ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
UNIVERSITÉ DU QUÉBEC

THESIS PRESENTED TO
ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR
THE DEGREE OF DOCTOR OF PHILOSOPHY
Ph.D.

BY
NAJIM DEHAK

DISCRIMINATIVE AND GENERATIVE APPROACHES FOR LONG- AND SHORT-TERM SPEAKER CHARACTERISTICS MODELING: APPLICATION TO SPEAKER VERIFICATION

MONTREAL, JUNE 23, 2009

BOARD OF EXAMINERS

THIS THESIS HAS BEEN EVALUATED

BY THE FOLLOWING BOARD OF EXAMINERS:

Mr. Pierre Dumouchel, Thesis Supervisor
Département de génie logiciel et des technologies de l'information

Mr. Patrick Kenny, Thesis Co-supervisor
Centre de recherche en informatique de Montreal

Mr. Richard Lepage, President of the Board of Examiners
Département de génie de la production automatisée à l'École de technologie supérieure

Mr. Douglas Reynolds, External Examiner
Massachusetts Institute of Technology Lincoln Laboratory

Mr. Eric Granger, Examiner
Département de génie de la production automatisée à l'École de technologie supérieure

THIS THESIS WAS PRESENTED AND DEFENDED

BEFORE A BOARD OF EXAMINERS AND PUBLIC

ON 21 MAY 2009

AT ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

ACKNOWLEDGMENTS

Je voudrais remercier infiniment mon directeur de thèse M. Pierre Dumouchel pour m'avoir fait confiance durant les travaux de cette thèse et pour son soutien moral et financier durant les trois dernières années.

Je tiens à exprimer toute ma gratitude à mon codirecteur de thèse M. Patrick Kenny qui m'a donné la possibilité de travailler avec lui et m'a permis d'approfondir mes connaissances dans ce domaine de recherche.

Je suis très reconnaissant à M. Richard Lepage d'avoir accepté de juger ce travail et de présider le comité de ma thèse.

Mes remerciements les plus sincères vont à M. Eric Granger et M. Douglas Reynolds pour m'avoir fait l'honneur d'accepter d'évaluer les travaux de cette thèse de doctorat.

Un grand merci va à Michel Comeau et Pierre Ouellet pour leur aide à corriger cette thèse.

Un remerciement particulier va à tous mes amis : Patrick, Sylvain, Arnaud, Christelle, Marie-France, Lucie, Nori, Khalifa, Rafik, Said, Mohammed, Yazid, El Hachemi, Maryse, Narjes, Pierre, Fred, Gilles et Julie.

Je tiens à remercier tout les membres du Centre de Recherche d'Informatique de Montréal (CRIM) et spécialement le personnel de support informatique : Ivan, Denis, Yves et Normand pour l'agréable gentillesse et l'aide fournie durant mon séjour au CRIM.

Je dédie cette thèse à mes parents et ma grand-mère qui m'ont beaucoup soutenu durant toute cette période, mon frère Réda à qui je dois énormément et avec qui j'ai beaucoup collaboré pendant ces quatre années, mon frère Mounir et ma petite soeur Ghizlene, mes deux belles-soeurs Amina et Soumia, et sans oublier le bout de chou de la famille Yacine.

Je voudrais dédier sincèrement ce manuscrit à mes anciens professeurs Benyettou Abdelkader et Benyettou Mohamed pour m'avoir montré le chemin de la recherche et pour leur amitié.

DISCRIMINATIVE AND GENERATIVE APPROACHES FOR LONG- AND SHORT-TERM SPEAKER CHARACTERISTICS MODELING: APPLICATION TO SPEAKER VERIFICATION

Najim DEHAK

RÉSUMÉ

Le problème de la vérification du locuteur consiste à vérifier si deux enregistrements de parole ont été produits par le même locuteur ou deux locuteurs différents. La majorité des systèmes de vérification du locuteur actuels sont basés sur le modèle de mélange de Gaussiennes. Ce modèle probabiliste permet de modéliser finement la distribution complexe des paramètres de la parole mais offre un niveau limité de discrimination, qui est pourtant un point majeur dans ce domaine. Dans le premier point de cette thèse, nous proposons de combiner un modèle discriminant qui est le séparateur à vaste marge avec deux approches génératives basées sur les modèles de mélange de Gaussiennes pour la vérification du locuteur. Dans la première approche générative, un locuteur est caractérisé à l'aide d'un modèle de mélange de Gaussiennes obtenu à partir d'une adaptation maximum *A Posteriori* d'un autre modèle de mélange de Gaussiennes nommé modèle du monde qui caractérise l'univers des locuteurs aux données du client. La deuxième approche générative est l'analyse jointe de facteur. Cette technique est devenue l'état de l'art dans le domaine de la vérification du locuteur durant ces trois dernières années. L'avantage de cette technique est de proposer des outils puissants pour modéliser la variabilité due au locuteur et au canal. Nous avons proposé et testé plusieurs fonctions noyaux pour chacun de ces deux combinions précédentes. Les meilleurs résultats sont obtenus lorsque les séparateurs à vaste marge ont été appliqués dans un nouvel espace appelé espace de la "variabilité totale" défini à l'aide de l'analyse de facteur. L'effet du canal dans cette modélisation a été traité par la combinaison d'une analyse discriminante linéaire et d'une technique de normalisation de la fonction noyau basée sur l'inverse de la matrice de covariance intra-classe du locuteur.

Le deuxième point traité dans cette thèse consiste à utiliser les caractéristiques prosodiques et spectrales à long terme du locuteur pour l'élaboration d'un nouveau système de vérification du locuteur. L'approche que nous proposons est basée sur l'approximation continue des contours prosodiques et cepstraux à l'aide d'un polynôme de Legendre utilisant les pseudo-syllabes comme unités de base. Les coefficients de ce polynôme sont représentés par un modèle de mélange de Gaussiennes. L'analyse jointe de facteur est utilisée pour traiter l'effet de la variabilité du canal et modéliser la variabilité entre les locuteurs. Finalement nous réalisons une fusion des scores entre les systèmes opérant dans les caractéristiques à long terme du locuteur avec ceux décrits plus haut utilisant les paramètres à court terme du locuteur.

Mot-clés : Vérification du locuteur, modèles de mélange de Gaussiennes, analyse de facteur jointe, séparateur à vaste marge, espace de la variabilité totale, polynôme de Legendre, pseudo-syllabes.

DISCRIMINATIVE AND GENERATIVE APPROACHES FOR LONG- AND SHORT-TERM SPEAKER CHARACTERISTICS MODELING: APPLICATION TO SPEAKER VERIFICATION

Najim DEHAK

ABSTRACT

The speaker verification problem can be stated as follows: given two speech recordings, determine whether or not they have been uttered by the same speaker. Most current speaker verification systems are based on Gaussian mixture models. This probabilistic representation allows to adequately model the complex distribution of the underlying speech feature parameters. It however represents an inadequate basis for discriminating between speakers, which is the key issue in the area of speaker verification. In the first part of this thesis, we attempt to overcome these difficulties by proposing to combine support vector machines, a well established discriminative modeling, with two generative approaches based on Gaussian mixture models. In the first generative approach, a target speaker is represented by a Gaussian mixture model corresponding to a Maximum *A Posteriori* adaptation of a large Gaussian mixture model, coined universal background model, to the target speaker data. The second generative approach is the Joint Factor Analysis that has become the state-of-the-art in the field of speaker verification during the last three years. The advantage of this technique is that it provides a framework of powerful tools for modeling the inter-speaker and channel variabilities. We propose and test several kernel functions that are integrated in the design of both previous combinations. The best results are obtained when the support vector machines are applied within a new space called the "total variability space", defined using the factor analysis. In this novel modeling approach, the channel effect is treated through a combination of linear discriminant analysis and kernel normalization based on the inverse of the within covariance matrix of the speaker.

In the second part of this thesis, we present a new approach to modeling the speaker's long-term prosodic and spectral characteristics. This novel approach is based on continuous approximations of the prosodic and cepstral contours contained in a pseudo-syllabic segment of speech. Each of these contours is fitted to a Legendre polynomial, whose coefficients are modeled by a Gaussian mixture model. The joint factor analysis is used to treat the speaker and channel variabilities. Finally, we perform a scores fusion between systems based on long-term speaker characteristics with those described above that use short-term speaker features.

Keywords: Speaker verification, Gaussian mixture model, joint factor analysis, support vector machines, total variability space, Legendre polynomial, pseudo-syllables.

CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

CMS	**C**epstral **M**ean **S**ubstraction
DCF	**D**etection **C**ost **F**unction
DET	**D**etection **E**rror **T**rade-off
EER	**E**qual **E**rror **R**ate
FA	**F**alse **A**larm
FR	**F**alse **R**ejection
GLDS	**G**eneralized **L**inear **D**iscriminant **S**equence
GMM	**G**aussian **M**ixture **M**odels
JFA	**J**oint **F**actor **A**nalysis
LDA	**L**inear **D**iscriminant **A**nalysis
MAP	**M**aximum **A** **P***osteriori*
MFCC	**M**el-**F**requency **C**epstral **C**oefficients
MinDCF	**M**inimum **D**etection **C**ost **F**unction
ML	**M**aximum **L**ikelihood
NAP	**N**uisance **A**ttribute **P**rojection
NERF	**N**onuniform **E**xtraction **R**egion **F**eatures
NIST	**N**ational **I**nstitute of **S**tandards and **T**echnology
PPRLM	**P**arallel **P**hone **R**ecognition **L**anguage **M**odeling
SNERF	**S**yllable-based **N**onuniform **E**xtraction **R**egion **F**eatures

SVM	**S**upport **V**ector **M**achine
SRE	**S**peaker **R**ecognition **E**valuation
UBM	**U**niversal **B**ackground **M**odel
WCCN	**W**ithin **C**lass **C**ovariance **N**ormalization

INTRODUCTION

Nowadays, applications such as money withdrawal in Automatic Teller Machines (ATM), require user identification. In the general case, this identification is performed using a smart card and a personal identification number, but it can also be accomplished using biometric characteristics. These can be the individual's fingerprint, hand shape, face or voice. However, when it comes to identifying the user at a distance through the use of a phone, for example, the possible way to carry out this identification is by the user's voice. The speech signal, in addition to containing language information used for human communication, also provides information about the speaker's identity. Automatic speaker recognition is divided into two main application functions. The first one is speaker identification, which consists in identifying, among several possible identities, the speaker who produced the speech test segment. The second one is speaker verification that ascertains whether the claimed identity of the target speaker is the same as that who produced the speech test segment. Speaker verification is the problem studied in this Thesis.

Voice servers integrating reliable speaker verification systems are destined to become mainstream in several areas for the purpose of protecting customer information by securing remote access through speaker verification. In addition to being used for security purposes, speaker verification systems can also be used for indexing multimedia content. Recent work in the field of speaker verification is focused on the problem of condition variability between training and test speech segments. These variations are in most cases caused by channel transmission effects. Another line of research involves the extraction of other features and information to better model the speaker characteristics in order to better recognize the individual over time. The most widely used features are based on information retrieved at the spectral level. Prosodic features are also exploited by speaker verification systems to model the intonation and speaking style of the speaker. It is well established that the fusion of several systems that operate on different sources of information will improve the performance of the speaker verification system (Reynolds et al., 2003) (Brummer et al., 2007).

Problem Statement

The most widely used model in speaker verification is the Gaussian Mixture Model (GMM). It is probabilistic in nature and has the advantage of adequately modeling the complex distribution of speech frames. It however represents an inadequate basis for discriminating between speakers, which is the main point of speaker verification. In the pattern recognition field and other areas as well, Support Vector Machines (SVM) have been shown to be exceptional discriminative models. This characteristic has been exploited in the field of speaker verification, where several SVM-based systems have been introduced. In this thesis we propose a new approach to combine speaker verification systems based on GMMs and SVMs. The second problem addressed in this thesis is the use of long-term prosodic and cepstral speaker characteristics. These characteristics model the speaker speaking style. The majority speaker verification systems are based on features that characterize the spectral envelope of the speech signal. This information is obtained using a short-term sliding windows (typically 25ms). The last ten years have seen the use of other information sources to model long-term speaker characteristics. This information is modeled in the spectral or prosodic domain. Most parameters are extracted through a discretization process. The fusion of different systems operating on different sources is commonly used in order to improve the speaker verification system's performance. We propose to model the speaker's long-term information using continuous approaches and fusing it with the short-term information.

For both of the issues addressed in this thesis, the goal is to improve the speaker verification system's performance in the context of the National Institute of Standards and Technology (NIST) Speaker Recognition Evaluation (SRE) campaign. Frequently, NIST organizes an evaluation campaign in order to compare the latest advances in the field. The majority of data used for this assessment are from the context of telephone conversations. In the 2008 competition however, data from several microphones were also used. At the end of each evaluation, a workshop is conducted in order to present the results and analyze the differences between systems based on their performances. There are several tasks each year; these are characterized by the difference between the train and test conditions. In each campaign, the participants receive the same labeled data to enroll the target speaker models and a series of blind test data (for which the identities of speakers that produce the test files are unknown). Participants test their systems on these test files and send their results to the NIST organization. The results are analyzed and classified according to the performance of the systems.

Objectives of the Research and Contributions

The objective of this thesis is twofold. The first point concerns the combination of generative and discriminative models in order to improve the performance of current speaker verification systems. The majority of state-of-the-art speaker verification systems are based on Gaussian mixture models. A speaker GMM is produced via adaptation of a large GMM, called universal background model (UBM), against the target speaker data. To this end, Maximum A Posteriori (MAP) and eigenvoice adaptation are the most widely used techniques. In our work we propose two systems that combine generative and discriminative approaches. The first system is a combination of SVM and the classical GMM-UBM based on MAP adaptation. The kernel used in our approach is nonlinear, derived from an approximated Kullback-Leibler distance between two GMMs. We have shown the importance of applying GMM normalization in the case of this kernel and have also tested a technique to deal with the intersession variability problem. This model can be applied to any other problem where MAP adaptation is appropriate. During the past three years, the Joint Factor Analysis (JFA) model has become the state-of-the-art in the speaker verification field. The novelty of this approach is that it allows modeling the inter-speaker and channel variabilities through two distinct space definitions. The first space depends on the speaker while the other depends on the channel. In order to develop the second system, we tested several combinations of SVM and JFA. The best results are obtained when simple factor analysis is used to define a single space called "total variability space" that contains both previous variabilities simultaneously. The channel variability is then addressed at the SVM level using linear discriminant analysis and within class covariance normalization techniques. The results obtained by this combination exceed those produced by current state-of-the-art systems.

The state-of-the-art in speaker verification systems is based on features which model the spectral envelope of the speech signal. These features model the short-term speaker characteristics because they are extracted through a short sliding window (25ms). For the second objective of this thesis, we propose to extract and model long-term prosodic and spectral speaker characteristics. These features depend on MFCCs, pitch and energy contours. The approach that we propose is based on the extraction of spectral and prosodic features at the pseudo-syllable level. The segmentation into pseudo-syllables is performed in an unsupervised manner, which represents an advantage compared to other systems that require a speech recognizer. The cepstral and prosodic contours in each segment are approximated by Legendre polynomials. The coefficients of the Legendre polynomials are then modeled by a Gaussian mixture model. The joint factor analysis model is applied to address the problem of variability between the speakers and also the variability between recording sessions. The use of JFA is important because it is the preferred method when few feature vectors are available to enroll the target speaker, which is the case in our new modeling based on the pseudo-syllables as unit which offers an average of 400 vectors per recording. We explored score fusion between systems based on long- and short-term speaker features. This fusion reveals that the long-term speaker feature systems provide complementary information to the short-term spectral speaker characteristics.

Organization of the Thesis

This thesis is organized as follows. In Chapter 1, we introduce the basic concepts of speaker verification, and introduce a state-of-the-art speaker verification system based on the GMM-UBM approach. This chapter also defines all the metrics used to evaluate the performance of these systems. Chapters 2 and 3 present, respectively, the core background behind joint factor analysis and support vector machines. In Chapter 4, we explain the combination between the SVM and GMM-UBM system based on Maximum *A Posteriori* adaptation. We define the kernel put to use and present a way to improve results by applying model normalization and nuisance attribute projection techniques. The combination between joint factor analysis and the SVM is the subject of Chapter 5. The results from several combinations are discussed. Chapter 6 introduces the long-term prosodic and cepstral speaker characteristics based on Legendre polynomial coefficients and joint factor analysis modeling. Finally, Chapter 7 discusses fusion between the long-term feature systems obtained in the previous chapter with the three other systems developed in Chapters 4 and 5.

CHAPTER 1

SPEAKER VERIFICATION PROBLEM

The present chapter gives a brief introduction to the field of speaker verification. This includes a description of the main components that make up a speaker verification system. We also present a general overview of state-of-the-art methods designed for robust speaker recognition. Particular attention is given to the channel variability problem, which is the main source of errors in speaker recognition systems. We conclude by describing the principal metrics used in assessing a speaker verification system's performance.

1.1 Biometrics Access Control Systems

Biometrics consists of science and technologies for recognizing humans uniquely based upon one or more intrinsic physiological or behavioral traits. A biometric system is a form of identity access management and access control based on human body characteristics such as fingerprints, retina and iris, voice patterns, signature, facial patterns, hand measurements, etc. Fingerprints are the most common physiological trait used in human identification. For other applications such as bank transactions, signature is the most widely used modality. With the development of cellular phone applications, voice-based biometric systems may prove to be the only feasible approach for remote access control.

1.2 Biometric System Based on Speech

A biometric system based on speech information can be split into two types: speaker identification and speaker verification. Unlike speaker identification, where the goal is to associate a given speech segment with a specific speaker chosen from a set of speakers, the goal in the speaker verification task is to determine whether or not a segment of speech belongs to the claimed speaker. Speaker Verification (SV) can be text-dependent or text-independent. In a text dependent SV, the speaker enunciates the same word, sentence or paragraph in the training and test steps. In a text independent SV, on the other hand, the content of produced speech is subject to no restriction whatsoever (free and spontaneous speech).

Speaker verification systems use different levels of speaker information. The first level of information incorporates parameters that model the acoustic characteristics of the human speech production system. These parameters are the most widely used in speaker recognition technology. Vocal features represent intrinsic physical traits that characterize the speaker's identity. Another source of information models phonetic characteristics. Using phonetic units have the advantage of modeling the behavior of the speaker traits during phonemes pronunciation. SV systems based on these features have the inconvenient to be language dependent. In designing a language independent SV system that analyzes phonetic information, we must train and adapt many such systems. A higher level of speech information is related to speech characteristics that are behavioral in nature. These characteristics are collectively referred to as prosody; they include speech intonation, melody and segment duration. Broadly speaking, prosodic information models the speaker's speaking style. It is related to the pitch (vibration of the vocal cords), sound duration and the energy used to produce speech sounds. A still higher level of information incorporates lexical, syntactic, semantic and pragmatic speaker characteristics. Unfortunately, in contemporary speaker verification applications, insufficient training data is available to model all these levels of information.

1.3 Speaker Verification System

Speaker verification systems are composed of three distinct parts. The first one is dedicated to feature extraction. A number of feature representations are possible, but the most widely used are cepstral parameters. The second part is the training module. The principal goal of this component is to build a speaker model. Several methods serve to model the speaker; they can be separated into the following two groups: generative and discriminative. The final part that makes up a speaker verification system includes the scoring and decision process. The scoring vary according to the method used to model the speaker. The figure 1.1 gives an example of speaker verification system. All these facets will be described in greater detail in what follows.

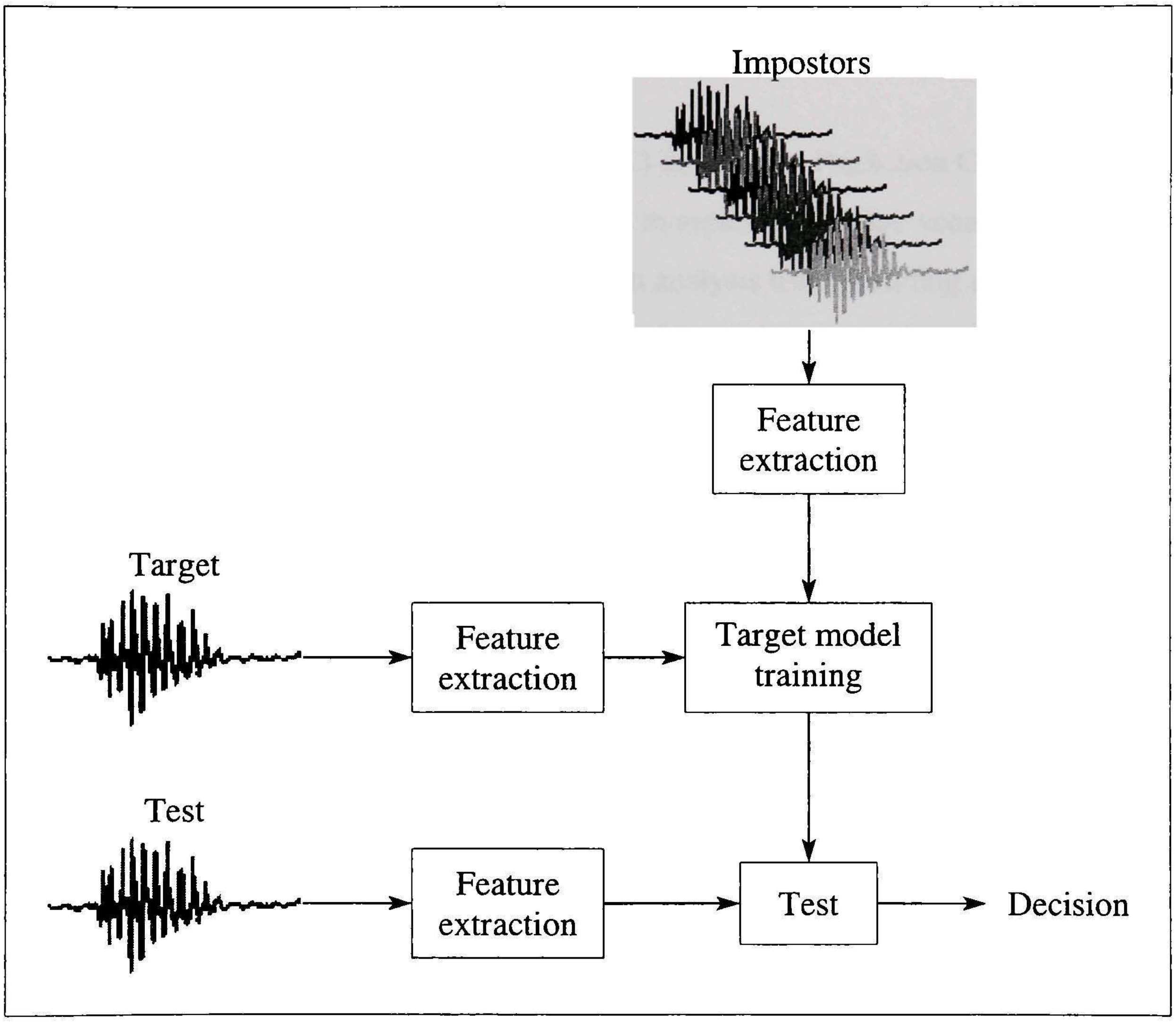

Figure 1.1 Speaker verification system.

1.4 Feature Extraction

The feature extraction module depends on the source-level information used by the system. In this part we will present parameters that model short-term vocal tract characteristics of the speaker. These features are the most widely used in the speaker verification field. In the second part of the thesis, we will describe prosodic features that model the speaker's long-term vocal tract characteristics, and their use in speaker verification.

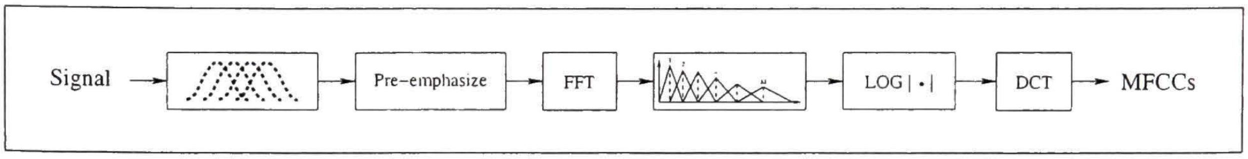

Figure 1.2 MFCC feature analysis.

The Mel Frequency Cepstral Coefficients (MFCC) and Linear Prediction Coding (LPC) parameters are the most important parameters used to represent speaker vocal tract characteristics. These parameters are based on a short-term analysis using a sliding analysis window. A feature vector is extracted for each placement of that window. Unlike the LPC analysis, which uses a linear process to predict the speech signal within each analysis window, MFCC analysis is based on filter banks applied to the spectrum of each window. The MFCCs are heuristic representations of acoustic properties, that simulate the human ear. More precisely, they mimic the perceptual representations of acoustic information conveyed by the human auditory system.

MFCCs are a short-term representation of the sound spectrum, defined as a real cepstrum of a windowed, short-time signal, derived from the FFT of that signal. The difference with the real cepstrum is that a non linear frequency scale is applied. It assumes that the sampled speech waveform is approximately stationary over short intervals of approximately 10 to 30 msec in duration. The feature analysis (Figure 1.2) procedure involves a sliding analysis window along the speech signal. For each window placement, the speech is pre-emphasized and the discrete spectrum is computed using the FFT algorithm. A filter bank with M triangular weighting filters is then used. Each filter computes the energy average around the center frequency of each triangle. The center frequencies are linearly spaced on a mel-frequency scale, which approximates the behavior of the human auditory system. Thereafter logarithmic compression of the filter bank outputs is performed. Finally, the Mel frequency cepstrum is then the discrete cosine transform of the logarithms of the M filter outputs. This transformation is used to reduce the correlation between pairs of features.

In speaker verification systems, an approximation to the first- and second-order time derivative (deltas and delta-deltas) can be appended to the MFCCs in order to capture the dynamic temporal information of speech.

The implementation of MFCC feature analysis used in this thesis uses a sliding window of 25 ms of duration. Window positions are updated by 10 ms increments. The discrete spectrum is computed using an FFT over a 4 kHz telephone bandwidth. A set of 24 filter bank energies ($M = 24$) are computed over the entire windowed spectrum. Each feature vector is the concatenation of 19 Mel Frequency Cepstral Coefficients extracted from the discrete cosine transform and an energy coefficient. We apply a feature normalization based on feature warping to the obtained feature vector components. This procedure involves mapping the feature vector components so that they follow a normal distribution over a sliding window that is 3 seconds in duration. More details can be found about this transformation in section 1.7.1. At the end, the first and second derivatives of the normalized vectors are computed. The final vectors are of 60-dimensional.

1.5 Generative Models

A Gaussian Mixture Model (GMM) is a generative model widely used in speaker verification. It represents the state-of-the-art in this field. This model was introduced and applied for the first time in speaker verification in (Reynolds et Rose, 1995)(Reynolds et al., 2000). It is a semi-parametric probabilistic method that offers the advantage of adequately representing speech signal variability. Frequently, speaker verification systems based on GMMs are combined with other systems based on other types of models to improve their performance.

Given a GMM λ modeling F-dimensional vectors, the likelihood of observing a feature vector x given this model λ is computed according to the following equation:

$$\boxed{\mathrm{P}\left(\mathbf{x}|\lambda\right) = \sum_{i=1}^{C} w_i \mathcal{N}\left(\mathbf{x}; \mu_i, \Sigma_i\right)} \tag{1.1}$$

Practical speaker verification systems use diagonal covariance matrices instead of full covariance matrices Σ_i to define GMM models. Full covariance matrices are not really necessary even if the features are not statistically independent, which is the case for MFCC parameters.

For a sequence of acoustic feature vectors $X = \{x_1, x_2, ..., x_T\}$ such as MFCC acoustic feature vectors representing the test utterance, we make the assumption that each observation (vector) is independent of other observations. As a result, the log-likelihood of the sequence X, given a GMM model λ, is the sum of the log-likelihoods of each feature vector x_i given that model. The corresponding likelihood is thus:

$$\boxed{\log \mathrm{P}\left(X|\lambda\right) = \sum_{t=1}^{T} \log \mathrm{P}\left(\mathrm{x_t}|\lambda\right)} \tag{1.2}$$

where $\mathrm{P}\left(x_t|\lambda\right)$ is the likelihood of feature vector x_t given GMM model λ ($cf.$ Eq. 1.1).

The Expectation Maximization (EM) algorithm (Dempster et al., 1997) is used to learn the GMM parameters $\lambda = (w_i, \mu_i, \Sigma_i)$ based on maximizing of the expected log-likelihood of the training data. In most speaker verification systems, we do not have enough data to train the speaker GMM using the EM algorithm. To overcome these difficulties, a speaker verification system uses a GMM Universal Background Model (UBM), under the assumption that this model will adequately describe the underlying characteristics of a large speaker population. Generally, the UBM is trained on a large set of speakers, the identities of whom are different from the target speaker. The speaker GMM model is then derived from the UBM by Maximum *A Posteriori* (MAP) adaptation using the target speaker data.

1.5.1 Training GMM-UBM : Maximum Likelihood GMM Parameter Estimation

The UBM is a large GMM trained to represent a speaker-independent distribution of features. The corresponding training utterances are selected according to the types and quality of speech, as well as the composition of speech expected to be encountered during recognition. For example, in NIST-SRE single speaker recognition tests, the gender of both the test and target speakers is known in advance; there are no cross-gender tests. So in this case, male speech utterances only are used to construct the male-dependent UBM; likewise, only female speech is used to build the female-dependent UBM.

The aim of the training step in GMM-UBM modeling is to estimate the parameters of the GMM λ, which in some sense best match the distribution of the training feature vectors. Several criteria are available for estimating the GMM parameters (McLachlan et al., 2000), the most popular approach being maximum likelihood (ML) estimation.

The goal of ML estimation is to find the model parameters that maximize the likelihood of the GMM, given the training data (Eq. 1.2). Unfortunately, there is no closed form expression for the ML estimation of GMM parameters. However, ML parameter estimates can be obtained iteratively using the EM algorithm. The basic idea is this, Given an initial model λ, estimate a new model $\overline{\lambda}$ such that the likelihood increases: $\mathrm{P}\left(X|\overline{\lambda}\right) \geq \mathrm{P}\left(X|\lambda\right)$. The updated model is then used as the initial model for the next iteration. The process is repeated until some convergence threshold is reached. For each iteration of the EM algorithm, the expressions of the ML estimates of the GMM parameters which guarantee a monotonic increase of the model's likelihood are as follows:

For each Gaussian i:

$$\boxed{\overline{w_i} = \frac{1}{T}\sum_{t=1}^{T} \mathrm{P}\left(i|x_t,\lambda\right)} \tag{1.3}$$

$$\boxed{\overline{\mu_i} = \frac{\sum_{t=1}^{T} \mathrm{P}\left(i|x_t,\lambda\right)\, x_t}{\sum_{t=1}^{T} \mathrm{P}\left(i|x_t,\lambda\right)}} \tag{1.4}$$

$$\boxed{\overline{\Sigma_i} = \frac{\sum_{t=1}^{T} \mathrm{P}\left(i|x_t,\lambda\right)\, x_t^2}{\sum_{t=1}^{T} \mathrm{P}\left(i|x_t,\lambda\right)} - \overline{\mu_i}^2} \tag{1.5}$$

where the mixture index i varies from 1 to C. The terms w_i, μ_i and Σ_i refer to the weight, mean vector and diagonal covariance matrix of the i^{th} Gaussian component of the initial

GMM λ respectively. Similarly, $\overline{w_i}$, $\overline{\mu_i}$ and $\overline{\Sigma_i}$ refer to the weight, mean vector and diagonal covariance matrix of the i^{th} Gaussian component of the updated GMM $\overline{\lambda}$ respectively. x_t is a F dimensional feature vector. The *a posteriori* probability for Gaussian i is given by

$$\mathrm{P}\left(i|x_t, \lambda\right) = \frac{w_i \mathcal{N}\left(x_t; \mu_i, \Sigma_i\right)}{\sum_{k=1}^{N} w_k \mathcal{N}\left(x_t; \mu_k, \Sigma_k\right)} \tag{1.6}$$

1.5.2 Training Speaker Models: Maximum *A Posteriori* Adaptation

Speaker-specific training data is typically too scarce to warrant reliable maximum-likelihood estimates of the underlying speaker-dependent models. In contrast, the generally large amounts of data used in estimating the speaker-independent UBM allows this model's parameters to serve as an appropriate starting point in adaptative speaker modeling. Accordingly, the parameters of a speaker-dependent model are determined via Maximum *A Posteriori* adaptation of the initial parameters of the prior model (UBM), using the target speaker training utterances. By virtue of the typically limited amount of corresponding data, the resulting MAP-adapted parameters will tend to be much more reliable than their ML-trained counterparts (EM-algorithm).

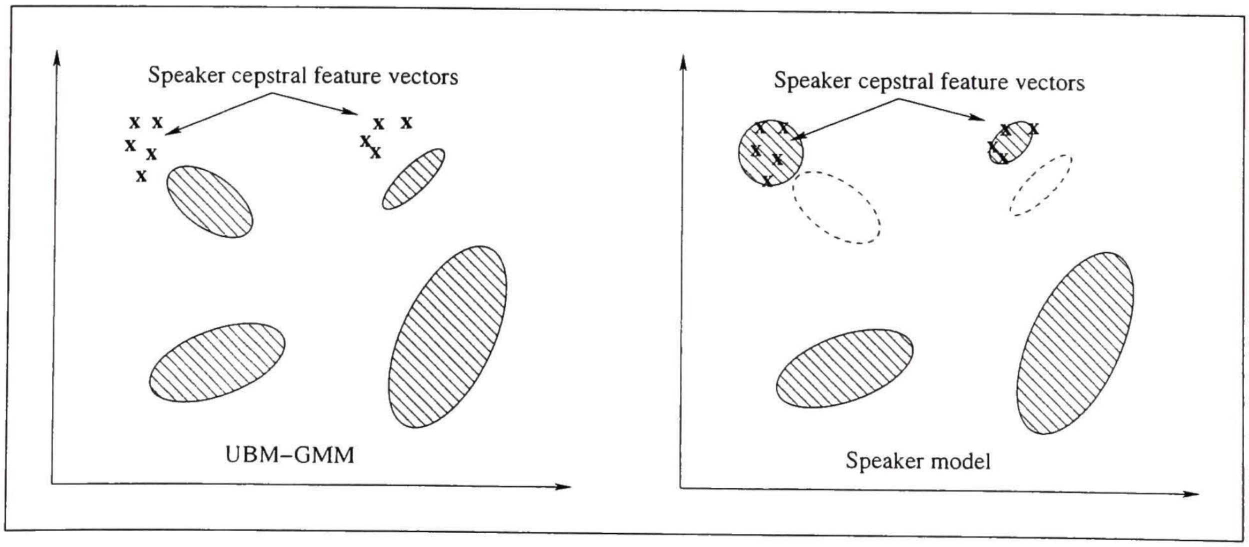
Speaker cepstral feature vectors
UBM–GMM
Speaker cepstral feature vectors
Speaker model

Figure 1.3 Maximum *A Posteriori* adaptation taken from (Reynolds et al., 2000).

The definition of MAP adaptation is as follows. Given a GMM-based UBM defined by $\lambda_\Omega = \{(w_i, \mu_i, \Sigma_i)\ i = 1..C\}$ and a series of acoustic vectors $X = \{x_1, x_2, ..., x_T\}$ corresponding to the hypothesized speaker X, we first compute the probabilistic alignment of the training vectors with respect to the UBM mixture components. For each mixture component i of the UBM, we compute its posterior distribution given the frame x_t:

$$\mathrm{P}\left(i|x_t, \lambda_\Omega\right) = \frac{w_i \mathcal{N}\left(x_t; \mu_i, \Sigma_i\right)}{\sum\limits_{j=1}^{C} w_j \mathcal{N}\left(x_t; \mu_j, \Sigma_j\right)} \tag{1.7}$$

We then use $\mathrm{P}(i|x_t, \lambda_\Omega)$ and x_t to compute the sufficient statistics for the weight, mean, and variance parameters:

$$\mathrm{P}\left(i|\lambda_\Omega\right) = \sum_{t=1}^{T} \mathrm{P}\left(i|x_t, \lambda_\Omega\right) \tag{1.8}$$

$$\mathrm{E}_i\left[x\right] = \frac{1}{\mathrm{P}\left(i|\lambda_\Omega\right)} \sum_{t=1}^{T} \mathrm{P}\left(i|x_t, \lambda_\Omega\right) x_t \tag{1.9}$$

$$\mathrm{E}_i\left[x\, x^t\right] = \frac{1}{\mathrm{P}\left(i|\lambda_\Omega\right)} \sum_{t=1}^{T} \mathrm{P}\left(i|x_t, \lambda_\Omega\right) x_t\, x_t^t \tag{1.10}$$

These sufficient statistics from the training data are used to update the UBM parameters λ_Ω. The adapted parameters for mixture i (Figure 1.3) are computed as follows:

$$\boxed{\tilde{w}_i = \left(\alpha_i \frac{\mathrm{P}\left(i|\lambda_\Omega\right)}{T} + \left(1 - \alpha_i^w\right) w_i\right) \gamma} \tag{1.11}$$

$$\boxed{\tilde{\mu}_i = \alpha_i\, \mathrm{E}_i\left[X\right] + \left(1 - \alpha_i^\mu\right) \mu_i} \tag{1.12}$$

$$\boxed{\tilde{\Sigma}_i = \alpha_i\, \mathrm{E}_i\left[x\, x^t\right] + \left(1 - \alpha_i^\Sigma\right)\left(\Sigma + \mu_i\, \mu_i^t\right) - \mu_i\, \mu_i^t} \tag{1.13}$$

The scale factor, γ, ensures that all adapted mixture weights sum to unity. The regularization parameter α_i control the balance between old and new estimates of the GMM parameters.

They are defined as:

$$\alpha_i = \frac{\mathrm{P}\left(i|\lambda_\Omega\right)}{\mathrm{P}\left(i|\lambda_\Omega\right) + r} \tag{1.14}$$

where r is a constant relevance factor. In practice, only mean vectors μ_i, $i = 1..C$ are adapted. Updated weights and covariance matrices do not significantly impact on system performance (Reynolds et al., 2000). Figure 1.3 shows an example of MAP adaptation when the mean and variance of observed Gaussians are adapted.

1.5.3 Log-Likelihood Ratio Scoring

The task in speaker verification is to ascertain whether or not a test set of speech frames $X = \{x_1, x_2, ..., x_T\}$ belongs to the claimed speaker s. With generative models, the aim is to test the following hypotheses:

- H_{tar}: X is uttered by speaker s.
- H_{non} : X is not uttered by speaker s.

The decision score is based on a likelihood ratio. It is evaluated by the following formula:

$$\boxed{S(X) = \frac{\mathrm{P}\left(X|H_{tar}\right)}{\mathrm{P}\left(X|H_{non}\right)} \begin{array}{l} \geq \theta \Longrightarrow H_{tar} \\ < \theta \Longrightarrow H_{non} \end{array}} \tag{1.15}$$

where $\mathrm{P}\left(X|H_{tar}\right)$ and $\mathrm{P}\left(X|H_{non}\right)$ are respectively the likelihood of X under the assumption that X is uttered or not by speaker s, and θ represents a decision threshold. If the computed score $S(X)$ is greater than the decision threshold θ, we conclude that test segment X is indeed uttered by speaker s. Otherwise, speaker s is deemed to be an impostor.

In practice, H_{tar} is the speaker model λ_s and H_{non} is the UBM λ_Ω defined previously. Using a logarithmic scale, the score defined in equation (1.15) can be rewritten as:

$$S(X, \lambda_s) = \log\left(\frac{\mathrm{P}\left(X|\lambda_s\right)}{\mathrm{P}\left(X|\lambda_\Omega\right)}\right) \tag{1.16}$$

$$\boxed{S(X, \lambda_s) = \log \mathrm{P}\left(X|\lambda_s\right) - \log \mathrm{P}\left(X|\lambda_\Omega\right)} \tag{1.17}$$

where $\mathrm{P}\left(X|\lambda_s\right)$ and $\mathrm{P}\left(X|\lambda_\Omega\right)$ are obtained using equation (1.2).

