proceedings AVIOS'00*. February, 2005.

NGSW. The National Gallery of the Spoken Word. Available at: http://www.ngsw.org/.

L. Nguyen, R. Schwartz, Y. Zhao and G. Zavaliagkos (1994). Is N-best dead? *Proceedings workshop on Human Language Technology, pages 411-414*. March, 1994.

H. Nyquist (1928). Certain topics in telegraph transmission theory. *Reprinted in Proceedings of the IEEE, volume 90, issue 2, pages 280-305*. February, 2002.

R.H. Ott (1977). Temporal radio frequency spectra of multifrequency waves in a turbulent atmosphere characterized by a complex refractive index. *IEEE Transactions on Antennas and Propagation, volume SP-25, issue 2, pages 254-260*. March, 1977.

D. Pallett (2003). A look at NIST's benchmark ASR tests: past, present, and future. *Proceedings ASRU'03, pages 483-488*. December, 2003.

D. Pearce (2000). Enabling new speech driven services for mobile devices: an overview of the ETSI standards activities for distributed speech recognition front-ends. *Proceedings AVIOS'00*. May, 2000.

L.C.W. Pols (1977). Spectral analysis and identification of Dutch vowels in monosyllabic words. *Ph.D. Dissertation, Free University of Amsterdam*. 1977.

G. Potamianos, C. Neti, G. Gravier, A. Garg and A.W. Senior (2003). Recent advances in the automatic recognition of audiovisual speech. *Proceedings of the IEEE, volume 91, issue 9, pages 1306-1326*. September, 2003.

W.H. Press, B.P. Flannery, S.A Teukolsky and W.T. Vetterling (1992). *Numerical Recipes in C: the art of scientific computing*. Cambridge University Press, second edition. October, 1992. Available at: http://www.nrbook.com/a/bookcpdf.php.

Y. Qian and P. Kabal (2004). Combining equalization and estimation for bandwidth extension of narrowband speech. *Proceedings ICASSP'04, volume 1, pages 713-716*. May, 2004.

T.F. Quatieri (2002). *Discrete-time speech signal processing: principles and practice*. Prentice Hall, Upper Saddle River, NJ, USA. 2002.

L. Rabiner (1989). A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE, volume 77, issue 2, pages 257-286*. February, 1989.

B. Raj, M.L. Seltzer and R.M. Stern (2004). Reconstruction of missing features for robust speech recognition. *Speech Communication, volume 43, issue 4, pages 275-296*. September, 2004.

D.R. Reddy (1976). Speech recognition by machine: a review. *Proceedings of the IEEE, volume 64, issue 4, pages 501-531*. April, 1976.

F. Richardson, M. Ostendorf and J.R. Rohlicek (1995). Lattice-based search strategies for large vocabulary speech recognition. *Proceedings ICASSP'95, volume 1, pages 576-579*. May, 1995.

B. Rivet, L. Girin and C. Jutten (2007). Log-Rayleigh distribution: a simple and efficient statistical representation of log-spectral coefficients. *IEEE Transactions on Audio, Speech and Language Processing, volume 15, issue 3, pages 796-802*. March, 2007.

H. Rogers (2000). *The sounds of language: an introduction to phonetics*. Pearson ESL, first edition. September, 2000.

H. Sakoe and S. Chiba (1978). Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustics, Speech and Signal Processing, volume 26, issue 1, pages 43-49*. February, 1978.

R. Schlüter and H. Ney (2001). Using phase spectrum information for improved speech recognition performance. *Proceedings ICASSP'01, volume 1, pages 133-136*. May, 2001.

P. Schwarz, P. Matějka and J. Černocký (2004). Towards lower error rates in phoneme recognition. *Proceedings International Conference on Text, Speech and Dialogue, pages 465-472*. September, 2004.

M. Seltzer and A. Acero (2005). Training wideband acoustic models using mixed-bandwidth training data via feature bandwidth extension. *Proceedings ICASSP'05, volume 1, pages 921-924*. March, 2005.