The decision threshold can be speaker-dependent or independent. In the context of the NIST-SRE campaign, we used a development dataset to estimate a speaker-independent decision threshold. This threshold satisfy the condition of minimizing the Detection Cost Function (DCF) which is used as performance measure of the SV systems. Subsection 1.8.1 introduces and gives more details about the DCF.

Figure 1.4 summarizes all components of the GMM-UBM system. Usually, many score normalization methods are applied in order to enhance the decision performance. These methods are intended to reduce the variability of intersession effects in the score space. They are presented in section 1.7.3.

1.6 Discriminative Models

As stated before, the most widely used methods in speaker verification are based on generative models, more precisely the Gaussian mixture model (Doddington et al., 2000) (Reynolds et al., 2000). However, these generic models are not discriminative. This is a consequence of speaker model training. Each speaker GMM is trained only on data from the same speaker. To solve this problem, new criteria have been developed that allow discriminative learning of generative models. It is also possible to combine generative models with discriminative methods such as GMM-based, support vector machine systems. In the following, we will outline the major discriminative approaches used in speaker verification.

1.6.1 Discriminative Training of Generative Models

The discrimination between the target speaker and the set of impostors is the most important problem connected with speaker model learning. The EM algorithm, which represents the standard approach to GMM training, incorporates target speaker data exclusively for estimat-

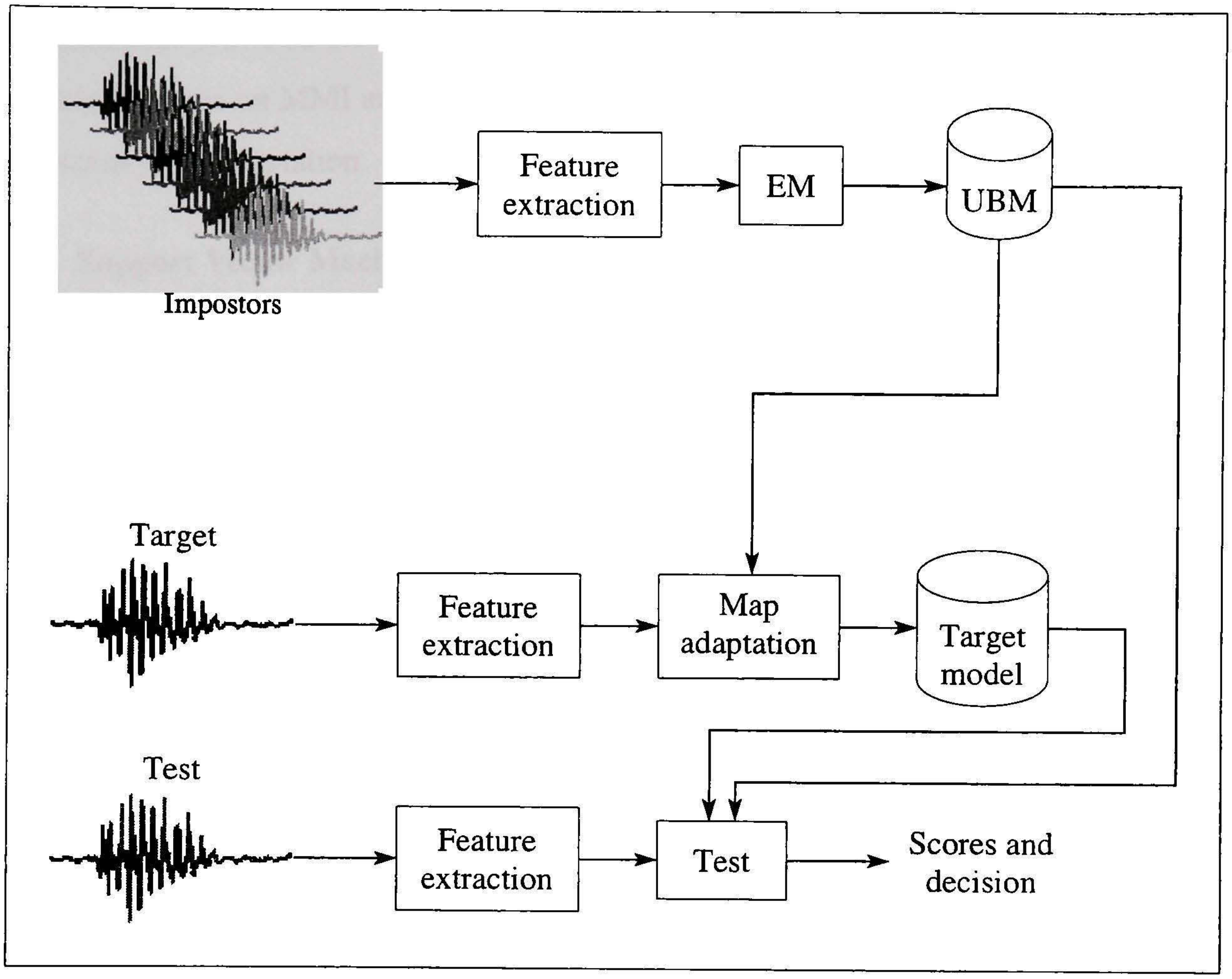

Figure 1.4 Architecture of the GMM-UBM system.

ing the GMM parameters. In state-of-the-art undertakings in the field of speaker verification, and to some extent the field of speech recognition, some attempts to achieve discriminative learning of generative models have been pursued. Some of these approaches are based on the Maximum Mutual Information (MMI) criterion, that was first introduced in the field of speech recognition (Normandin, 1991) (Gunawardana, 2001) (Gunawardana et Byrne, 2001); the methodology can be used to train a target model's parameters, while explicitly taking other classes of data into account. To date, it also has been applied to speaker verification and compared to other discriminative training criteria, such as Minimum Classification Error in (Ma et Chang, 2003). Preti *et al.* (Preti et al., 2006) applied the MMI criterion to adapt Gaussian weights exclusively, and compared this approach to two other adaptation procedures for Gaussian weights. The first one employs the Maximum Likelihood (ML) criterion, while the other procedure draws on the Maximum A Posteriori (MAP) criterion. The results reveal that the methods based on MMI and ML criteria yield equivalent performance, and these in turn outperform MAP adaptation.

1.6.2 Support Vector Machines for Speaker Verification

In the pattern recognition and machine learning communities, the method of Support Vector Machines (SVM) is acknowledged as one of the preeminent discriminative approaches. They are binary classifiers (based on finding a discriminating surface between two classes) that can be extended to n classes. The original linear approach has been extended to nonlinear classification, which has shown to be extremely useful in dealing with a number of classification problems. Processing a nonlinear problem incorporates kernel functions that project the input data to another feature space; as a result the problem is converted to a linear one in this new space. SVM will be described in greater detail in Chapter 3.

Two different approaches have been experimented with in order to use SVMs for the purpose of speaker verification. The first approach consists in performing a combination between generative models and SVMs. Several types of combination have been proposed. A case in point is the work presented in (Dong et Zhaohui, 2001), which performs discriminating training of GMMs through the use of a continuous density SVM. Another form of combination consists in using SVMs in a post-processing of the GMMs models, using Fisher mapping (Wan et Renals, 2003) (Wan et Renals, 2005). This treatment produces high-dimensional vectors, with the number of dimensions equals to the number of parameters of the GMM. These vectors are subsequently used by SVMs to achieve discrimination and decision. Finally, the most commonly used, and in addition most powerful methods (Campbell et al., 2006a) (Dehak et Chollet, 2006) (Dehak et al., 2007a), exploit the advantages of combining GMMs and SVMs into a single system. This construct uses a probabilistic distance kernel derived from the Kullback-Leibler (KL) divergence between GMMs. With these last methods, the SVM input space coincides with the GMM means. As a result, the GMM training procedure is used as feature extraction for SVM methods.

The second class of approaches consists in applying SVMs directly to the acoustic data. The method implemented in (Schmidt et Gish, 1996) trains SVMs directly on the acoustic vectors that characterize the client and impostor data. During testing, the segment score is obtained by averaging the scores of the SVM output for each frame. There exists other applications of SVM in speaker verification that operate on kernel sequences. The generalized linear discriminant sequence (GLDS) kernel is the most widely used kernel function. Proposed by William Campbell (Campbell, 2002), this kernel offers the advantage of eliminating context variability by averaging features over the entire projected vectors.

1.7 Robust Speaker Verification System

Tracking variability represents a major challenge of robust speaker verification methodology. The speech is a complex signal; it is very sensitive to changing channel conditions in acquisition and transmission steps. These conditions introduce distortions in the speech signal. Several methods have been used in speaker verification to remove channel variability. These methods apply different operations at each step of speaker verification. In feature space, Cepstral Mean Substraction (Furui, 1981), Feature Warping (Pelecanos et Sridharan, 2001) and Feature Mapping (Reynolds, 2003) are the best known methods. Joint Factor Analysis (Kenny et Dumouchel, 2004) and Nuisance Attribute Projection (Campbell et al., 2006a) are used in GMM parameter space. There also exist many score normalization methods which are used in score space in order to track variability.

1.7.1 Feature Space

Cepstral Mean Substraction

Cepstral Mean Subtraction (CMS) (Furui, 1981) is a method for normalizing cepstral features. The idea behind CMS is to obtain a centered feature: the mean feature vector computed over complete segment is subtracted from each individual feature vector. The principal objective of this operation is to reduce the noise caused by stationary convolution transmission channel effects. CMS can also be applied using a sliding window to reflect the time variation of transmission channel effects in a single recording. In essence, the CMS is a technique for feature normalization which compensates for convolution stationary noise. Other types of noise are not compensated for.

Feature Warping

Gaussianization or *Feature Warping* (Pelecanos et Sridharan, 2001) is a method that involves a mapping of feature component amplitudes so that each component exhibits a normal distribution over a sliding window (in practice 3 seconds in duration). This transformation is performed using a table that establishes the correspondence between the acoustic feature distribution and the normal distribution. Feature warping is motivated by the fact that distortions caused by additive and convolution noise affect the distribution of cepstral features. This method leads to slightly better performance than the CMS method.

Feature Mapping

Feature Mapping is one of the first methods introduced for addressing the problem of variability between data acquisition conditions during the training and testing steps. This technique was first introduced by Reynolds in (Reynolds, 2003). It is used to normalize the cepstral features using a channel-independent UBM model and several channel-dependent UBM models. The channel-independent UBM is built using a large corpus of speech recordings under different acquisition conditions. Thereafter, for each channel type, the channel-dependent UBM is derived from the independent-channel UBM via adaptation to the channel-specific training data (see MAP training Section 1.5.2). For the purpose of normalizing the feature vectors of a test file, the recording's channel type must first be assessed. This is done by computing the likelihoods of the test utterance against the channel-dependent UBMs. The channel is inferred from the channel-dependent UBM with the highest likelihood. Normalizing a feature vector x is achieved by the following formula:

$$\hat{x}_t = \frac{\sigma_{G_c^i}}{\sigma_{G_c^d}}(x_t - \mu_{G_c^d}) + \mu_{G_c^i} \tag{1.18}$$

where G_c^i is the Gaussian component of the channel-independent UBM with highest likelihood; $\mu_{G_c^i}$ and $\sigma_{G_c^i}$ represent, respectively, the mean and standard deviation of this Gaussian component. The corresponding Gaussian component in the channel-dependent UBM is denoted by G_c^d; $\mu_{G_c^d}$ and $\sigma_{G_c^d}$ are, respectively, the mean and standard deviation of this Gaussian component.

1.7.2 GMM Parameter Space

The related methods operate on the GMM parameter space. Since only mean vectors are adapted, the other parameters stay unchanged and are equal to the UBM ones. Accordingly, these methods operate on the GMM mean supervector space. Each GMM supervector is the concatenation of mean vectors from each Gaussian component.

Joint Factor Analysis

Joint Factor Analysis (JFA) (Kenny et Dumouchel, 2004) (Kenny et al., 2005b) is a method used for modeling channel and speaker variabilities in GMM parameter space. With JFA, we assume that each GMM supervector M for a given utterance is the result of two independent components. The first supervector s is speaker-dependent and the second one, c is channel-dependent:

$$M = s + c \tag{1.19}$$

Unlike feature mapping, which addresses channel effect problems through a discrete solution, JFA uses a continuous situation. The channel supervector is defined in continuous space. Details will be presented in Chapter 2.

Vair *et al.* (Vair et al., 2006) propose a new method based on estimating the effect of the channel in the GMM parameter space then using that to normalize the feature vectors in the acoustic parameters space.

Nuisance Attribute Projection

In (Solomonoff et al., 2004), the authors propose a technique named Nuisance Attribute Projection (NAP) for channel compensation in the context of SVM for speaker verification. Recently this technique is applied in the GMM supervector space (Campbell et al., 2006a) to design a new SV system based on combination of SVM and GMM approaches. The principal idea of the NAP algorithm is to use a projection matrix P in the SVM feature space in order to cancel the channel effect. The underlying assumption of this method is that the GMM supervector space is a combination of two orthogonal subspaces: the first one represents the channel information and the second is immunized against the impact of the channel. The projection matrix P is defined as follows:

$$\boxed{P = I - vv^t} \text{ with } \|v\| = 1 \tag{1.20}$$

where v is a vector from the channel subspace basis. The purpose of this matrix is to project all vectors onto a space immunized against the impact of the channel. This projection matrix is defined in the feature space rather than in the input space. So in the case where $\phi(x)$ is the mapping function of the input vector x, the new mapping function that uses the projection matrix P will be:

$$\boxed{\widehat{\phi}(x) = P\,\phi(x)} \tag{1.21}$$

The kernel function between two input vectors x_i,x_j is defined as the scalar product of the two mappings $\widehat{\phi}(x_i)$, $\widehat{\phi}(x_j)$:

$$\widehat{K}_{ij} = \widehat{\phi}(x_i)^t \widehat{\phi}(x_j) \tag{1.22}$$

Introducing $A = [\phi(x_1), \phi(x_2), ..., \phi(x_N)]$, the original Gram matrix defined by $\mathbf{K} = A^t A$ will be:

$$\begin{aligned} \widehat{\mathbf{K}} &= (P\,A)^t\,(P\,A) && (1.23) \\ &= K - A^t v v^t A && (1.24) \end{aligned}$$

If we have many channel types $\{c_1, c_2, ..., c_d\}$, NAP consists in finding the projection matrix that minimizes the distance between the projection of two feature vectors of the same speaker but with different channel effects:

$$\widetilde{P} = \arg\min_{P} \sum_{i,j \in \{c_1, c_2, ..., c_d\}} b_{ij} \left\| P\left(\phi(x_j) - \phi(x_i)\right)\right\|^2 \tag{1.25}$$

where b_{ij} is a weight value equal to one if x_i and x_j represent the same speaker; zero otherwise.

in (Solomonoff et al., 2004), the authors show that this problem is equivalent to an eigenvalue problem:

$$K\,Z\,K\,v = \lambda\,K\,v \tag{1.26}$$

where $Z = \text{diag}(B\mathbf{1}) - B$; B is the matrix of weights b_{ij}. $\mathbf{1}$ is a column vector of unit value, v corresponds to the eigenvector with highest eigenvalue of this problem. Additional details about this method are given in Chapter 4.

1.7.3 Score Space

Score normalization methods are applied to reduce the variability of the decision scores. These techniques are based on the assumption that the distribution of target speaker and impostor scores follow two distinct normal distributions. The normalization processing is performed as follows:

$$\boxed{S\left(X, \lambda_L\right)_{norm} = \frac{S\left(X, \lambda_L\right) - \mu}{\sigma}} \tag{1.27}$$

where X is the test segment and L is the proclaimed identity. The definition of μ and σ depends on the score normalization method.

Z-Norm

The z-norm score normalization addresses the problem of speaker score variability. It allows finding a decision threshold that is independent of the target speaker. For z-norm, we consider a set of impostor segments $X_1, X_2, ..., X_J$. For each proclaimed identity L, we compute a speaker-dependent μ_L and σ_L as follows:

$$\mu_L = \frac{1}{J}\sum_{j=1}^{J} S\left(X_j, \lambda_L\right) \tag{1.28}$$

$$\sigma_L = \sqrt{\frac{1}{J}\sum_{j=1}^{J}\left(S\left(X_j, \lambda_L\right) - \mu_{\lambda_L}\right)^2} \tag{1.29}$$

T-Norm

The t-norm addresses the problem of session variability. It compensates for differences between the training and testing conditions. For t-norm, we consider a set of impostor models

$\lambda_1, \lambda_2, ..., \lambda_N$. For each test segment X, we compute a test-dependent μ_X and σ_X as follows:

$$\mu_X = \frac{1}{N}\sum_{n=1}^{N} S(X, \lambda_n) \tag{1.30}$$

$$\sigma_X = \sqrt{\frac{1}{N}\sum_{n=1}^{N} \left(S(X, \lambda_n) - \mu_X\right)^2} \tag{1.31}$$

Remark:

It is possible to combine the z-norm and t-norm score normalizations. There is two combinations, the first one named zt-normalization. It consists in first applying a z-norm, followed by a t-norm. This normalization is the most widely used because it offers better performance than application of one normalization only. The second one is the tz-normalization which consists to apply t-norm first followed by z-norm.

1.8 Performance Measurement

To measure the performance of a speaker verification system, we analyze two types of errors:

- False Acceptance (FA): this occurs when the system grants access to an impostor.
- False Rejection (FR): this occurs when the system denies access to an enrolled speaker.

We generally analyze the rate of FA, R_{FA}, and that of FR, R_{FR}. These rates are computed as follows:

$$\boxed{R_{FA} = \frac{\text{Number of FA}}{\text{Number of impostors accesses}}} \tag{1.32}$$

$$\boxed{R_{FR} = \frac{\text{Number of FR}}{\text{Number of target accesses}}} \tag{1.33}$$

These two rates depend on the decision threshold. For higher decision thresholds, we will accept fewer accesses; false acceptances will be fewer but false rejections will be more common. For lower decision thresholds, we will accept more access requests; false acceptances will be higher but false rejections will be fewer. An operational system needs to ajust the decision threshold in order to find a compromise between both operating rates.

1.8.1 Detection Cost Function

In order to measure the performance of a speaker verification system given a fixed decision threshold, we define a Detection Cost Function (DCF), which is a weighted sum of the FA and FR rates. These weights correspond to the costs C_{FR} and C_{FA} associated with the R_{FR}, R_{FA} respectively and the *a priori* probability of impostor P_{non} and the target speaker P_{tar} trials. The detection cost function is defined as follows:

$$\boxed{DCF = C_{FR}\,P_{tar}\,R_{FR} + C_{FA}\,P_{non}\,R_{FA}} \tag{1.34}$$

The cost values and *a priori* probabilities used to evaluate the DCF are fixed depending on the application context. In the NIST speaker recognition evaluation, these parameters are specified by the NIST evaluation plan[1] : $C_{FR} = 10$, $C_{FA} = 1$, $P_{tar} = 0.001$ and $P_{non} = 1 - P_{tar}$. The value of DCF depends on the value of the decision threshold. The MinDCF is the minimum value of the DCF obtained when the decision threshold is changed. This last value is used as the principal metric on the NIST speaker recognition evaluation campaign.

1.8.2 Equal Error Rate

The Equal Error Rate (EER) is another criterion used to compare the performance of speaker verification systems. It represents the operating point where the false acceptance rate is equal to the false rejection rate.

[1] `http://www.nist.gov/speech/tests/spk/index.htm`

1.8.3 DET Curve

The criteria presented so far give the performance of speaker verification systems at an operating point (corresponding to a fixed decision threshold). Another method for viewing the performance at different points on the same curve is the Detection Error Tradeoff (DET) curve (Figure 1.5), introduced by Martin *et al.* (Martin et al., 1997). It is a variant of the Receiver Operating Characteristics (ROC) curve that plots the variation of the FR rate to FA rate according to different decision thresholds. The EER represents the point on the curve where both rates are equal.

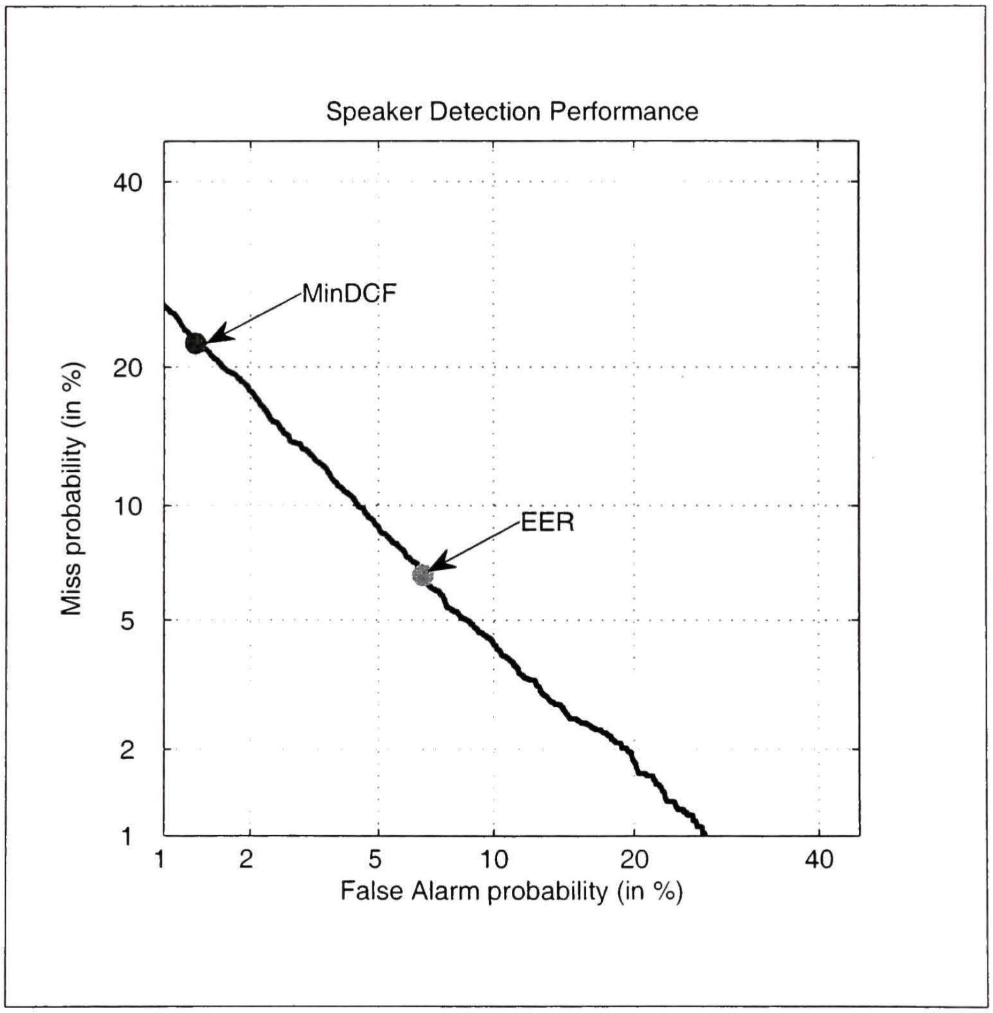

Figure 1.5 DET Curve showing the results of a speaker verification system

CHAPTER 2

JOINT FACTOR ANALYSIS

This chapter describes the joint factor analysis (JFA). We detail how the JFA approach allows to model the speaker and intersession variabilities. We also present the underlying steps involved in the JFA of a speaker verification system.

2.1 Joint Factor Analysis

In a generative approach based on Gaussian mixture models, each speaker is represented by a GMM composed of C Gaussians. These Gaussians are learned in a continuous parameter space of dimension F. Each Gaussian is characterized by a mean vector, a diagonal covariance matrix and a weight. A target speaker GMM is built by adapting the GMM components of the Universal Background Model (UBM) to the considered speaker's frames. The UBM is trained on a large set of speaker training data. Joint factor analysis (Kenny et al., 2007a,b, 2008b) is a model that takes into account speaker and intersession variabilities in the context of the GMM framework. Traditionally used in conjunction with cepstral features, its application can be extended to other continuous features where GMM modeling is appropriate. The JFA model is based on a combination of classical MAP adaptation and eigenvoice for modeling speaker variability, and eigenchannel MAP for modeling intersession variability. The intersession variability in the spectral speech features is generally caused by channel transmission effects. This is the reason for using the term channel variability rather than intersession variability in the context of spectral features. The key assumption in joint factor analysis is that the GMM supervector of speaker- and channel-dependent M for a given utterance can be broken down into a sum of two supervectors:

$$\boxed{M = s + c} \tag{2.1}$$

where supervector s depends on the speaker and supervector c depends on the channel.

The GMM supervector is a CF-dimensional vector obtained via concatenation of all Gaussian component means. In the following sections, we will outline how the speaker and channel supervectors are determined.

2.2 Speaker Variability Modeling

In traditional MAP adaptation as used in speaker verification (Reynolds et al., 2000), the prior distribution of a GMM speaker supervector s is normally distributed with mean vector $\mathrm{E}\left[s\right] = m$ and covariance diagonal matrix $\mathrm{Cov}\left(s, s\right) = \frac{1}{\tau}\Sigma$, where m is the mean supervector of the universal background model, Σ is a block-diagonal matrix where the blocks correspond to the diagonal covariance matrices of the UBM and τ is the relevance factor. In (Reynolds et al., 2000), the authors empirically fit the value of τ to find a compromise between the prior distribution speaker variance and UBM variance. Instead of using empirical estimation of the relevance factor, Kenny *et al.* proposed in (Kenny et al., 2007a,b) a ML-based estimation of the *a priori* variance of the speaker population within a training corpus. In this new modeling, the supervector s of a randomly chosen speaker can be written in the form of hidden variables as follows:

$$\boxed{s = m + Dz} \tag{2.2}$$

where m is the the speaker- and channel-independent supervector of dimension CF. The vector z is a hidden vector of dimension CF, *a priori* associated with a standard normal distribution $\mathrm{P}\left(z\right) \sim \mathcal{N}\left(z|0, I\right)$, and D is a diagonal matrix of dimension $CF \times CF$. In order to calculate the posterior distribution of speaker supervector s, we need to know the *a priori* probability of supervector s. The prior distribution of this supervector is normally distributed with the following parameters:

$$\begin{aligned}
\mathrm{E}\left[s\right] &= \mathrm{E}\left[m + Dz\right] && (2.3)\\
&= m + D\,\mathrm{E}\left[z\right] && (2.4)\\
\mathrm{Cov}\left(s, s\right) &= \mathrm{E}\left[\left(s - \mathrm{E}\left[s\right]\right)\left(s - \mathrm{E}\left[s\right]\right)^t\right] && (2.5)\\
&= \mathrm{E}\left[\left(Dz - D\,\mathrm{E}\left[z\right]\right)\left(z^tD^t - \mathrm{E}\left[z\right]^t D^t\right)\right] && (2.6)\\
&= \mathrm{E}\Big[Dzz^tD^t - Dz\,\mathrm{E}\left[z\right]^t D^t - D\,\mathrm{E}\left[z\right] z^tD^t \\
&\qquad + D\,\mathrm{E}\left[z\right]\mathrm{E}\left[z\right]^t D^t\Big] && (2.7)\\
&= D\,\mathrm{E}\left[\left(z - \mathrm{E}\left[z\right]\right)\left(z - \mathrm{E}\left[z\right]\right)^t\right] D^t && (2.8)\\
&= D\,\mathrm{Cov}\left(z, z\right) D^t && (2.9)
\end{aligned}$$

We already know that the prior distribution of hidden variable z is a standard normal distribution, so the mean vector and covariance matrix of the *a priori* distribution of supervector s are simplified to

$$\begin{aligned}
\text{Prior expectation of } s &= m && (2.10)\\
\text{Prior covariance matrix of } s &= DD^t && (2.11)
\end{aligned}$$

The matrix D is derived from the *a priori* distribution of speaker supervectors; it is estimated in an iterative fashion from the training corpus comprised of speaker-specific sets of audio recordings. In (Kenny et al., 2008b), the authors also proposed parameter updates of supervector m using the same data used to train the diagonal matrix D.

Given a sequence of speaker training observations and model parameters m and D, the posterior distribution of speaker supervector s is based on the calculation of the posterior probability of the hidden variable associated with that same speaker. The calculation of the posterior distribution of the hidden variable is described in Appendix A. The posterior distribution of the latent variable z is modeled by the mean vector $\mathrm{E}\left[z\right]$ and covariance matrix $\mathrm{Cov}\left(z, z\right)$, so the posterior distribution of supervector s is modeled by a mean vector and covariance matrix derived, respectively, in the same manner as for equations (2.4) and (2.9):

$$\begin{aligned} \mathrm{E}[s] &= m + D\,\mathrm{E}[z] \qquad (2.12) \\ \mathrm{Cov}(s, s) &= D\,\mathrm{Cov}(z, z)\,D^t \qquad (2.13) \end{aligned}$$

The expectation vector $\mathrm{E}[s]$ of the target speaker posterior probability is the corresponding speaker GMM supervector estimated via MAP adaptation. Unlike classical MAP adaptation (Reynolds et al., 2000), this new MAP modeling (Kenny et al., 2007a,b) allows taking into account the uncertainty associated with the estimation of the speaker's GMM. Statistically speaking, the covariance matrix $\mathrm{Cov}(s, s)$ models the uncertainty associated with MAP estimation of the speaker's GMM. When the number of target speaker training frames increases, the influence of $\mathrm{Cov}(s, s)$ decreases. Provided the matrix D is well-conditioned, MAP adaptation using the *a priori* distribution is equivalent to Maximum Likelihood training of the speakers, when sufficient speaker data are available for adaptation. The use of MAP adaptation with prior diagonal covariance matrix D does not model correlations between Gaussian components of a single GMM. As a result, only observed Gaussians are adapted; other Gaussians remain unchanged. Had we imposed $D^2 = \frac{1}{\tau}\Sigma$, then the MAP adaptation proposed in (Kenny et al., 2007a,b) would reduce to the classical MAP adaptation (Reynolds et al., 2000).

We now describe another adaptation technique for speaker GMM estimation. The technique, called eigenvoice adaptation, is rooted in a definition of speaker population space. Given the availability of speaker recordings, the aim of this adaptation is to locate the speaker within the speaker space. Eigenvoice adaptation operates on the assumption of a low rank rectangular matrix V of dimension $CF \times R$, with $R \ll CF$, that defines a representation of the speaker space. The supervector s of a randomly chosen speaker is obtained by:

$$\boxed{s = m + Vy} \qquad (2.14)$$

where m corresponds to the UBM mean supervector and y is a hidden vector of dimension R having a standard normal prior distribution $\mathrm{P}(y) \sim \mathcal{N}(y|0, I)$. Referring once again to equations (2.4) and (2.9), it is readily shown that the expectation and covariance matrices of the prior distribution of supervector s are obtained by:

$$\text{prior expectation of } s = m \tag{2.15}$$

$$\text{prior covariance matrix of } s = VV^t \tag{2.16}$$

The prior distribution of supervector s is used to estimate its posterior distribution. The posterior distribution of a speaker supervector in the case of eigenvoice adaptation is modeled by a mean vector $\mathrm{E}[s]$ and covariance matrix $\mathrm{Cov}(s, s)$ derived, respectively, in the same manner as for equations (2.4) and (2.9):

$$\mathrm{E}[s] = m + V\,\mathrm{E}[y] \tag{2.17}$$

$$\mathrm{Cov}(s, s) = V\,\mathrm{Cov}(y, y)\,V^t \tag{2.18}$$

When few observations are available, eigenvoice adaptation is more powerful than MAP adaptation for estimating speaker GMMs. This stems from the much lower dimension of latent vector y used in eigenvoice adaptation, compared to that of latent vector z used in MAP adaptation; as a result very little data is required to adequately estimate the posterior probability of the eigenvoice-based vector y. Unlike MAP adaptation, eigenvoices model correlations between GMM components, which allows us to adapt non-observed Gaussians. Eigenvoice adaptation is based on the assumption that the rank R of estimated matrix V is less than or equal to the number of speakers in the training corpus (Kenny et al., 2007a). It is necessary to have a significant number of speakers to estimate this matrix well enough. One final, important point concerning eigenvoice adaptation: when large amounts of data are used to enroll the speaker model, it cannot be proven that model will behave properly (Kenny et al., 2007a).

It is clear that both adaptation methods (classical MAP and eigenvoices) are complementary. Classical MAP adaptation is appropriate in cases where we have sufficient data to enroll the target model, whereas eigenvoice adaptation is the method of choice when data are scarce. It would be interesting to consider combination of both adaptation techniques. In this case, the supervector s of a randomly chosen speaker is distributed according to:

$$\boxed{s = m + Vy + Dz} \tag{2.19}$$

The two hidden vectors y and z are mutually independent and each vector has a standard normal prior distribution. The supervector s follows a prior normal distribution characterized by mean m and covariance matrix $VV^t + D^2$. This last modeling corresponds to the factor analysis as proposed for speaker verification (Kenny et al., 2007b). We refer to the components of y as speaker factors and to the components of z as common factors.

2.3 Channel Variability Modeling

The factor analysis proposed by in (Kenny et al., 2008b) is based on modeling channel effect. As is the case with speaker space, joint factor analysis also models the channel space. The supervector c represents the channel supervector. It models channel effects in the given recording. This supervector is written as follows:

$$\boxed{c = Ux} \tag{2.20}$$

where U is a low-rank rectangular matrix $R_c \ll CF$ whose columns represent the eigenvectors of the channel covariance matrix. The matrix U defines the channel space. The hidden variable x has a standard normal prior distribution $\mathrm{P}(x) \sim \mathcal{N}(x|0, I)$. This is equivalent to stating that supervector c follows a normal prior distribution with mean vector equal to zero and covariance matrix UU^t. This technique is referred to as eigenchannel adaptation (Kenny et al., 2007a), which has the same form as the eigenvoice adaptation procedure outlined in the preceding section. The components of the vector x are called the channel factors. When the speaker and channel factors are both taken into consideration for modeling the system, we refer to the resulting model as Joint Factor Analysis (JFA).

2.4 Joint Factor Analysis-based Speaker Verification System

In this section, we outline the sequence of steps required to produce a speaker verification system based on the joint factor analysis approach.

2.4.1 Universal Background Model

In this step, the universal background model Ω is determined by estimating its underlying parameters according to an iterative EM algorithm. The UBM is a GMM composed of C Gaussian components trained on F-dimensional feature frames. This GMM is characterized by Gaussian mixture weights, its supervector m of dimension CF and covariance matrix Σ of dimension $CF \times CF$. The diagonal blocks of this covariance matrix correspond to the diagonal covariance matrices of each Gaussian $\Sigma_c \quad (c = 1, ..., C)$.

The UBM is used to extract first- and second-order Baum-Welch statistics, for subsequent use by the joint factor analysis modeling. Suppose we have a sequence of T frames $\{x_1, x_2, ..., x_T\}$ and a UBM composed of C Gaussians. To extract the Baum-Welch statistics, we define the variable X_t^c which is given by:

$$\boxed{X_t^c = \mathrm{P}\left(c|x_t, \Omega\right) x_t} \tag{2.21}$$

where c is the Gaussian index. The sufficient statistics are obtained by an alignment of the frames using the UBM Gaussians.

- The statistics of order zero :

$$\boxed{N_c = \sum_{t=1}^{T} \mathrm{P}\left(c|x_t, \Omega\right)} \tag{2.22}$$

- The first order statistics :

$$\boxed{F_c = \sum_{t=1}^{T} X_t^c} \tag{2.23}$$

- The second order statistics :

$$\boxed{S_c = \mathrm{diag}\left(\sum_{t=1}^{T} \mathrm{P}\left(c|x_t, \Omega\right) x_t x_t^t\right)} \tag{2.24}$$

2.4.2 Training of Joint Factor Analysis Hyperparameters

The joint factor analysis hyperparameters are given by $\lambda = (m, V, D, U, \Sigma)$. The diagonal covariance matrix Σ, of dimension $CF \times CF$, models the unresolved variability of the speaker and channel matrix representations (Kenny, 2005). The diagonal blocks of this matrix are denoted by $\Sigma_c \quad (c = 1, ..., C)$. A block element Σ_c is a diagonal covariance matrix associated with Gaussian mixture component c used to estimate the GMM log likelihood function. In the classical GMM-UBM based system, the covariance matrices $\Sigma_c \quad (c = 1, ..., C)$ are taken from the universal background model components in order to represent speaker uncertainties generated by MAP adaptation. However, in the joint factor analysis approach, the Σ matrix is trained on data that takes into account the variability of models associated with the speaker and channel supervector distributions. All the JFA hyperparameters are estimated iteratively using an EM algorithm in order to maximize the likelihood of the training corpus. The training database is composed of many speakers, and each speaker has several recordings under different channel conditions. The EM algorithm is performed in two steps. In the first step, we evaluate the posterior distribution of the hidden variables, given the speaker-sufficient statistics and current hyperparameter estimation. The second step consists in updating the joint factor analysis hyperparameters, based on the expectations and covariance matrices of the hidden variables obtained in the previous step. EM training is initiated by first initializing the JFA hyperparmeters. A random initial guess of the eigenvoice matrix V, eigenchannel matrix U and diagonal matrix D works well in practice. The supervector and covariance matrices of the universal background model can be used as initial estimates of supervector m and residual covariance matrices $\Sigma_c \quad (c = 1, ..., C)$ (Kenny et al., 2008b). The following two paragraphs outline in greater details, the EM steps involved in JFA training.

Posterior Distribution of Hidden Variables

Based on the current estimate of JFA hyperparameters $\lambda_0 = (m, V, D, U, \Sigma)$, prior speaker distribution and channel supervectors, the posterior distribution of the background speaker utterances [1] is computed using their Baum-Welch statistics. The evaluation of all joint factor analysis latent variables (y, z, x), given the Baum-Welch statistics of an utterance, is described in Appendix A.

Re-estimation of the Hyperparameters

Updated hyperparameters are conditioned by the current joint factor analysis hyperparameter estimate λ_0 and speaker posterior distributions obtained in the previous step. The speaker posterior distribution is characterized by the expectation and covariance matrix of the hidden variables. Two criteria are applied to re-estimate the joint factor analysis hyperparameters λ. The first estimate is based on maximum likelihood, while a second estimate is required to satisfy the minimum divergence criterion (Kenny, 2005).

In (Kenny, 2005), the authors start by training the hyperparameters related to the speaker supervectors which are m, V, D and Σ. In order to carry out this estimation, the Baum-Welch statistics of each speaker's utterances are pooled together. Pooling the statistics is motivated by the fact that averaging the statistics over all utterances of each speaker removes the chan-

[1] Background speakers are taken from all other available databases which do not contain the target speaker.

nel effects. These statistics are used in order to estimate the speaker and common space. The first version of the joint factor analysis is based on the joint estimation of the eigenvoice matrix V and of the diagonal matrix D (Kenny, 2005). Recently Kenny *et al.* (Kenny et al., 2008b) proposed a decoupled estimation of these two matrices. In the first step, we estimate the eigenvoice matrix, the supervector m and the residual variance Σ using a subset of the background data. In the second step, the diagonal matrix D is trained on another subset of data after removing the speaker effects, already modeled through eigenvoices, from the sufficient statistics. In this way, the diagonal matrix models the residual speaker variability not captured by the eigenvoices. We also re-estimate the residual covariance matrix Σ after removing the variability modeled by the diagonal matrix D. After training the hyperparameters which model the speaker supervector distribution, we estimate the eigenchannel matrix U and re-estimate the diagonal covariance matrix Σ in order to take into account the channel variability captured by the eigenchannels. The eigenchannel matrix is computed after centralizing the sufficient statistics for each utterance of each speaker with respect to their corresponding speaker supervectors.

2.4.3 Speaker Enrollment

When the full joint factor analysis model $\lambda_0 = (m, V, D, U, \Sigma)$ is used, the target speaker enrollment is based on the Baum-Welch statistics and the prior distribution of the speaker supervector s $\mathrm{P}(s) \sim \mathcal{N}(s|m, VV^t + D^2)$ and channel supervector c $\mathrm{P}(c) \sim \mathcal{N}(c|, 0, UU^t)$. The joint posterior distribution of all hidden variables y, z and x can be computed in the manner described in Appendix A. As explained in previous sections, the posterior distribution of a specific speaker is a normal distribution with expectation supervector $\mathrm{E}[s] = m + V\,\mathrm{E}[y] + D\,\mathrm{E}[z]$ and covariance matrix $\mathrm{Cov}(s, s) = V\,\mathrm{Cov}(y, y)\,V^t + D\,\mathrm{Cov}(z, z)\,D^t$ (Kenny, 2005). The target speaker supervector can be directly computed from the posterior distribution of the hidden variables. The computation of the expectation vectors and covariance matrices of all hidden variable posterior distributions are given in Appendix A. Figure 2.1 summarizes the estimation of the posterior speaker supervector distribution.

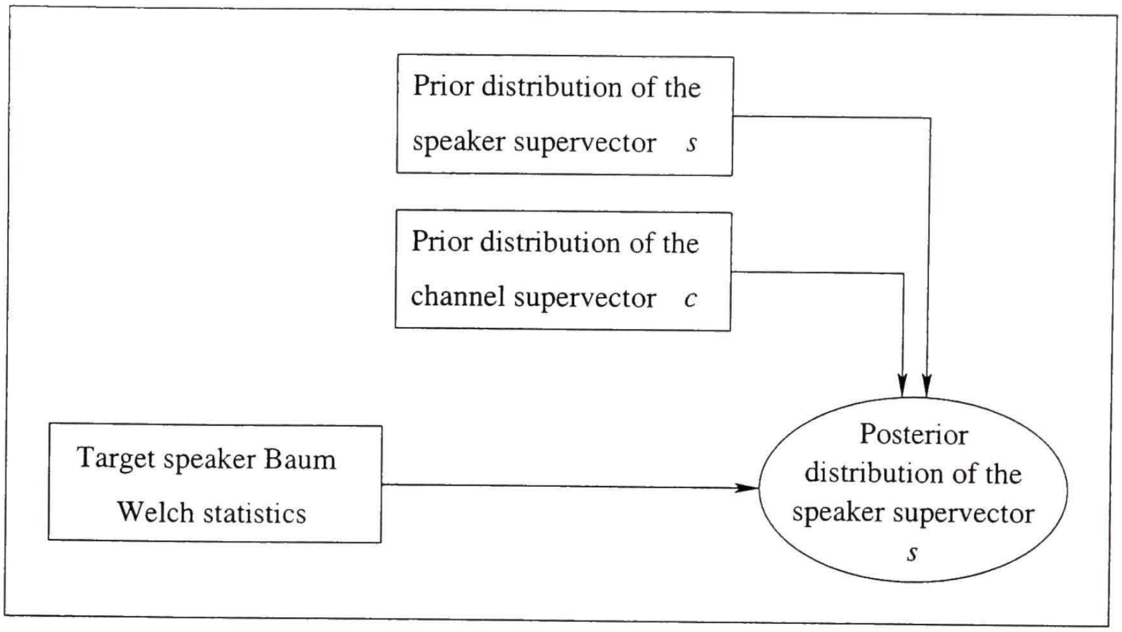
Prior distribution of the speaker supervector s
Prior distribution of the channel supervector c
Target speaker Baum Welch statistics
Posterior distribution of the speaker supervector s

Figure 2.1 Posterior distribution of target speaker supervector.

2.4.4 Test and Final Decision

This step consists in evaluating the log-likelihood ratio based on the target speaker and universal background models described in Chapter 1. The resulting score is then compared to a threshold in order to take the final decision. Given a target speaker supervector s and test utterance χ, and assuming that the test recording is produced by the target speaker, the GMM supervector (Kenny et al., 2007a) of this test utterance is given by:

$$M = s + Ux \tag{2.25}$$

where U is the eigenchannel matrix and x is the vector of channel factors. If we suppose x to be known, then it is straightforward to compute the conditional likelihood of the test utterance given the target speaker supervector and channel factor components. In practice, however, x is a hidden variable which we only know to be represented by a standard normal prior distribution. In this case, the likelihood of the test recording, given the claimed speaker, is given by integrating over all channel factors:

$$\boxed{\mathrm{P}(\chi|s) = \int \mathrm{P}(\chi|s,x)\,\mathcal{N}(x|0,I)\,dx} \tag{2.26}$$

where $\mathcal{N}(.|0,I)$ is the standard Gaussian kernel. Proposition 2 in (Kenny et al., 2005a) explains how to obtain a closed form for this expression based on Baum-Welch statistics. The final expression used to estimate this likelihood is derived as follows.

Let us first introduce the terms that will be used to evaluate the likelihood in the case of joint factor analysis. N is a diagonal matrix of dimension $CF \times CF$, with diagonal blocks $N_c I \quad (c = 1, ..., C)$ and I is the identity matrix of dimension $F \times F$. The vector $\mathbf{F}$ is of dimension CF; it is formed by the concatenation of the F_c statistics. The diagonal matrix S is a matrix of size $CF \times CF$; its diagonal blocks are the S_c statistics. We define the expectation of the first-order and second-order Baum-Welch statistics as follows:

$$\begin{aligned} \mathrm{E}[F_s] &= \mathbf{F} - N\,\mathrm{E}[s] && (2.27) \\ \mathrm{E}[S_s] &= S - 2\,\mathrm{diag}\left(\mathbf{F}\,\mathrm{E}\left[s^t\right]\right) + \mathrm{diag}\left(N\left(\mathrm{E}[s]\,\mathrm{E}\left[s^t\right] + \mathrm{Cov}(s,s)\right)\right) && (2.28) \end{aligned}$$

Let us define the matrix $l = I + U^t\Sigma^{-1}NU$ and its Cholesky decomposition $l^{1/2}$. The log likelihood of test utterance χ given the target supervector s is given by the following equation (Kenny et al., 2007a,b):

$$\boxed{\begin{aligned} \log \mathrm{P}(\chi|s) = \sum_{c=1}^{C} N_c \log \frac{1}{(2\pi)\,|\Sigma_c|^{1/2}} - \frac{1}{2}\,\mathrm{tr}\left(\Sigma^{-1}\,\mathrm{E}[S_s]\right) \\ - \frac{1}{2}\log|l| + \frac{1}{2}\left\|l^{-1/2}U^t\Sigma^{-1}\,\mathrm{E}[F_s]\right\|^2 \end{aligned}} \tag{2.29}$$

where $\mathrm{E}[s]$ and $\mathrm{Cov}(s,s)$ are the expectation and covariance matrix of the posterior distribution of speaker supervector s. In joint factor analysis, score normalization plays an important part towards improving system performance (Kenny et al., 2007a,b, 2008a). In this dissertation all our JFA systems used zt-norm score normalization as described in Chapter 1.

CHAPTER 3

SUPPORT VECTOR MACHINES

This chapter presents the theory of support vector machines and gives the most popular kernels used for speaker verification.

3.1 Support Vector Machines

Support Vector Machines (SVMs) are supervised binary classifiers (Vapnick, 1995). They are based on the idea of finding, from a set of learning examples $X = \{(x_1, y_1), (x_2, y_2), ..., (x_M, y_M)\}$, the best linear separator H to distinguish between the positive examples $(y_i = +1)$ and negative examples $(y_i = -1)$. The linear separator is defined by the following function f:

$$\begin{aligned} f : \mathbb{R}^N &\rightarrow \mathbb{R} \\ x &\mapsto f(x) = w^t x + b \end{aligned} \tag{3.1}$$

where x is an input vector and (w, b) are the SVM parameters chosen during the training. The classification of a new example x is based on the sign of the function $f(x)$:

$$h(x) = \text{sign}\left(f\left(x\right) = w^t x + b\right) \tag{3.2}$$

In support vector machines, the hyperplane separator H has the characteristic of maximizing the minimum distance between the hyperplane and all example points of the training set. We use the term *margin* to refer to this distance. The classification margin ρ of an example x is determined by:

$$\rho_f\left(x, y\right) = y f\left(x\right) \tag{3.3}$$

The classifier margin is determined by the minimum value of $\rho_f\left(x, y\right)$ for all training points:

$$\rho_f = \min_{1 \leq i \leq N} \rho_f\left(x_i, y_i\right) \tag{3.4}$$

The hyperplane that maximizes equation (3.4) is the optimal separator. The training of the function f depends only on the example points which are located in the decision border. All these points are called *support vectors*. Figure 3.1 shows an example of optimal linear separation between two classes which maximizes the margin between the support vector points which are closer to the boundary. When we use the primal form of the SVM optimization problem, training is equivalent to solving the following problem:

$$\begin{cases} \min \frac{1}{2}\|w\|^2 \\ \text{under the constraints} \\ y_i\left(w^t x_i + w_0\right) \geq 1 \qquad i = 1, ..., M \end{cases} \tag{3.5}$$

where (x_i, y_i) are the learning examples and their respective label classes; M is the number of examples and $y_i \in \{+1, -1\}$.

In optimization theory, a problem that involves an objective function and strictly convex constraints can be reformulated in terms of a dual problem. The resolution of this dual problem is then equivalent to solving the primal one. The expression of a dual optimization problem for SVM is defined by:

$$\begin{cases} \max\left\{\sum_{i=1}^{M} \alpha_i - \frac{1}{2}\sum_{i,j=1}^{M} \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle\right\} \\ \text{under the constraints:} \\ \alpha_i \geq 0 \qquad\qquad i = 1, ..., M \\ \sum_{i=1}^{M} \alpha_i y_i = 0 \end{cases} \tag{3.6}$$

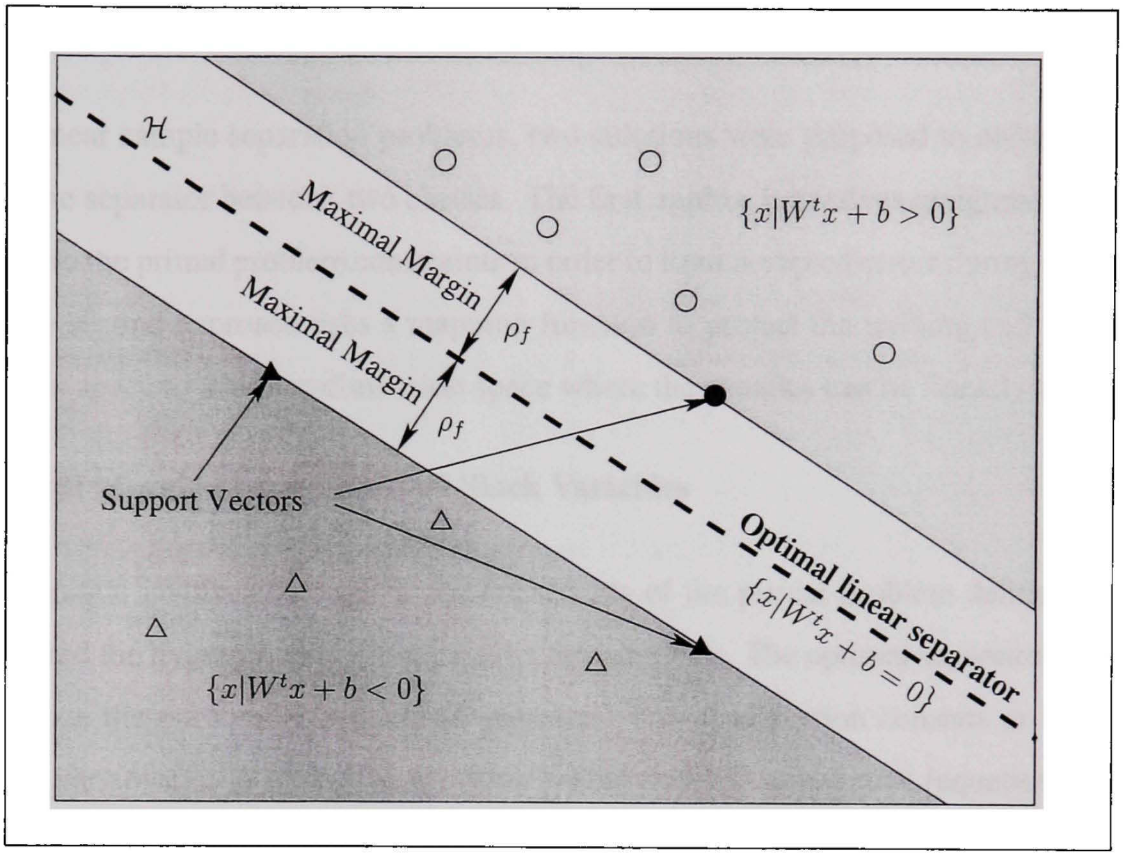

Figure 3.1 Optimal linear separation between two classes.

The optimal hyperplane separator for the SVM in the case of the dual representation is as follows:

$$f(x) = \sum_{i=1}^{m} \alpha_i^* y_i \langle x, x_i \rangle + w_0^* \tag{3.7}$$

where α_i^* and w_0^* are the SVM parameters set during the training step. The parameters α_i^* correspond to the Lagrangian multipliers which are used to solve the SVM dual problem. In the dual representation of support vector machines, two points are worthy of mention. Firstly, the estimation of the optimal hyperplane involves only the evaluation of inner products of vectors in the input space. The second point is that the dual problem of an SVM does not depend on the dimension of the examples, but only on the number of samples M.

3.2 Nonlinear Separation

For nonlinear sample separation problems, two solutions were proposed in order to find the hyperplane separator between two classes. The first approach involves assignments of slack variables to the primal problem constraints in order to limit accepted errors during the training step. The second approach uses a mapping function to project the training and test vectors from input space to a higher-dimension space where the samples can be linearly separable.

3.2.1 Soft Margin Hyperplane and Slack Variables

This technique involves changes to the constraints of the primal problem defined above in order to find the hyperplane that tolerates the fewest errors. The optimal separator is required to minimize the number of committed mistakes. The modification consists in introducing the slack variables $\zeta_i \geq 0$ into the previous primal problem constraints (equation 3.5). The modified criterion of the primal problem is given by:

$$\begin{cases} \min \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{M} \zeta_i \\ \text{under the constraints} \\ y_i\left(w^t x_i + w_0\right) \geq 1 - \zeta_i \quad i = 1, ..., M \\ \zeta_i \geq 0 \quad i = 1, ..., M \end{cases} \tag{3.8}$$

where $C > 0$ is a constant. The dual formulation of the new primal problem using the slack variables can be rewritten as follows:

$$\begin{cases} \max\left\{\sum_{i=1}^{M} \alpha_i - \frac{1}{2}\sum_{i,j=1}^{M} \alpha_i\alpha_j y_i y_j \langle x_i, x_j\rangle\right\} \\ \text{under the constraints:} \\ 0 \leq \alpha_i \leq C \quad i = 1, ..., M \\ \sum_{i=1}^{M} \alpha_i y_i = 0 \end{cases} \tag{3.9}$$

In this new optimization, the constant C controls the compromise between the optimal margin and the number of allowed errors.

3.2.2 Mapping and Kernel Functions

The rationale of projecting data from a low-dimension input space to a higher dimension *feature space*, is to transform a nonlinear separation problem in the initial space into a linearly separable problem in the feature space. The SVMs exploit a nonlinear transformation $\varphi : \mathbb{R}^N \rightarrow \mathbb{R}^D$ that converts all examples $X = \{(x_1, y_1), (x_2, y_2), ..., (x_M, y_M)\}$ of the input space $\mathbb{R}^N$ to a feature space $\mathbb{R}^D$ of higher, and potentially infinite dimension $D \gg N$ in which it is in principle possible to find a linear separator.

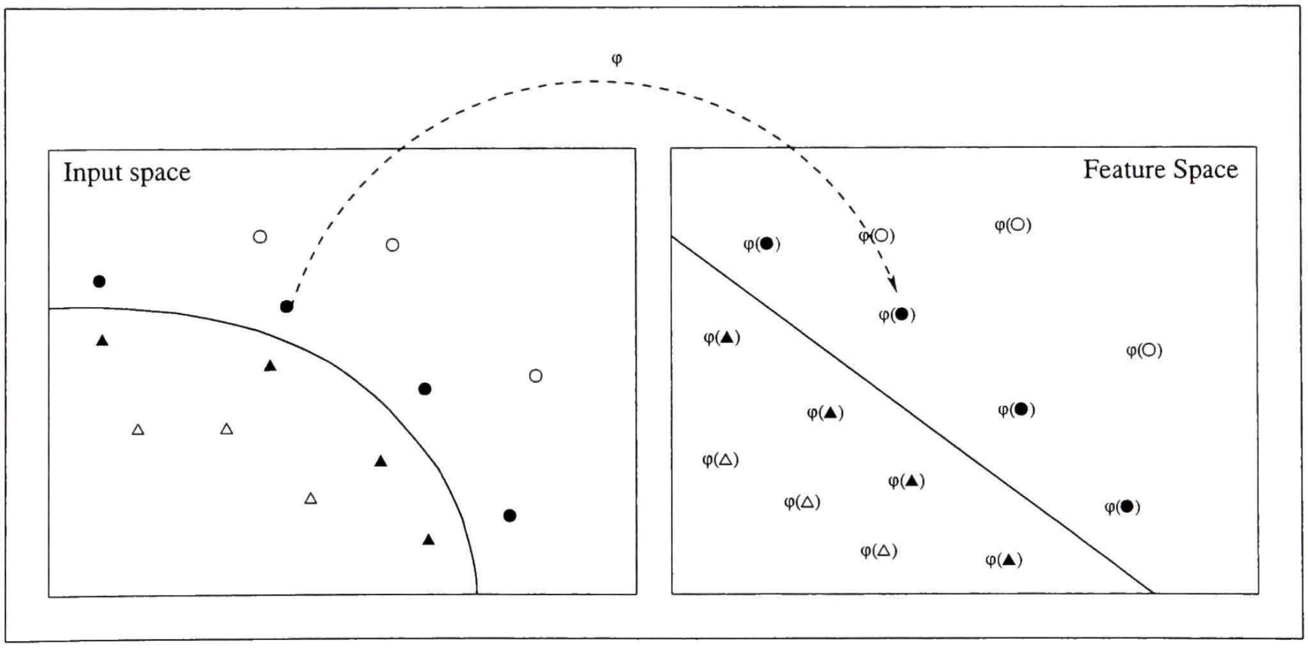

Figure 3.2 Mapping function projection.

It is easy to imagine the application of SVMs in high-dimension feature space. The dual form of the support vector machines optimization problem based on the mapping function φ which allowed us to go from the initial space to the feature space is given by:

$$\begin{cases} \max\left\{\sum_{i=1}^{M}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{M}\alpha_i\alpha_j y_i y_j \left\langle \varphi\left(x_i\right), \varphi\left(x_j\right)\right\rangle\right\} & \\ \text{under the constraints:} & \\ \alpha_i \geq 0 & i = 1, ..., M \\ \sum_{i=1}^{M}\alpha_i y_i = 0 & \end{cases} \tag{3.10}$$

where $\left\langle \varphi\left(x_i\right), \varphi\left(x_j\right)\right\rangle$ is the inner product in the feature space between two projected samples. The formula of the optimal separator in this new space has the following form:

$$h\left(x\right) = \sum_{i=1}^{m}\alpha_i^* y_i \left\langle \varphi\left(x\right), \varphi\left(x_i\right)\right\rangle + w_0^* \tag{3.11}$$

where α_i^* and w_0 are the optimal solutions of the SVM dual problem in the feature space.

From a practical point of view, the formulation of SVMs presented until now presents a very challenging problem, *viz.* the evaluation of the inner product $\left\langle \varphi\left(x_i\right), \varphi\left(x_j\right)\right\rangle$ in some feature space. We must note that this space may be of quite high dimensionality, indeed infinite in principle, which makes the evaluation of the inner product in this new space unfeasible. These difficulties can be circumvented through the use of kernel functions $k\left(x_i, x_j\right)$ in order to evaluate this inner product. The kernel functions are bilinear symmetric and positive functions which satisfy the Mercer conditions (Shawe-Taylor et Cristianini, 2004). They are easy to compute in the input space and it can be shown that, for a large-dimension feature space - a Hilbert space (Shawe-Taylor et Cristianini, 2004) in general - they correspond to an inner product $k\left(x_i, x_j\right) = \left\langle \varphi\left(x_i\right), \varphi\left(x_j\right)\right\rangle$. The new optimization problem of SVMs based on the kernel function is as follows:

$$\begin{cases} \max\left\{\sum_{i=1}^{M}\alpha_i-\frac{1}{2}\sum_{i,j=1}^{M}\alpha_i\alpha_j y_i y_j k\left(x_i,x_j\right)\right\} & \\ \text{under the constraints:} & \\ \alpha_i\geq 0 & i=1,...,M \\ \sum_{i=1}^{M}\alpha_i y_i=0 & \end{cases} \tag{3.12}$$

The new separator is given by the following formula:

$$\boxed{h\left(x\right)=\sum_{i=1}^{m}\alpha_i^* y_i k\left(x,x_i\right)+w_0^*} \tag{3.13}$$

The kernel function allows us to calculate the inner product in the high-dimension feature space via operations performed in the low-dimension input space $\mathbb{R}^N$. Furthermore, the kernel evaluation does not need to know the exact expression of the mapping function. These two kernel advantages greatly simplify the application of support vector machines for non-linearly separable problems. The choice of an adequate kernel function for a given problem is however critical. The user can test several classical kernel functions which are already guaranteed to correspond to inner products in a feature space, and choose the kernel function that yields the best performance. It is also possible for the user to construct an appropriate *ad hoc* kernel function, but he needs to prove that this function corresponds to an inner product in a given space. In order to prove that a given symmetric and positive function corresponds to a kernel function or inner product in the feature space, we test it against Mercer's theorem (Schölkopf et Smola, 2001) (Shawe-Taylor et Cristianini, 2004). The slack variables can also be applied in the kernel function SVM optimization problem (equation 3.12) in order to relax the constraints and tolerate some errors during hyperplane separator training. The final SVM formalism problem can be written as:

$$\boxed{\begin{cases} \max\left\{\sum_{i=1}^{M}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{M}\alpha_i\alpha_j y_i y_j k\left(x_i, x_j\right)\right\} & \\ \text{under the constraints:} & \\ 0 \leq \alpha_i \leq C & i = 1, ..., M \\ \sum_{i=1}^{M}\alpha_i y_i = 0 & \end{cases}} \tag{3.14}$$

The last optimization problem given in (Eq. 3.14) is the most popular criterion used to build pattern recognition systems based on support vector machines. Without exception, all the SVM experiments carried out in this thesis are based on this last optimization problem.

3.3 Speaker Verification Kernels

In this section, we present the most popular kernel functions applied to speaker verification systems based on support vector machine models.

3.3.1 Fisher Kernel

The Fisher mapping kernel is considered as the state-of-the-art approach to combine generative models and support vector machines. The SVM input vectors are derived from the generative model itself. In the case of speaker verification, the Gaussian mixture model plays the role of the generative model which is used to extract the SVM input vectors (Wan et Renals, 2003). The size of these vectors depends on the number of GMM parameters. Given a GMM speaker s parameterized by Θ and an utterance sequence X, the Fisher mapping kernel function based on the first derivative of the GMM parameters is obtained by:

$$\varphi_{\text{fisher}}\left(X\right) : X \mapsto \nabla_{\theta}\log P\left(X|s, \theta\right) \tag{3.15}$$

The Fisher mapping was first successfully introduced and applied to biological sequence processing by Jaakkola and Haussler (Jaakkola et Haussler, 1999). The Fisher kernel function between two utterances is computed as follows:

$$\boxed{k\left(X^a, X^b\right) = \varphi_{\text{fisher}}\left(X^a\right) R^{-1} \varphi_{\text{fisher}}\left(X^b\right)} \tag{3.16}$$

where, in the general case, R is the covariance matrix of the data in the Fisher mapping space $R = E\left[\varphi_{\text{fisher}}\left(X^a\right)\varphi_{\text{fisher}}\left(X^b\right)\right]$.

3.3.2 Generalized Linear Discriminant Sequence Kernel

The Generalized Linear Discriminant Sequence (GLDS) is a linear kernel, proposed by William Campbell (Campbell, 2002). This kernel is evaluated directly using the sequence of speech cepstral frames. Specifically, given a cepstral vector sequence $X = \{x_1, x_2, ..., x_l\}$, the mapping function of the GLDS kernel φ_{GLDS} is expressed as follows:

$$\varphi_{\text{GLDS}} : X \mapsto \frac{1}{l} \sum_{i=1}^{l} b\left(x_i\right) \tag{3.17}$$

where $b\left(x_i\right)$ is the polynomial expansion of each speech frame x_i (Campbell et Assaleh, 1999). The GLDS kernel function k_{GLDS} for two frame sequences X^a and X^b is defined by:

$$\boxed{k_{\text{GLDS}}\left(X^a, X^b\right) = \varphi_{\text{GLDS}}\left(X^a\right) R^{-1} \varphi_{\text{GLDS}}\left(X^b\right)} \tag{3.18}$$

where R is a normalization matrix obtained by $R = M^t M$ and M is given by:

$$M = \begin{bmatrix} \varphi_{\text{GLDS}}\left(X^{s_1}\right) \\ \varphi_{\text{GLDS}}\left(X^{s_2}\right) \\ \dots \\ \varphi_{\text{GLDS}}\left(X^{s_{N_S}}\right) \\ \varphi_{\text{GLDS}}\left(X^{z_1}\right) \\ \varphi_{\text{GLDS}}\left(X^{z_2}\right) \\ \dots \\ \varphi_{\text{GLDS}}\left(X^{z_{N_I}}\right) \end{bmatrix} \tag{3.19}$$

where $\varphi_{\text{GLDS}}\left(X^{s_i}\right)$ and $\varphi_{\text{GLDS}}\left(X^{z_i}\right)$ are the polynomial expansion of the speaker's and impostor's data sequences respectively. The terms N_S and N_I represent the number of speaker and impostor sequences. A noteworthy characteristic of this kernel is that an average of all projected vectors removes the context variability caused by the phonemic context. This operation results in a significant loss of information. Nevertheless, the results obtained using this kernel are promising (Campbell, 2002).

3.3.3 SVM-GMM Kernels

In this subsection we outline an approach that combines support vector machines and GMM-UBM systems. In this new combination, the SVMs are applied in GMM space. The GMMs for the relevant speakers are obtained using MAP adaptation of the component means of the UBM to the target data. The proposed kernel functions are based on an approximation of the Kullback-Leibler (KL) distance between two GMMs. Two kernel functions were tested. In (Campbell et al., 2006a,b), the authors propose a linear kernel that exploits the inner product of the KL distance between two GMMs s^a and s^b:

$$\boxed{k_{\text{lin}}\left(s^a, s^b\right) = \sum_{i=1}^{C} \left(\sqrt{w_i}\Sigma_i^{-1/2}\mu_i^a\right)^t \left(\sqrt{w_i}\Sigma_i^{-1/2}\mu_i^b\right)} \tag{3.20}$$

where w_i, Σ_i are respectively the weight and covariance matrix of the i^{th} Gaussian of the UBM. μ_i^a and μ_i^b are the mean components of the i^{th} Gaussian for each speaker and C is number of mixture components. During the same period of time that this work took place, we proposed a nonlinear kernel between two GMMs (Dehak et Chollet, 2006) based on the same KL distance as the previous kernel.

$$\boxed{k_{\text{nonlinear}}\left(s^a, s^b\right) = e^{-\sum_{i=1}^{C} w_i\left(\mu_i^a-\mu_i^b\right)^t\Sigma_i^{-1}\left(\mu_i^a-\mu_i^b\right)}} \tag{3.21}$$

We will present the details underlying this combination in the next chapter.

CHAPTER 4

SUPPORT VECTOR MACHINES AND GAUSSIAN MIXTURE MODELS

Methods based on support vector machines (SVM) have been widely applied in the field of pattern recognition (Schölkopf et Smola, 2001). In recent NIST speaker recognition evaluation campaigns, one of the most effective systems combined SVM with the classical GMM-UBM system to enhance overall performance (Campbell et al., 2006a).

We proposed a new nonlinear kernel defined in the GMM parameter space. This kernel was built from an approximated Kullback-Leiber divergence between two GMM models. These results were published in (Dehak et Chollet, 2006). Further work on the comparison between this nonlinear kernel and linear kernel (Campbell et al., 2006a) was also presented in (Dehak et al., 2007c). We also applied a channel compensation technique called nuisance attribute projection to both kernels. A comparison of results between the combination SVM-GMM approach and joint factor analysis was presented in (Dehak et al., 2008a).

4.1 Distance Between two GMMs

In this section, we define a weighted scaled Euclidean distance between two GMMs for speaker verification. The Kullback-Leibler Divergence between two probabilistic distribution models $\mathrm{P}^a(x)$ and $\mathrm{P}^b(x)$, corresponding respectively to speakers a and b, is formulated as follows:

$$\boxed{\mathrm{KL}\left(\mathrm{P}^a \,\|\, \mathrm{P}^b\right) = \int\limits_{\mathbb{R}} \mathrm{P}^a(x) \log\left(\frac{\mathrm{P}^a(x)}{\mathrm{P}^b(x)}\right) dx} \tag{4.1}$$

This KL divergence doesn't possess the symmetric property. In order to define a symmetric distance based on the divergence between two probabilistic models, we rewrite the KL distance in its symmetric form:

$$\mathrm{KL}\,2\left(\mathrm{P}^a \,\|\, \mathrm{P}^b\right) = \mathrm{KL}\left(\mathrm{P}^a \,\|\, \mathrm{P}^b\right) + \mathrm{KL}\left(\mathrm{P}^b \,\|\, \mathrm{P}^a\right) \tag{4.2}$$

In the case of Bayesian maximum *a posteriori* adaptation and according to the general property of KL divergence (Do, 2003), the Kullback-Leiber divergence between two GMMs with $\langle (\{w_i^a\}, \{\mu_i^a\}, \{\Sigma_i^a\}), (\{w_i^b\}, \{\mu_i^b\}, \{\Sigma_i^b\}), i = 1..C \rangle$ as respective parameters, is bounded by the following formula:

$$\mathrm{KL}\left(\mathrm{P}^a \,\|\, \mathrm{P}^b\right) \leq \mathrm{KL}\left(w^a \| w^b\right) + \sum_{i=1}^{C} w_i^a \, \mathrm{KL}\left(\mathcal{N}\left(.; \mu_i^a, \Sigma_i^a\right) \| \mathcal{N}\left(.; \mu_i^b, \Sigma_i^b\right)\right) \tag{4.3}$$

The first term $\mathrm{KL}\left(w^a \| w^b\right)$ is the KL divergence between the weights w^a and w^b. The second term $\mathrm{KL}\left(\mathcal{N}\left(.; \mu_i^a, \Sigma_i^a\right) \| \mathcal{N}\left(.; \mu_i^b, \Sigma_i^b\right)\right)$ is the divergence between the i^{th} Gaussian of model P^a and i^{th} Gaussian of model P^b. The formula is correct when the i^{th} Gaussians of both speaker GMMs correspond. This condition is implicit in the case of MAP adaptation since in that case each i^{th} Gaussian of each speaker GMM is adapted from the same Gaussian i of the UBM. In the case of GMM-UBM speaker-verification based models where only Gaussian means are adapted (i.e. $w_i^a = w_i^b$ and $\Sigma_i^a = \Sigma_i^b$, $i = 1, ..., C$), and using the symmetrical KL divergence, the last equation can be rewritten as follows:

$$\begin{aligned} \mathrm{KL}\,2\left(\mathrm{P}^a \,\|\, \mathrm{P}^b\right) &\leq \mathrm{KL}\left(\mathrm{P}^a \,\|\, \mathrm{P}^b\right) + \mathrm{KL}\left(\mathrm{P}^b \,\|\, \mathrm{P}^a\right) && (4.4) \\ &\leq \sum_{i=1}^{C} w_i \left(\mu_i^a - \mu_i^b\right)^t \Sigma_i^{-1} \left(\mu_i^a - \mu_i^b\right) && (4.5) \\ &\leq D_e^2\left(s^a, s^b\right) && (4.6) \end{aligned}$$

where

$$\boxed{D_e^2\left(s^a, s^b\right) = \sum_{i=1}^{C} w_i \left(\mu_i^a - \mu_i^b\right)^t \Sigma_i^{-1} \left(\mu_i^a - \mu_i^b\right)} \tag{4.7}$$

The right-hand term $D_e^2\left(s^a, s^b\right)$ of the last inequality gives a similarity measure between two GMM supervectors s^a and s^b for which only the Gaussian means are adapted. This term is homogeneous with the square of a Euclidean distance between two points in GMM supervector space. This distance is an upper bound of the KL distance. So if the distance between the two GMMs is small, the corresponding KL distance is also small and the opposite is also true.

4.2 Kernel Between two GMMs

4.2.1 Linear Kernel

The linear kernel between two GMMs was proposed by Campbell *et al.* (2006a). This kernel was derived from the Kullback-Leibler divergence between two GMMs (Campbell et al., 2006a,b). In the case of MAP adaptation with diagonal covariance matrices and when only the GMM mean components are adapted from the UBM, the weighted Euclidean distance between two scaled GMM supervectors s^a and s^b is given by equation (4.7). The KL linear kernel is defined as the corresponding inner product of the Euclidean distance between two GMMs:

$$k_{\text{lin}}\left(s^a, s^b\right) = \sum_{i=1}^{C} w_i \left(\mu_i^a\right)^t \Sigma_i^{-1} \mu_i^b \tag{4.8}$$

$$\boxed{k_{\text{lin}}\left(s^a, s^b\right) = \sum_{i=1}^{C} \left(\sqrt{w_i}\Sigma_i^{-1/2}\mu_i^a\right)^t \left(\sqrt{w_i}\Sigma_i^{-1/2}\mu_i^b\right)} \tag{4.9}$$

In kernel machines, the distance in feature space between two corresponding mapped vectors $\varphi\left(x_a\right)$ and $\varphi\left(x_b\right)$ can be computed using the kernel function:

$$\boxed{D\left(\varphi\left(x_a\right), \varphi\left(x_b\right)\right) = \sqrt{k\left(x_a, x_a\right) - 2k\left(x_a, x_b\right) + k\left(x_b, x_b\right)}} \tag{4.10}$$

If we apply the previous KL linear kernel between two GMMs k_{lin} in Equation 4.10, we obtain the same Euclidean distance defined in equation 4.7. The Kullback-Leibler linear kernel is a linear product in the GMM supervector space and its feature-mapping functional effect is just to scale the original Gaussian mean components by $\left(\sqrt{w_i}\Sigma^{-1/2}\right)$, where w_i,

Σ_i are respectively the weight and covariance matrix of the i^{th} Gaussian. This new kernel satisfies the Mercer condition (Shawe-Taylor et Cristianini, 2004).

4.2.2 Nonlinear Kernel

Another way to obtain a kernel based on distance D_e^2 defined in Equation 4.7 is to use the exponential function of the distance (Shawe-Taylor et Cristianini, 2004):

$$k_{\text{nonlinear}}\left(s^a, s^b\right) = e^{-D_e^2\left(s^a, s^b\right)} \tag{4.11}$$

$$\boxed{k_{\text{nonlinear}}\left(s^a, s^b\right) = e^{-\sum\limits_{i=1}^{C} w_i\left(\mu_i^a - \mu_i^b\right)^t \Sigma^{-1}\left(\mu_i^a - \mu_i^b\right)}} \tag{4.12}$$

We proposed this nonlinear kernel for speaker verification in (Dehak et Chollet, 2006). This kernel is equivalent to a Gaussian kernel applied in GMM space. This kind of kernel was already applied for speaker identification (Moreno et Ho, 2003) and multimedia classification (Moreno et al., 2003). In all of these applications, the authors used a KL divergence between two probabilistic distributions in order to define the kernel function. The difference between both kernels is that the nonlinear kernel is a normalized form of the exponential of the linear one. As mentioned above, the distance in feature space with respect to the linear kernel is just the scaled Euclidean distance between two GMMs. The distance in feature space between two vectors points $\varphi(x_a)$ and $\varphi(x_b)$ based on the nonlinear kernel is however different and can be derived using equation 4.10 such as:

$$\boxed{D\left(\varphi\left(x_a\right), \varphi\left(x_b\right)\right) = \sqrt{2 - 2e^{-D_e^2\left(x_a, x_b\right)}}} \tag{4.13}$$

The expanded feature space of the Gaussian kernel has infinite dimension (Shawe-Taylor et Cristianini, 2004) and its mapping function $\varphi(.)$ (Shawe-Taylor et Cristianini, 2004) can be computed as follows:

$$s \mapsto \varphi(s) = k(s, .) = e^{-\frac{\|s - .\|^2}{2\sigma^2}} \tag{4.14}$$

4.3 SVM-GMM Architecture

In the new system that we proposed, the MAP adaptation plays the role of features extraction as well as the MFCCs extraction. Given a target-speaker speech utterance and a set of impostor utterances, we first extract the MFCC frames; then we adapt the UBM Gaussian means to the target speaker and each impostor's recording frames in order to obtain the overall GMM supervectors. The linear or Gaussian Kullback-Leibler kernel can be used in order to find the hyperplane separator between target and impostor supervectors. When the test recording is available, we extract the MFCC vectors; then we use MAP adaptation to adapt the UBM means to the test frames in order to produce the test GMM supervector. The decision score is calculated by comparing this test supervector with the SVM separator established in the SVM training step. The final decision is obtained by comparing the decision score with a threshold. If the score is greater or equal to the threshold, the test utterance is assigned to the target speaker; otherwise, we assume the test utterance not to have originated from the latter. We can also apply score normalization in order to compensate for session variability between enrollment and test recordings. Figure 4.1 displays a diagram depicting the architecture of SVM-GMM systems.

4.4 Decision Score in the GMM Space

Decision scores in the classical GMM-UBM system are based on the log likelihood ratio, defined as follows:

$$\text{Score}_s\left(X\right) = \frac{1}{N}\sum_{i=1}^{N} \log\left(\frac{\mathrm{P}\left(x_i|s\right)}{\mathrm{P}\left(x_i|\Omega\right)}\right) \tag{4.15}$$

where s and Ω represent respectively the target speaker and the universal background models. The sequence $X = \{x_1, x_2, ..., x_N\}$ corresponds to the test utterance frames. In (Ben, 2004), Ben proved that this score can be rewritten using the KL distance between the GMM test model and the UBM and between the same test model and target speaker model such as:

$$\boxed{\text{Score}_s\left(X\right) = D_e^2\left(X_M, \Omega\right) - D_e^2\left(X_M, s\right)} \tag{4.16}$$

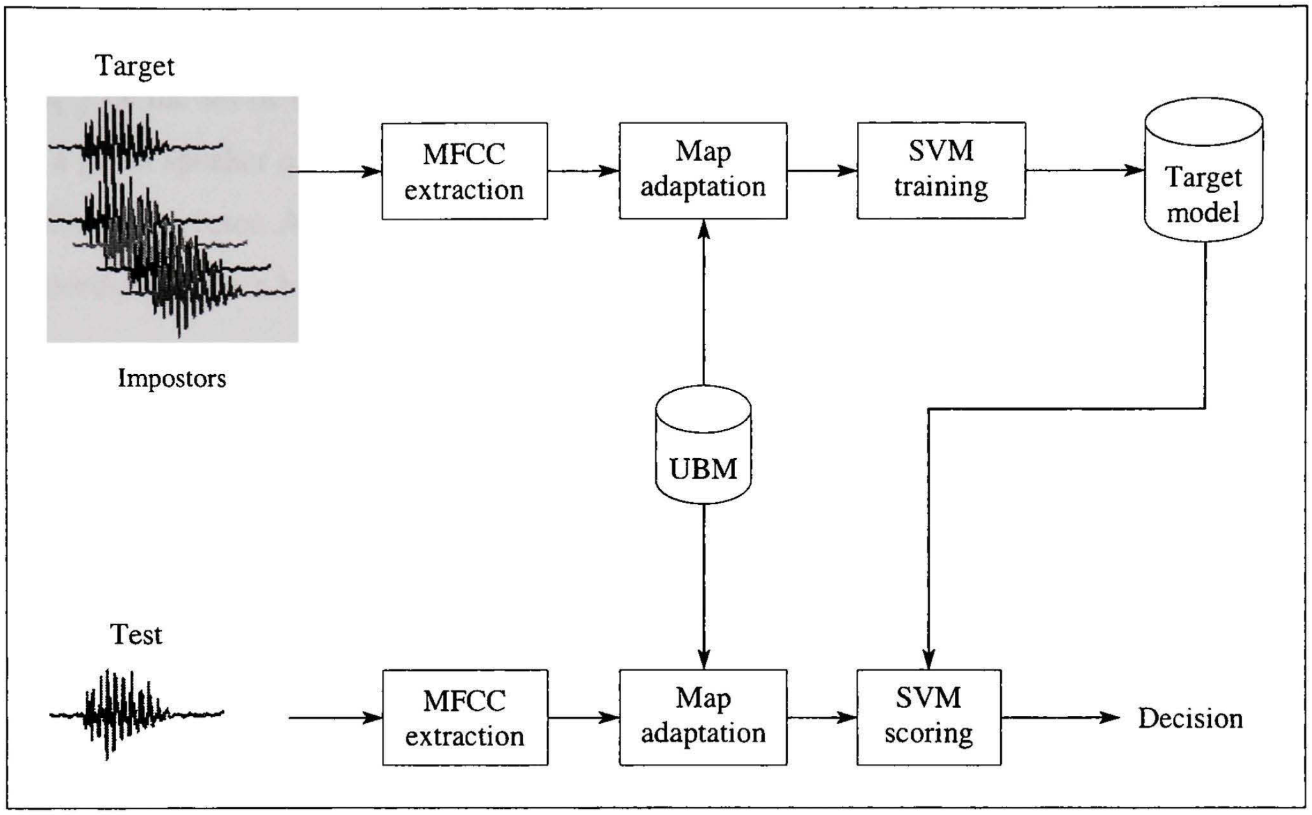

Figure 4.1 Architecture of the SVM-GMM system.

where X_M is the GMM supervector corresponding to the test model which is obtained by adapting the UBM mean components to the test utterance using MAP. This adaptation has the same form as to the target speaker model adaptation. The decision is taken by comparing this new score to a threshold.