M. Seltzer, A. Acero and J. Droppo (2005). Robust bandwidth extension of noise-corrupted narrowband speech. *Proceedings Interspeech'05, pages 1509-1512*. September, 2005.

M. Seltzer and A. Acero (2007). Training wideband acoustic models using mixed-bandwidth training data for speech recognition. *IEEE Transactions on Audio, Speech and Language Processing, volume, 15, issue, 1, pages 235-245*. January, 2007.

C. Shanon (1949). Communication in the presence of noise. *Proceedings of the IRE volume 37, issue 1, pages 10-21. Reprinted in Proceedings of IEEE, volume 86, issue 2, pages 447-457*. January, 1949.

G. Shi, M.M. Shanechi and P. Aarabi (2006). On the importance of phase in human speech recognition. *IEEE Transactions on Audio, Speech and Language Processing, volume 14, issue 5, pages 1867-2006*. September, 2006.

K. Shikano, K.-F. Lee and D. Reddy (1986). Speaker adaptation through vector quantization. *Proceedings ICASSP'86, volume 11, pages 2643-2646*. April, 1986.

SONIC. SONIC: large vocabulary continuous speech recognition system. Available at: http://cslr.colorado.edu/beginweb/speech_recognition/sonic_main.html

Sphinx. CMU Sphinx project page. Available at: http://cmusphinx.sourceforge.net/html/cmusphinx.php

M.R. Spiegel and L. Abellanas (1998). Fórmulas y tablas de matemática aplicada. Translation of: *Mathematical Handbook of Formulas and Tables*. First edition. McGraw-Hill, Madrid. 1988.

S. Stevens and J. Volkmann (1940). The relation of pitch to frequency: a revised scale. *The American Journal of Psychology, volume 53, issue 3, pages 329-353*. July, 1940.

D.T. Toledano, A. Moreno, J. Colás and J. Garrido (2005). Acoustic-phonetic decoding of different types of spontaneous speech in Spanish. *Proceedings DiSS'05, pages 165-168*. September, 2005.

V. Tomico, N. Morales, E. Campos, J. Tejedor, D. Bolaños, S. Jiménez, J. Garrido and J. Colás (2003). Vocal platform for telephone voice portals and internet based interfaces. *Proceedings m-ICTE 2003, pages 1889-1893*. December, 2003.

R. Vergin, D. O'Shaughnessy and V. Gupta (1996). Compensated Mel frequency cepstrum coefficients. *Proceedings ICASSP'96, volume 1, pages 323-326*. May, 1996.

ViaVoice. IBM Pro USB Edition. Available at: http://www.nuance.com/viavoice/pro.

A. Viterbi (1967). Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory, volume 13, issue 2, pages 260-269*. April, 1967.

R.M. Warren, K.R. Riener, J.A. Bashford and B.S. Brubaker (1995). Spectral redundancy: intelligibility of sentences heard through narrow spectral slits. *Perception & Psychophysics, volume 57, issue 2, pages 175-182*. 1995.

F. Wessel, R. Schlüter, K. Macherey and H. Ney (2001). Confidence measures for large vocabulary continuous speech recognition. *IEEE Transactions on Speech and Audio Processing, volume 9, pages 288-298*. March, 2001.

H. Yasukawa (1996). Restoration of wide band signal from telephone speech using linear prediction error processing. *Proceedings ICSLP'96, volume 2, pages 901-904*. October, 1996.

H. Yilmaz (1967). A theory of speech perception. *Bulletin of Mathematical Biophysics, volume 29, issue 4, pages 793-825*. 1967

H. Yilmaz (1972). Statistical theory of speech perception. *Proceedings Conference on Speech Communication and Processing, pages 226-229*. 1972.

S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev and P. Woodland (2005). *The HTK Book (for HTK version 3.3)*. Cambridge University Engineering Department. April, 2005.