4.5 Model Normalization: M-norm

It has already been proven that, in the context of kernel machines, scaling (Sarle, 1997) the values of data or normalizing the input vectors transform them into a spherical area (Wan et Renals, 2005), helps, and improve SVM performance. In speaker verification, we demonstrated the effectiveness of model normalization (M-norm) applied to GMMs, especially for nonlinear kernels (Ben, 2004; Dehak et Chollet, 2006). M-norm was first introduced by Ben (2004). The objective of this approach is to modify the GMM mean vectors or GMM supervectors so that the distance between all final normalized supervectors and the UBM supervector Ω is a constant distance D_{ref}, called reference distance, which is equal to one. Let $\{\mu_i^\Omega\}$ be the set of UBM mean components and let $\{\mu_i^a\}$ be the set of GMM mean vectors of a given speaker a. Let $D_e(s^a, \Omega)$ denote the distance between the speaker GMM and the UBM supervector. Applied to a particular GMM mean vector for speaker a, the normalization procedure is given by:

$$\boxed{\mu_k^a \leftarrow \frac{D_{\text{ref}}}{D_e(s^a, \Omega)}\mu_k^a + \left(1 - \frac{D_{\text{ref}}}{D_e(s^a, \Omega)}\right)\mu_k^\Omega} \tag{4.17}$$

Figure 4.2 shows the way in which all GMMs were moved in order that they live in a spherical area in GMM space, centered around the UBM mean supervector.

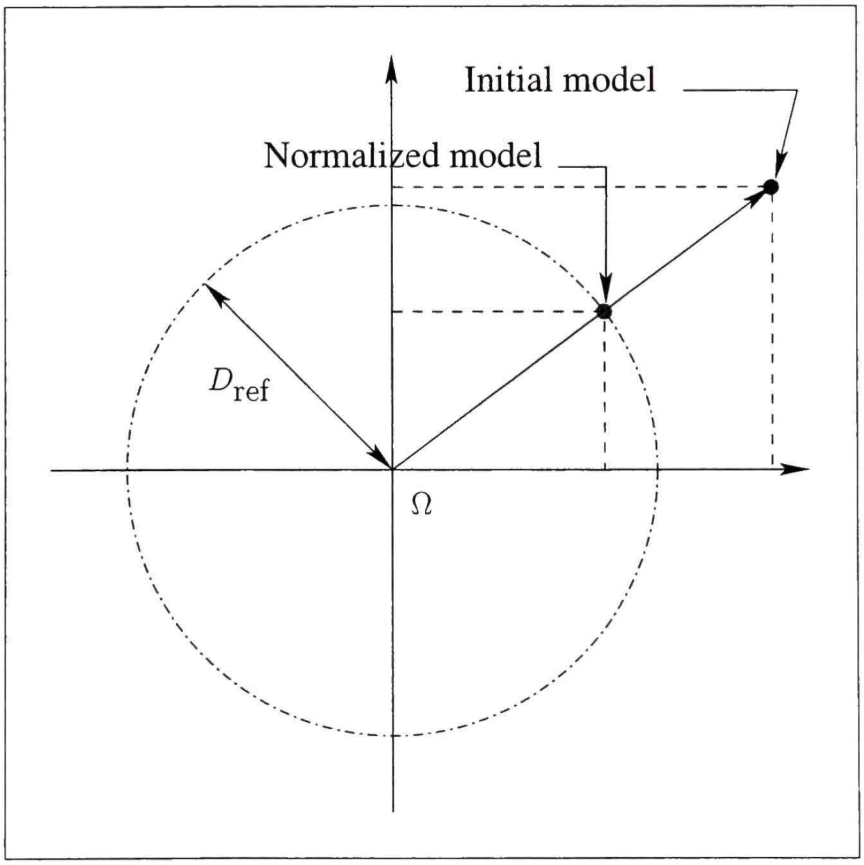

Figure 4.2 Model normalization in the GMM supervector space.

4.6 Nuisance Attribute Projection

As introduced in Chapter 1, the Nuisance Attribute Projection (NAP) approach was proposed to deal with the session variability problem in the framework of support vector machines applied to speaker verification. The NAP was first introduced in (Solomonoff et al., 2005, 2004) and successfully applied by Campbell *et al.* (2006a) to the Kullback-Leibler linear kernel given in Equation (4.9). This method uses an appropriate projection matrix $P = I - vv^t$ in linear kernel feature space in order to remove unwanted variability (such as channel effects). Given two GMM supervectors s^a and s^b, the new kernel can be expressed as follows:

$$\begin{aligned} k\left(s^a, s^b\right) &= \left\langle P\varphi\left(s^a\right), P\varphi\left(s^b\right)\right\rangle && (4.18) \\ &= \varphi\left(s^a\right) P\varphi\left(s^b\right) && (4.19) \end{aligned}$$

$$\boxed{k\left(s^a, s^b\right) = \varphi\left(s^a\right)\left(I - vv^t\right)\varphi\left(s^b\right)} \qquad (4.20)$$

where v is a rectangular matrix of low rank whose columns are orthonormal. These columns correspond to eigenvectors that represent the nuisance or channel effects that need to be removed from the feature space. The idea behind NAP is to project out the intersession variability in order to minimize the distortion between GMM supervectors belonging to the same speaker. The optimal projection matrix P and corresponding eigenvector matrix v can be determined by applying the following criterion:

$$\widetilde{P} = \arg\min_P \sum_{i,j} B_{i,j} \left\| P\left(\varphi\left(s^i\right) - \varphi\left(s^j\right)\right)\right\|^2 \qquad (4.21)$$

where $\varphi\left(s^i\right)$ is the kernel-mapping function. In order to train the projection matrix $\widetilde{P}$, we require a multi-speaker database comprised of many recordings over several sessions using different channel settings for each speaker. The $\{s^i\}$ are the speaker background dataset.

The matrix $B_{i,j}$ contains the information weights; its underlying structure can be chosen according to the information that we aim to remove:

- channel compensation:

$$B_{i,j} = \begin{cases} 1 & \text{if channel}\left(s^i\right) \neq \text{ channel}\left(s^j\right) \\ 0 & \text{otherwise} \end{cases} \tag{4.22}$$

- maximizing the variance between speakers:

$$B_{i,j} = \begin{cases} -1 & \text{if speaker}\left(s^i\right) \neq \text{ speaker}\left(s^j\right) \\ 0 & \text{otherwise} \end{cases} \tag{4.23}$$

- minimize the intersession variability:

$$B_{i,j} = \begin{cases} 1 & \text{if speaker}\left(s^i\right) = \text{ speaker}\left(s^j\right) \\ 0 & \text{otherwise} \end{cases} \tag{4.24}$$

The combination of these three matrices has already been tested in Solomonoff *et al.* (Solomonoff et al., 2004). Our work is restricted to the third case only.

In (Campbell et al., 2006a), the authors pointed out that when the KL linear kernel is applied to GMM supervector space and the nuisance variable for the session variability, the NAP subspace defined by matrix v is equivalent to the channel subspace modeled in joint factor analysis (Kenny et al., 2008b). The optimal matrix v is composed of the k eigenvectors having the k largest eigenvalues of the within-class covariance matrix which corresponds to the channel covariance:

$$\boxed{W = \frac{1}{S} \sum_{j=1}^{S} \frac{1}{n_j} \sum_{i=1}^{n_j} \left(s_i^j - \widetilde{s}_j\right) \left(s_i^j - \widetilde{s}_j\right)^t} \tag{4.25}$$

where s_i^j represents the GMM supervector corresponding to the i^{th} session of the j^{th} speaker. S is the number of speakers in our background database; n_j represents the number of recordings of speaker j and $\widetilde{s_j}$ is the mean of the set of GMM supervectors from speaker j:

$$\widetilde{s_j} = \frac{1}{n_j} \sum_{i=1}^{n_j} s_i^j \tag{4.26}$$

In order to estimate the within-class covariance matrix, we need many speakers, each one having recording over several sessions. This covariance matrix W is equivalent to the matrix UU^t, which is the channel covariance matrix of the channel supervector c in joint factor analysis (Kenny et al., 2008b). There is a difference between applying NAP in linear and in Gaussian kernels. In linear kernel, NAP is applied in feature space because feature space is equivalent to input space. However, in Gaussian kernel, we carry out the NAP in input space rather than in feature space.

4.7 Experiment with SVM-GMM

4.7.1 Experimental Set-up

In our experiments, we used cepstral features, extracted using a 25 ms Hamming window. Nineteen Mel Frequency Cepstral Coefficients together with log energy are calculated every 10 ms. This 20-dimensional feature vector was subjected to feature warping (Pelecanos et Sridharan, 2001) using a 3-seconds sliding window. Delta and double delta coefficients were then computed using a 5-frame window, giving a 60-dimensional feature vector. The resulting feature vectors were modeled using the GMM approach.

In all the experiments, two gender-dependent universal background models are used. Each UBM contains 2048 Gaussian components. They are trained using LDC releases of Switchboard II, Phases 2 and 3; Switchboard Cellular, Parts 1 and 2; and NIST 2004 speaker recognition evaluation data.

The SVM was trained using 1000 impostors for female and 1000 impostors for male, extracted from the same data used in the UBM training. The decision scores obtained with the SVM-GMM were normalized using t-norm based on 227 males and 283 females. The nuisance attribute projection matrix was trained with the same data used to train the UBM.

We carried out our experiments on the core condition of the NIST 2006 speaker recognition evaluation (SRE) [1]. This evaluation set contains 350 male and 461 female speakers; the number of tests is 51448 (Usually we use the term *trials* to refer to the these tests). For each target speaker model, a five-minute telephone conversation recording is available which contains roughly two minutes of speech from that specific speaker.

4.7.2 Results

Performance of the SVM-GMM System

In order to study the effects of combining support vector machines with Gaussian mixture models, we built four systems trained on the same data set.

- The first system is based on the classical GMM-UBM approach. Specifically, each target speaker had a specific GMM adapted from the UBM to the target speech frames using MAP adaptation. The decision scores were obtained using the log likelihood ratio as shown in Equation 4.15. We carried score normalization based on zt-normalization technique.
- The second system is also based on the GMM-UBM approach; however, the decision scores were obtained by using the distance between two GMMs. The final decision score was obtained using Equation 4.16.
- The third system is a SVM-GMM system based on a linear kernel.
- The final system is a SVM-GMM system based on a Gaussian kernel.

[1] http://www.nist.gov/speech/tests/spk/index.htm

All these systems were tested on the core condition of NIST 2006 SRE. The results of these experiments are given in Table 4.1. These results show that we achieve around 4% absolute improvement in EER when we compare the results obtained with the support vector machines based on the KL linear kernel and GMM-UBM log likelihood ratio scoring. This performance demonstrates the efficiency of applying SVM in GMM space. However, we found that the linear kernel gave the best results, around 2% in EER absolute improvement, compared to the Gaussian kernel. The worst results were obtained when the scoring using the distance between GMMs was applied.

Table 4.1

Comparison between classical GMM-UBM and SVM-GMM approaches. Results given for both genders, English trials of the core condition of NIST2006 SRE

	EER	MinDCF
GMM-UBM system	11.03%	0.03920
Distance between models	35.42%	0.09999
Linear kernel	**7.09%**	**0.03625**
Gaussian kernel	8.83%	0.04046

The Influence of Model Normalization

For this section, we conducted a series of experiments in order to measure the influence of model normalization on the performance of our systems based on GMM distance scoring, Gaussian and linear kernels. We tested the last three systems designed in the previous section, with and without applying M-norm. This normalization is applied to the distance between the UBM and all the speaker, test and impostor models.

System performance on English and all trials of the NIST206 SRE are presented respectively in Tables 4.2 and 4.3. These results reveal the effectiveness of model normalization for both kernels; however, the improvement is more marked in the Gaussian kernel and distance

Table 4.2

Influence of Model normalization. Results given for both genders, English trials of the core condition of NIST2006 SRE

	EER	MinDCF
GMM-UBM system	11.03%	0.03920
Distance between models	35.42%	0.09999
Linear kernel	7.09%	0.03625
Gaussian kernel	8.83%	0.04046
Distance between models **with MNorm**	11.64%	0.05424
Linear kernel **with MNorm**	**6.59%**	**0.03119**
Gaussian kernel **with MNorm**	7.34%	0.03539

Table 4.3

Influence of the Model normalization. Results given for both genders, all trials of the core condition of NIST2006 SRE

	EER	MinDCF
GMM-UBM system	11.64%	0.04555
Distance between models	36.97%	0.09999
Linear kernel	8.70%	0.04128
Gaussian kernel	10.52%	0.04779
Distance between models **with MNorm**	13.07%	0.06122
Linear kernel **with MNorm**	**8.14%**	**0.03745**
Gaussian kernel **with MNorm**	9.05%	0.04152

between GMMs scoring. We obtain respectively 1.5% and more than 20% absolute improvement in EER in both trial conditions using the Gaussian kernel and GMM distance scoring. The SVM-GMM system based on the KL linear kernel achieved the best results, 8.14% in EER and 0.03745 in MinDCF for all trials of the NIST 2006 SRE.

Importance of Nuisance Attribute Projection

This section presents the results when the nuisance attribute projection algorithm is applied. We replicated the same experiment as described in the previous section. We first tested the performance of the all systems when the NAP was applied without M-norm. In the last approach, model normalization was applied after removing the nuisance by projecting the initial GMM supervectors using the NAP matrix. We began by choosing the optimal number of channel eigenvectors for the matrix V which compose the nuisance attribute projection matrix P. We referred to this number as the NAP corank. We carried out several experiments by varying the NAP corank using both kernels until we found the optimal corank. Figure 4.3 depicts the influence of the NAP corank on EER in all trials of the core condition of the NIST 2006 SRE.

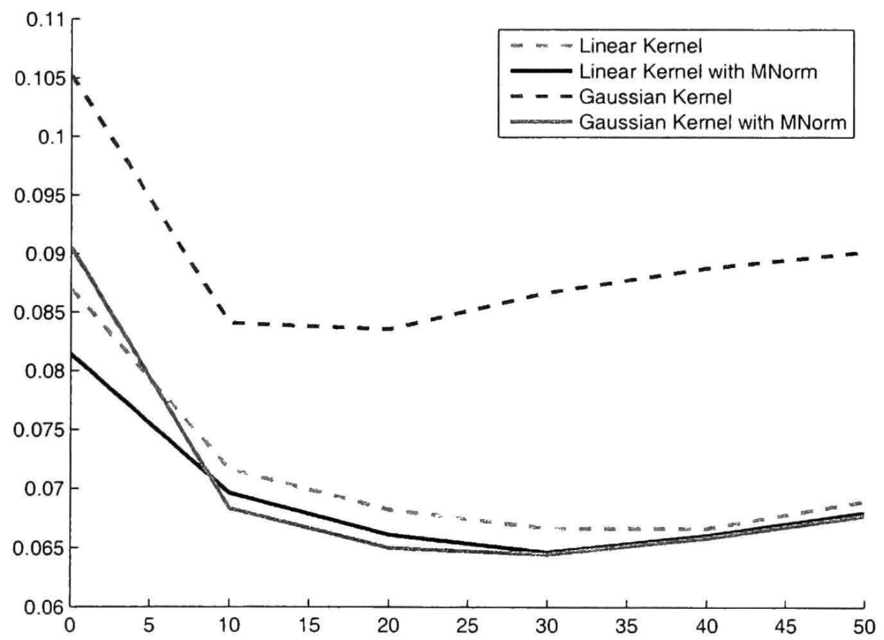

Figure 4.3 Influence of the NAP corank on the SVM-GMM supervector systems tested on NIST 2006 SRE (all trials).

Table 4.4

NAP influence on SVM performance. Results given for both genders, English trials of the core condition of NIST2006 SRE

	EER	MinDCF
Distance between models + NAP	45.29%	0.09981
Linear kernel + NAP	4.99%	0.02581
Gaussian kernel + NAP	6.54%	0.03064
Distance between models **with MNorm** + NAP	45.61%	0.09986
Linear kernel **with MNorm** + NAP	4.72%	0.02508
Gaussian kernel **with MNorm** + NAP	**4.70%**	**0.02426**

Table 4.5

NAP influence on SVM performance. Results given for both genders, all trials of the core condition of NIST2006 SRE

	EER	MinDCF
Distance between models + NAP	46.84%	0.09999
Linear kernel + NAP	6.66%	0.03441
Gaussian kernel + NAP	8.87%	0.04118
Distance between models **with MNorm** + NAP	46.90%	0.09999
Linear kernel **with MNorm** + NAP	6.60%	0.03437
Gaussian kernel **with MNorm** + NAP	**6.58%**	**0.03384**

The results given in Tables 4.4 and 4.5 were obtained with NAP corank= 30. We can conclude from these tables that application of the NAP in the SVM system lead to improve results as compared to the last ones obtained in the previous section. We went from an EER of 7.09% (without M-norm and NAP) given in table 4.2 to 4.99% (with NAP only) for English trials of the 2006 SRE. The same conclusion can be drawn for the all-trial task. The best results were obtained when M-norm was applied after NAP. In this experiment, the Gaussian and linear kernels produce equivalent results. However, the linear kernel is less computationally complex than the Gaussian kernel.

4.8 Comparison Between SVM-GMM and Joint Factor Analysis

In this section, we compare the best results obtained on the core condition of the NIST 2006 SRE using both linear and nonlinear kernels (as described in section 4.7.2) with those obtained via three joint factor analysis systems using different configurations (with and without speaker factors).

4.8.1 Experiments

Joint Factor Analysis Training

We used gender-dependent UBMs with 2048 Gaussians. The UBMs were trained using the LDC releases of Switchboard II, Phases 2 and 3; Switchboard Cellular, Parts 1 and 2; and NIST 2004 speaker recognition evaluation data. The gender-dependent factor analysis models were trained on the same data as the UBM. The diagonal matrix D and eigenvoice matrix V were estimed in a decoupled manner. The eigenvoice matrix V was trained on the same data used for UBM training, minus the NIST 2004 SRE dataset; the D matrix was trained on the 2004 SRE dataset. We used the same t-norm impostors as for SVM-GMM systems. The impostors used to train SVMs were also used to carry out the z-norm JFA score normalization. The JFA was trained on exactly the same data as the SVM-GMM.

Results

The results, summarized in Tables 4.6 and 4.7, reveal the following points. When speaker factors are used, the joint factor analysis configurations yield substantially better results than the SVM-GMM system for the NIST evaluation dataset. However, both SVM-GMM kernels lead to better EER (but not MinDCF) in both conditions of the NIST evaluation as compared to the factor analysis without speaker factors. The JFA that incorporates no speaker factors coincides closely with the SVM-GMM system: In the absence of speaker factors, the target speaker enrollment procedure through factor analysis modeling is similar to traditional MAP adaptation, which is the first step in enrolling a target speaker in a GMM-SVM system. The nuisance attribute projection applied to the SVM can be seen as a dual representation of the eigenchannel for JFA. Clearly, the best performing system incorporates 300 speaker factors, in comparison to the speaker factors absent JFA and all other SVM-GMM systems. We can conclude from these results that speaker factors play an important role in the target speaker enrollment. Additional results highlighting the effectiveness of speaker factor inclusion can be found in (Kenny et al., 2008a).

Table 4.6

Results on English trials of the core condition of the NIST 2006 SRE

	EER	MinDCF
Linear kernel with MNorm + NAP	4.72%	0.0250
Gaussian kernel with MNorm + NAP	4.70%	0.0242
0 speaker factors, 100 channel factors, $D \neq 0$	4.98%	0.0199
300 speaker factors, 100 channel factors, $D = 0$	2.04%	0.0132
300 speaker factors, 100 channel factors, $D \neq 0$	**1.46%**	**0.0092**

Table 4.7

Results on all trials of the core condition of the NIST 2006 SRE

	EER	MinDCF
Linear kernel with MNorm + NAP	6.60%	0.0250
Gaussian kernel with MNorm + NAP	6.58%	0.0242
0 speaker factors, 100 channel factors, $D \neq 0$	7.63%	0.0320
300 speaker factors, 100 channel factors, $D = 0$	3.73%	0.0208
300 speaker factors, 100 channel factors, $D \neq 0$	**2.98%**	**0.0170**

4.9 Discussion

In this chapter, we introduced a new nonlinear kernel application for support vector machines in Gaussian mixture model supervector space. The proposed kernel is based on the

Kullback-Leibler approximation distance between two GMMs. The results obtained with this combination outperform the GMM-UBM results. The new kernel's performance was compared to that offered by the linear kernel of Campbell *et al.* (2006a) which, by design, also implements the same GMM distances. We demonstrated the effectiveness of combining model normalization with nuisance attribute projection in both kernels. Comparing results of this new modeling with those obtained by joint factor analysis reveals the effectiveness of speaker factors in speaker model enrollment. These results motivated us to combine support vector machines with joint factor analysis in order to create a speaker verification system.

CHAPTER 5

SUPPORT VECTOR MACHINES AND JOINT FACTOR ANALYSIS

In this chapter, we will present several ways to carry out the combination between Support Vector Machines and Joint Factor Analysis. The first approach is similar to the classical SVM-GMM presented in the preceding chapter (Campbell et al., 2006a), which consists in using the speaker GMM supervectors as input to an SVM. The second set of methods that we tested is based on new kernels that are functions of configuration-dependent JFA factors. We also proposed a new factor analysis scoring based on the cosine kernel as decision score.

5.1 SVM-JFA in GMM Supervector Space

The first approach that we propose for combining SVM with JFA uses the GMM speaker supervector produced by JFA modeling (Kenny et al., 2008b) as input for an SVM based on the classical linear Kullback-Leibler kernel defined in the preceding chapter. This kernel applied in GMM supervector space is based on the Kullback-Leibler divergence between two GMMs (Campbell et al., 2006a). This distance corresponds to the Euclidean distance between scaled GMM supervectors s^a and s^b:

$$D_e^2\left(s^a, s^b\right) = \sum_{i=1}^{C} w_i \left(\mu_i^a - \mu_i^b\right)^t \Sigma_i^{-1} \left(\mu_i^a - \mu_i^b\right) \tag{5.1}$$

where w_i and Σ_i are the weight and diagonal covariance matrix of the i^{th} UBM mixture component respectively, μ_i^a corresponds to the mean of Gaussian i of GMM speaker a. C is the number of Gaussian mixture components. The derived linear kernel is defined as the corresponding inner product of the preceding distance equation 5.1:

$$k_{\text{lin}}\left(s^a, s^b\right) = \sum_{i=1}^{C} \left(\sqrt{w_i}\Sigma_i^{-1/2}\mu_i^a\right)^t \left(\sqrt{w_i}\Sigma_i^{-1/2}\mu_i^b\right) \tag{5.2}$$

The preceding chapter describes this kernel in greater detail. Figure 5.1 describes the architecture of the SVM-JFA system when the GMM supervectors are used.

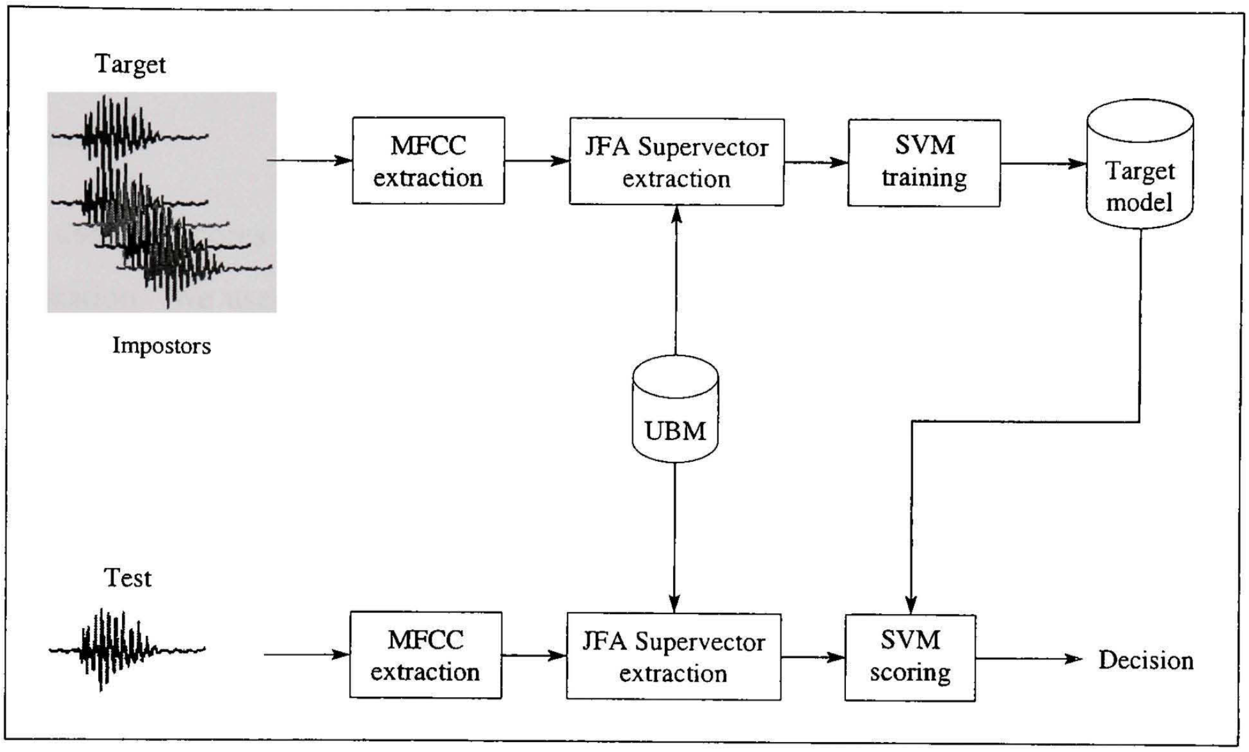

Figure 5.1 Architecture of the SVM-JFA system when the supervectors are used.

5.1.1 Experiments

Experimental Set-up

Our experiments operate on cepstral features, extracted using a 25 ms Hamming window. Nineteen Mel Frequency Cepstral Coefficients together with log energy were calculated every 10 ms. This 20-dimensional feature vector was subjected to feature warping (Pelecanos et Sridharan, 2001) using a 3 s sliding window. Delta and double delta coefficients were then calculated using a 5-frame window to produce 60-dimensional feature vectors. These feature vectors were modeled using GMM and factor analysis was used to address the problem of speaker and session variability.

We used two gender-dependent universal background models containing 2048 Gaussians. These UBMs were trained using LDC releases of Switchboard II, Phases 2 and 3; Switchboard Cellular, Parts 1 and 2; and NIST 2004-2005 speaker recognition evaluation data. The gender-dependent joint factor analysis models were trained on the same data as the UBM training.

The decision scores obtained with joint factor analysis were normalized using zt-norm normalization. We used 300 t-norm models for female trials. We used around 1000 z-norm utterances for females. All these impostors were taken from the same dataset as the UBM training list.

In our SVM system, we used 307 t-norm female models taken from the NIST 2005 SRE and 1292 female SVM impostor models trained on Switchboard II, Phases 2 and 3; Switchboard Cellular, Parts 1 and 2; and NIST 2004 SRE data.

We used two joint factor analysis configurations. The first JFA is made up of 300 speaker factors and 100 channel factors only. The second configuration is full: we added the diagonal matrix D in order to have speaker and common factors. When the diagonal matrix was estimated, we used decoupled estimation of the eigenvoice matrix V and diagonal matrix D (Kenny et al., 2008b). The eigenvoice matrix V was trained on all the UBM training data, except the NIST 2004 SRE data. The D matrix was trained on the 2004 SRE data. The experiments was carried out in the core condition of the NIST 2006 SRE dataset. This evaluation set contains 350 male and 461 female speakers; the number of test utterances is 51448. For each target speaker model, a five-minute telephone conversation recording is available which contains roughly two minutes of speech from that specific speaker.

Results

The results of our experiments are reported only on female trials of the NIST 2006 SRE. In the SVM-JFA system, we used the speaker GMM supervectors obtained using both JFA configurations (with or without common factors) as input for the SVM approach. The results are given in Table 5.1. These results are compared to JFA scoring based on integration over all channel factors (equation 2.29).

Table 5.1

Scoring result comparison between SVM-JFA in speaker supervector space and JFA scoring. The results are given on EER and MinDCF of the female part of the core condition of NIST 2006 SRE

System	English		All trials	
	EER	MinDCF	EER	MinDCF
JFA: $s = m + Vy$	1.74%	0.0121	3.84%	0.0223
JFA: $s = m + Vy + Dz$	**1.64%**	**0.0120**	**3.15%**	**0.0189**
SVM-JFA: $s = m + Vy$	5.14%	0.0420	6.28%	0.0466
SVM-JFA: $s = m + Vy + Dz$	5.58%	0.0427	6.36%	0.0472

The results show that application of the SVM in GMM supervector space yields significantly worse performance than that obtained by JFA scoring computed via integration over all channel factors. These results can be explained by recognizing that the linear KL kernel is not appropriate for GMM supervectors obtained from the JFA approach. This is because the assumption of GMM Gaussian independence in the case of MAP adaptation (see Chapter 1), does not apply for eigenvoice-based adaptation. Specifically, correlations between the Gaussians are unavoidable in the case of eigenvoice adaptation. The results also reveal that the addition of common factors does not improve performance in the case of SVM -JFA as compared to JFA scoring.

5.2 SVM-JFA in Speaker Factor Space

In this section, we discuss the use of speaker factors as feature vector input to SVM. The speaker factor coefficients correspond to speaker coordinates in the speaker space defined by the eigenvoice matrix. The advantage of using speaker factors stems from the low dimension of these vectors (typically 300), thus making the decision process faster. We tested these vectors with three classical kernel types: linear, Gaussian and cosine. The corresponding kernels between two speaker factor vectors y_1 and y_2 are given respectively by the following equations:

$$\boxed{k\left(y_1, y_2\right) = \left\langle y_1, y_2 \right\rangle} \tag{5.3}$$

$$\boxed{k\left(y_1, y_2\right) = \exp\left(-\frac{1}{\sigma^2}\left\|y_1 - y_2\right\|^2\right)} \tag{5.4}$$

$$\boxed{k\left(y_1, y_2\right) = \frac{\left\langle y_1, y_2 \right\rangle}{\left\|y_1\right\| \left\|y_2\right\|}} \tag{5.5}$$

Note that the cosine kernel consists in normalizing the linear kernel by the norm of both speaker factor vectors. The motivation behind the use of the linear kernel is that the speaker factor vectors are normally distributed with zero mean and identity covariance matrix. In order to establish the speaker factors for this new modeling, we used the JFA configuration associated with speaker and channel factors only. There are no common factors z (see Equation 2.19).

5.2.1 Kernel Normalization

In this novel approach, we proposed applying kernel normalization in speaker space based on the Within Class Covariance Normalization algorithm (WCCN) (Hatch et al., 2006). This algorithm can be interpreted as another intersession compensation step in speaker space. The first step is carried out by estimating the channel factors in GMM supervector space.

Within Class Covariance Normalization

This approach is applied in SVM modeling based on linear separation between target speaker and impostors using a one-versus-all decision. Linear separation in the context of SVMs is equivalent to using a linear kernel as defined in the previous section. The idea behind WCCN is to minimize the expectation of the error rate of false alarms and false rejections during the SVM training step. In order to minimize the error rate, the author in (Hatch et al., 2006) defines a set of upper bounds in the classification error metric.

The optimized solution to this problem is found by minimizing these upper bounds which, by the same token, minimizes the classification error. This optimization procedure allows us to alter the hard-margin separation formalism of the SVM. The resulting solution is given by a generalized linear kernel of the form:

$$k\left(y_1, y_2\right) = y_1^t R y_2 \tag{5.6}$$

where R is a symmetric, positive semidefinite matrix. The optimal normalized kernel matrix is given by $R = W^{-1}$, where W is the within class covariance matrix computed using all impostor utterances in our background. We assume that all utterances of a given speaker represents a class.

$$\boxed{W = \frac{1}{S}\sum_{s=1}^{S}\frac{1}{n_s}\sum_{i=1}^{n_s}\left(y_i^s - \overline{y_s}\right)\left(y_i^s - \overline{y_s}\right)^t} \tag{5.7}$$

where $\overline{y_s} = \frac{1}{n_s}\sum_{i=1}^{n_s} y_i^s$ is the mean of speaker factor vectors of speaker s, S is the number of speakers, and n_s is the number of utterances of speaker s. In order to keep the inner product context of the linear and cosine kernels, a feature mapping function φ can be defined as follows:

$$\varphi\left(y\right) = A^t y \tag{5.8}$$

where A is obtained using Cholesky decomposition of the matrix $W^{-1} = AA^t$. In our approach, the WCCN algorithm is applied to the linear and cosine kernels. The new versions of these kernels are given by the following equations:

$$\boxed{k\left(y_1, y_2\right) = \left(A^t y_1\right)^t \left(A^t y_2\right)} \tag{5.9}$$

$$\boxed{k\left(y_1, y_2\right) = \frac{\left(A^t y_1\right)^t \left(A^t y_2\right)}{\sqrt{\left(A^t y_1\right)^t \left(A^t y_1\right)} \sqrt{\left(A^t y_2\right)^t \left(A^t y_2\right)}}} \tag{5.10}$$

Our application of the WCCN algorithm in speaker space is motivated by the premise that speaker factors are, at the outset, low-dimension vectors. Consequently, removing additional directions could be detrimental. The WCCN algorithm uses the within class covariance matrix to normalize the linear kernel functions in order to compensate for intersession variability, while guaranteeing conservation of directions in space, in contrast with other approaches such as Nuisance Attribute Projection and Linear Discriminant Analysis.

5.2.2 Experiments

Experimental Set-up

We used exactly the same experimental set-up as in the previous experiments based on gender-dependent UBM and JFA trained on the same data. However, in this section, we implement a restricted JFA configuration based on 300 speaker factors and 100 channel factors. We do not make use of common factors ($D = 0$). The within class covariance matrix was trained on NIST 2004 and 2005 SRE datasets. The experiments were carried out on the core condition female trials of the NIST 2006 and 2008 SRE telephone data. The NIST 2008 SRE dataset contains 648 male and 1140 female speakers; the number of trials or tests is 37050. For each speaker, we have around two minutes of speech to train and test the model.

Results

In this section, we present the results obtained with the linear, Gaussian and cosine kernels applied to the speaker factor space. These new results are compared to the previous ones obtained by application of SVM-JFA in GMM supervector space and JFA scoring via integration over channel factors as proposed in (Kenny et al., 2008b). The results for the female data of the the NIST 2006 SRE core condition are shown in Table 5.2 and 5.3.

Table 5.2

Scoring result comparison between SVM-JFA in speaker factor space and GMM supervector space. The results are given on EER for the female English trials of the NIST 2006 SRE core condition

	English		
	No-norm	T-norm	ZT-norm
JFA: $s = m + Vy$	4.04%	-	**1.74%**
KL-kernel: supervectors	5.51%	5.14%	-
Linear kernel	2.89%	2.55%	-
Gaussian kernel	2.75%	2.56%	-
Cosine kernel	2.56%	2.38%	-

Table 5.3

Scoring result comparison between SVM-JFA in speaker factor space and GMM supervector space. The results are given on EER for all trials of the NIST 2006 SRE core condition

	All trials		
	No-norm	T-norm	ZT-norm
JFA: $s = m + Vy$	7.17%	-	**3.84%**
KL-kernel: supervectors	6.58%	6.28%	-
Linear kernel	4.57%	4.37%	-
Gaussian kernel	4.89%	4.91%	-
Cosine kernel	4.40%	4.44%	-

Three remarks are in order for Tables 5.2 and 5.3. To begin with, application of the SVM in speaker factor space gives better results than its application in GMM supervector space. Secondly, there is a marked linear separation between the speakers in the speaker space as seen when comparing the results for the cosine and Gaussian kernels. And finally, score normalization does not lead to a large improvement in the case of cosine and Gaussian kernels as compared to joint factor analysis scoring.

Within Class Covariance Normalization

We will now discuss the performance achieved using the WCCN technique in the case of linear and cosine kernels applied to speaker factor space. Tables 5.4 and 5.5 compare the results obtained with WCCN with the results of JFA scoring based on integration over channel factors.

Table 5.4

Scoring result comparison between JFA and SVM-JFA in speaker factor space when WCCN is applied. The JFA and SVM-JFA scores are respectively zt-norm and t-norm normalized. The results are given on EER and MinDCF for the female part of the NIST 2006 SRE core condition, for English and all trials

	English trials		All trials	
	EER	MinDCF	EER	MinDCF
Linear kernel	2.55%	0.0148	4.37%	0.0223
Linear kernel + WCCN	1.94%	0.0132	3.06%	0.0175
Cosine kernel	2.38%	0.0134	4.44%	0.0230
Cosine kernel + WCCN	**1.67%**	**0.0123**	**3.02%**	**0.0174**
JFA: $s = m + Vy$	1.74%	0.0121	3.84%	0.0223

Table 5.5

Scoring result comparison between JFA and SVM-JFA in speaker factor space when WCCN is applied. The JFA and SVM-JFA scores are respectively zt-norm and t-norm normalized. The results are given on EER and MinDCF for the female part of the NIST 2008 SRE core condition, for English and all trials

	English trials		All trials	
	EER	MinDCF	EER	MinDCF
Linear kernel + WCCN	4.26%	0.0206	7.24%	0.0361
Cosine kernel + WCCN	4.20%	0.0176	7.27%	0.0367
JFA: $s = m + Vy$	**3.68%**	**0.0159**	**6.3%**	**0.0327**

The results shown in Table 5.4 show that with WCCN, we achieve around 30% relative improvements for both kernels for the English trials of the NIST 2006 SRE. The results obtained with WCCN applied to the cosine kernel are superior to the JFA scoring results, especially in the all trials case of the NIST 2006 SRE. If we compare the performance of the same systems on the NIST 2008 SRE, we find that JFA outperforms our systems. An explaination of this behaviour is that the NIST 2006 SRE dataset resembles more closely the NIST 2004 and 2005 SRE datasets, which are used for JFA training; there are however some differences between the NIST 2008 setting and other NIST-SRE datasets. These results indicate that, insofar as the speaker space is well defined for the test data, we achieve better results than JFA scoring.

5.3 SVM-JFA in Speaker and Common Factor Space

When speaker and common factors were available, we proposed and compared two techniques that combine these two sources of information. The first approach applies SVM in each space (speaker factor space and common factor space). Thereafter we linearly combine these two SVM scores. The fusion weights are obtained from a logistic regression (Brummer et al., 2007). The second approach is to define a new kernel which is a linear combination of two initial kernels: the first kernel is applied in the speaker factor space, while the second kernel is applied in the common factor space. The kernel combination weights are chosen to maximize the margin between target speaker and impostor models. This technique has already been applied in speaker verification (Dehak et al., 2008b). The main difference between these two fusion approaches is that, in score fusion, we require extra development data to estimate the score fusion weights; however, we don't need any development data when combining the two kernels.

5.3.1 Score Fusion

The main idea behind score fusion is to fuse the multiple scores produced by different subsystems into a single score for the decision. Score fusion techniques have proved to be beneficial to speaker verification performance (Brummer et al., 2007), especially when the subsystems were applied on several speaker information sources (Brummer et al., 2007). Many approaches have been used for combining the scores of several systems. In (Campbell et al., 2007), Campbell *et al.* applied a neural network model to combine system scores. The model was trained to minimize the DCF. Equal fusion weights are already used by Kajarekar (2005) for the fusion of four different SVM systems. The most commonly used fusion approach during the NIST 2008 speaker recognition evaluation campaign was based on linear score fusion carried out through logistic regression training. The resulting score weights were optimal with respect to DCF minimization. The linear score fusion is computed using the following equation:

$$\boxed{s_f = w_0 + \sum_{l=1}^{M} w_l s_l} \tag{5.11}$$

where s_f is the final fused score, s_l is the score for the l^{th} subsystem, M is the number of fused subsystems and $w = (w_0, w_1, ..., w_M)$ is a vector comprised of the individual score fusion weights. We will describe in greater detail score fusion based on logistic regression in Chapter 7. In our modeling, we built two separate SVM systems for speaker and common factors. The first system employs a cosine kernel in terms of speaker factor vectors. Based on the results obtained in section 5.2, we applied within class covariance normalization technique in order to normalize the resulting kernel. The second SVM system is also based on a cosine kernel, applied to common factor vectors. Figure 5.2 illustrates the architecture of our score fusion system.

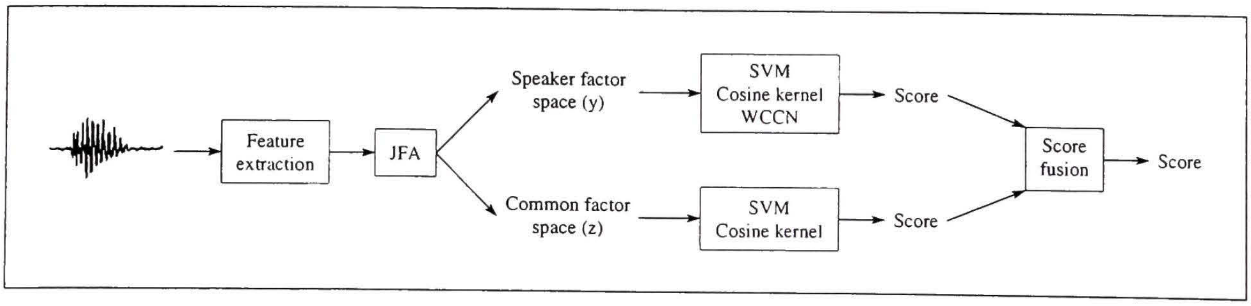

Figure 5.2 Architecture of the score fusion system.

5.3.2 Kernel Combination

SVM system performance is very sensitive to the choice of kernel function; its design thus represents a critical step. Several kernel functions were already applied in speaker verification (see Chapter 3). As in the score fusion approach, it will prove favorable to combine all kernel functions in order to produce a new kernel that encompasses the information conveyed by each kernel. The most straightforward solution is to carry out a simple linear combination of all kernels. It has already been proved that the sum of many kernels produces a kernel that satisfies Mercer's condition (Shawe-Taylor et Cristianini, 2004). The new kernel k_f is based on the combination of M kernel functions using a weighted linear combination.

$$\boxed{k_f\left(x_i, x_j\right) = \sum_{l=1}^{M} \lambda_l\, k_l\left(x_i, x_j\right)} \tag{5.12}$$

where λ_l are the kernel weights. The problem of defining the best kernel weights has been addressed in (Lanckriet et al., 2004) and consists in choosing the parameters λ_l that maximize the margin between the support vectors of two classes. This technique was successfully introduced and applied in speaker verification (Dehak et al., 2008b). We begin by reiterating the classical support vector machine optimization problem, based on the error penalty parameter C. Then we will show how to reformulate this problem in order to take into account the kernel combination weights. As outlined in chapter 3, the dual support vector machine optimization problem can be expressed as:

$$\begin{aligned} w_c\left(K\right) &= \max_{\alpha}\left(2\alpha^t\mathbf{1} - \alpha^t G\left(K\right)\alpha - \frac{1}{C}\alpha^t\alpha\right) \\ &\quad \text{under the constraints } \alpha \geq 0,\ \alpha^t y = 0 \end{aligned} \tag{5.13}$$

Here, $\mathbf{1}$ is the n-dimensional vector of ones, K is the $n \times n$ kernel Gram matrix which, is symmetric positive semidefinite. The elements of this matrix correspond to kernel function coupling values between n training input vectors. $G\left(K\right)$ is defined by $g_{ij}\left(K\right) = K_{ij}y_iy_j$,

$\alpha \in \mathbb{R}^n$. Furthermore, $\alpha_i > 0$, $y_i = \pm 1$, $i = 1, ..., n$, correspond to the training weight and class label associated with a given input vector i respectively. When the kernel Gram matrix K_f of the new kernel function based on weighted linear kernel fusion is used, the previous optimization problem is extended in order to seek the optimal parameters λ_l that maximize the margin. Maximizing the margin with respect to the parameters λ_l is equivalent to minimizing the problem $w_c\left(K\right)$ given in Equation 5.13 over the convex cone $\mathcal{K}$ of symmetric, positive definite matrices $\mathcal{K} = \{K \in \mathbb{R}^{n \times n} | K = K^t,\ K \succeq 0\}$. The corresponding equations are:

$$\begin{aligned} \min_{K \in \mathcal{K}} \max_{\alpha \in \mathbb{R}^n} \quad & \left(2\alpha^t 1 - \alpha^t G\left(K_f\right)\alpha - \frac{1}{C}\alpha^t\alpha\right) \\ & \text{under the constraints } \operatorname{tr}\left(K_f\right) = c \\ & K_f = \sum_{l=1}^{M} \lambda_l K_l \end{aligned} \tag{5.14}$$

where $\{K_1, ..., K_M\}$ are the initial kernel Gram matrices and $c \geq 0$ is a parameter that conditions the trace of the resulting new Gram matrix. The interesting property of this new problem is that it involves the same optimization criterion for both the boundary weights α_i, $i = 1..n$ and kernel weights λ_l, $l = 1..M$. Both sets of weights combine the information provided by all initial kernel functions in order to maximize the margin. If we pick $\lambda_l \geq 0$, $l = 1..M$, the optimization problem can be reduced to:

$$\begin{aligned} \min_{\lambda \in \mathbb{R}^{+M}} \max_{\alpha \in \mathbb{R}^n} \quad & \left(2\alpha^t 1 - \alpha^t G\left(K_f\right)\alpha - \frac{1}{C}\alpha^t\alpha\right) \\ & \text{under the constraints } \sum_{l=1}^{M} \operatorname{tr}\left(K_l\right)\lambda_l = c \end{aligned} \tag{5.15}$$

This problem can be transposed into the constrained quadratic program (Dehak et al., 2008b; Lanckriet et al., 2004) whose primal-dual solution indicates the optimal weights λ_l and the boundary parameters α_i:

$$\begin{aligned}
&\max_{\alpha,\rho}\left(2\alpha^t 1 - \frac{1}{C}\alpha^t\alpha - c\rho\right) && (5.16)\\
&\text{under the constraints } \rho \geq \frac{1}{\operatorname{tr}(K_l)}\alpha^t G(K_l)\,\alpha,\ l = 1..M\\
&\alpha^t y = 0\\
&\alpha \geq 0
\end{aligned}$$

The optimal kernel combination weight λ_l corresponds to the dual variable given by the l^{th} constraint in the optimization problem. This problem can be solved efficiently with algorithms given in (Sturm, 1999) or (Andersen et Andersen, 2000). Prior to solving the last optimization problem, all initial kernel Gram matrices need to be centered and normalized. Given the kernel Gram matrix K of dimension $n \times n$, the centering and normalization steps are obtained by:

$$\begin{aligned}
\text{Centering: } K_{ij} &\longleftarrow K_{ij} + \frac{1}{n^2}\sum_{m,o=1}^{n} K_{mo} - \frac{1}{n}\sum_{m=1}^{n}(K_{im} + K_{jm}) && (5.17)\\
\text{Normalization: } K_{ij} &\longleftarrow \frac{K_{ij}}{K_{ii}\,K_{jj}} && (5.18)
\end{aligned}$$

In our approach, we proposed to combine two kernels. The first was a cosine kernel applied in the speaker factor space. Within class covariance normalization was used to normalize this kernel. The second kernel was also a cosine kernel, applied in common factor space. Figure 5.3 illustrates the structure of the kernel combination applied to speaker and common factor space.

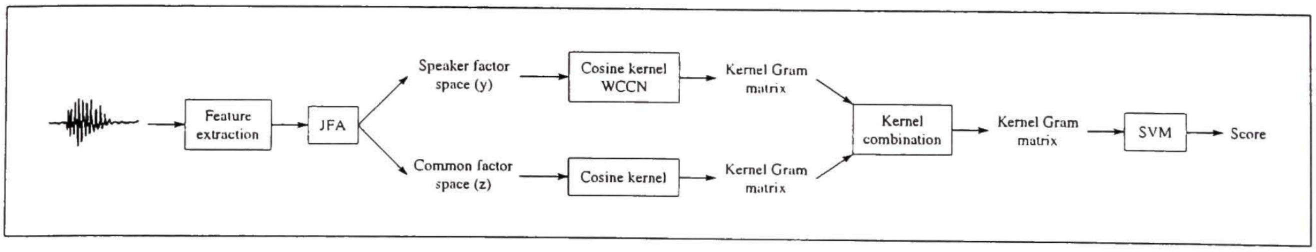

Figure 5.3 Architecture of the kernel combination system.

5.3.3 Experiments

Experimental Set-up

We used the exact same feature frames as in the previous experiments. These experiments are based on gender-dependent UBM and JFA. The UBM was trained on LDC releases of Switchboard II, Phases 2 and 3; Switchboard Cellular, Parts 1 and 2; and NIST 2004 and 2005 SRE data. The full JFA configuration was used. This configuration was made up of 300 speaker factors, 100 channel factors and common factors. We applied decoupled estimation of the eigenvoice matrix V and diagonal matrix D. The eigenvoice matrix V was trained on the same data used in UBM training except NIST 2004 SRE; the matrix D was trained on 2004 SRE data. The eigenchannel matrix was trained on the same data as the UBM. The same SVM and score normalization impostors of previous experiments were used. The within class covariance matrix was trained on NIST 2004 and 2005 SRE datasets. We carried out experiments on female trials of the core condition (telephone data) of the NIST 2006 and 2008 SRE. We carried out two cross-validations when score fusion was applied. We trained the fusion weights on NIST 2006 SRE and we tested on the 2008 dataset. We did the reverse when we tested on the 2006 evaluation data. The kernel combination does not need extra data for choosing the kernel weights, as compared to score fusion.

Results

We present a comparison of results obtained with score fusion and kernel combination applied in the speaker and common factor spaces. In both fusion techniques, we applied the cosine kernel in both spaces. We also used WCCN to normalize the speaker factor cosine kernel. The results are given in Table 5.6.

By examining these results, we first conclude that in both fusion approaches, the common factor vectors give information complementary to that on speaker factors. We decrease the DCF on NIST 2006 SRE from 0.0127, obtained using the cosine kernel applied to speaker

Table 5.6

Scoring result comparison between score fusion and kernel combination for SVM-JFA system. The results are given on EER and MinDCF for the English trials of the female part of the NIST 2006 core condition and 2008 SRE

	NIST 2006 SRE		NIST 2008 SRE	
	EER	MinDCF	EER	MinDCF
Cosine kernel on y + WCCN	1.92%	0.0127	3.68%	0.0190
Cosine kernel on z	3.75%	0.0189	8.68%	0.0282
Linear score fusion	1.74%	0.0101	3.68%	0.0164
Kernel combination	**1.64%**	**0.0096**	3.37%	0.0154
JFA: $s = m + Vy + Dz$	1.64%	0.0120	**3.17%**	**0.0150**

factors, to 0.0096 in the case of the kernel combination. We also note that the kernel combination technique achieves better results than score fusion for both datasets. An advantage of using kernel combination is that development data is not required to train the kernel combination weights, as compared to score fusion. If we compare the results between kernel combination and JFA scoring, we can conclude that kernel combination gives the best DCF 0.0096 for English trials of the NIST2006 SRE and EER equivalent to JFA scoring. However, on 2008 SRE data, both techniques produce equivalent DCF and little improvement in EER (3.37% to 3.17%) for JFA scoring.

5.4 SVM-JFA in Channel Factor Space

Up to this point, we have applied support vector machines in the speaker and common factor spaces. In this section, we conduct an experiment in order to apply the SVM in the channel factor space. Our motivation is to identify any loss of speaker information when channel factors are estimated. We used the same cepstral feature as for previous experiments. The JFA configuration consists of 300 speaker factors and 100 channel factors. The cosine kernel was evaluated on the channel factors in order to achieve the combination between JFA and SVM. The same SVM and t-norm impostor models as in previous experiments were used. We carried out the experiments for female English trials of the core condition (telephone data) of the NIST 2006 SRE.

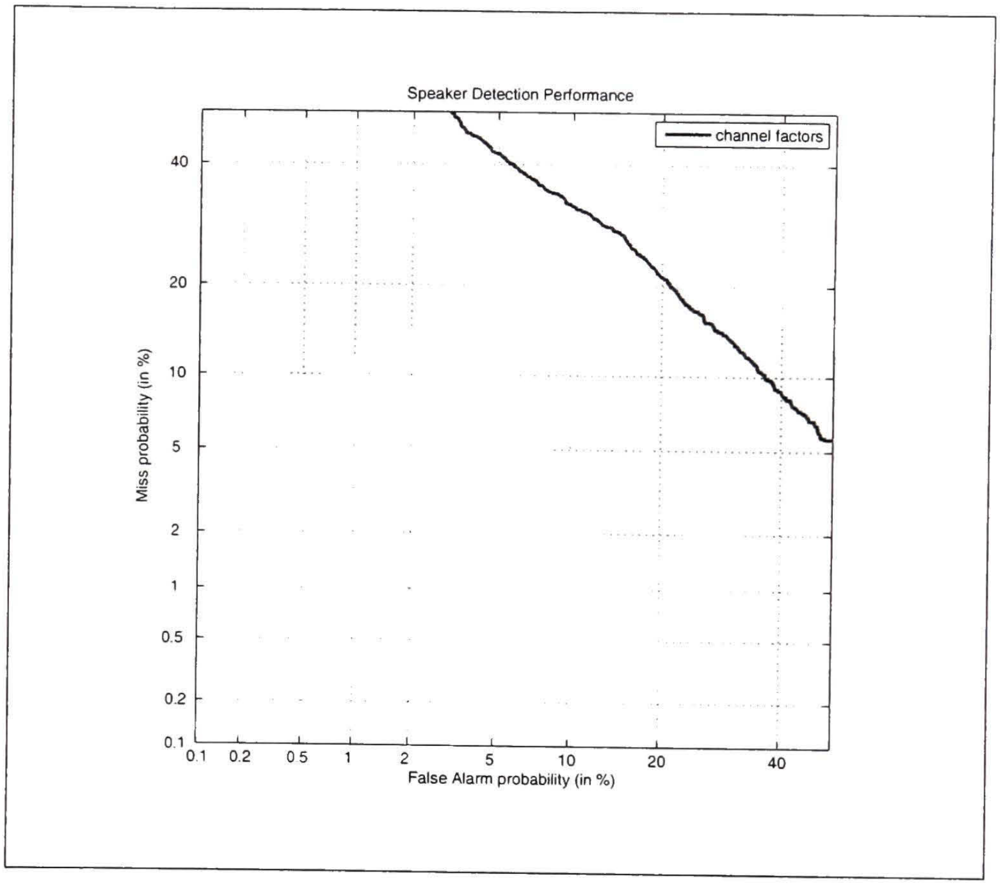

Figure 5.4 DET curves showing the results of the SVM applied in channel space for the female English trials of the core condition of the NIST 2006 SRE.

The channel factors are assumed to contain only channel information and no speaker information. If this assumption is correct, the expected EER of our SVM applied in the channel space is 50%. The DET-curve given in Figure 5.4 shows the performance of our experiment. We achieved an EER of 20% in female English trials of the NIST 2006 SRE, which reveals that channel factors also contain speaker information. In the next section, we will propose several techniques designed to restore the speaker information hidden in the channel factors.

5.5 Restoring Lost Speaker Information from Channel Factors

The experiment of the previous section showed that the channel factors contain information about the speaker. In order to restore these speaker information, two techniques are proposed. The first one involves application of support vector machines in each JFA factor space (speaker, common and channel factors). The scores obtained by all these systems are then fused before making the final decision. The second approach consists in modifying JFA learning in order to combine the eigenvoice and eigenchannel matrices into a single matrix that models both speaker and channel variabilities. In this new approach, we tested several techniques to compensate for the channel effect applied in the new space designated as "the total variability space".

5.5.1 Score Fusion

In our experiments, we employed the restricted joint factor analysis configuration, comprised of speaker and channel factors; common factors were not applied. We implemented two separate support vector machines in both JFA factors space. The first applied SVM used a cosine kernel in the speaker space. We also applied within class covariance normalization technique to normalize the kernel in this space. The second SVM is also based on a cosine kernel, evaluated in the channel factors space. The linear scores fusion is carried out using logistic regression function in the same manner as in section 5.3.1 and as described in Chapter 7.

Experiment

Experimental Set-up

We used the exact same MFCC feature extraction and gender-dependent UBM and JFA as in previous experiments. The UBM and JFA were trained in LDC releases of Switchboard II, Phases 2 and 3; switchboard Cellular, Parts 1 and 2 and NIST 2004 and 2005 SRE. The JFA was made up of 300 speaker factors and 100 channel factors. The within class covariance matrix was trained on NIST 2004 and 2005 SRE. The score fusion weights were trained on NIST 2006 SRE.

Results

These fusion weights were tested in the female English trials of telephone data of the core condition of the NIST 2008 SRE.

Table 5.7

Score fusion results are given as EER and MinDCF on the female part of the core condition of the NIST 2006 and 2008 SRE, English trials

	NIST 2006 SRE		NIST 2008 SRE	
	EER	MinDCF	EER	MinDCF
Cosine kernel on y + WCCN	1.74%	0.0123	4.21%	0.0176
Cosine kernel on x	20.66%	0.0723	25.7%	0.0880
Linear score fusion	**1.74%**	**0.0120**	3.95%	0.0183
JFA scoring	1.74%	0.0121	**3.68%**	**0.0159**

Score fusion performance is given in Table 5.7. These results show that the fusion between two SVMs, applied respectively to the speaker and channel factors, yields a relative improvement in the EER and a slight increase in the DCF. We also note that fusion performance does not exceed that obtained by joint factor analysis scoring. JFA gives the best results, especially for DCF.

5.5.2 Total Variability

Classical joint factor analysis modeling based on speaker and channel factors consists in defining two distinct spaces: the speaker space defined by the eigenvoice matrix V and the channel space defined by the eigenchannel matrix U. The approach that we propose is based on defining only a single space, instead of two separate spaces. This new space, which we refer to as the total variability space, simultaneously contains the speaker and channel variabilities. It is defined by the total variability matrix that contains the eigenvectors of the total variability covariance matrix with the largest eigenvalues. In the new model, we make no distinction between the effect of the speaker and the effect of the channel in GMM supervector space. Given an utterance, the new speaker -and channel-dependent GMM supervector defined by the Equation 2.1 in the JFA framework, is rewritten as follows:

$$\boxed{M = m + Tw} \tag{5.19}$$

where m is the speaker- and channel-independent supervector (which can be taken to be the UBM supervector), T is a rectangular matrix of low rank, and w is a random vector of dimension D having a standard normal distribution $\mathcal{N}(0, I)$. The components of the vector w are the total factors. In other words, M is assumed to be normally distributed with mean vector m and covariance matrix TT^t. The process of training the total variability matrix T is equivalent to learning the eigenvoice V matrix (see section 2.4.2), except for one important difference: in eigenvoice training, all the recordings of a given speaker are considered to belong to the same person. In the case of the total variability matrix, however, a given speaker's entire set of utterances are regarded as having been produced by different speakers. The new model that we propose can be seen as a Principal Component Analysis (PCA) that allows us to project the speech frames onto the total variability space. The total factors are than used as input for the SVM based on the cosine kernel. Figure 5.5 describes all steps of this new SVM-JFA system.

Intersession Compensation

In this new modeling based on total variability space, we propose carrying out channel compensation in the total factor space rather than in the GMM supervector space, as is the case for classical JFA modeling. The advantage of applying channel compensation in the total factor space is the low dimension of these vectors, as compared to GMM supervectors which results

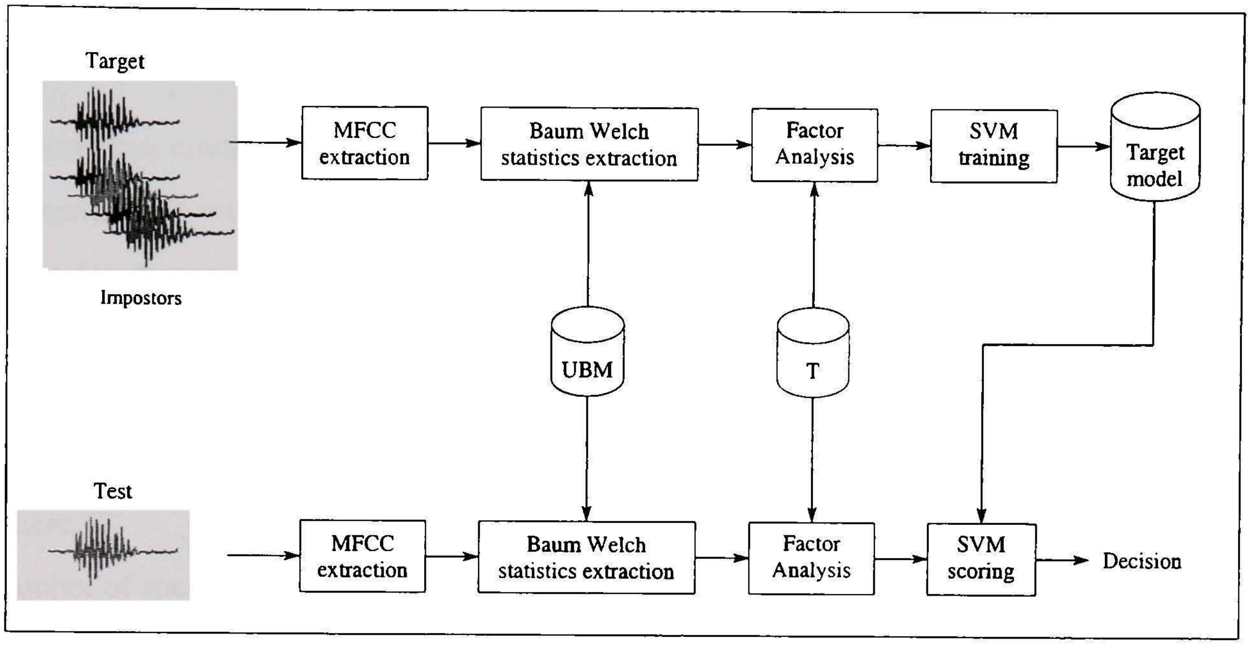

Figure 5.5 Architecture of the SVM-JFA system when the total factors are used.

in a less expensive computation. We tested three channel compensation techniques in the total variability space in order to remove nuisance effects. The first approach is within class covariance normalization which we already applied in the speaker factor space (Section 5.2.1). This technique used the inverse of the within class covariance to normalize the cosine kernel. The second approach is Linear Discriminant Analysis (LDA). The motivation for using this technique is that, in the case where all utterances of a given speaker are assumed to represent one class, LDA attempts to define new special axes that minimize the intra-class variance caused by channel effects, and to maximize the variance between speakers. The advantage of the LDA approach is based on discriminative criteria designed to remove unwanted directions and to minimize the removed information about variance between speakers. Similar work was carried out for speaker verification based on a discriminative version of the Nuisance Attribute Projection (NAP) algorithm without any success (Vogt et al., 2008). The last approach is the NAP presented in the preceding chapter (Section 4.6). This technique proposed a channel space definition based on the best eigenvector of the within class covariance computed in the total factor speakers background. The total factor vectors are projected onto the orthogonal complementary channel space, which is the speaker space.

Within Class Covariance Normalization

Within class covariance normalization is presented in detail in Section 5.2.1. It consists in computing the within class covariance matrix in the total factor space using a set of background impostors. The computation of this matrix is given by:

$$W = \frac{1}{S}\sum_{s=1}^{S}\frac{1}{n_s}\sum_{i=1}^{n_s}\left(w_i^s - \overline{w_s}\right)\left(w_i^s - \overline{w_s}\right)^t \tag{5.20}$$

where $\overline{w_s} = \frac{1}{n_s}\sum_{i=1}^{n_s} w_i^s$ is the mean of the total factor vectors of each speaker s, S is the number of speakers and n_s is the number of utterances of each speaker s. We use the inverse of this matrix to normalize the direction of the total factor components, without removing any nuisance direction. The new cosine kernel is given by the following equation:

$$\boxed{k\left(w_1, w_2\right) = \frac{w_1^t W^{-1} w_2}{\sqrt{w_1^t W^{-1} w_1}\,\sqrt{w_2^t W^{-1} w_2}}} \tag{5.21}$$

where w_1 and w_2 are two total variability factor vectors.

Linear Discriminant Analysis

Linear discriminant analysis is a technique for dimensionality reduction that is widely used in the field of pattern recognition. The idea behind this approach is to seek new orthogonal axes to better discriminate between different classes. The axes found must satisfy the requirement of simultaneously maximizing between class variance and minimizing intra-class variance. In our modeling, each class is made up of all the recordings of a single speaker. The LDA optimization problem can be defined according to the following ratio:

$$\boxed{J\left(v\right) = \frac{v^t S_b v}{v^t S_w v}} \tag{5.22}$$

This ratio is often referred to as the Rayleigh coefficient for space direction v. It represents the amount of information ratio of the between class variance and within class variance, given space direction v. The matrices S_b and S_w correspond respectively to the between class and within class covariance matrix. These are calculated as follows:

$$S_b = \sum_{i=1}^{S} (w_i - \overline{w}) (w_i - \overline{w})^t \tag{5.23}$$

$$S_w = \sum_{s=1}^{S} \frac{1}{n_s} \sum_{i=1}^{n_s} (w_i^s - \overline{w_s}) (w_i^s - \overline{w_s})^t \tag{5.24}$$

where $\overline{w_s} = \frac{1}{n_s} \sum_{i=1}^{n_s} w_i^s$ is the mean of total factor vectors for each speaker, S is the number of speakers and n_s is the number of utterances for each speaker s. In the case of total factor vectors, the mean vector of the total speaker population $\overline{w}$ is equal to the zero vector since, in factor analysis, the total factors have a standard normal distribution $w \sim \mathcal{N}(0, I)$ with zero mean and identity covariance matrix. The purpose of LDA is to maximize the Rayleigh coefficient. This maximization serves to define a projection matrix A of size $D \times k$ composed of the best eigenvectors (those with highest eigenvalues) of the general eigenvalue equation:

$$\boxed{S_b\, v = \lambda S_w\, v} \tag{5.25}$$

where λ is the diagonal matrix of eigenvalues. When the within class covariance is a non-singular matrix, this problem can be reduced to finding the best k eigenvectors of the matrix $S_w^{-1}\, S_b$. The total factor vectors are submitted to the projection matrix A obtained by LDA. The new cosine kernel between two total factor vectors w_1 and w_2 can be rewritten as:

$$\boxed{k(w_1, w_2) = \frac{(A^t w_1)^t\, (A^t w_2)}{\sqrt{(A^t w_1)^t\, (A^t w_1)}\, \sqrt{(A^t w_2)^t\, (A^t w_2)}}} \tag{5.26}$$

The motivation in using LDA is that it allows us to define a new projection matrix aimed at minimizing the intra-class variance and maximizing the between class variance, i.e. the variance between speakers, which is the key requirement in speaker verification.

Nuisance Attribute Projection

The nuisance attribute projection algorithm was presented in section 4.6. It is based on finding a projection matrix appropriate for removing the nuisance direction. The projection matrix carries out an orthogonal projection in the channel's complementary space, which depends only on the speaker. The projection matrix is formulated as:

$$\boxed{P = I - vv^t} \tag{5.27}$$

where v is a rectangular matrix of low rank whose columns are the k best eigenvectors of the same within class covariance matrix (or channel covariance) given in Equation 5.20.

These eigenvectors define the channel space. The cosine kernel based on the NAP matrix is given as follows:

$$\boxed{k\left(w_1, w_2\right) = \frac{\left(Pw_1\right)^t \left(Pw_2\right)}{\sqrt{\left(Pw_1\right)^t \left(Pw_1\right)}\sqrt{\left(Pw_2\right)^t \left(Pw_2\right)}}} \tag{5.28}$$

where w_1 and w_2 are two total variability factor vectors.

Experiment

Experimental Set-up

We used exactly the same MFCC feature extraction and gender-dependent UBMs as for the previous experiments. The total variability matrix was trained on LDC releases of Switchboard II, Phases 2 and 3; Switchboard Cellular, Parts 1 and 2; and NIST 2004 and 2005 SRE data. We used 400 total factors. The within class covariance and LDA projection matrices were trained on NIST 2004 and 2005 SRE data. The NAP matrix was trained on the same data as the total variability matrix. The same SVM and t-norm impostors as in previous experiments were also used. In all our experiments, we used the cosine kernel in order to build the SVM systems. All the results are reported on the female part of the core condition of the NIST 2006 and 2008 SRE for telephone data.

Results

Within Class Covariance Normalization The experiments carried out in this section compare the results obtained with and without application of within class covariance normalization in total variability factor space. We also present results given by the joint factor analysis scoring based on integration over channel factors. The results are given in tables 5.8 and 5.9.

Table 5.8

WCCN performance in the total factor space. The results are given on EER and MinDCF on the female part of the NIST 2006 SRE core condition

	English trials		All trials	
	EER	MinDCF	EER	MinDCF
JFA : $s = m + Vy$	1.74%	0.0121	3.84%	0.0223
Cosine kernel without WCCN	3.29%	0.0211	5.39%	0.0313
Cosine kernel with WCCN	**1.87%**	**0.0116**	**2.76%**	**0.0171**

If we compare the results with and without WCCN, we find that its application helps to compensate for channel variability in the total factor space. This improvement was very marked for the NIST 2006 SRE data, especially for the all trials condition. We obtained an EER of 2.76%, which represents a 1% absolute improvement compared to JFA scoring. However, when we compare the same performance on NIST 2008 SRE data, we can conclude that the classical JFA scoring based on integration over channel factors yields the best results. This is the same conclusion drawn in section 5.2.2. It can be explained by the fact that when

Table 5.9

WCCN performance in the total factor space. The results are given on EER and MinDCF on the female part of the NIST 2008 SRE core condition

	English trials		All trials	
	EER	MinDCF	EER	MinDCF
JFA : $s = m + Vy$	**3.68%**	**0.0159**	**6.3%**	**0.0327**
Cosine kernel without WCCN	5.33%	0.0208	8.40%	0.0406
Cosine kernel with WCCN	4.73%	0.0180	7.32%	0.0351

the total variability space is adequate for the test data, we can achieve very interesting results using an SVM with WCCN applied in the total variability factor space.

Linear Discriminant Analysis This section presents the results obtained with linear discriminant analysis applied in the total variability factor space in order to compensate for channel effects. We carried out several experiments using different LDA dimension reductions, in order to show the effectiveness of this technique in removing the unwanted nuisance directions. The results given in table 5.10 were obtained on the NIST 2006 speaker recognition evaluation data.

These results show the efficiency of LDA to compensate for channel effects. A first important remark is that application of LDA to rotate space in order to minimize the within speaker variance, without any dimensionality reduction (dim $=$ 400), improves performance in the case of the cosine kernel. If we try to minimize the DCF as requested in the NIST competition, the best results are obtained by reducing dimensionality to (dim $=$ 250). When no channel compensation is applied, we obtain a DCF value of 0.0313. Applying a dimensional reduction from size 400 to 250 significantly improves performance, as shown by the resulting DCF value of 0.0115. However, if we compare the EER obtained by LDA with that obtained by WCCN, we find that the latter approach gives better results than the first one. This observation motivated us to combine both techniques simultaneously. We performed several

Table 5.10

The LDA dimensionality reduction results are given on EER and MinDCF on the female part of the core condition of the NIST 2006 SRE

	English trials		All trials	
	EER	MinDCF	EER	MinDCF
JFA : $s = m + Vy$	1.74%	0.0121	3.84%	0.0223
No channel compensation	3.29%	0.0211	5.39%	0.0313
WCCN	**1.87%**	**0.0116**	**2.76%**	**0.0171**
LDA dim = 400	2.38%	0.0133	3.61%	0.0205
LDA dim = 350	2.25%	0.0134	3.56%	0.0207
LDA dim = 300	2.31%	0.0134	3.46%	0.0198
LDA dim = 250	2.38%	0.0115	3.31%	0.0189
LDA dim = 200	2.56%	0.0132	3.48%	0.0190
LDA dim = 150	2.65%	0.0130	3.36%	0.0186
LDA dim = 100	2.84%	0.0130	3.61%	0.0189

experiments where, in a preliminary step, we applied LDA to remove nuisance directions; thereafter we used WCCN in the reduced space in order to normalize the new cosine kernel. During the training step, we began by training the LDA projection matrix on the NIST 2004 and 2005 SRE datasets, then we projected the same data in the reduced space in order to compute the within class covariance matrix. Figure 5.6 shows the value of MinDCF versus the number of spatial dimensions defined by the LDA, in order to find the optimal dimension of the new space. These results were computed on the NIST 2006 SRE dataset.

The best MinDCF achieved by the combination of LDA and WCCN is 0.0107 for English trials and 0.0164 for all trials. These results were obtained with a new space dimension of dim = 200. Table 5.11 and 5.12 compare these results with those obtained with JFA scoring, WCCN alone and LDA alone on the NIST 2006 and 2008 SRE datasets. We first note that applying WCCN in the LDA projected space helps to improve performance as compared to LDA alone. If we compare the performance of LDA and WCCN combination with that

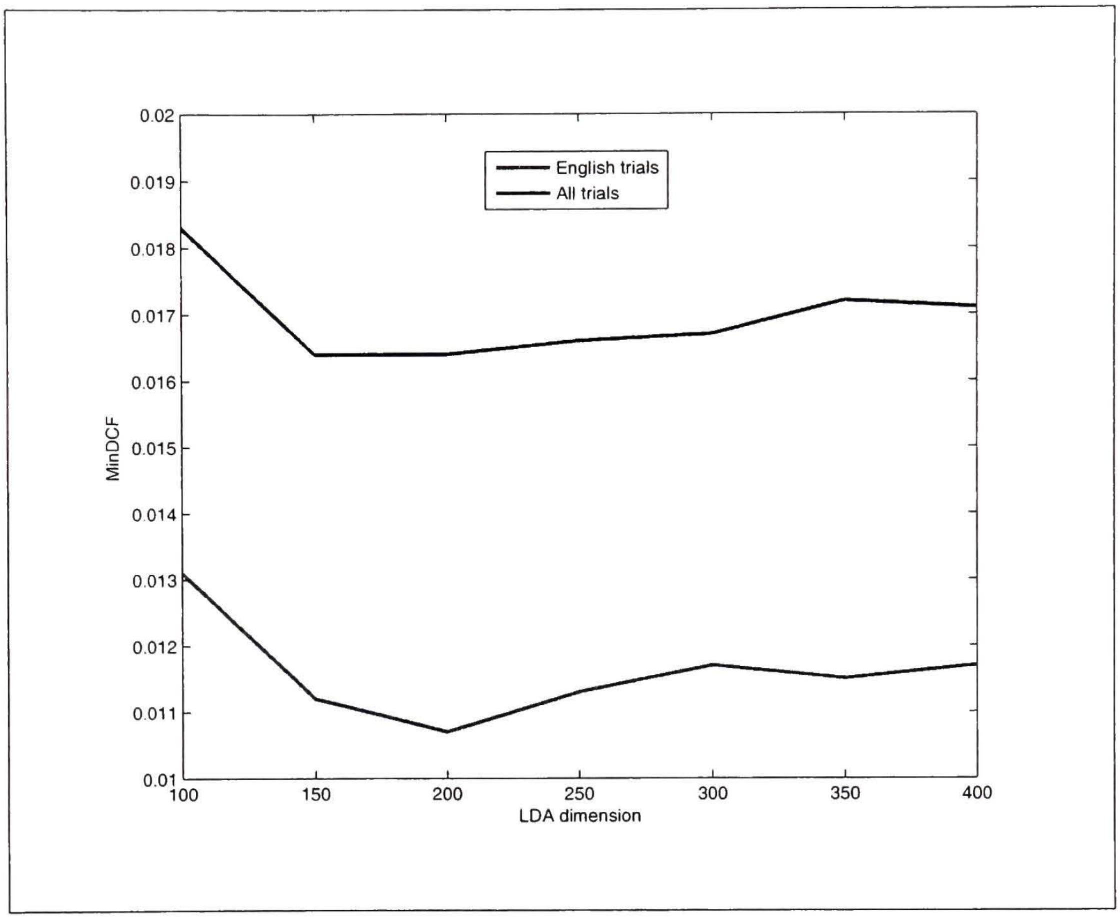

Figure 5.6 MinDCF for the NIST 2006 SRE of the SVM-JFA system based on the LDA technique.

obtained with JFA scoring and WCCN alone, we find that this combination achieves the best MinDCF for the English and all trial conditions for both the NIST 2006 and 2008 SRE datasets. We can see that this combination also yields the best EER on the all trials condition for both datasets.

Nuisance Attribute Projection The same experiment as LDA before was carried out in order to show the performance of the nuisance attribute projection technique to compensate for channel effects. We begin by presenting the results obtained for NAP based on several corank numbers which represent the number of removed dimensions. Table 5.13 gives the results of these experiments for the female trials of the core condition of the NIST 2006 SRE.

Table 5.11

Comparison of results between JFA scoring and several SVM-JFA channel compensation techniques based on LDA and WCCN. The results are given on EER and MinDCF on the female part of the core condition of the NIST 2006 SRE

	English trials		All trials	
	EER	MinDCF	EER	MinDCF
JFA : $s = m + Vy$	**1.74%**	0.0121	3.84%	0.0223
WCCN	1.87%	0.0116	2.76%	0.0171
LDA dim = 250	2.38%	0.0115	3.31%	0.0189
LDA dim = 200 + WCCN	2.05%	**0.0107**	**2.72%**	**0.0164**

Table 5.12

Comparison of results between JFA scoring and several SVM-JFA channel compensation techniques based on LDA and WCCN. The results are given on EER and MinDCF on the female part of the core condition of the NIST 2008 SRE

	English trials		All trials	
	EER	MinDCF	EER	MinDCF
JFA : $s = m + Vy$	**3.68%**	0.0159	6.3%	0.0327
WCCN	4.73%	0.0180	7.32%	0.0351
LDA dim = 200 + WCCN	3.95%	**0.0147**	**6.09%**	**0.0326**

These results prove that application of NAP to compensate for channel effects helps to improve the performance of the SVM applied on the total factor space. We decreased the MinDCF for the English trials from 0.0211 when no channel compensation was applied, to 0.0115 when NAP corank was equal to 200. As in the case of LDA, we also found that WCCN gave better results than NAP, which again persuaded us to combine NAP and WCCN. To apply this new approach, we first start by training the NAP matrix in the same manner as before, using all the data used in training the total factor matrix (see previous experimental

Table 5.13

The results obtained with several NAP coranks. These results are given on EER and MinDCF on the female part of the core condition of the NIST 2006 SRE

	English trials		All trials	
	EER	MinDCF	EER	MinDCF
JFA : $s = m + Vy$	1.74%	0.0121	3.84%	0.0223
No channel compensation	3.29%	0.0211	5.39%	0.0313
WCCN	1.87%	0.0116	2.76%	0.0171
NAP corank = 10	2.92%	0.0175	4.45%	0.0249
NAP corank = 60	2.63%	0.0141	4.05%	0.0210
NAP corank = 100	2.50%	0.0133	3.81%	0.0196
NAP corank = 150	2.29%	0.0118	3.38%	0.0174
NAP corank = 200	**2.29%**	**0.0115**	**3.27%**	**0.0178**
NAP corank = 250	2.19%	0.0130	3.51%	0.0187
NAP corank = 300	2.83%	0.0144	4.13%	0.0199

set-up section), then we compute the WCCN matrix in the new projected space. The MinDCF of this combination versus the number of the NAP corank is given in figure 5.7.

The best MinDCF achieved by this combination, based on the nuisance attribute projection and within class covariance normalization, is 0.0107 for English trials and 0.0164 for all trials. These results were obtained with a NAP corank of 150. Tables 5.14 and 5.15 compare these results with those obtained with JFA scoring and the WCCN technique on both the NIST 2006 and 2008 SRE datasets. The same remark as for LDA is applicable in the NAP case, which is the combination of WCCN and NAP to improve the performance compared to NAP applied alone. If we compare the performance of NAP and WCCN combination with that obtained with JFA scoring and WCCN alone in 2008 SRE dataset, we find that this combination achieved better MinDCF than the JFA scoring; however, the JFA achives the best EER.

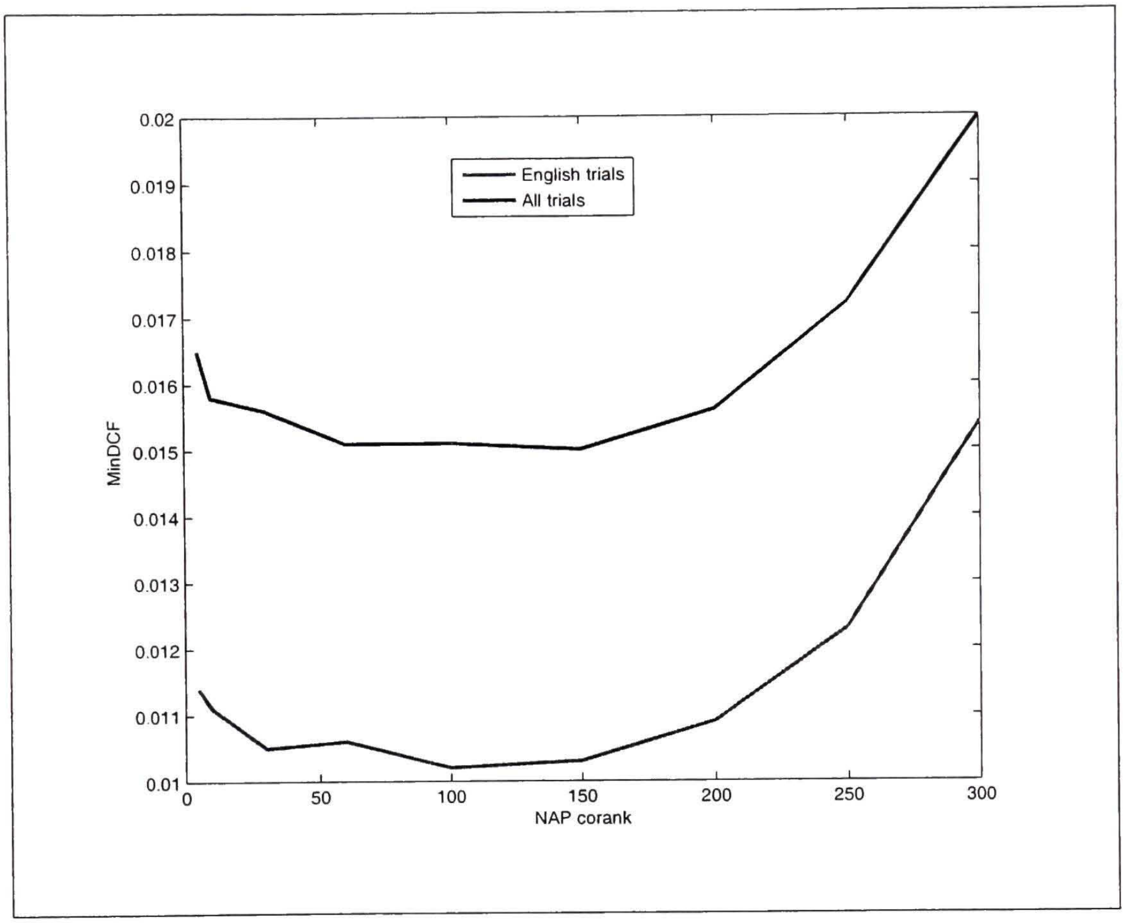

Figure 5.7 MinDCF for the NIST 2006 SRE of the SVM-JFA system based on the NAP technique.

Table 5.16 summarizes the results obtained for JFA scoring and SVM-JFA based on WCCN, the LDA and WCCN combination, and NAP combined with WCCN. These results show that the LDA and WCCN combination gives the best MinDCF (**0.0147**) for English trials and also the best EER in all trials; however the NAP and WCCN combination gave the best MinDCF on all trials.

Results for Both Genders

In this section, we present the results for both genders (male and female speakers) obtained through the application of support vector machines in total factor space. We used exactly the same universal background model and factor analysis configuration (400 total factors) as

Table 5.14

Comparison results between JFA scoring and several SVM-JFA channel compensation techniques based the NAP and WCCN. The results are given on EER and MinDCF on the female part of the core condition of the NIST 2006 SRE

	English trials		All trials	
	EER	MinDCF	EER	MinDCF
JFA : $s = m + Vy$	**1.74%**	0.0121	3.84%	0.0223
WCCN	1.87%	0.0116	2.76%	0.0171
NAP corank = 150	2.29%	0.0118	3.38%	0.0174
NAP corank = 150 + WCCN	1.83%	**0.0103**	**2.66%**	**0.0150**

Table 5.15

Comparison results between JFA scoring and several SVM-JFA channel compensation techniques based on the NAP and WCCN. The results are given on EER and MinDCF on the female part of the core condition of the NIST 2008 SRE

	English trials		All trials	
	EER	MinDCF	EER	MinDCF
JFA : $s = m + Vy$	**3.68%**	0.0159	**6.3%**	0.0327
WCCN	4.73%	0.0180	7.32%	0.0351
NAP corank = 150 + WCCN	4.73%	**0.0157**	6.70%	**0.0309**

in the last two previous experiments. The only difference lies in the amount of data used to train the total variability matrix T for both genders. We added the Fisher English database Parts 1 and 2 to the previous used data, namely LDC releases of Switchboard II, Phases 2 and 3; Switchboard Cellular, Parts 1 and 2; and NIST 2004-2005 SRE datasets, in order to capture a greater extent of variability. We applied linear discriminant analysis and nuisance attribute projection, in combination with within class covariance normalization to compensate for channel effects. We used the same female impostors to estimate the SVM model and to carry out the score normalization as described in previous experiments. The UBM for male

Table 5.16

Summary of results obtained with JFA scoring and several SVM-JFA channel compensation techniques. The results are given on EER and MinDCF on the female part of the core condition of the NIST 2008 SRE

	English trials		All trials	
	EER	MinDCF	EER	MinDCF
JFA : $s = m + Vy$	**3.68%**	0.0159	**6.3%**	0.0327
WCCN	4.73%	0.0180	7.32%	0.0351
LDA dim = 200 + WCCN	3.95%	**0.0147**	**6.09%**	0.0326
NAP corank = 150 + WCCN	4.73%	0.0157	6.70%	**0.0309**

speakers was trained on the same corpus as female speakers. We used 1007 impostors to train the support vector machines. These impostors are taken from the same data used for UBM training except for the NIST 2005 SRE dataset. We applied t-norm score normalization based on 204 impostors taken from the NIST 2005 SRE. The experiments were carried out on the telephone data of the core condition of the NIST 2008 SRE. Tables 5.17 and 5.18 compare results between SVM-JFA and JFA scoring based on both configurations (with and without common factors).

Table 5.17

Comparison results between JFA scoring and several SVM-JFA channel compensation techniques. The results are given on EER and MinDCF on the female part of the core condition of the NIST 2008 SRE

	English trials		All trials	
	EER	MinDCF	EER	MinDCF
JFA: $s = m + Vy + Dz$	3.17%	0.0150	6.15%	0.0319
JFA: $s = m + Vy$	3.68%	0.0159	6.38%	0.0327
LDA dim = 200 + WCCN	**3.68%**	**0.0150**	**6.02%**	**0.0319**
NAP corank = 150 + WCCN	3.95%	0.0157	6.36%	0.0321

Table 5.18

Comparison results between JFA scoring and several SVM-JFA channel compensation techniques. The results are given on EER and MinDCF on the male part of the core condition of the NIST 2008 SRE

	English trials		All trials	
	EER	MinDCF	EER	MinDCF
JFA: $s = m + Vy + Dz$	2.64%	0.0111	5.15%	0.0273
LDA dim = 200 + WCCN	**1.28%**	**0.0095**	**4.57%**	**0.0241**
NAP corank = 150 + WCCN	1.51%	0.0108	4.58%	0.0241

Inspection of the tabulated results reveals that, in the case of the SVM-JFA system, the LDA and WCCN combination achieves better performance than the NAP and WCCN combination. Adding more training data to estimate total variability matrix T (Adding Fisher database) improves the performance of the SVM-JFA system. The EER values for the NIST 2008 SRE English trials decreases from 3.95% (Table 5.12) to 3.68% (Table 5.17) when LDA and WCCN are applied. Finally, the SVM-JFA achieves better results than the full configuration of joint factor analysis (with speaker and common factors), especially for male trials. We obtain 1.23% absolute improvement in EER for the English trials of the NIST 2008 SRE. For female trials, the JFA achieves a better EER for English trials (a value of 3.17% in EER for JFA scoring compared to 3.68% for the SVM-JFA); however, the SVM-JFA produced a better MinDCF on both trial conditions and a better EER on all trials (6.02% in EER for SVM-JFA compared to 6.15% in EER for JFA scoring). In conclusion, the application of SVM in the total factor space leads to commendable results compared to those obtained with the full JFA configuration, despite the absence of common factors in our new SVM-JFA modeling. This having been said, it would be interesting in future work to combine the total and common factors in the same SVM-JFA system.

5.6 Cosine Distance Scoring

In this section and based on the results obtained with SVM applied in the total variability space using the cosine kernel, we propose a new scoring technique which directly use the value of the cosine kernel between the target speaker total factor vector w_{target} and the test total factor vector w_{test} as decision score:

$$\text{score}\left(w_{\text{target}}, w_{\text{test}}\right) = \frac{\left\langle w_{\text{target}}, w_{\text{test}} \right\rangle}{\left\| w_{\text{target}} \right\| \left\| w_{\text{test}} \right\|} \gtrless \theta \tag{5.29}$$

The value of this kernel is then compared to the threshold θ in order to take the final decision. The advantage of this scoring is that no speaker enrollement is required. The use of the cosine kernel as a decision score for speaker verification makes the process faster and less complex than other JFA scoring (Glembek et al., 2009). Figure 5.8 present the cosine distance scoring system.

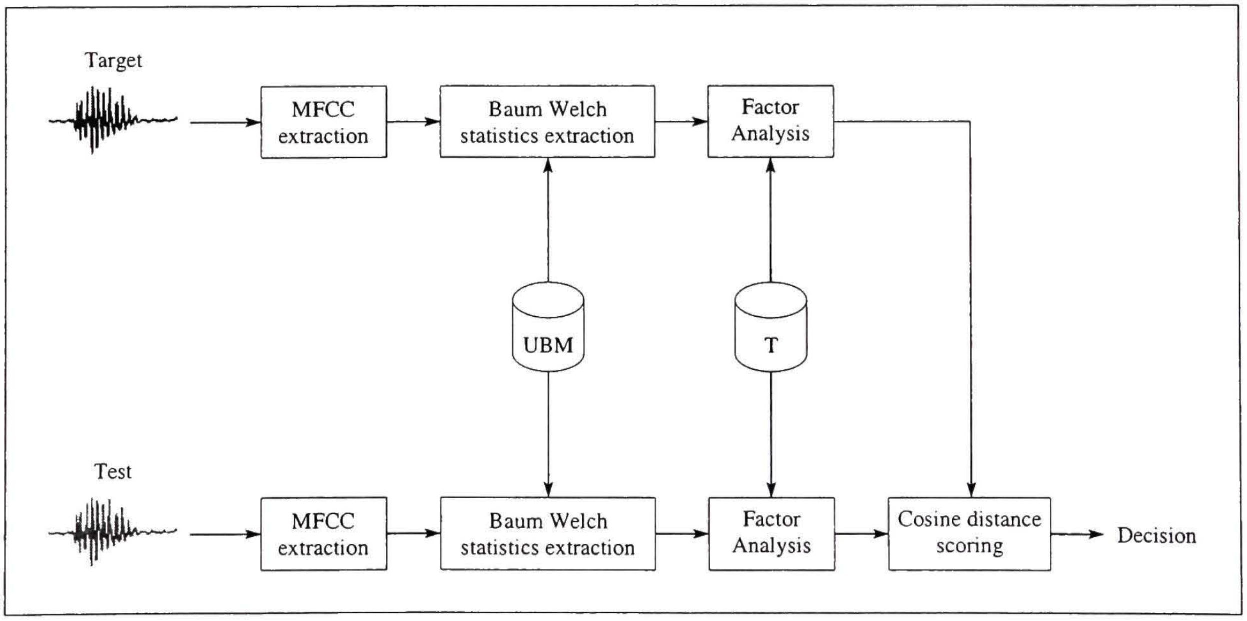

Figure 5.8 Architecture of the cosine distance scoring system.

5.6.1 Experiments

The cosine distance scoring is based on the same total variability matrix and total factor vectors as the previous SVM-JFA system (when the Fisher data are used to train the total variability matrix T). In this modeling, the scores are normalized using the zt-norm technique based on the same t-norm model impostors as in the SVM-JFA system. Data from the preceding training SVM impostors are used as z-norm utterances. we used the same LDA and WCCN combination matrix as the SVM-JFA system.

The experiments are carried out on short2-short3 (core condition), short2-10sec and 10sec-10sec conditions of the NIST 2008 SRE dataset. We used exactely the same cosine distance scoring and channel compensation for all these condtions.

Short2-short3 Condition

Table 5.19 and 5.20 present the results obtained with cosine distance scoring and JFA scoring for both genders on the core condition of telephone Telephone data of the NIST 2008 SRE dataset. We used the same channel compensation techniques as in the SVM-JFA experiments.

The results given in both tables show that cosine distance scoring based on total factor vectors definitively gave the best results in all conditions of the NIST evaluation compared to

Table 5.19

Comparison of results from JFA scoring and cosine distance scoring with several channel compensation techniques. The results are given as EER and DCF on the female trials of the core condition of the NIST 2008 SRE dataset

	English trials		All trials	
	EER	MinDCF	EER	MinDCF
JFA scoring	3.17%	0.0150	6.15%	0.0319
LDA (200) + WCCN	**2.90%**	**0.0124**	**5.76%**	**0.0322**

Table 5.20

Comparison of results from JFA scoring and cosine distance scoring with several channel compensation techniques. The results are given as EER and DCF on the male trials of the core condition of the NIST 2008 SRE dataset

	English trials		All trials	
	EER	MinDCF	EER	MinDCF
JFA scoring	2.64%	0.0111	5.15%	0.0273
LDA (200) + WCCN	**1.12%**	**0.0094**	**4.48%**	**0.0247**

JFA scoring. If we compare these results with those obtained with SVM-JFA system in tables 5.17 and 5.18, we find that cosine distance scoring achieves the best results, especially for female trials. Using cosine distance scoring, we obtained an EER of 2.90% and MinDCF of 0.0124 for English trials versus an EER of 3.68% and MinDCF of 0.0150 for the SVM-JFA system. The main contribution of both new modelings is the use of the cosine kernel on new features, which are the total variability factors extracted using a simple factor analysis.

Short2-10sec Condition

Table 5.21 and 5.22 present the results obtained with cosine distance scoring and JFA scoring for both genders. The experiments are carried out on telephone data of the short2-10sec condition. In this condition we have around 2 m of speech to enroll the speaker and 10 s for testing. We used the same channel compensation techniques as in the SVM-JFA experiments.

Both tables reveal that cosine distance scoring achieves better results than the full joint factor analysis configuration (with speaker and common factors) especially in female trials. We obtain around 2% absolute improvement in EER of the English trials. However, the improvment is not very significant for males trials compared to the female ones.

Table 5.21

Comparison of results from JFA scoring and cosine distance scoring with several channel compensation techniques. The results are given as EER and DCF on the female trials of short2-10sec condition of the NIST 2008 SRE dataset

	English trials		All trials	
	EER	MinDCF	EER	MinDCF
JFA scoring	7.89%	0.0354	11.19%	0.0643
LDA (200) + WCCN	**5.91%**	**0.0347**	**9.59%**	**0.0506**

Table 5.22

Comparison of results from JFA scoring and cosine distance scoring with several channel compensation techniques. The results are given as EER and DCF on the male trials of short2-10sec condition of the NIST 2008 SRE dataset

	English trials		All trials	
	EER	MinDCF	EER	MinDCF
JFA scoring	5.36%	0.0275	8.09%	0.0382
LDA (200) + WCCN	**5.18%**	**0.0263**	**7.38%**	**0.0365**

10sec-10sec Condition

Table 5.23 and 5.24 present the results obtained with cosine distance scoring and full JFA scoring for both genders on the 10sec-10sec conditon of the NIST 2008 SRE data. In this condition, we have only 10 seconds of speech to enroll the target speaker model and also 10 seconds for testing which make the recognition process . We used the same LDA and WCCN combination to compensate for the channel effects as in the SVM-JFA experiments.

The results given in both tables show an absolute improvement around 4% in the EER for Both genders. The EER on the English trials goes from 16.01% to 12.19% for females and

Table 5.23

Comparison of results from JFA scoring and cosine distance scoring with several channel compensation techniques. The results are given as EER and DCF on the female trials of 10sec-10sec condition of the NIST 2008 SRE dataset

	English trials		All trials	
	EER	MinDCF	EER	MinDCF
JFA scoring	16.01%	0.0647	17.99%	0.0758
LDA (200) + WCCN	**12.19%**	**0.0573**	**16.59%**	**0.0725**

Table 5.24

Comparison of results from JFA scoring and cosine distance scoring with several channel compensation techniques. The results are given as EER and DCF on the male trials of 10sec-10sec condition of the NIST 2008 SRE dataset

	English trials		All trials	
	EER	MinDCF	EER	MinDCF
JFA scoring	15.20%	0.0575	15.45%	0.0686
LDA (200) + WCCN	**11.09%**	**0.0473**	**14.44%**	**0.0632**

15.20% to 11.09% for males. We also note a quite significant improvement in DCF. In our knowledge's, these results are the best results ever obtained in the 10sec-10sec condition. The reason for obtaining these extraordinary results can be explained by the fact that in our modeling we have few parameters to estimate only 400 total factors compared to JFA when the common factors are also used.

5.7 Discussion

In this chapter, we discussed the combination between SVMs and JFA based on the cosine kernel for use in speaker verification. We proposed also a new fast scoring based on the last kernel values as decision scores. We have shown that using speaker factors, which represent the spatial coordinates of the target speaker in the speaker space, as input to the SVM produces better results compared to using the Gaussian mixture model supervector. This process leads to a quite good linear separation in speaker space. We also proved that it is not necessary to compensate for the channel effect in the GMM supervector space via the eigenchannel modeling proposition of classical JFA. Instead, we reformulate the problem by assigning both the channel and speaker variabilities to a low dimension space called the total variability space. This space is defined using a low rank matrix that we refer to as the total variability matrix, that simultaneously includes the two eigenvoice and eigenchannel matrices of the JFA approach. In this new space, we tested three techniques to compensate for the intersession problem. These techniques are linear discriminant analysis, nuisance attribute projection and within class covariance normalization. The best results were obtained with the combination between LDA and WCCN. The advantage of using the LDA is to remove the nuisance direction and maximize the variance between the speakers, which is the key point in speaker verification. The results obtained with cosine distance scoring outperforms those obtained with both SVM-FA and classical JFA scorings on several NIST evaluation condition. However, the best improvement is obtained on 10sec-10sec condition of the NIST 2008 SRE dataset (we achieved an absolute improvement of 4% for both gender).

CHAPTER 6

LONG-TERM SPEAKER CHARACTERISTICS

In this chapter a novel approach for modeling prosodic and cepstral long-term speaker features is presented. The modeling is based on a syllable segmentation carried out in an unsupervised manner. Segments are delimited by energy-based contour regions. Acoustic feature (energy, pitch, formant) contours of each segmented unit, or pseudo-syllable, are parameterized using Legendre polynomials. The corresponding sets of expansion coefficients, including unit duration, are modeled using Gaussian mixture models. Speaker and intersession variabilities are modeled through joint factor analysis. A part of this work was published in (Dehak et al., 2007a,b).

6.1 Long-term Speaker Characteristics

This section presents an overview of methods aimed at modeling long-term speaker information, also called high-level information. The relevant acoustic characteristics can be extracted from the spectral or prosodic speaker information domains.

6.1.1 Spectral Domain

Modeling Speaker Phoneme Information

The methods proposed here are based on speaker characteristics extracted at the phoneme level. These methods offer the advantage of allowing modeling of the phoneme pronunciations of each speaker for a given language. These approaches require phoneme segmentation of speech. The desired result is usually achieved by using an acoustico-phonetic decoder or speech recognizer system. The disadvantage of applying a speech recognition system is that the system produced is language-dependent. An approach that allows carrying out multi-language phonetic decoding was originally proposed by Zissman (1996) for use in language identification. This method, named Parallel Phone Recognition Language Modeling

(PPRLM), uses several phonetic decoders or multiple speech recognizers in parallel to produce a phonetic segmentation. Among the speaker verification methods based on phoneme units, we cite the works of Andrew *et al.* (2001). These authors modeled the target speaker using multiple GMMs. Each GMM describes a specific phoneme. In (Campbell et al., 2004a), Campbell *et al.* proposed an approach based on support vector machines to model phonetic unit dynamics. The kernel function used in that work is based on phoneme N-grams frequencies. This approach is inspired by the work in the field of language identification (Zissman, 1996). In (Vair et al., 2007), Vair *et al.* propose a GMM-based speaker verification system. Each GMM is trained on a different phoneme. Phonetic segmentation was accomplished through a hybrid phonetic decoder composed of two models: a Hidden Markov Model (HMM) and Neural Networks. Intersession compensation was applied using eigenchannels. This system achieved the best results at the phonetic feature level.

Modeling Speaker Syllabic Information

The work presented in (Baker et al., 2005) is based on the use of pseudo-syllables as basic units. These pseudo-syllables are produced by a technique proposed in (Martin et al., 2006) and correspond to a sequence of language-independent phonetic units which are grouped among the following four phonetic classes: nasal/semi-vowel, vowel/diphthong, fricatives and plosives/silence. In (Baker et al., 2005), the authors modeled the spectral features of these pseudo-syllables using a Gaussian mixture model. The results obtained with this approach are satisfactory; they were, however, able to improve on system performance by replacing the GMMs with HMM (Baker et Sridharan, 2006) for describing the temporal evolution of spectral parameters at the syllabic level.

Modeling Speaker Information in Frequency Stability Area

Recently, various studies have shown that high-level speaker features can be combined with low-level information (short-term spectral features) to increase the robustness of speaker verification systems. These parameters are usually extracted by analyzing the phoneme sequences produced by automatic speech recognition. Two of the main problems that arise when phonetic systems are developed relate to potential differences between the development and evaluation data and lack of transcribed databases. To solve these two problems El Hannani *et al.* proposed in (Hannani et Petrovska-Delacrétaz, 2005, 2007; Hannani et al., 2006) an approach based on automatic language-independent segmentation (Automatic Language-Independent Speech Processing - ALISP -) to extract high-level information. The authors have shown that ALISPs units can be subjected to the same type of analysis as phonemes for the purpose of extracting high level information (Hannani et Petrovska-Delacrétaz, 2005; Hannani et al., 2006). In (Hannani et Petrovska-Delacrétaz, 2007), the authors propose three high-level systems that operate on ALISP segmentation of speech and demonstrate that fusion of these high-level systems with a classical short-term GMM-UBM system improves performance significantly. The results presented in this work were validated on natural speech databases in the context of the NIST speaker recognition evaluation.

6.1.2 Prosodic Domain

Frequently used, prosodic features are based on pitch and energy contour statistics. For example in (Sönmez et al., 1997), the authors show that pitch has a log-normal distribution and they propose a speaker verification system based on distances between pitch histogram values. The same authors propose a pitch contour stylization technique (Sönmez et al., 1998) based on segmentation of the pitch contour. In each segment they extract a set of parameters, such as the median, the slope of the pitch contour, and the segment feature duration. Each feature is modeled with a Gaussian distribution. In (Adami et al., 2003), Adami *et al.* used an n-gram approach for modeling segments obtained by pitch and energy contour stylization. The objective of the n-gram approach is to model the speaker's speaking style. The authors also proposed the application of dynamic time warping between pitch contours extracted from two different recordings with the same context (same word or sentence). This approach improves results, but it requires both word alignment and detection.

The work presented in Kajarekar *et al.* (2004) introduces a novel approach based on Nonuniform Extraction Region Features (NERFs). A NER is a region of a speech utterance between two consecutive pauses larger than a threshold. The pause duration threshold generally used is 500 ms. The use of long-term pauses as segmentation boundaries is motivated by the fact that this kind of pause affects the characteristic of speech prosody. For example, the speakers tend to lengthen the last syllable of a phrase located near the long pause. In (Kajarekar et al., 2004), the authors extract a set of 32 features from each NER (although not all features can be extracted in all cases). This feature set corresponds to statistics of pitch contour evolution, and information concerning phone durations (or higher-level units). The advantage of using NERFs stems from the long-term speaker characteristics of the extracted features. The suggested model for these features is a Gaussian mixture model. A drawback of this approach is that feature extraction requires phone duration values in each NER, and this information must be acquired through a phonetic alignment (Kajarekar et al., 2004).

Another variant of NERFs uses syllables as the basic units. This variant, named Syllable-based Nonuniform Extraction Region Features (SNERFs), was introduced by Shriberg *et al.* and applied in (Shriberg et al., 2005, 2004). The syllabic segmentation is performed using a speech recognition system which makes this technique language-dependent. The approach proposed by Shriberg *et al.* consists in extracting a set of 140 features for each syllable. Discretization of the prosodic syllable features is performed using several bins. The resulting features are then modeled with a SVM based on an n-gram kernel. The results obtained with SNERFs (Shriberg et al., 2005) are given only on the English trials of the NIST speaker recognition evaluation dataset, because the authors used an English speech recognition system. The approach could be extended to other languages by using, in parallel, several speech recognition systems with various languages; an approach similar to parallel phone recognition language modeling for language identification (Zissman, 1996). This approach represents the state of the art in prosodic feature modeling and fusing this system with a cepstral-based system improves the performance of the latter (Shriberg et al., 2005).

In recent works (Ferrer et al., 2007a,b), the authors introduced a technique based on GMM background models that transforms the prosodic feature sequence of syllable-based nonuniform extraction regions into vectors of fixed length, which are subsequently used as input for the support vector machines. In this new approach, the Gaussian mixture model components are equivalent to the bins used in the classical SNERF modeling approach (Shriberg et al., 2005, 2004). The SVM is based on a linear kernel which is evaluated using the Gaussian components weights. For each target and test utterance, the Gaussian weights are obtained by adapting the background model weights via MAP adaptation to these utterances. Two estimation techniques to the GMM background models were tested. The first uses the standard approach based on the EM algorithm and the second is based on vector quantization. The motivation behind the use of vector quantization is that application of the EM algorithm maximizes the likelihood in a way which generates a GMM model with overlapping Gaussians to better approximate the prosodic feature distribution. This overlap produces correlated Gaussian weights that are ill-suited for performing discrimination. On other hand, vector quantization produces better-separated Gaussians. In (Ferrer et al., 2007b), the authors also proposed applying kernel smoothing in order to improve the results.

6.2 Modeling Long-term Prosodic Features with Joint Factor Analysis

The majority of the methods outlined in the preceding section are based on discrete modeling of pitch and energy contours. In this chapter, we propose to implement continuous modeling of these contours instead. This approach offers several advantages: the continuous prosodic contour model can, at the outset, be represented by existing continuous models developed in the speaker recognition literature. In addition, joint factor analysis (Kenny et al., 2005b) can be used to address the effects of speaker and intersession variability in prosodic features. Continuous prosodic contour modeling based on Legendre polynomial expansions has been successfully applied in the field of language identification (Lin et Wang, 2005) and in quantitative phonetics (Grabe et al., 2003). We extract pitch and energy at 10 ms intervals and we break the contours into pseudo-syllabic units (or *segments* for short). We approximate the pitch and energy contours in each segment by Legendre polynomial expansions. The Legendre polynomial coefficients for pitch and energy, together with segment duration, form the prosodic feature set. We calculate one feature vector for each pseudo-syllable. We then model these features using GMMs and compensate for speaker and session variability effects using joint factor analysis. The speaker factors play a crucial role here, since the number of feature vectors corresponding to the given enrollment utterances (400 on average) may be too small for classical MAP estimation to perform reliably. In our initial investigations, our segmentation into pseudo-syllable units does not rely on the output of a speech recognition system as is the case with the SNERF approach; nevertheless, the results obtained with our modeling are comparable to those obtained with SNERF systems.

6.2.1 Feature Extraction

We extracted the log pitch and log energy values calculated at 10 ms intervals with the *praat* package (Boersma, 2001). Pitch is calculated with the autocorrelation method proposed in (Boersma, 2001). We used only the voiced part of the speech signal in our modeling. The log-energy is normalized on an utterance by utterance basis by subtracting the maximum value for the whole utterance from each utterance frame. In the following section, we will describe how the pitch and energy contours (containing more than one syllable) are segmented into pseudo-syllables based on unsupervised segmentation according to the energy contour only.

Segmentation

In order to model the prosodic contours based on the pseudo-syllable as a unit, we segment the long prosodic contours into syllable-like regions in the same way as in (Lin et Wang, 2005). This method is based on detecting the valley points of the energy of the voiced speech contour. In general, these valley points serve as segment boundaries; however, we impose a minimum duration constraint of 60 ms. This enables us to calculate six-term Legendre polynomial expansions. An example of log pitch and normalized log energy segmentation is given in Figure 6.1.

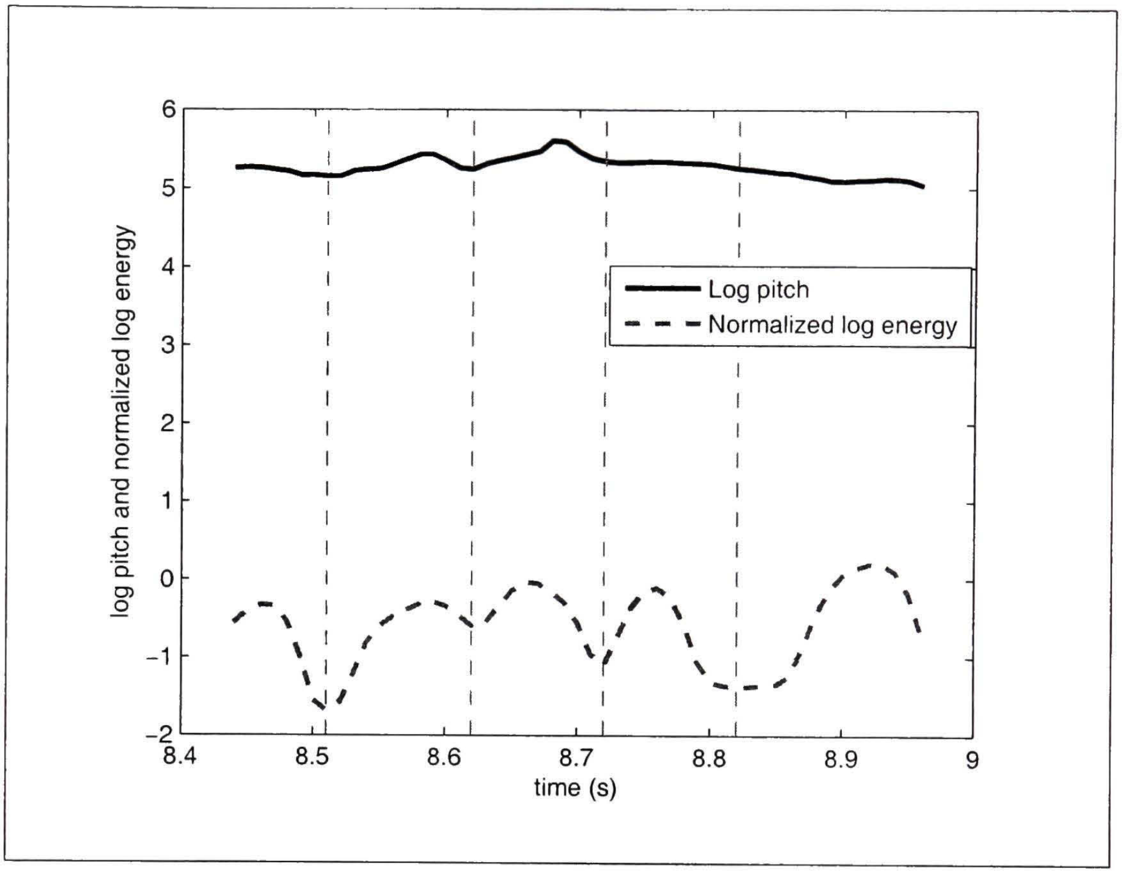

Figure 6.1 Example of segmentation of the log pitch and normalized log energy contours extracted from voiced speech.

We will show in the next paragraph how the pitch and energy contours (based on pseudo-syllable units) are approximated by Legendre polynomials.

Approximation and Time Normalization

For each generated segment, we carry out an approximation of the pitch and energy contour by taking the M leading terms in a Legendre polynomial expansion. That is, each contour $f(t)$ (where t represents time) is approximated as:

$$\boxed{f(t) = \sum_{i=1}^{M} a_i \, P_i(t)} \tag{6.1}$$

where $P_i(t)$ is the i^{th} Legendre polynomial, and we set $M = 5$ in our implementation. Figure 6.2 shows how Legendre polynomials (P_i) model a log pitch contour. Each coefficient models a particular aspect of the contour. For example, a_0 is interpreted as the mean of the segment, a_1 is the slope, a_2 gives information about the curvature of the segment, and a_3, a_4, a_5 model the fine details.

However, in order for these coefficients to be comparable across segments, it is important to carry out time normalization. All the segment coefficients must be scaled and mapped onto the same interval $[-1, +1]$. This technique of prosodic contour approximation has been successfully applied in quantitative phonetics (Grabe et al., 2003) and in language identification applications (Lin et Wang, 2005).

For each segment, we used six coefficients to represent the pitch contour and six coefficients to represent the energy contour. These pitch and energy features, with the addition of the segment duration, produce a 13-dimensional feature vector for each segment. These are the prosodic feature vectors that we use for GMM and factor analysis modeling. Note that since we used only the voiced part of the speech signal and we imposed a pseudo-syllable minimum duration of 60 ms, the total number of feature vectors within an utterance (an utterance is a five-minute telephone conversation) was much less than the number of corresponding MFCC frames. There is an average of 400 prosodic vectors per utterance.

6.2.2 Joint Factor Analysis as a Model for Prosody

Joint factor analysis is a model of speaker and session variability in GMMs. Although it is traditionally used with cepstral-type features, it can be applied to any type of continuous features for which Gaussian mixture modeling is appropriate. In our modeling, we used exactly the same JFA as used for the cepstral features. In cepstral JFA modeling, the term "channel variability" is used to describe the variability between several recording sessions of a given speaker because, in the majority of cases this variability is caused by channel effects. However, for high-level features as used in our modeling, the term "intersession variability"

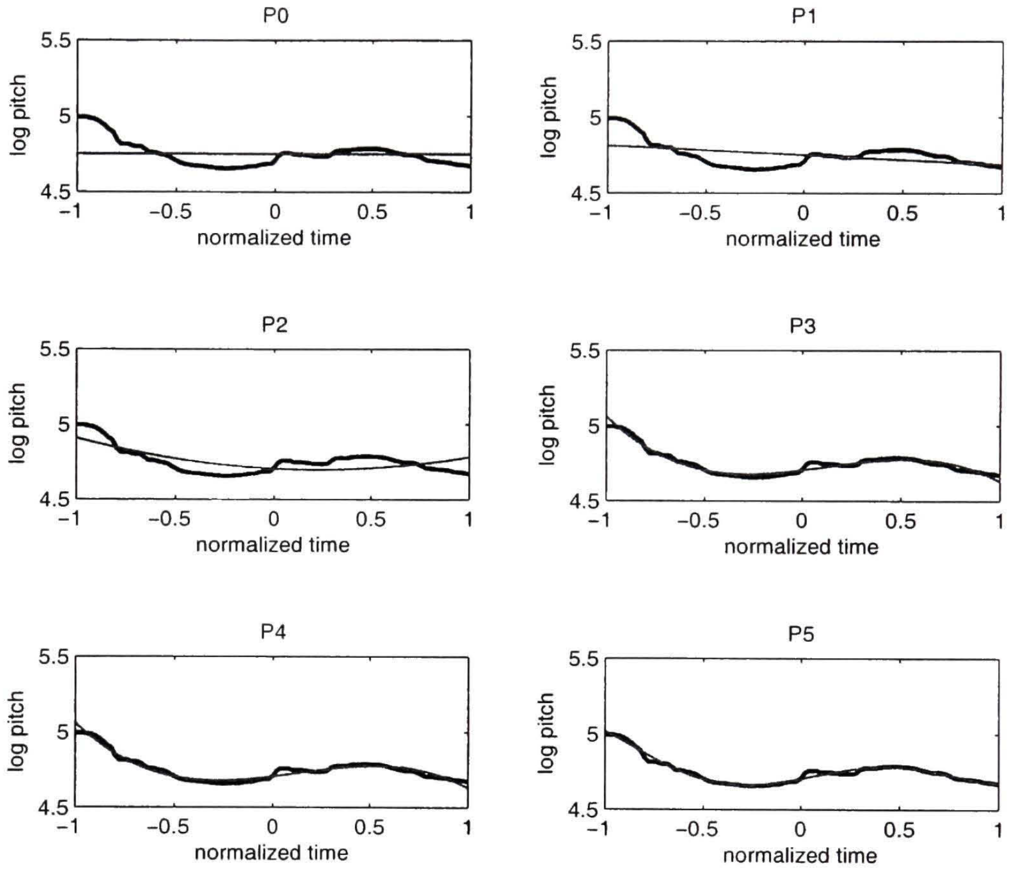

Figure 6.2 Approximation of the log pitch contour using Legendre polynomials with different orders.

is probably more appropriate than "channel variability". For this work, joint factor analysis with prosodic features is implemented essentially in the same way as standard joint factor analysis with cepstral features (only the features are different).

6.3 Experiments with Long-term Prosodic Features

6.3.1 Database

We carried out our experiments on the core condition of the NIST 2006 speaker recognition evaluation (SRE) [1]. This evaluation set contains 350 male and 461 female speakers; the number of test utterances is 51448. For each target speaker model, a five-minute recording is available which contains roughly two minutes of speech from that speaker. We used a universal background model made up of 512 Gaussians, trained on the (13-dimensional) prosodic features extracted from LDC releases of Switchboard II, Phases 2 and 3; Switchboard Cellular, Parts 1 and 2 ;and NIST 2004-2005 speaker recognition evaluation datasets. The same data was used to train the factor analysis model. In the joint factor analysis framework, it is necessary to use this kind of dataset to model intersession variability since each training speaker has to be recorded several times (ideally under a wide variety of recording conditions). Where speaker and common factors were used, we carried out decoupled estimation (Kenny et al., 2008b) of the eigenvoice matrix and diagonal matrix D. The zt-norm technique has proved to be useful in the joint factor analysis framework (Kenny et al., 2007a; Vogt et al., 2005). Accordingly, verification scores were normalized using zt-norm normalization with 280 t-norm models and 1000 z-norm utterances from the same dataset for each gender.

6.3.2 Joint Factor Analysis with Prosodic Features

The experiments carried out in this section aim to find the best configuration for the joint factor analysis model (i.e, the optimal number of speaker and intersession factors) for the 13-dimensional prosodic features presented in section 6.2.1. The results for the female subset of the English and all trials core condition of the NIST 2006 SRE dataset are summarized in Table 6.1 and Table 6.2.

[1] http://www.nist.gov/speech/tests/spk/index.htm

Table 6.1

Results (on EER and MinDCF) obtained on English trials of the core condition of the female subset of the NIST 2006 Evaluation dataset using prosodic joint factor analysis with several configurations

	Speaker factors	Channel factors	Common factors	EER	MinDCF
1	50	20	Yes	12.00%	0.0622
2	50	20	No	11.54%	0.0608
3	70	30	No	10.98%	0.0589
4	90	40	No	10.69%	0.0592
5	100	40	No	10.69%	0.0590
6	100	50	No	**10.60%**	**0.0589**
7	0	0	No	28.06%	0.0994
8	100	0	No	25.33%	0.0960
9	0	50	No	25.24%	0.0979

Table 6.2

Results (on EER and MinDCF) obtained on all trials of the core condition of the female subset of the NIST 2006 Evaluation dataset using prosodic joint factor analysis with several configurations

	Speaker factor	Channel factors	Common factors	EER	MinDCF
1	50	20	Yes	14.88%	0.0701
2	50	20	No	15.10%	0.0706
3	70	30	No	14.39%	0.0713
4	90	40	No	14.14%	0.0698
5	100	40	No	13.97%	0.0698
6	100	50	No	**13.90%**	**0.0698**
7	0	0	No	28.54%	0.0992
8	100	0	No	26.71%	0.0982
9	0	50	No	27.09%	0.0987

If we first compare the results obtained in lines 1 and 2 of both Tables 6.1 and 6.2, we see that inclusion of common factors does not contribute to improving performance on English trials; a marginal improvement is however achieved in the all trials case. On the basis of those results, we chose not to include common factors and diagonal matrix D in future prosodic JFA configurations. The best configuration in both trials conditions was found to have 100 speaker factors and 50 intersession factors (see lines 2 to 6 of both tables 6.1 and 6.2). For prosodic features, we used fewer speaker factor components than for cepstral features (300 speaker factors) (Kenny et al., 2007a,b). This is based on the observation that both the speaker and intersession eigenvalues show a steep initial decrease followed by an exponential decrease towards zero, as shown in Figure 6.3. This figure also underscores the preponderance of speaker variability (as measured by the sum of the eigenvalues) over intersession variability. This confirms that our prosodic features are less sensitive to intersession effects, but vary considerably from one speaker to another.

In order to show the effectiveness of the speaker and intersession factors, we carried out three experiments with and without speaker and intersession factors. The results of these experiments for English trials are given in lines 7, 8 and 9 of Table 6.1 (the same behavior is also observed in all trial experiments, as shown in Table 6.2).

- Line 7 of Table 6.1 corresponds to an experiment where speaker and intersession factors are ignored ($U = 0,\ V = 0,\ D = 0$). This is basically equivalent to the standard GMM-UBM approach for speaker verification (Reynolds et al., 2000). The results in lines 6 and 7 of Table 6.1 show that disregarding speaker and intersession factors results in a significant degradation in performance (10.60% EER with speaker and intersession factors versus 28.06% EER with no speaker and with no intersession factors).

- Line 8 of Table 6.1 corresponds to an experiment using only 100 speaker factors and no intersession factors ($U = 0,\ V \neq 0,\ D = 0$). This modeling is based on eigenvoice MAP adaptation. The purpose of this experiment is to highlight the importance of the intersession factors. The results given in lines 6 and 8 of Table 6.1 show that inclusion of intersession factors improves the EER performance from 26.71% without intersession factors to 10.60% with intersession factors.

- Line 9 of Table 6.1 corresponds to an experiment that uses classical MAP adaptation for enrollment, intersession factors, but no speaker factors $(U \neq 0,\ V = 0,\ D = 0)$. The purpose of this experiment is to ascertain the contribution stemming from the speaker factor component. The results given in lines 6 and 9 of Table 6.1 show that the use of speaker factors improves the performance from an EER of 27.09% without speaker factors to 10.60% with speaker factors.

This last experiment leads us to conclude that speaker factors play an important part in enrolling target speakers. From the point of view of our approach, this result can be explained by noting that we have few feature vectors to estimate the target speaker model (an average of 400 vectors per enrollment). It is important to note that in classical MAP adaptation, only the Gaussians observed in the enrollment data are adapted. This is because in traditional MAP adaptation, the GMM supervector covariance matrix is assumed to be diagonal; there is thus no correlation between the Gaussians in this type of GMM. However, in joint factor analysis modeling, the GMM supervector covariance matrix is given by $VV^t + D^2$ with diagonal matrix D and a low rank rectangular matrix V. The matrix V takes into account the correlations between Gaussians in a speaker model. Gaussians that are not observed in the enrollment data are also adapted by using statistics from the other Gaussians. The number of speaker factors whose values need to be estimated in enrollment is much smaller than the number of parameters estimated in classical MAP adaptation. Thus, the method is effective even with very limited amounts of enrollment data. The use of intersession factors also proves to be important in our approach because they model session variability (see lines 6 and 8 of both Tables 6.1 and 6.2). Our prosodic factor analysis system gives the best results when both speaker and intersession factors are taken into account.

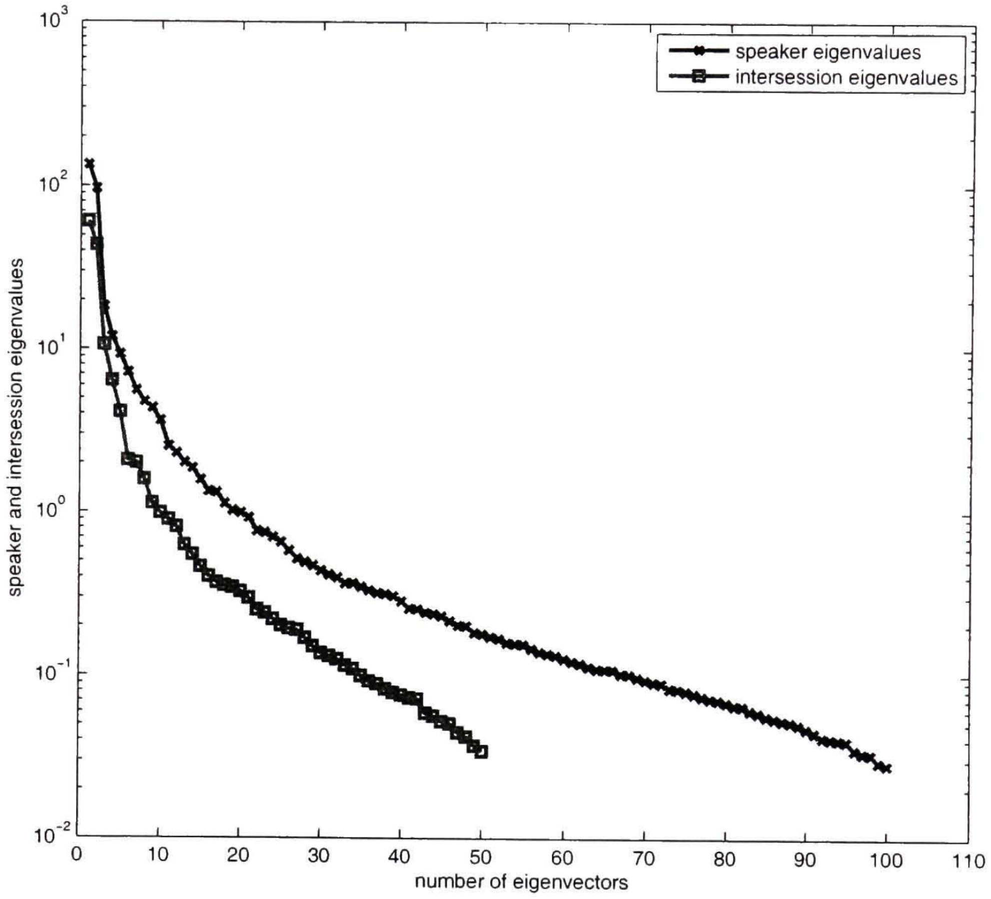

Figure 6.3 The eigenvalues of VV^t (the speaker eigenvalues, upper curve) and the eigenvalues of UU^t (the intersession eigenvalues, lower curve) obtained by the prosodic joint factor analysis.

6.3.3 Importance of Energy, Duration, and Pitch

This subsection shows the effectiveness of Legendre polynomials for modeling the prosodic contours and the importance of the information given by the energy for speaker modeling. For the purpose of comparing with other approaches to prosodic feature extraction and modeling, we performed three experiments on the female subset of the NIST 2006 evaluation data (core condition, English and all trials), varying the feature set as follows:

1 In the first experiment we computed, for each segment, the slope and curvature of the pitch and energy contours, as well as the segment duration, as features in a manner similar to (Adami et al., 2003). Note that the slope and curvature correspond to the coefficients a_1 and a_2 of Equation (6.1). The result is given in line 1 of Table 6.3.

2 In the second experiment, we used as segment features the Legendre polynomial coefficients of the pitch contour (all 6 coefficients) and the duration of the segment. The energy contour was not used. This modeling was similar to (Sönmez et al., 1997). Line 2 of Table 6.3 gives the results for this experiment.

3 In the last experiment, we used all 13 prosodic features, as described in Section 6.2.1. The result for this experiment is given in line 3 of Table 6.3.

Table 6.3

Results obtained on English and all trials of the core condition of the female subset of the NIST 2006 Evaluation dataset using joint factor analysis with several types of prosodic features

	Features	English trial		All trials	
		EER	MinDCF	EER	MinDCF
1	slope + curvature + duration	19.56%	0.0812	22.55%	0.0864
2	pitch + duration	13.67%	0.0704	17,63%	0.0849
3	pitch + energy + duration	**10.60%**	**0.0589**	**13.90%**	**0.0698**

Our best performance over the various prosodic feature sets considered was obtained with the full 13 dimensional feature set (see line 3 of Table 6.3). The energy contour clearly adds a substantial amount of information to the pitch contour (a 3% absolute reduction in EER for the English condition, comparing the results of lines 2 and 3 of Table 6.3). The same conclusion is found in SNERF modeling (Shriberg et al., 2005). Shriberg *et al.* found that using information about the pitch, energy, and duration of different units gives the best performance. Table 6.3 shows that the slope and curvature representation of the pitch and energy contour is less effective than using all the Legendre polynomial coefficients (comparing the results for lines 1 and 3 of Table 6.3).

6.3.4 Results for Both Genders

We tested the joint factor analysis model with the 13-prosodic coefficients for both genders on the core condition of the NIST 2006 speaker recognition evaluation dataset. We used the same joint factor analysis configuration for each gender (100 speaker factors and 50 intersession factors). The UBM size is 512 Gaussians for each gender. The decision scores are normalized with zt-norm. The results obtained (under the two conditions: English only, and all trials) are given in Table 6.4.

Table 6.4

Results obtained with gender-dependent prosodic factor analysis on the core condition of the NIST 2006 Evaluation dataset

	English trials		All trials	
	EER	MinDCF	EER	MinDCF
Females	10.60%	0.0589	13.90%	0.0698
Males	11.02%	0.0557	13.14%	0.0638
Both genders	10.91%	0.0587	13.63%	0.0672

The results show that for English trials these prosodic features are better for female speakers than for male speakers. The opposite is true of our cepstral-based joint factor analysis system. Ferrer *et al.* have recently published EERs in the range 12.09% - 13.65% on the English subset (both genders) of the NIST 2006 evaluation data, results obtained with three systems based on three different sequence transforms of the SNERF features modeled by a smoothed SVM classifier (Ferrer et al., 2007a,b). If we restrict ourselves to the English subset of the NIST 2006 speaker recognition evaluation dataset, then our EER is 10.91%. The results obtained with our prosodic system are much better than those obtained with other systems based on the SNERF approach. The advantage of our approach is that segmentation into pseudo-syllabic units is carried out in an unsupervised manner, taking solely into consideration the energy contour of voiced speech. Furthermore, a speech recognition system is required for the SNERF approach. Although the results obtained with these systems are reported on the English trials only, our system is limited by no language restrictions whatsoever. As part of her thesis (Ferrer, 2009), Lucianna Ferrer compared her approach, based on the GMM prosodic feature transform, with Stanford Research Institute (SRI) joint factor analysis implementation. Both techniques achieved equivalent performance (an ERR around 14.70% on 2006 NIST SRE data); these results are however inferior to those obtained with our prosodic JFA. To explain this discrepancy, we first note that our JFA uses more data in order to better estimate the speaker and intersession factors. Another factor that plays a major role in our system performance relates to the nature of our JFA implementation: the likelihood of sequence of speech frames for a given speaker model is obtained by integrating over intersession factors (equation 2.29), which allows modeling some uncertainty in the speaker model estimation. In contrast, the SRI implementation uses a linear approximation of the likelihood. In the case when few speaker frames are available, it is more appropriate to model some model estimation uncertainty, because the fewer vectors we have to estimate the speaker model the greater uncertainty there is in the model estimation.

For comparison purposes, the Queensland University of Technology (QUT) provided us with the results of their prosodic system [2], which is based on an approach similar to (Adami et al., 2003). On the core condition of the NIST 2006 speaker recognition evaluation dataset, EERs of 21.1% (English trials) and 22.5% (all trials) were obtained. It is clear that our approach produces better results.

6.4 Modeling Long-term Vocal Tract Features with Joint Factor Analysis

In this section we discuss application of joint factor analysis model long-term speaker vocal tract varation. We used the same Legendre polynomial approximation as given in the preced-

[2] `http://www.stat.cmu.edu/~minka/papers/logreg/`

ing section to approximate two sets of vocal tract feature contours: the first features are the formants and the second are the Mel Frequency Cepstral Coefficient Components.

6.4.1 Formants

We begin by modeling the $F1$ and $F2$ formant contours in the same way as the pitch and energy contours by using Legendre polynomials. The motivation of using the formants is to introduce phonetic context information to the pseudo-syllables which will lead to a better modeling of these entities. The formants are usually used to detect and classify the vowels. Formants have already been used in speaker verification (Mezghani et O'Shaughnessy, 2005; Tanabian et al., 2005). In (Mezghani et O'Shaughnessy, 2005), Mezghani *et al.* combine the formants with the cepstral coefficients (MFCC). Tanabian *et al.* propose to model the formants' trajectory using decision tree modeling and a neural network approach for speaker recognition (Tanabian et al., 2005). The formants are extracted using *praat* package (Boersma, 2001). This software is based on Burg approach as described in (Childers, 1978).

Experiment

Experimental Set-up

The experiments are carried out for the core condition of the NIST 2006 Speaker Recognition Evaluation. The universal background model and joint factor were trained on the same dataset as in the previous experiment. The decision scores are normalized using zt-norm normalization based on 1000 impostors for z-norm and 280 impostors for t-norm, taken from the same data used for UBM training. Similar to the previous experiments with prosodic features, we extract log pitch, log energy and log formant ($F1$, $F2$) at 10 ms intervals with the *praat* package (Boersma, 2001). Pitch is acquired using the autocorrelation method proposed in (Boersma, 2001) and is defined only in voiced regions. For each utterance, the energy is normalized by subtracting the maximum value of the same utterance. The use of actual formant values results in very large Legendre polynomial coefficient values, producing very large variances. To circumvent this problem, we used the log of formants in order to ensure numerical stability of the Legendre polynomial coefficient values.

Results

We carried out three experiments with the aim of showing the advantage of combining the formants with other prosodic features in order to model long-term speaker characteristics.

1 The first experiment consists in interpreting the following information: contour of pitch and energy, along with duration of pseudo-syllables, in the same way as in previous experiments. The resulting 13-dimensional feature vector consists of six coefficients from a Legendre polynomial for the pitch contour, six coefficients for the energy, plus the pseudo-syllable duration. GMMs are used to model these feature vectors. Joint factor analysis is also used to deal with speaker and intersession effects. We used a two gender-dependant UBMs with 512 Gaussians and joint factor analysis composed of 100 speaker and 50 intersession factors.

2 In the second experiment, we modeled long-term formant characteristics. We used six Legendre polynomial coefficients for the log of the formant $F1$ and six coefficients for the log of the formant $F2$, using the same pseudo-syllable units as the first experiment. To these twelve coefficients, we added the pseudo-syllable duration. The final feature vectors are of dimension 13. The relevant UBM contains 1024 Gaussians (formants introduce context-related information from the pseudo-syllables, which requires more Gaussians to be properly taken into account). The joint factor analysis configuration is the same as that of the first experiment.

3 The third experiment employs the 13 features of the first experiment, extended by the six Legendre polynomial coefficients for log formant $F1$ and six coefficients for log formant $F2$. The final feature vectors are of dimension 25. The UBM used contains

1024 Gaussians. We used the same joint factor analysis configuration as in the first two experiments.

The results obtained in these three experiments are given in Tables 6.5 and 6.6. The first noteworthy point is that prosodic features achieved better results for females than for males. However, when formant features are used, we discriminate better among male than among female speakers. We can conclude that prosodic information can be used to improve performance for female trials, while minimizing performance differences between males and females. These results also show that contours of the $F1$ and $F2$ formants contribute significantly as supplemental information to the pitch and energy contours: the addition of the formants $F1$ and $F2$ results in significant performance improvement, especially in the male case (an absolute improvement of approximately 4% in EER for the English trials). The formants introduce information that helps to better discriminate between the pseudo-syllables and thus to better model them.

Table 6.5

Joint factor analysis results applied to long-term prosodic and formant features in English trials of the core condition of the NIST 2006 Evaluation dataset

		Males		Females		Both genders	
		EER	MinDCF	EER	MinDCF	EER	MinDCF
1	Prosodic features	11.02%	0.0557	10.60%	0.0589	10.91%	0.0587
2	Formant features	12.37%	0.0572	18.01%	0.0683	15.81%	0.0668
3	Formant and prosodic features	**6.93%**	**0.0324**	**8.23%**	**0.0613**	**7.69%**	**0.0394**

6.4.2 Mel Frequency Cepstral Coefficients

In this section, we model the evolution of vocal tract characteristics by using MFCC coefficients. This approach is quite similar in spirit to (Adami et al., 2003; Sönmez et al., 1997). In this modeling, we begin by extracting a set of MFCC components using a sliding window of

Table 6.6

Joint factor analysis results applied to long-term prosodic and formant features in all trials of the core condition of the NIST 2006 Evaluation dataset

		Males		Females		Both genders	
		EER	MinDCF	EER	MinDCF	EER	MinDCF
1	Prosodic features	13.14%	0.0638	13.90%	0.0698	13.63%	0.0672
2	Formant features	15.37%	0.0670	19.83%	0.0704	17.86%	0.0729
3	Formant and prosodic features	**9.09%**	**0.0451**	**11.83%**	**0.0613**	**10.64%**	**0.0549**

duration 25 ms and an overlap of 10 ms. We then apply the feature warping transform without removing silence, using a sliding window of 3 seconds. Thereafter, we process the time evolution of the various cepstral coefficients independently. For each pseudo-syllable, we approximate each MFCC component using six Legendre polynomial coefficients. We used the same pseudo-syllable segmentation as in the prosodic system. In this modeling, the Legendre polynomial coefficients better model the MFCC component evolution than the classic calculation of the first and second time derivatives of the MFCC. We also approximate the energy contour using a Legendre polynomial in the same manner as for the prosodic features. The resulting feature vectors were modeled using Gaussian mixture models and joint factor analysis was applied in order to model the speaker and intersession variability.

Experiments

Experimental Set-up

We used the same data as in the formant experiments to train the UBM and JFA. We used the same impostors to carry out the zt-norm score normalization. The experiments are tested on the core condition of the NIST 2006 SRE.

Results

We carried out three experiments to show the importance of combining prosodic features with long-term MFCC features and also the influence of adding formants to these latter features.

1 The first experiment is based on 12-MFCC frames. We used the same pseudo-syllable segmentation as in Secttion 6.3 in order to approximate each MFCC component contour using Legendre Polynomials of order five. For each pseudo-syllable, we obtained feature vectors of dimension 79 (12 × 6 for MFCC + 6 for log-energy + 1 for pseudo-syllable duration). For joint factor analysis modeling, we used gender-dependent UBMs with 1024 Gaussians and 200 speaker factors and 50 intersession factors.

2 The second experiment differs from the first one only in that we augmented the 79-dimensional feature vectors by adding the 6 Legendre polynomial pitch coefficients. This experiment was carried out to ascertain wether MFCCs contain pitch-related information.

3 The third and last experiment used feature vectors of dimension 97, comprised of 12×6 for MFCC, 6 for pitch, 6 for formant $F1$, 6 for formant $F2$, 6 for log-energy and 1 for pseudo-syllable duration. We used the same UBM and JFA configuration as in the two previous experiments. We carried out this experiment to ascertain wether formants contribute extra information to the MFCCs.

The results for these three experiments are given in Tables 6.7 and 6.8. In (Vair et al., 2007), the authors propose a speaker verification system based on several GMMs trained on different phonemes. Their phonetic segmentation was obtained using a hybrid phonetic decoder composed of two model types: Hidden Markov Model and Neural Networks. To our knowledge, this system produces the best results in cases where high-level phonetic information is used. If we compare our EER obtained on the all trials condition (both genders) of the NIST 2006

Table 6.7

Joint factor analysis results applied to long-term MFCC, prosodic and formant features in English trials of the core condition of the NIST 2006 Evaluation dataset

		Male		Female	
		EER	MinDCF	EER	MinDCF
1	MFCC+E+D	5.11%	0.0241	5.39%	0.0270
2	MFCC+P+E+D	**4.53%**	**0.0213**	**4.47%**	**0.0239**
3	MFCC+P+F1+F2+D	4.64%	0.0246	4.54%	0.0242

Table 6.8

Joint factor analysis results applied to long-term MFCC, prosodic and formant features in all trials of the core condition of the NIST 2006 Evaluation dataset

		Male		Female	
		EER	MinDCF	EER	MinDCF
1	MFCC+E+D	7.65%	0.0375	8.50%	0.0436
2	MFCC+P+E+D	**6.91%**	**0.0332**	**7.91%**	**0.0434**
3	MFCC+P+F1+F2+D	7.11%	0.0364	8.14%	0.0447

SRE (7.9%) with that obtained with the phonetic GMMs system (6.0%), we can conclude that our preliminary long-term MFCC modeling achieved respectable results compared to the phonetic GMMs modeling. An important point to note is that, in our modeling, we carried out an unsupervised segmentation, without recourse to any phonetic or syllabic decoder. The two other experiments reveal that adding the pitch contour to the MFCCs improves performance (especially for female trials), which demonstrates that MFCCs may not contain, at the outset, information about the pitch. However, combining the MFCCs with formant contours leads to no significant improvement, revealing that MFCCs implicitly model formant-related information. We arrived at the same conclusion in (Dehak et al., 2007b) when the formant and MFCC scores were fused.

6.5 Discussion

Although the most successful approaches to speaker recognition rely on short-term spectral features such as MFCCs, it has long been recognized that prosodic contours contain complementary information to the short-term cepstral data. In order to exploit prosodic information, many systems have been developed which use sophisticated modeling techniques such as n-gram modeling of stylized pitch contours (Adami et al., 2003), or complex language-dependent features, which can only be extracted with the aid of a speech recognizer (Kajarekar et al., 2004; Shriberg et al., 2005). However, recent work in language identification (Lin et Wang, 2005) and quantitative phonetics (Grabe et al., 2003) has shown that a simple approach to prosodic feature extraction, namely fitting pitch and energy contours with Legendre polynomial expansions, can be very effective. In this chapter, we have explored the application of this type of prosodic feature extraction to speaker verification. The experiments showed that our modeling achieved the best results compared to other prosodic systems. An interesting characteristic of our modeling is that the prosodic features performed better for females than for males for the English trials of the NIST evaluation (the opposite is true of our cepstral-based system). A key aspect of the coefficients in the Legendre polynomial expansion is that they define a continuous rather than a discrete feature set. Thus, they are amenable to modeling with the methods that have already been developed for modeling cepstral features in state-of-the-art speaker recognition systems, such as Gaussian mixture modeling (Reynolds et al., 2000) and joint factor analysis (Kenny et al., 2007b). Our experiments showed that both speaker and intersession factors play a useful role. Speaker factors are helpful because the number of prosodic feature vectors available for enrolling a target speaker is relatively small. There is only one feature vector per pseudo-syllable, rather than one vector per 10 ms for cepstral features. Intersession factors are useful because the Legendre coefficients are not entirely robust to session variability.

We proposed to the use of long-term formant and MFCC features with prosodic contours for speaker recognition. These combinations yield to significant performance improvement compared with the performances obtained with the prosodic features only. However when we combined the long-term cepstral features with formant contours, we observed no improvement, which means that the MFCCs already contain information about formants. In the next chapter, we will present linear scores fusion of the long-term cepstral and prosodic features with state-of-the-art short-term cepstral features. This score fusion is based on logistic regression training.

CHAPTER 7

SCORE FUSION

This chapter presents score fusion performance for three speaker verification systems which combine long-term and short-term features developed in this dissertation. The first system is based on the classical joint factor analysis model. The second is the combination of the support vector machines and the joint factors analysis presented in Chapter 5. Finally, the third system is the combination of support vector machines and traditional Gaussian mixture models based on an universal background model.

7.1 Score Fusion

As already introduced in section 5.3.1, the linear score fusion technique consists in combining the scores of several subsystems into a single, definitive score. This score is compared to a threshold in order to take the final decision. In the recent NIST speaker recognition evaluation competitions, the best performance was achieved by systems that combine several subsystems. These subsystems are based on different modelings and operate on different features of speaker information (Brummer et al., 2007). Several approaches have been applied in order to combine the scores of different systems. In (Campbell et al., 2007), Campbell *et al.* used a neural network model to estimate the score fusion weights. These fusion weights are trained to minimize the DCF. Naïve Bayes score fusion, based on equal fusion weights, has already been tested by Kajarekar (Kajarekar, 2005) for fusing four different support vector machine systems. Support vector machines (Ferrer et al., 2006) have already been applied to estimation of fusion weights based on polynomial kernels of orders 1, 2, and 3. The final decision score is obtained by averaging the three SVM scores obtained from the three kernels. The most popular fusion approach used during the NIST 2008 speaker recognition evaluation campaign was based on training the linear score fusion weights using logistic regression training. The linear score fusion is evaluated as follows:

$$\boxed{s_f\,(w) = w_0 + \sum_{l=1}^{M} w_l\, s_l} \tag{7.1}$$

where s_f is the final fused score, s_l is the score for the l^{th} subsystem, M is the number of fused subsystems and $w = (w_0, w_1, ..., w_M)$ is a vector comprised of the score fusion weights.

7.2 Logistic Regression

The logistic regression used to estimate the score fusion weights (Brummer et al., 2007; Leeuwen et Brummer, 2007) is based on supervised training. In this approach, we require a set of labeled scores for each sub-system in the same corpus. The score labels correspond to the target and non-target trials. During a given NIST speaker recognition evaluation campaign, it is usually possible to exploit previous NIST evaluations datasets to estimate the fusion weights, since at the end of each competition, NIST distributes the test labels. Using logistic regression to estimate the fusion weights was motivated by the fact that it improves the discriminative power of the fused system and it serves also as a basis for defining final calibrated scores (Brummer et du Preez, 2006).

Logistic regression can be used to fuse the scores from several systems (as described in equation 7.1) and also to calibrate the decision scores for a single system (we set $M = 1$ in equation 7.1). By calibration, we mean finding a linear transformation for the scores from a single system which projects the scores in the same range and onto setting speaker-independent universal decision threshold. Given a new speaker, it is not necessary to find a corresponding new decision threshold: it is sufficient to transform the underlying scores using the weights obtained by logistic regression to allow for proper comparison with the universal threshold.

In speaker verification, we are seeking to calibrate or to fuse scores that can be interpreted as a log-likelihood ratio. Given a test utterance X, the new calibrated score $s_f(w)$ can be represented according to the following log-likelihood ratio:

$$\boxed{s_f(w) = \log\left(\frac{\mathrm{P}(X|H_{\text{tar}})}{\mathrm{P}(X|H_{\text{non}})}\right)} \tag{7.2}$$

where H_{tar} and H_{non} represent, respectively, the target speaker and impostor access hypotheses. This modeling is different from traditional logistic regression usually that is used to model the score as a formulation of posterior log-odds:

$$\begin{aligned}
s_f(w) &= \log\left(\frac{\mathrm{P}(H_{\text{tar}}|X)}{\mathrm{P}(H_{\text{non}}|X)}\right) && (7.3)\\
&= \log\left(\frac{\mathrm{P}(X|H_{\text{tar}})}{\mathrm{P}(X|H_{\text{non}})}\right) + \log\left(\frac{\mathrm{P}_{\text{tar}}}{\mathrm{P}_{\text{non}}}\right) && (7.4)\\
&= \log\left(\frac{\mathrm{P}(X|H_{\text{tar}})}{\mathrm{P}(X|H_{\text{non}})}\right) + \log\left(\frac{\mathrm{P}_{\text{tar}}}{1-\mathrm{P}_{\text{tar}}}\right) && (7.5)
\end{aligned}$$

$$\boxed{s_f(w) = \log\left(\frac{\mathrm{P}(X|H_{\text{tar}})}{\mathrm{P}(X|H_{\text{non}})}\right) + \operatorname{logit}(\mathrm{P}_{\text{tar}})} \tag{7.6}$$

The difference between the scores given in equations 7.2 and 7.6 lies in the additive term of prior log-odds ($\operatorname{logit}(\mathrm{P}_{\text{tar}})$) in the case of the *a posteriori* modeling scores. The absence of the prior in the case of log-likelihood-based modeling can be circumvented by modifying the objective function that is to be optimized (Brummer et al., 2007; Leeuwen et Brummer, 2007):

$$\boxed{\begin{aligned}
o(w, \mathrm{P}_{\text{tar}}) = \frac{\mathrm{P}_{\text{tar}}}{\|\chi_{\text{tar}}\|} \sum_{x \in \chi_{\text{tar}}} \log\left(1 + e^{-s(w) - \operatorname{logit}(\mathrm{P}_{\text{tar}})}\right) \\
+ \frac{1 - \mathrm{P}_{\text{tar}}}{\|\chi_{\text{non}}\|} \sum_{x \in \chi_{\text{non}}} \log\left(1 + e^{-s(w) + \operatorname{logit}(\mathrm{P}_{\text{tar}})}\right)
\end{aligned}} \tag{7.7}$$

where $\|\chi_{\text{tar}}\|$ and $\|\chi_{\text{non}}\|$ represent, respectively, the number of target and impostor trials. This version of the objective function is given with respect to to the speaker's score prior, which allows us to first set distribution-independent speaker data parameters and to minimize the DCF (see section 1.8.1); a metric used in the NIST competition. The target speaker's new optimal prior log odds score is given by:

$$\boxed{\operatorname{logit}\left(\mathrm{P_{tar}}\right) = \operatorname{logit}\left(\mathrm{P'_{tar}}\right) + \log\left(\frac{C_{\mathrm{FR}}}{C_{\mathrm{FA}}}\right)} \tag{7.8}$$

where $\mathrm{P'_{tar}} = 0.01$, $C_{\mathrm{FR}} = 10$ and $C_{\mathrm{FA}} = 1$ are parameters representing, respectively, the initial prior of the speaker target trial, the relative cost of false rejection errors and false acceptance errors. These parameters are given in the NIST evaluation plan [1]. The values for the new *a priori* probability of the target speaker trial, based on these parameters, are given by $\mathrm{P_{tar}} = 0.0917$.

As described above, the main purpose of score calibration in the speaker verification systems is to exploit a speaker-independent decision threshold. The logistic regression score calibration approach allows for a theoretical determination of this threshold, based on the following equation:

$$\boxed{\theta_{\mathrm{DCF}} = -\operatorname{logit}\left(\mathrm{P'_{tar}}\right) - \log\left(\frac{C_{\mathrm{FR}}}{C_{\mathrm{FA}}}\right)} \tag{7.9}$$

where $\mathrm{P'_{tar}} = 0.01$, $C_{\mathrm{FR}} = 10$ and $C_{\mathrm{FA}} = 1$ are the parameters described above. The value of the optimal decision threshold that minimizes the DCF is given by $\theta_{\mathrm{DET}} = 2.29$. In all of our fusion experiments, we used the Focal toolkit [2] developed by Niko Brummer for estimating the score fusion weights. This software implements a conjugate gradient algorithm, based on the work of Minka [3], which optimizes the convex objective function described in equation 7.7.

[1] http://www.nist.gov/speech/tests/spk/index.htm

[2] http://www.dsp.sun.ac.za/~nbrummer/focal/

[3] http://www.stat.cmu.edu/~minka/papers/logreg/

7.3 Missing Trials

In the case of logistic regression-based score fusion, it may happen, for some reason, that subsystems do not produce scores for test files (e.g., if a problem occurred during feature extraction). We then need to start by calibrating the scores from individual subsystem in the manner described in the previous section; thereafter, we add the missing tests with a score equal to zero. This technique allows *a priori* knowledge of the scores from all trials to be used for learning the fusion weights. The same approach is used in the event of missing scores during the test step.

7.4 Experiment

In this section, we present the results of score fusion of the prosodic-based systems and spectral long-term features, as described in the preceding chapter, with the systems operating on short-term speaker features. The score fusion weights were estimated on the core condition of the NIST 2006 SRE and tested on the telephone data from the core condition of the NIST 2008 SRE. We carried out gender-dependent score fusion. In the case of the all trial NIST evaluation condition, we estimated two separate sets of fusion weights. The first set of weights concerns the English trials, while the second pertains to the non-English trials. This approach is motivated by the results obtained by the SRI group during the NIST 2008 SRE (Kajarekar et al., 2009). For real-life applications, this type of approach is inadequate because we actually need to know the speaker's language to apply the appropriate score fusion weights. However, in the case of the NIST evaluation campaign, it is allowed to exploit the language information of the recording since this information is given at the outset. In the next sections, we will describe the systems that are applied to the two features categories.

7.4.1 Short-term Speaker Characteristic

Feature Extraction

Our systems operate on cepstral features, extracted using a 25 ms Hamming window. 19 Mel Frequency Cepstral Coefficients together with log-energy are calculated every 10 ms. This 20-dimensional feature vector was subjected to feature warping (Kajarekar, 2005) using a 3 s sliding window. Delta and double delta coefficients were then calculated using a 5-frame window to produce a 60-dimensional feature vectors.

Joint Factor Analysis

We used gender-dependent universal background models each containing 2048 Gaussians. These UBMs were trained using LDC releases of Switchboard II, Phases 2 and 3; Switchboard Cellular, Parts 1 and 2; and NIST 2004-2005 speaker recognition evaluation. We applied gender-dependent joint factor analysis models comprised of 300 speaker factors and 100 channel factors and common factors. These JFA were trained on the same amounts of data as the corresponding UBMs. We used decoupled estimation of the eigenvoice matrix V and diagonal matrix D. The eigenvoice matrix V was trained using all the UBM training, except the NIST 2004 SRE data. The D matrix was trained on NIST 2004 SRE data. The decision scores obtained with joint factor analysis were normalized using zt-norm normalization. We used 300 t-norm models and approximately 1000 z-norm utterances for each gender. All these impostors were taken from the same dataset as used for UBM training.

Support Vector Machines and Joint Factor Analysis

This system was introduced in section 5.5.2. It is based on the application of support vector machines in total factor space. We used the same gender-dependent universal background models as in the previous joint factor analysis models. The total variability matrix T of the factor analysis was trained using LDC releases of Switchboard II, Phases 2 and 3; Switchboard Cellular, Parts 1 and 2; Fisher English database Part 1 and 2; and NIST 2004-2005 speaker recognition evaluation. We used 400 total factors. The cosine kernel was used in order to build the SVM system. Linear discriminant analysis and within class covariance normalization were used to compensate for channel effects. The linear discriminant analysis projection matrix was trained on the same data as the total variability matrix except the Fisher English databases. The within class covariance matrix was trained on the NIST 2004 and 2005 SRE data. We used approximately 300 t-norm models taken from NIST 2005 SRE and about 1200 impostor models to train SVM taken from the other databases for each gender.

Cosine Distance Scoring

We used the same total variability matrix T and total factor vectors as the previous SVM-JFA system. The same SVM-JFA t-norm impostor models are used. The impostors used to train the SVM in the preceding system are applied as z-norm utterances.

Support Vector Machines and Gaussian Mixture Models

This new system is a combination of the SVM and the GMM-UBM system described in Chapter 4. It is based on the Gaussian kernel derived from a Kullback-Leibler divergence between two GMMs. In this experiment, two gender-dependent universal background models are used. Each UBM contains 2048 Gaussian components and trained using LDC releases of Switchboard II, Phases 2 and 3; Switchboard Cellular, Parts 1 and 2; and NIST 2004 SRE. The SVM was trained using 1000 impostors for each gender, extracted from the UBM training data. The decision scores obtained with the SVM-GMM were normalized using t-norm normalization based on 227 male and 283 female speakers. The nuisance attribute projection matrix of corank $= 30$ was trained with the same data used to train the UBM.

7.4.2 Long-term Speaker Characteristics

Feature Extraction

We extract log-pitch, log-energy and twelve MFCCs at 10 ms intervals. We segment the contours into pseudo-syllable-like regions as described in Chapter 6. This method is based on detecting the valley points in the energy contour of the voiced speech component. In general, these valley points serve as segment boundaries, but we also impose a minimum duration constraint of 60 ms. We approximated each contour using Legendre polynomial expansions of order five (six terms). The coefficients of the Legendre polynomials were used as input for the joint factor analysis model.

Prosodic System

We used six coefficients for log-pitch and six coefficients for log-energy and pseudo-syllable duration as feature representations. These features were modeled using gender-dependent UBMs comprised of 512 Gaussians. The UBMs were trained on LDC releases of Switchboard II, Phases 2 and 3; Switchboard Cellular, Parts 1 and 2; and NIST 2004-2005. Joint factor analysis was used to deal with speaker and channel variabilities. The JFA is composed of 100 speaker factors, 50 channel factors and no common factors. We used 300 t-norm models and approximately 1000 z-norm utterances for each gender taken from UBM training data in order to carry out zt-norm score normalization.

Cepstral System

We used six coefficients for each MFCC contour and six coefficients for log-energy and pseudo-syllable duration as features representations. These features were modeled using gender-dependent UBMs comprised of 1024 Gaussians. The UBMs were trained in LDC releases of Switchboard II, Phases 2 and 3; Switchboard Cellular, Parts 1 and 2; and NIST 2004-2005. The joint factor analysis model is comprised of 200 speaker factors, 50 channel factors and no common factors. We used the same t-norm and z-norm impostor as in the previous prosodic system.

Prosodic and Cepstral System

This system is similar to the previous one, except that we include six additional Legendre polynomial coefficients associated with the pitch. We used exactly the same UBM and JFA configurations as for the cepstral system.

7.4.3 Results

In this section we present the results of the short-term speaker characteristic systems and their fusion with the long-term feature systems. The results for the individual long-term speaker feature systems are given in Appendix B.

Joint Factor Analysis System

We begin by presenting results from the fusion of the three long-term speaker characteristic systems with joint factor analysis based on the short-term MFCC features. This fusion is carried out on the core condition of the NIST 2006 and 2008 speaker recognition evaluation. At the outset, we should point out that the results pertaining to NIST 2006 SRE should be interpreted with caution, because we have trained and tested the fusion weights on the same dataset. The score fusion results are given in Tables 7.1 and 7.2.

The score fusion results show a very slight improvement in the case of female speakers, especially in the all trial condition. However, a greater decrease in the EER and the DCF is observed in the case of male speakers. We obtain an EER of 1.75% on English trials and 4.80% on all trials, which translates to a relative improvement of approximately 33% and 20% respectively. The DET curves given in Figure 7.1 illustrate the improvement of score fusion in the case for males in both conditions.

Table 7.1

Score fusion results between JFA and long-term speaker feature systems are given on EER and MinDCF for the female part of the core condition of the NIST 2006 and 2008 SRE, English trials

Language condition	System	NIST 2006		NIST 2008	
		EER	MinDCF	EER	MinDCF
English trials	JFA: $s = m + Vy + Dz$	1.64%	0.0120	3.15%	0.0150
	JFA + long-term features	**1.37%**	**0.0100**	**2.89%**	**0.0147**
All trials	JFA: $s = m + Vy + Dz$	3.11%	0.0189	6.15%	0.0315
	JFA + long-term features	**3.06%**	**0.0181**	**6.15%**	**0.0305**

Table 7.2

Score fusion results between JFA and long-term speaker feature systems are given on EER and MinDCF for the male part of the core condition of the NIST 2006 and 2008 SRE, all trials

Language condition	System	NIST 2006		NIST 2008	
		EER	MinDCF	EER	MinDCF
English trials	JFA: $s = m + Vy + Dz$	1.19%	0.0058	2.63%	0.0112
	JFA + long-term features	**0.93%**	**0.0049**	**1.75%**	**0.0108**
All trials	JFA: $s = m + Vy + Dz$	2.76%	0.0136	5.26%	0.0272
	JFA + long-term features	**2.25%**	**0.0130**	**4.80%**	**0.0261**

Support Vector Machines and Joint Factor Analysis System

The results of score fusion of SVM-JFA applied to the short-term MFCC features with the other systems based on long-term cepstral and prosodic features are given in Tables 7.3 and 7.4. These results were obtained on the core condition of the NIST 2006 and 2008 SRE datasets.

The fusion of scores from the three systems based on long-term speaker features and the system based on the combination of SVM and JFA shows significantly improvements in

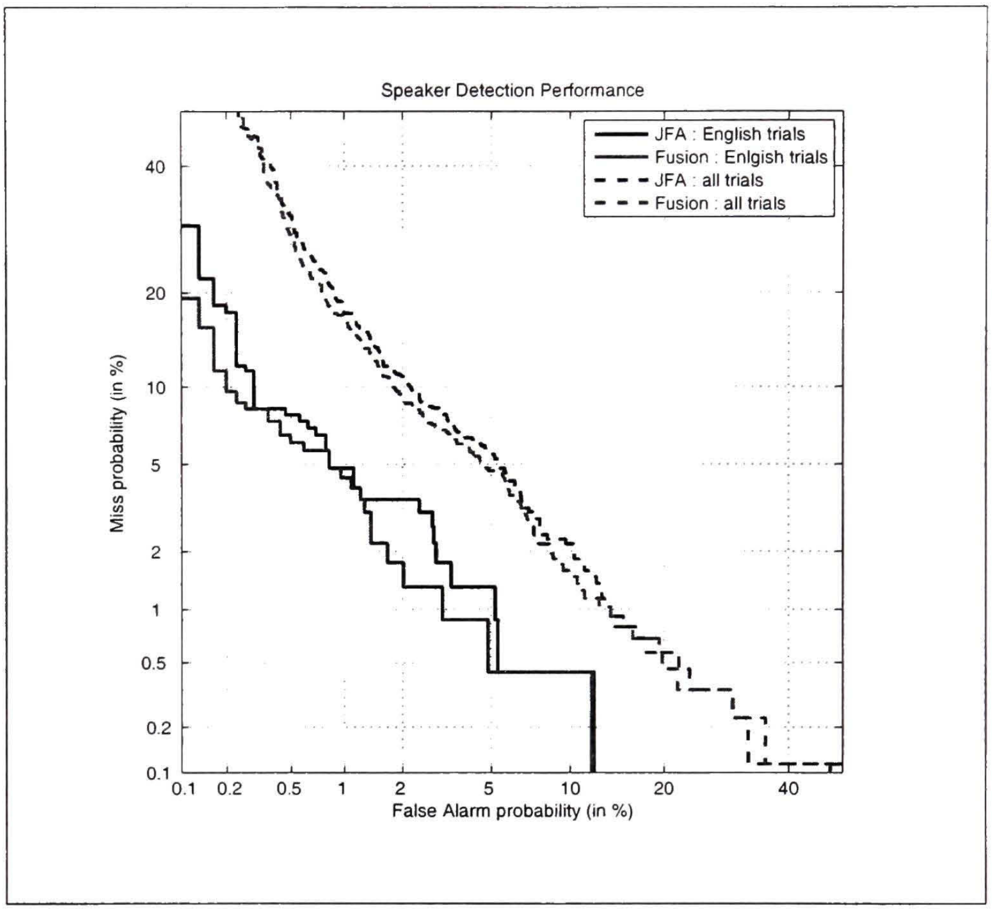

Figure 7.1 Comparison results between JFA alone and fused system comprised of JFA and long-term speaker characteristic systems. The results are given for the male part of the core condition of the NIST 2008 speaker recognition evaluation.

MinDCF for the female case. For the English trials, the MinDCF value decreases from 0.0150 to 0.0134. For all trials, it goes from 0.0322 to 0.0316. In the case of the male speakers, on the other hand, we observe no improvement for the English trials and a slight improvement in EER for all trial condition (a 5% relative improvement). In the case of male spakers, if we compare the results obtained with the SVM-JFA only with the results from score fusion between classical JFA and long-term speaker feature systems (Tables 7.1 and 7.2), we find that the JFA-SVM yields better results than the fusion of the other four systems (1.28% in EER for the SVM-JFA compared to 1.75% for the score fusion given in Table 7.2). In the case of female speakers, the fusion of the four systems achieves a slight improvement in MinDCF

Table 7.3

Score fusion results between SVM-JFA and long-term speaker feature systems are given on EER and MinDCF for the female part of the core condition of the NIST 2006 and 2008 SRE, English trials

Language condition	System	NIST 2006		NIST 2008	
		EER	MinDCF	EER	MinDCF
English trials	SVM-JFA: (LDA + WCCN)	1.55%	0.0095	3.68%	0.0150
	SVM-JFA + long-term features	**1.37%**	**0.0082**	**3.68%**	**0.0134**
All trials	SVM-JFA: (LDA + WCCN)	2.42%	0.0142	6.04%	0.0322
	SVM-JFA + long-term features	**2.27%**	**0.0135**	**5.93%**	**0.0316**

Table 7.4

Score fusion results between SVM-JFA and long-term speaker feature systems are given on EER and MinDCF for the male part of the core condition of the NIST 2006 and 2008 SRE, all trials

Language condition	System	NIST 2006		NIST 2008	
		EER	MinDCF	EER	MinDCF
English trials	SVM-JFA: (LDA + WCCN)	1.59%	0.0102	1.28%	0.0095
	SVM-JFA + long-term features	**1.19%**	**0.0076**	**1.28%**	**0.0095**
All trials	SVM-JFA: (LDA + WCCN)	2.63%	0.0143	4.57%	0.0238
	SVM-JFA + long-term features	**2.25%**	**0.0127**	**4.34%**	**0.0234**

(0.0147 compared to 0.0150 in the case of the SVM-JFA system) but a higher EER (2.89% for the score fusion given in Table 7.1 compared to 3.68% for the SVM-JFA). In the context of the NIST speaker recognition evaluation, where we need to minimize the DCF, the SVM-JFA performance remains very competitive with that of other systems based on multi-system fusion.

Cosine Distance Scoring

The score fusion results obtained with cosine distance scoring which is applied on short-term MFCC features and long-term cepstral and prosodic feature systems are given in Tables 7.5 and 7.6. These results were obtained on the core condition of the NIST 2006 and 2008 SRE datasets.

Table 7.5

Score fusion results between cosine distance scoring and long-term speaker feature systems are given on EER and MinDCF for the female part of the core condition of the NIST 2006 and 2008 SRE, English trials

Language condition	System	NIST 2006		NIST 2008	
		EER	MinDCF	EER	MinDCF
English trials	Cosine distance: (LDA + WCCN)	1.46%	0.0102	2.90%	0.0124
	Cosine distance + long-term features	**1.19%**	**0.0092**	**2.89%**	**0.0124**
All trials	Cosine distance: (LDA + WCCN)	2.54%	0.0162	5.76%	0.0322
	Cosine distance + long-term features	**2.34%**	**0.0138**	**2.90%**	**0.0316**

Table 7.6

Score fusion results cosine distance scoring and long-term speaker feature systems are given on EER and MinDCF for the male part of the core condition of the NIST 2006 and 2008 SRE, all trials

Language condition	System	NIST 2006		NIST 2008	
		EER	MinDCF	EER	MinDCF
English trials	Cosine distance: (LDA + WCCN)	1.59%	0.0093	1.12%	0.0094
	Cosine distance + long-term features	**1.06%**	**0.0071**	**1.15%**	**0.0095**
All trials	Cosine distance: (LDA + WCCN)	2.63%	0.0147	4.48%	0.0247
	Cosine distance + long-term features	**2.15%**	**0.0121**	**4.48%**	**0.0239**

The results of score fusion between cosine distance scoring and long-term feature systems do not achieve significant improvements. This is quite the same behavior as for SVM-JFA systems. However, individual system based on the cosine distance scoring gives the best results compared to the score fusion between short-term JFA scoring and long-term feature systems.

Support Vector Machines and Gaussian Mixture Model System

In this section, we compare the performance between two different categories of systems based on different modelings and feature representations. The first category is based on JFA modeling applied to long-term cepstral and prosodic features. The second category is based on the combination of SVM and GMM-UBM systems applied to MFCC frame features. The results of this fusion are given in Tables 7.7 and 7.8.

Table 7.7

Score fusion results between SVM-GMM and long-term speaker feature systems are given on EER and MinDCF for the female part of the core condition of the NIST 2006 and 2008 SRE, English trials

Language condition	System	NIST 2006		NIST 2008	
		EER	MinDCF	EER	MinDCF
English trials	SVM-GMM: (Gaussian kernel)	4.75%	0.0238	7.10%	0.0238
	SVM-GMM + long-term features	**2.83%**	**0.0162**	**4.73%**	**0.0332**
All trials	SVM-GMM: (Gaussian kernel)	6.92%	0.0366	10.64%	0.0540
	SVM-GMM + long-term features	**5.68%**	**0.0294**	**8.92%**	**0.0450**

In both Tables 7.7 and 7.8, we note a substantial reduction of the EER and MinDCF for both genders in comparison with the previous experiments. The margin for improvement is greater for the GMM-UBM system, since its initial performance was lower than that of the SVM-JFA system. If we compare the results obtained from the fusion between the GMM-UBM and long-term feature systems with those already obtained in the two previous experiments, we find that, in general, this last fusion does not perform as well as the first ones.

Table 7.8

Score fusion results between SVM-GMM and long-term speaker feature systems are given on EER and MinDCF for the male part of the core condition of the NIST 2006 and 2008 SRE, all trials

Language condition	System	NIST 2006		NIST 2008	
		EER	MinDCF	EER	MinDCF
English trials	SVM-GMM: (Gaussian kernel)	4.65%	0.0243	4.82%	0.0289
	SVM-GMM + long-term features	**2.39%**	**0.0147**	**3.94%**	**0.0177**
All trials	SVM-GMM: (Gaussian kernel)	6.08%	0.0300	8.46%	0.0446
	SVM-GMM + long-term features	**4.26%**	**0.0229**	**6.86%**	**0.0356**

7.5 Discussion

In this chapter, we have discussed the influence of score fusion of systems using long- and short-term speaker characteristics. The long-term features are extracted from spectral and prosodic characteristics. These features were modeled thereafter using joint factor analysis. The short-term features are based on MFCC frames modeled using several models. The fusion results show that the long-term feature systems fuse well with the classical short-term JFA system, especially in male trials. However, the individual systems, based on the cosine distance scoring and SVM-JFA applied to MFCC frames achieve very competitive results, compared to the fused system.

CONCLUSION

The major contribution of the first part of this thesis is our presentation of a speaker verification system designed as a combination of joint factor analysis and support vector machines. In this new modeling, each speech recording is characterized by a vector of dimension 400. This vector, designated as "total factors", represents the coordinate of this recording in a new space referred to as "the total variability space", which is also defined using factor analysis model. Unlike the classical joint factor analysis model based on the definition of two distinguished spaces, where the first models the variability between speakers and the second depends on the channel, our modeling uses the JFA to define a single space that simultaneously includes both speaker and channel variabilities. In this approach, the JFA plays the role of a feature extractor and can be regarded as a Principal Component Analysis that compresses the sequence of speech frames. The support vector machines are then applied in the total variability space using a cosine kernel. We have also shown that we can remove the SVM from the decision process and used directly the cosine kernel values as decision scores. In order to cancel out the channel effect, we tested several techniques; these are: Linear Discriminant Analysis, Nuisance Attribute Projection and Within Class Covariance Normalization. The best results were obtained when we applied LDA followed by WCCN. The advantage of using LDA is to define a new space which minimizes channel variability and maximizes the variability between the speakers. The results achieved by both cosine distance scoring and SVM-JFA systems are superior to those obtained with existing state-of-the-art systems and are also very competitive with those obtained from the fusion of multiple systems.

In the same kind of challenge that combines generative and discriminative models, we proposed a new Gaussian kernel to combine support vector machines and the classical GMM-UBM based on MAP adaptation for speaker verification. The nonlinear kernel was derived from approximated Kullback-Leibler divergence between two GMMs. The results for this new kernel were compared with those obtained from a linear kernel, proposed during the same time in (Campbell et al., 2006a), which is derived from the same distance between two

GMMs as our kernel. We have demonstrated the importance of addressing channel effects in both kernels, by applying the NAP algorithm in GMM space. We have also proved the usefulness of model normalization in the case of the Gaussian kernel. The performance obtained with the Gaussian kernel are equivalent to those obtained with the linear kernel.

In the second part of this thesis, we proposed testing a new approach for modeling long-term cepstral and prosodic information. This approach is based on the use of Legendre polynomials to approximate the spectral and prosodic contours as units that can be viewed as pseudo-syllables. These entities are segmented using the energy contour only. The advantage of this modeling is that it requires no phonetic nor word alignment, in contrast to most other approaches. The coefficients of the Legendre polynomials are then modeled with a Gaussian mixture model and joint factor analysis is applied to model the intersession and inter-speaker variabilities. The advantage of using JFA to model these new features is that we deal with a limited set of vectors for each speech recording (400 on the average) to adapt the universal background model to the target speaker data. It is important to note that in classical MAP adaptation, only the observed Gaussians are updated. However, in the eigenvoice adaptation used by joint factor analysis, the Gaussians which are not observed are also adapted using the statistical correlations with other Gaussians. The performance obtained with the joint factor analysis applied in the prosodic features has become the state of the art in this field. The advantage of these new prosodic features compared to other proposed features in the literature is that we do not require a speech recognizer in order to carry out the speech segmentation. The score fusion between the long-term speaker feature systems and the classical short-term JFA system produced satisfactory results, especially in male trials. However, the new system that we propose based on the combination of SVM and JFA produces better results than the fusion of long- and short-term JFA systems.

Future work

In this thesis, we proposed a new speaker verification architecture based on the combination of support vector machines and joint factor analysis models. In this novel combination, we used total factor vectors as input for SVM. In future work, it will be interesting to study the effect of combining total factor information as well as common factors, as was already done in Chapter 5. The new joint factor analysis configuration is based on two matrix representations: the first matrix is the total variability matrix T and the second is the diagonal matrix D. The experiments that we carried out using the SVM-JFA are based on telephone speech data. It would be interesting to extend this approach to the microphone and interview data that is currently available from the NIST 2008 speaker recognition evaluation. Using the SVM-JFA system, we achieved better results for male trials as compared to female trials. This indicates that the cosine kernel may not be appropriate for the female total variability space. More kernels need to be designed and investigated in this space.

Regarding long-term speaker characteristics, our proposed approach does not model the evolution of features between successive pseudo-syllables. Modeling these dynamics will better capture the speaker's style of speech. One possible avenue for modeling thes dynamics are the use of Hidden Markov Models instead of a Gaussian Mixture Models, since it is already known that hidden Markov models have the advantage of taking into account the temporal aspect of pseudo-syllable sequences. These models have been successfully applied to similar prosodic features for language identification.

APPENDIX A

Posterior distribution of the Joint factor analysis latent variables

Enrolling the target speaker in the case of joint factor analysis is based on the computation of the posterior distribution of the speaker supervector s given the Baum-Welch statistics. This distribution is related to the evaluation of the posterior distribution of the hidden variables. Let X be composed of the latent variables x,y and z

$$X = \begin{pmatrix} x \\ y \\ z \end{pmatrix} \tag{A.1}$$

the posterior distribution of X is a Gaussian distribution described in Proposition 2 of (Kenny et al., 2005a). Let define the matrices B and L as:

$$\begin{aligned} B &= \begin{pmatrix} U & V & D \end{pmatrix} && \text{(A.2)} \\ L &= I + B^t\,\Sigma^{-1}\,N\,B && \text{(A.3)} \end{aligned}$$

the covariance matrix and mean vector of the posterior distribution of X is given respectively by L^{-1} and $L^{-1}\,B^t\,\Sigma^{-1}\,(F - N\,m)$. The inverse of the matrix L is evaluated as follows:

$$\begin{pmatrix} a & b & c \\ b^t & I + V^t\Sigma^{-1}NV & V^t\Sigma^{-1}ND^2 \\ c^t & D\Sigma^{-1}NV & I + \Sigma^{-1}ND^2 \end{pmatrix} \tag{A.4}$$

where

$$\begin{aligned} a &= I + U^t\Sigma^{-1}NU && \text{(A.5)} \\ b &= U^t\Sigma^{-1}NV && \text{(A.6)} \\ c &= U^t\Sigma^{-1}ND && \text{(A.7)} \end{aligned}$$

The inverse of matrix L can be evaluated using the following step

$$\begin{pmatrix} \alpha & \beta \\ \beta^t & \gamma \end{pmatrix}^{-1} = \begin{pmatrix} \zeta^{-1} & -\zeta^{-1}\beta\gamma^{-1} \\ -\gamma^{-1}\beta^t\zeta^{-1} & \gamma^{-1} + \gamma^{-1}\beta^t\zeta^{-1}\beta\gamma^{-1} \end{pmatrix} \tag{A.8}$$

where

$$\zeta = \alpha - \beta\gamma^{-1}\beta^t \tag{A.9}$$

$$\alpha = \begin{pmatrix} a & b \\ b^t & I + V^t\Sigma^{-1}NV \end{pmatrix} \tag{A.10}$$

$$\beta = \begin{pmatrix} c \\ V^t\Sigma ND \end{pmatrix} \tag{A.11}$$

$$\gamma = I + \Sigma^{-1}ND^2 \tag{A.12}$$

APPENDIX B

The results for long- and short-term speaker feature systems

We present in this appendix the results of each individual long- and short-term feature system used in score fusion.

Table B.1

The results are given on EER and MinDCF on the male part of the core condition of the NIST 2006 SRE

	English trials		All trials	
	EER	MinDCF	EER	MinDCF
JFA: $s = m + Vy + Dz$	1.19%	0.0058	2.76%	0.0136
SVM-JFA: (LDA+WCCN)	1.59%	0.0102	2.63%	0.0143
SVM-GMM: Gaussian kernel	4.65%	0.0243	6.08%	0.0300
Pitch + energy + duration	11.02%	0.0557	13.14%	0.0638
Long-term MFCC + energy+duration	5.11%	0.0241	7.65%	0.0375
Long-term MFCC + pitch + energy + duration	4.53%	0.0213	6.91%	0.0331

Table B.2

The results are given on EER and MinDCF on the female part of the core condition of the NIST 2006 SRE

	English trials		All trials	
	EER	MinDCF	EER	MinDCF
JFA: $s = m + Vy + Dz$	1.64%	0.0120	3.11%	0.0189
SVM-JFA: (LDA + WCCN)	1.55%	0.0095	2.42%	0.0142
SVM-GMM: Gaussian kernel	4.75%	0.0238	6.92%	0.0366
Pitch + energy + duration	10.60%	0.0589	13.90%	0.0697
Long-term MFCC + energy + duration	5.39%	0.0270	8.50%	0.0436
Long-term MFCC + pitch + energy + duration	4.47%	0.0239	7.91%	0.0434

Table B.3

The results are given on EER and MinDCF on the male part of the core condition of the NIST 2008 SRE

	English trials		All trials	
	EER	MinDCF	EER	MinDCF
JFA: $s = m + Vy + Dz$	2.63%	0.0112	5.26%	0.0272
SVM-JFA: (LDA + WCCN)	1.28%	0.0095	4.57%	0.0238
SVM-GMM: Gaussian kernel	4.82%	0.0289	8.46%	0.0446
Pitch + energy + duration	13.04%	0.0618	13.85%	0.0754
Long-term MFCC + energy + duration	6.17%	0.0287	9.76%	0.0450
Long-term MFCC + pitch + energy + duration	4.39%	0.0220	8.47%	0.0398

Table B.4

The results are given on EER and MinDCF on the female part of the core condition of the NIST 2008 SRE

	English trials		All trials	
	EER	MinDCF	EER	MinDCF
JFA: $s = m + Vy + Dz$	3.15%	0.0150	6.15%	0.0315
SVM-JFA: (LDA + WCCN)	3.68%	0.0150	6.04%	0.0322
SVM-GMM: Gaussian kernel	7.10%	0.0238	10.64%	0.0540
Pitch + energy + duration	12.81%	0.0638	15.44%	0.0799
Long-term MFCC + energy + duration	6.63%	0.0385	11.16%	0.0528
Long-term MFCC + pitch + energy + duration	6.36%	0.0316	10.43%	0.0532

BIBLIOGRAPHY

Adami, A., Mihaescu, R., Reynolds, D., et Godfrey, J. (2003). « Modeling Prosodic Dynamics For Speaker Recognition. » *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 788–791, Hong Kong.

Andersen, E. et Andersen, A. (2000). « The MOSEK Interior Point Optimizer for Linear Programming: An Implementation of the Homogeneous Algorithm. » Frenk, H., Roos, K., Terlaky, T., et Zhang, S., editors, *High Performance Optimization*, pages 197–232. Kluwer Academic Publishers.

Baker, B. et Sridharan, S. (2006). « Speaker Verification Using Hidden Markov Models in a Multilingual Text-constrained Framework. » *IEEE Odyssey: The Speaker and Language Recognition Workshop*, San Juan, Puerto Rico.

Baker, B., Vogt, R., et Sridharan, S. (2005). « Gaussian Mixture Modeling of Broad Phonetic and Syllabic Events For Text Independant Speaker Verification. » *European Conference on Speech Communication and Technology*, pages 2429–2432, Lisbon, Portugal.

Ben, M. (2004). « Approches Robustes pour la Vérification Automatique du Locuteur par Normalisation et Adaptation Hiéarchique. » PhD thesis, University of Renne I.

Boersma, P. (2001). « Praat: Doing Phonetics by Computer. » *Glot International*, 5(9/10):341–345.

Brummer, N., Burget, L., Cernocky, J., Glembek, O., Grezl, F., Karaat, M., Leeuwen, D. V., Matejka, P., Schwarz, P., et Strasheim, A. (2007). « Fusion of Heterogeneous Speaker Recognition Systems in the STBU Submission for the NIST Speaker Recognition Evaluation 2006. » *IEEE Transaction On Audio, Speech and Language Processing*, 15(7):2072–2084.

Brummer, N. et du Preez, J. (2006). « Application-Independent Evaluation of Speaker Detection. » *Computer Speech and Language*, 20(2-3):230–275.

Campbell, W. et Assaleh, K. (1999). « Polynomial Classifier Techniques for Speaker Verification. » *ICASSP*, pages 321–324.

Campbell, W., Campbell, J., Reynolds, D., Jones, D., et Leek, T. (2004a). « Phonetic Speaker Recognition with Support Vector Machines. » *Advances in Neural Information Processing Systems*, pages 1377–1384. MIT Press.

Campbell, W., Sturim, D., Navratil, J., Shen, W., et Reynolds, D. (2007). « The MIT-LL/IBM 2006 Speaker Recognition System: High-Performance Reduced-Complexity Recognition. » *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 217–220, Honolulu, Hawaii.

Campbell, W., Sturim, D., Reynolds, D., et Solomonoff, A. (2006a). « SVM Based Speaker Verification using a GMM Supervector Kernel and NAP Variability Compensation. » *IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages 97–100, Toulouse.

Campbell, W. M. (2002). « Generalized Linear Discriminant Sequence Kernels for Speaker Recognition. » *IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages 161–164.

Campbell, W. M., Sturim, D. E., et Reynolds, D. (2006b). « Support Vector Machines Using GMM Supervectors for Speaker Verification. » *IEEE Signal Processing Letters*, 13(5):308–311.

Childers, D. G. (1978). « Modern Spectrum Analysis. » IEEE press.

Dehak, N. et Chollet, G. (2006). « Support Vector GMMs for Speaker Verification. » *IEEE Odyssey: The Speaker and Language Recognition Workshop*, San Juan, Puerto Rico.

Dehak, N., Dehak, R., Kenny, P., et Dumouchel, P. (2008a). « Comparison Between Factor Analysis and GMM Support Vector Machines for Speaker Verification. » *IEEE Odyssey: The Speaker and Language Recognition Workshop*, Stellenbosch, South Africa.

Dehak, N., Dumouchel, P., et Kenny, P. (2007a). « Modeling Prosodic Features With Joint Factor Analysis For Speaker Verification. » *IEEE Transaction on Speech and Signal Processing*, 15(7):2095–2103.

Dehak, N., Kenny, P., et Dumouchel, P. (2007b). « Continous Prosodic Features and Formant With Joint Factor Analysis for Speaker Verification. » *Interspeech*, Antwerp, Belgium.

Dehak, R., Dehak, N., Kenny, P., et Dumouchel, P. (2007c). « Linear and Non Linear Kernel GMM SuperVector Machines for Speaker Verification. » *Interspeech*, Antwerp, Belgium.

Dehak, R., Dehak, N., Kenny, P., et Dumouchel, P. (2008b). « Kernel Combination for SVM Speaker Verification. » *IEEE Odyssey: The Speaker and Language Recognition Workshop*, Stellenbosch, South Africa.

Dempster, A., Laird, N., et Robin, D. (1997). « Maximum Likelihood from Incomplete Data via the EM Algorithm. » *Journal of the Royal Statistical Society*, B:1–38.

Do, M. (2003). « Fast Approximation of Kullback-Leibler Distance for Dependence Trees and Hidden Markov Models. » *IEEE Signal Processing Letters*, 10(4):115–118.

Doddington, G., Przybocki, M., Martin, A., et Reynolds, D. (2000). « The NIST Speaker Recognition Evaluation: Overview, Methodology, Systems, Results, Perspectives. » *Speech Communication*, volume 31, pages 225–254.

Dong, X. et Zhaohui, W. (2001). « Speaker Recognition using Continuous Density Support Vector Machines. » *Electronics Letters*, 37(17):1099–1101.

Ferrer, L. (2009). « Statistical modeling of heterogeneous features for speech processing tasks. » PhD thesis, Stanford University.

Ferrer, L., Shriberg, E., Kajarekar, S., et Sönmez, K. (2007a). « Parametrezation of Prosodic Feature Distributions for SVM Modeling in Speaker Recognition. » *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Honolulu, Hawaii.

Ferrer, L., Shriberg, E., Kajarekar, S., Stolcke, A., Sönmez, K., Venkataraman, A., et Bratt, H. (2006). « The Contribution of Cepstral and Stylistic Features to SRI's 2005 NIST Speaker Recognition Evaluation System. » *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Toulouse.

Ferrer, L., Sönmez, K., et Shriberg, E. (2007b). « A Smoothing Kernel for Spatially Related Features and Its Application to Speaker Verification. » *Interspeech*, Antwerp, Belgium.

Furui, S. (1981). « Cepstral Analysis Technique For Automatic Speaker Verification. » *IEEE Transaction on Speech and Signal Processing*, 29:254–272.

Glembek, O., Burget, L., Brummer, N., et Kenny, P. (2009). « Comparaison of Scoring Methods used in Speaker Recognition with Joint Factor Analysis. » *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Taipei, Taiwan.

Grabe, E., Kochanski, G., et Coleman, J. (2003). « Quantitative Modelling of Intonational Variation. » *Speech Analysis and Recognition in Technology, Linguistics and Medicine*.

Gunawardana, A. (2001). « Maximum Mutual Information Estimation of Acoustic HMM Emission Densities. » Technical report 40, Center for Language and Speech Processing, Johns Hopkins University, Baltimore.

Gunawardana, A. et Byrne, W. (2001). « Discriminative Speaker Adaptation with Conditional Maximum Likelihood Linear Regression. » *International Conference on Speech Communication and Technology*, pages 1203–1206.

Hannani, A. E. et Petrovska-Delacrétaz, D. (2005). « Improving Speaker Verification System Using Alisp-based Specific GMMs. » *Audio- and Video-Based Biometric Person Authentication*, Hilton Rye Town, NY, USA. Springer.

Hannani, A. E. et Petrovska-Delacrétaz, D. (2007). « Data-Driven High-Level Information For Text-Independent Speaker Verification. » *IEEE Workshop on Automatic Identification Advanced Technologies*, pages 209–213, Alghero, Italy.

Hannani, A. E., Toledano, D. T., Petrovska-Delacrétaz, D., Montero-Asenjo, A., et Hennebert, J. (2006). « Using Data-driven And Phonetic Units For Speaker Verification. » *IEEE Odyssey: The Speaker and Language Recognition Workshop*, San Juan, Puerto Rico.

Hatch, A., Kajarekar, S., et Stolcke, A. (2006). « Within-Class Covariance Normalization for SVM-Based Speaker Recognition. » *International Conference on Spoken Language Processing*, Pittsburgh, PA, USA.

Jaakkola, T. et Haussler, D. (1999). « Exploiting Generative Models in Descriminative Classifiers. » M.S. Kearns, S. S. et Cohn, D., editors, *Advances in Neural Information Processing Systems*, volume 11. MIT Press.

Kajarekar, S. (2005). « Four Weightings and a Fusion: A Cepstral-SVM System for Speaker Recognition. » *IEEE Speech Recognition and Understanding Workshop*, pages 17–22, San Juan, Puerto Rico.

Kajarekar, S., Ferrer, L., Sönmez, K., Zheng, J., Shriberg, E., et Stolcke, A. (2004). « Modeling NERFs for Speaker Recognition. » *IEEE Odyssey: The Speaker and Language Recognition Workshop*, pages 51–56, Toledo, Spain.

Kajarekar, S., Scheffer, N., Graciarena, M., Shriberg, E., Stolcke, A., Ferrer, L., et Bocklet, T. (2009). « The SRI NIST 2008 Speaker Recognition Evaluation System. » *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Taipei, Taiwan.

Kenny, P. (2005). « Joint Factor Analysis of Speaker and Session Variability: Theory and Algorithms. » Technical report CRIM-06/08-13, CRIM, Montreal.

Kenny, P., Boulianne, G., et Dumouchel, P. (2005a). « Eigenvoice modeling with sparse training data. » *IEEE Trans. Speech Audio Processing*, 13(3).

Kenny, P., Boulianne, G., Ouellet, P., et Dumouchel, P. (2005b). « Factor Analysis Simplified. » *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 637–640, Philadelphia, PA.

Kenny, P., Boulianne, G., Ouellet, P., et Dumouchel, P. (2007a). « Joint Factor Analysis versus Eigenchannels in Speaker Recognition. » *IEEE Transaction on Audio Speech and Language Processing*, 15(4):1435–1447.

Kenny, P., Boulianne, G., Ouellet, P., et Dumouchel, P. (2007b). « Speaker and Session Variability in GMM-Based Speaker Verification. » *IEEE Transaction on Audio Speech and Language Processing*, 15(4):1435–1447.

Kenny, P., Dehak, N., Gupta, V., et Dumouchel, P. (2008a). « The Role of Speaker Factors in the NIST Extended Data Task. » *IEEE Odyssey: The Speaker and Language Recognition Workshop*, Stellenbosch, South Africa.

Kenny, P. et Dumouchel, P. (2004). « Disentangling Speaker and Channel Effects in Speaker Verification. » *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Montreal, Canada.

Kenny, P., Ouellet, P., Dehak, N., Gupta, V., et Dumouchel, P. (2008b). « A Study of Interspeaker Variability in Speaker Verification. » *IEEE Transaction on Audio, Speech and Language*, 16(5):980–988.

Lanckriet, G., Cristianini, N., Bartlett, P., Ghaoui, L. E., et Jordan, M. (2004). « Learning the Kernel Matrix with Semidefinite Programming. » *Journal of Machine Learning Reasearch*, 5:27–72.

Leeuwen, D. V. et Brummer, N. (2007). « An Introduction to Application-Independent Evaluation of Speaker Recognition Systems. » *Speaker Classification I: Fundamentals, Features, and Methods*, pages 330–353, Berlin, Heidelberg. Springer-Verlag.

Lin, C.-Y. et Wang, H.-C. (2005). « Language Identification Using Pitch Contour Information. » *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 601–604, Philadelphia, PA.

Ma, C. et Chang, E. (2003). « Comparaison of Discriminative Training Methods for Speaker Verification. » *IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages 192–195.

Martin, A., Doddington, G., Kamm, T., Ordowski, M., et Przybocki, M. (1997). « The DET Curve in Assessment of Detection Task Performance. » *European Conference on Speech Communication and Technology*, volume 4, pages 1895–1898.

Martin, T., Baker, B., Wong, E., et Sridharan, S. (2006). « A Syllabes-scale Farmework For Language Identification. » *Computer Speech and Language*, 20:276–302.

McLachlan, G., Peel, D., et Peel, D. (2000). « Finite Mixture Models. » Wiley series in probability and statistics. John Wiley & Sons, New York.

Mezghani, A. et O'Shaughnessy, D. (2005). « Speaker Verification Using a New Representation Based on a Combination of MFCC and Formants. » *IEEE Canadian Conference on Electrical and Computer Engineering*, Saskatoon, SK.

Moreno, P. et Ho, P. (2003). « A New SVM Approach to Speaker Identification and Verification Using Probabilistic Distance Kernels. » *European Conference on Speech Communication and Technology*, pages 2965–2968, Geneva, Swizerland.

Moreno, P., Ho, P., et Vasconcelos, N. (2003). « A Generative Model Based Kernel for SVM Classification in Multimedia Applications. » *Neural Information Processing Systems*.

Normandin, Y. (1991). « Hidden Markov Models, Maximum Mutual Information, and the Speech Recognition Problem. » PhD thesis, McGill University, Montreal.

Pelecanos, J. et Sridharan, S. (2001). « Feature Warping for Robust Speaker Verification. » *IEEE Odyssey: The Speaker and Language Recognition Workshop*, pages 213–218, Crete, Greece.

Preti, A., Scheffer, N., et Bonastre, J.-F. (2006). « Discriminant Approaches for GMM Based Speaker Detection Systems. » *International Workshop on Multimodal User Authentication, MMUA*, pages 50–56, Toulouse, France.

Reynolds, D. (2003). « Channel Robust Speaker Verification Via Feature Mapping. » *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Hon Kong, Chine.

Reynolds, D., Andrews, W., Campbell, J., Navratil, J., Peskin, B., Adami, A., Jin, Q., Klusacek, D., Abramson, J., Mihaescu, R., Godfrey, J., Jones, D., et Xiang, B. (2003). « The Supersid Project: Exploiting High-level Information for High-accuracy Speaker Recognition. » *IEEE International Conference on Acoustics, Speech, and Signal Processing*.

Reynolds, D., Quatieri, T., et Dunn, R. (2000). « Speaker Verification using Adapted Gaussian Mixture Models. » *Digital Signal Processing*, 10:19–41.

Reynolds, D. et Rose, R. (1995). « Robust Text-Independent Speaker Identification Using Gaussian Mixture Model. » *IEEE Transaction on Speech and Audio Processing*, 3(1):72–83.

Sarle, W. (1997). « Neural Network FAQ. Periodic posting to the Usenet newsgroup. »

Schmidt, M. et Gish, H. (1996). « Speaker Identification via Support Vector Machines. » *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 105–108.

Schölkopf, B. et Smola, A. (2001). « Learning with Kernels: Support Vector Machines, Regularization, Optimization and Beyond. » MIT Press.

Shawe-Taylor, J. et Cristianini, N. (2004). « Kernel Methods for Pattern Analysis. » Cambrige.

Shriberg, E., Ferrer, L., Kajarekar, S., Venkataraman, A., et Stocke, A. (2005). « Modeling Prosodic Feature Sequences for Speaker Recognition. » *Speech Communication*, pages 455–472.

Shriberg, E., Ferrer, L., Venkataraman, A., et Kajarekar, S. (2004). « SVM Modeling of SNRF-Gram for Speaker Recognition. » *International Conference on Spoken Language Processing*, Jeju Island, Korea.

Solomonoff, A., Campbell, W., et Boardman, I. (2005). « Advances in Channel Compensation For SVM Speaker Recognition. » *IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages 629–632.

Solomonoff, A., Quillen, C., et Campbell, W. (2004). « Channel Compensation for SVM Speaker Recognition. » *IEEE Odyssey: The Speaker and Language Recognition Workshop*, Spain.

Sönmez, K., Heck, L., Weintraub, M., et Shriberg, E. (1997). « A Lognormal Tied Mixture Model of Pitch for Prosody-Based Speaker Recognition. » *European Conference on Speech Communication and Technology*, pages 1391–1394, Rhodes, Greece.

Sönmez, K., Shriberg, E., Heck, L., et Weintraub, M. (1998). « Modeling Dynamic Prosodic Variation for Speaker Verification. » *International Conference on Spoken Language Processing*, pages 2631–2634, Sydney, Australia.

Sturm, J. (1999). « Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. » *Optimization Methods and Software*, 11–12:625–653. Special issue on Interior Point Methods (CD supplement with software).

Tanabian, M.-M., Tierney, P., et Zahirazami, B. (2005). « Automatic Speaker Recognition with Formant Trajectory Tracking using CART and Neural Networks. » *IEEE Canadian Conference on Electrical and Computer Engineering*, Saskatoon, SK.

Vair, C., Colibro, D., Castaldo, F., Dalmasso, E., et Laface, P. (2007). « Loquendo-Politecnico di Torino's 2006 NIST Speaker Recognition Evaluation System. » *Interspeech*.

Vair, C., Colibro, D., Castaldo, F., Dalmassoy, E., et Lafacey, P. (2006). « Channel Factors Compensation in Model Feature Domain for Speaker Recognition. » *IEEE Odyssey: The Speaker and Language Recognition Workshop*, San Juan, Puerto Rico.

Vapnick, V. (1995). « The Nature of Statistical Learning. » Springer.

Vogt, R., Baker, B., et Sridharan, S. (2005). « Modelling Session Variability in Text-Independent Speaker Verification. » *Interspeech*, pages 3117–3120, Lisboa.

Vogt, R., Kajarekar, S., et Sridharan, S. (2008). « Discriminat NAP for SVM Speaker Recognition. » *IEEE Odyssey: The Speaker and Language Recognition Workshop*, Stellenbosch, South Africa.

Wan, V. et Renals, S. (2003). « SVMSVM: Support Vector Machine Speaker Verification Methodology. » *IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 2, pages 221–224, Hong Kong.

Wan, V. et Renals, S. (2005). « Speaker Verification Using Sequence Discriminant Support Vector Machines. » *IEEE Transaction on Speech and Audio Processing*, 13(2):203–210.

Zissman, M. (1996). « Comparison of Four Approaches to Automatic Language Identification of Telephone Speech. » *IEEE Transaction on Speech and Audio Processing*, 4(1):54–58.